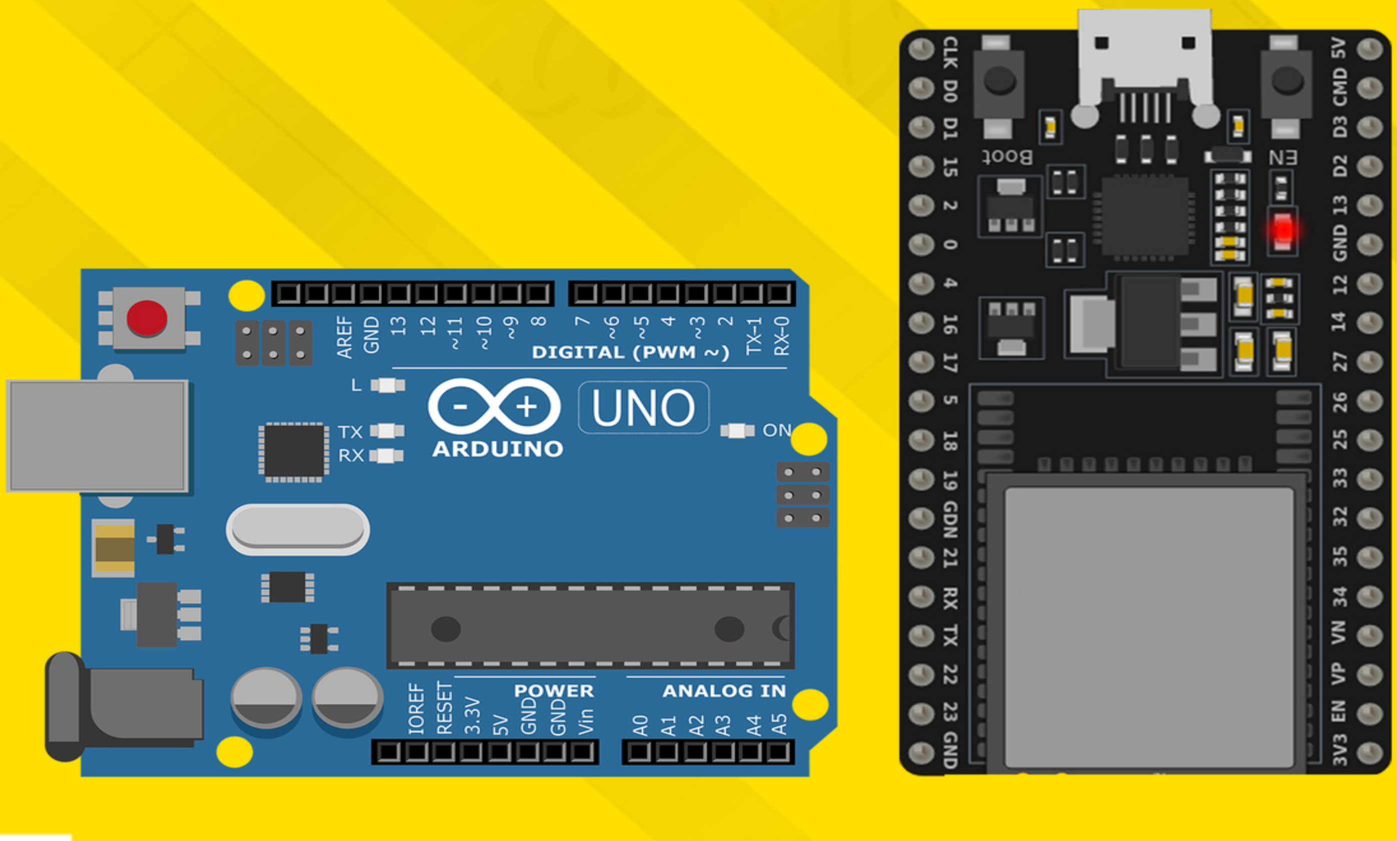
HANDBOOK OF ESP32 USING THE ARDUINO IDE

A Practical Guide to Development RTOD and Example Dummy Code



HANDBOOK OF ESP32 USING THE ARDUINO IDE

A Practical Guide to Development RTOD and Example Dummy Code

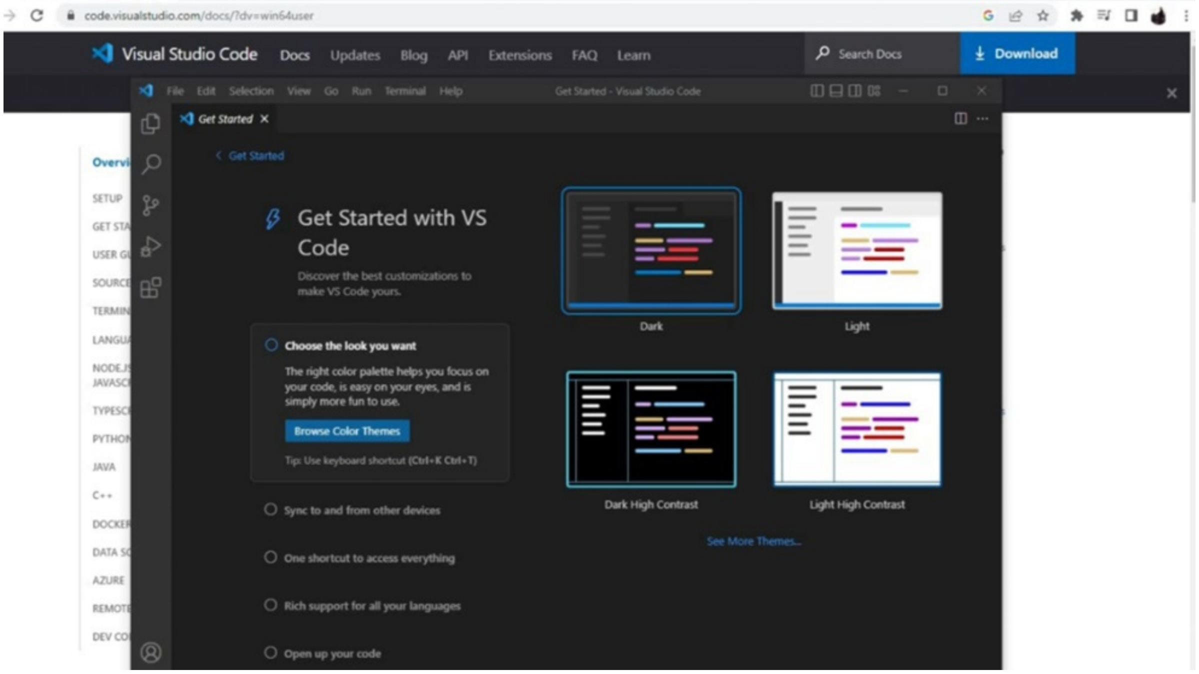
By

Jansa Selvam

TABLE OF CONTENTS

INSTALLING VSC AND LAUNCHING PROJECT

I will tell you how to write or doing programs for ESP 32 using the free Arctos real time operating system. To develop programs, I suggest you to use VS code with the platform IO extension, which can have more advanced functionality compared to the original Arduino IDE. At the same time, all sketches made in VS code are fully compatible with the Arduino ID and can be easily transferred from one ID to another.



code.visualstudio.com/docs/?dv=win64user

Get Started - Visual Studio Code

Get Started with VS Code

Discover the best customizations to make VS Code yours.

○ Choose the look you want

The right color palette helps you focus on your code, is easy on your eyes, and is simply more fun to use.

Browse Color Themes

Tip: Use keyboard shortcut (Ctrl+K Ctrl+T)

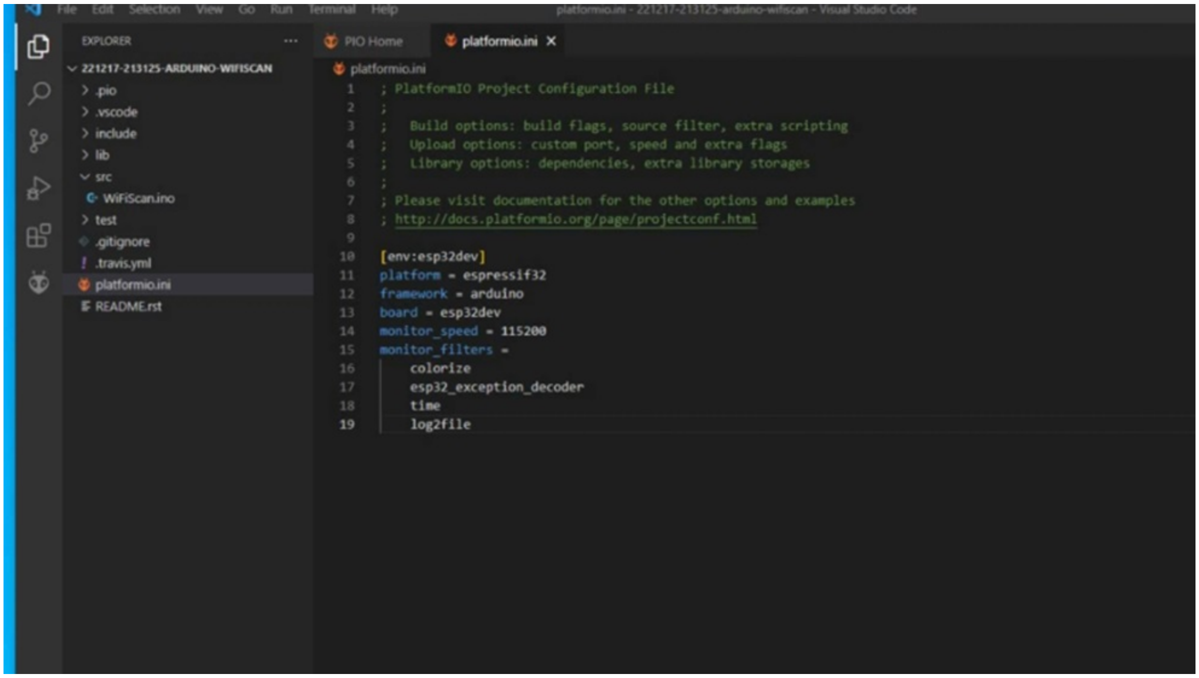
○ Sync to and from other devices

○ One shortcut to access everything

○ Rich support for all your languages

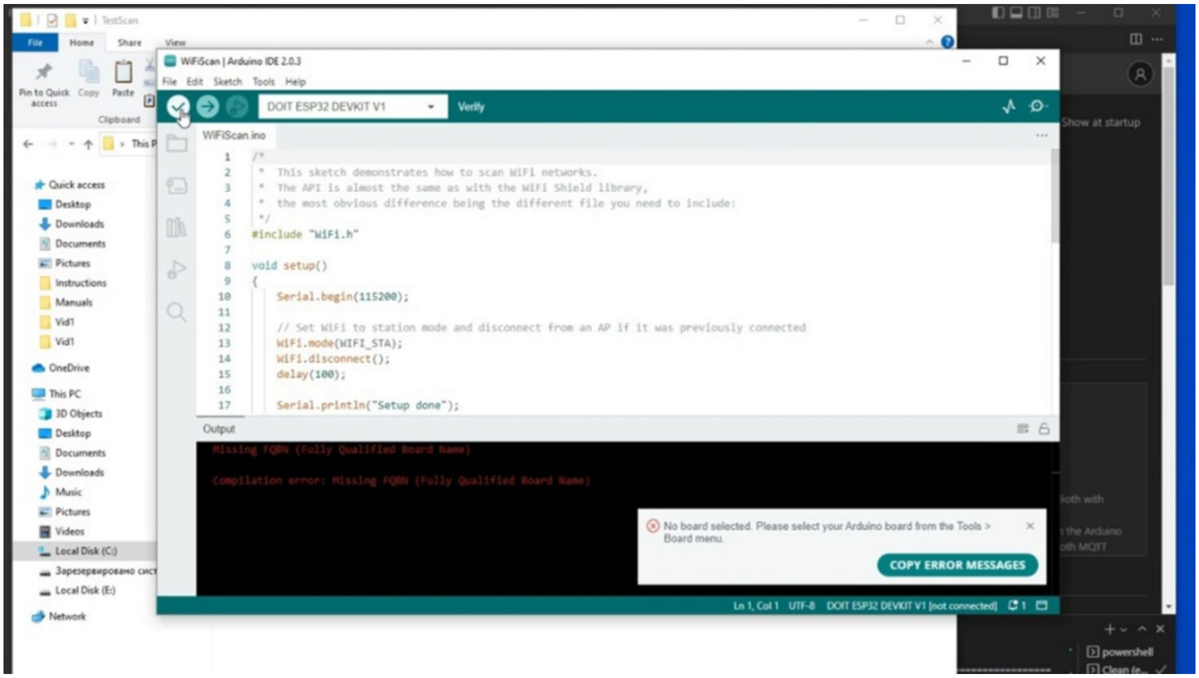
○ Open up your code

The same goes for libraries all libraries installed it in their doing ID can be used in VS code let's install VS code and I'll show you how to configure it for Arduino download the installer you can find the link to it in the attachment to this project I select the it for 64 bit Windows stopped and stolen install it next step is to install platform IO extension go to Extensions and find platformio here and install it bullet one mio is installed let's start with code here we can see what for my workstation next we will open some are doing a project to do this navigate to open and here select project examples and here we can select some example let it be doing away face calm import USA trust router Let's send Lucha into a bit let's take a look at the platform IO project. folder the files will need a source code or the ino file this one and a platform IO ini file this one you can see the IntelliSense warning that the does not support the ino file so we can rename it to CPP. Now IntelliSense will work let's look at platformio ini file we can see four different configurations or four different boards in this file.



```
; PlatformIO Project Configuration File
;
;   Build options: build flags, source filter, extra scripting
;   Upload options: custom port, speed and extra flags
;   Library options: dependencies, extra library storages
;
; Please visit documentation for the other options and examples
; http://docs.platformio.org/page/projectconf.html

[env:esp32dev]
platform = espressif32
framework = arduino
board = esp32dev
monitor_speed = 115200
monitor_filters =
    colorize
    esp32_exception_decoder
    time
    log2file
```

I'm using the first version of ESP 32 This is the first configuration We don't need to the others I can just delete them let's add another parameter to this configuration this parameter will enable logging all the monitored text information with time integration and decoding of ESP 32 exceptions further I will show how to use it let's compile this project and see how it works to do this I go to platform IO icon and first I choose Korean and then build built a successful cranium lucky accuses necessary audio the first configuration then it's not needed next we can upload the program to ESP 32 Login let's open terminal to sit down when you're stuck to ESP start this is my next Wi Fi and it's against construct as we can see everything is broken let's stop monitor and look at the clock file log file is here I can open it this here's everything that what was right in to monitor let's move for this project to the original Arduino IDE fuse to let's see where platform IO projects are located. Hover the cursor over the file header and see the path to it. This path to it open this folder and copy the CPP file this my platform IO projects this Wi Fi is calm SFC why first comm CPP copy and paste it to a new another folder I pasted here let's rename it back to No okay now we have Arduino file double click it I agree coke and Arduino creates a folder with the name of our file. Wi Fi is calm.



```
/*
 *  This sketch demonstrates how to scan WiFi networks.
 *  The API is almost the same as with the WiFi Shield library,
 *  the most obvious difference being the different file you need to include:
 */
#include "WiFi.h"

void setup()
{
    Serial.begin(115200);

    // Set WiFi to station mode and disconnect from an AP if it was previously connected
    WiFi.mode(WIFI_STA);
    WiFi.disconnect();
    delay(100);

    Serial.println("Setup done");
```

```
Missing FQBN (Fully Qualified Board Name)

Compilation error: Missing FQBN (Fully Qualified Board Name)
```

No board selected. Please select your Arduino board from the Tools > Board menu.

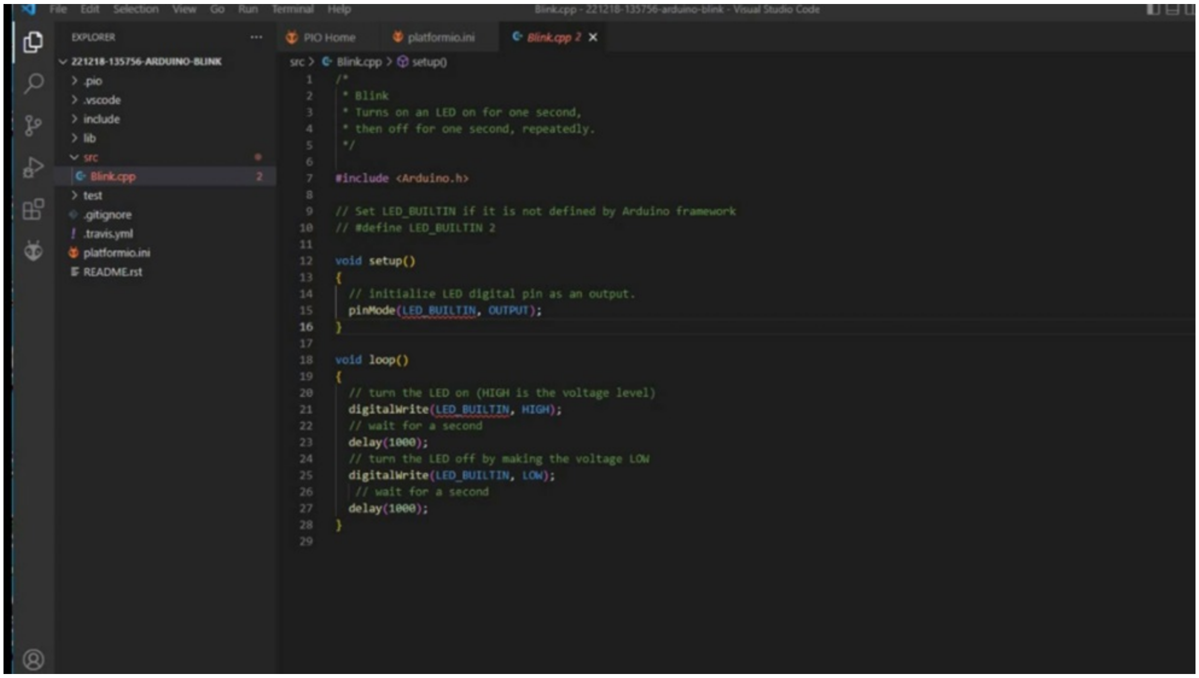
COPY ERROR MESSAGES

Ln 1, Col 1   UTF-8   DOIT ESP32 DEVKIT V1 [not connected]

You can see it Wi Fi is calm we can come compile now we have to select a boat Okay we'll do it here specifically to Jeff good vision one okay against come again compile okay it's compiled it let's upload it to ESP 32 the pseudocode report and upload and then we start monitor sudo ammonia to enter the stock USB as we can see it's we can start scammed on this my Wi Fi in it let's open a new example program in Arduino IDE and import it in platform IO here's many examples I opened something simple think timer repeat timer let's see if I'm doing okay repeat timer documents are doing the Perth repeat timer okay and close this hole this also glows and then we imported this project a lot for my oh open import Arduino project Well let's find out the both to our project users name of PC documents or do Ina and our repeat timer repeat timer no input NO NO NO NO siliq The boat My boat is do it the sun and sit this use libraries installed by Arduino ag if you said this, we can use all libraries installed in Arduino. And now input now we can see repeat time. Let's look at Platform io e file. Here we can see that the birth of the Arduino Libraries has been aged. We also need to specify the monitor settings now we can compile and upload carry on the first and then upload and monitor. First compiling, then upload and then starts monitor.

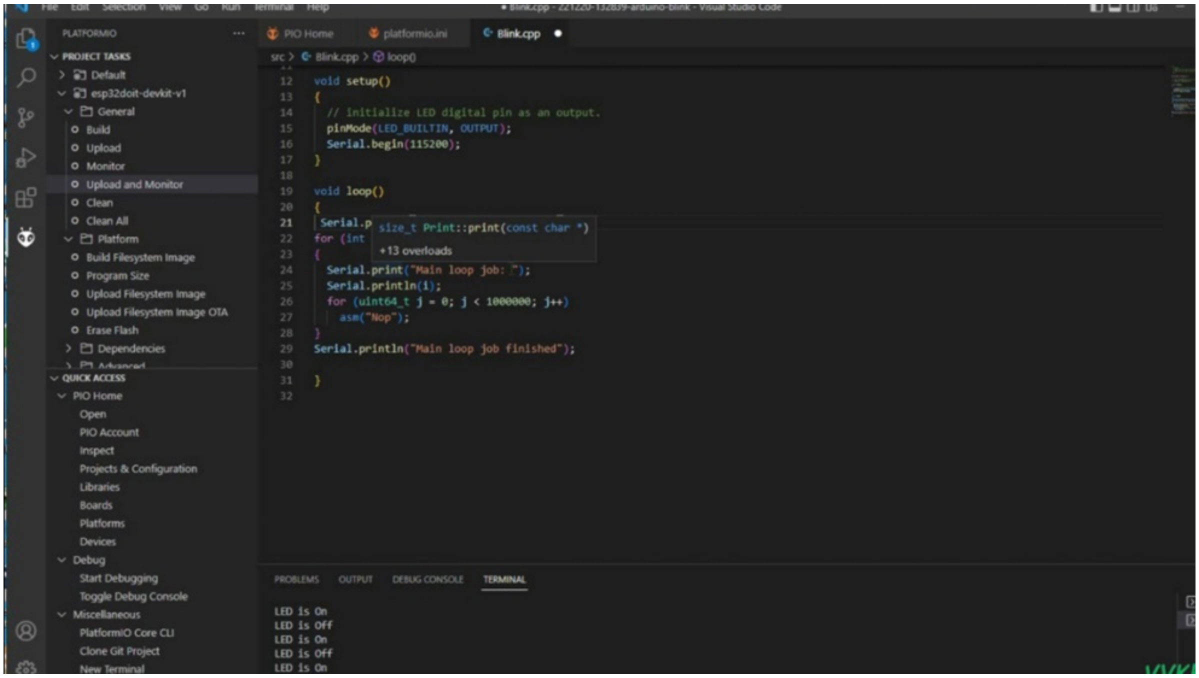
SIMPLE MULTITHREADING PROGRAM

we'll create a simple program for Arduino and FreeRTOS let's first input and are doing a blink example project that flesh synlett Then we'll modify it click platformio icon open and project examples how to do nebulin kidsfirst import let's take a look at Platform IO ini file there are four configurations in this file I need only the second one because my vote is this one that's why I deleted the others save it and switch to blink CPP this code is very simple there is the initialization of the pin in the setup function then LED turns on and off in one second on one second off and one second again.



```
/*
 * Blink
 * Turns on an LED on for one second,
 * then off for one second, repeatedly.
 */

#include <Arduino.h>

// Set LED_BUILTIN if it is not defined by Arduino framework
// #define LED_BUILTIN 2

void setup()
{
  // initialize LED digital pin as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop()
{
  // turn the LED on (HIGH is the voltage level)
  digitalWrite(LED_BUILTIN, HIGH);
  // wait for a second
  delay(1000);
  // turn the LED off by making the voltage LOW
  digitalWrite(LED_BUILTIN, LOW);
   // wait for a second
  delay(1000);
}
```

In order to see the action of the program in the monitor I will add the text messages about turning the lid on and off first we need to initialization the serial port and sort of the discord and then we can add messages here now you can see in the monitor when beauty is on or off now you can compile it clean and build okay upload and monitor at the start ESP 32 Here we can see all messages when leaders flushing Okay, the program is welcome. Now let's imagine that our program does not drink with toilet, but before some long term job. In this case, the delays between loop switching can be used as simulation of this long term operation. Let's change the code so that it simulates this job more clearly. I will replace the inflation rate with a message output and delay in this case, I will make the Julie through the for loop this one so the simulation of blockchain work is complete. Let's call it this job. main loop job. Here I output the main loop job message. This one and the iteration number.

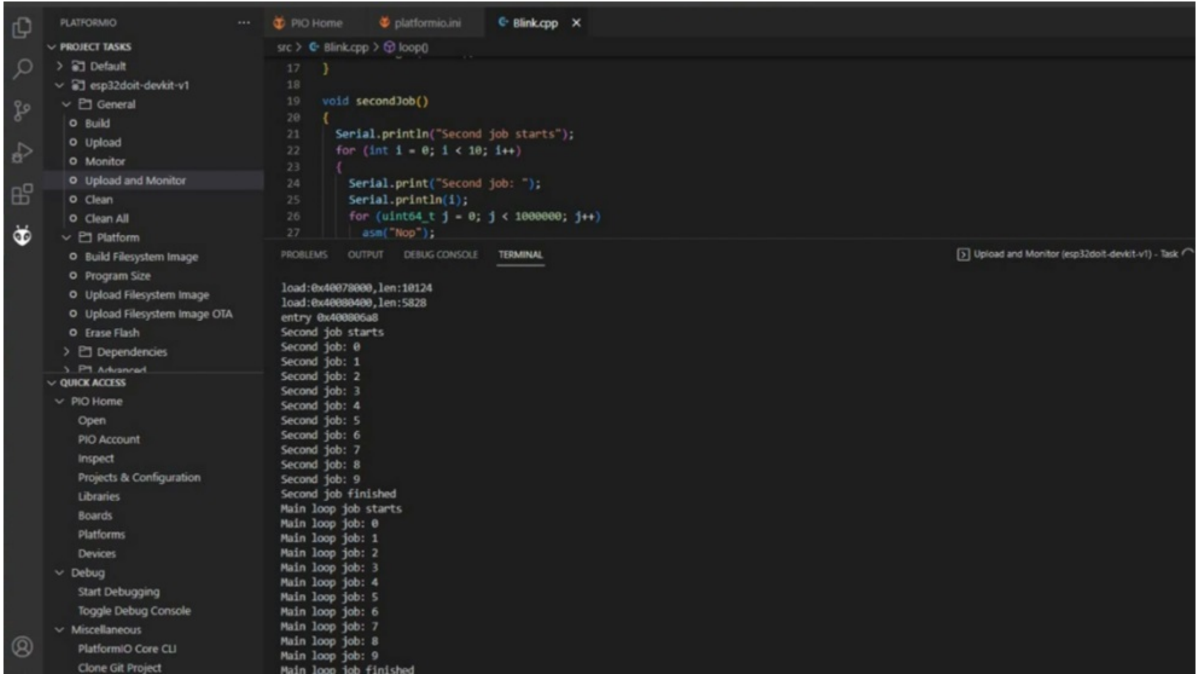


```
void setup()
{
  // initialize LED digital pin as an output.
  pinMode(LED_BUILTIN, OUTPUT);
  Serial.begin(115200);
}

void loop()
{
 Serial.p
for (int
{
  Serial.print("Main loop job: ");
  Serial.println(i);
  for (uint64_t j = 0; j < 1000000; j++)
    asm("Nop");
}
Serial.println("Main loop job finished");

}
```

```
LED is On
LED is Off
LED is On
LED is Off
LED is On
```

10 times in the loop it's let's all drop. Let's compile and test it Just start to USB yes we see this we can let's hit the

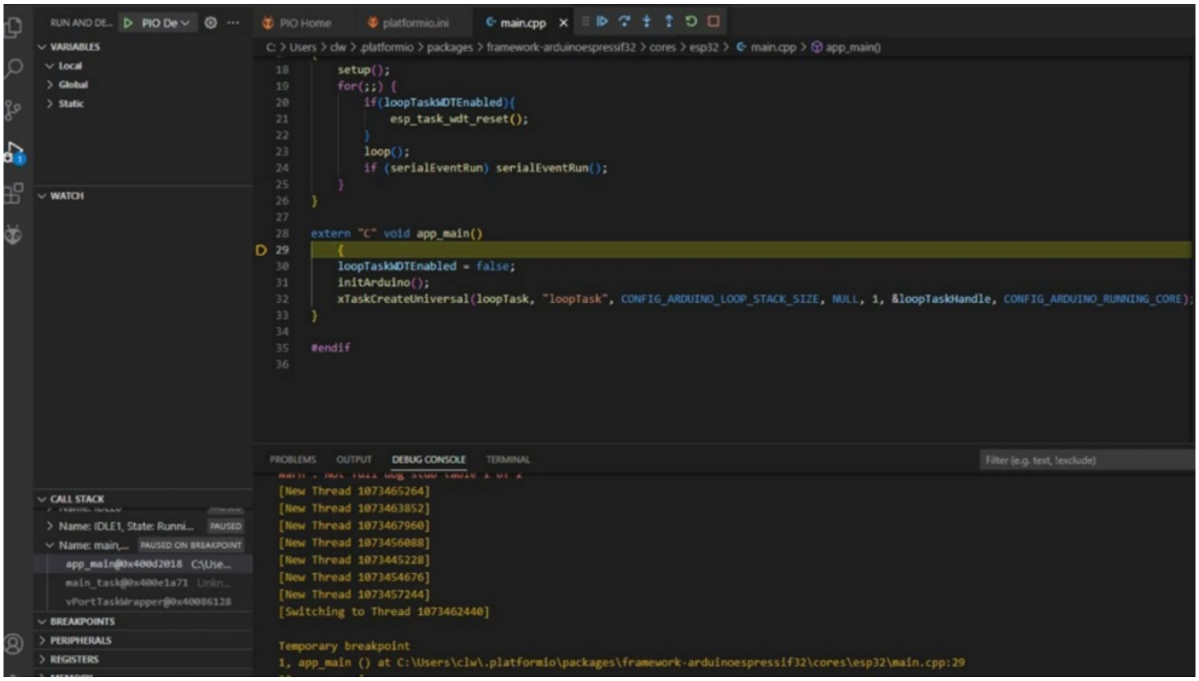
infinite loop to stop the program now let's add another similar job and design it is a function let's call it second job this same job let's ensure that before executing the main job hear upload and monitor comport is busy let's closer with it and again restart the SP as you can see in such program jobs can be only performed sequentially one after another the second shop and then main loop shop but what if you need to do them at the same time.



```
}

void secondJob()
{
  Serial.println("Second job starts");
  for (int i = 0; i < 10; i++)
  {
    Serial.print("Second job: ");
    Serial.println(i);
    for (uint64_t j = 0; j < 1000000; j++)
      asm("Nop");
```

```
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Second job starts
Second job: 0
Second job: 1
Second job: 2
Second job: 3
Second job: 4
Second job: 5
Second job: 6
Second job: 7
Second job: 8
Second job: 9
Second job finished
Main loop job starts
Main loop job: 0
Main loop job: 1
Main loop job: 2
Main loop job: 3
Main loop job: 4
Main loop job: 5
Main loop job: 6
Main loop job: 7
Main loop job: 8
Main loop job: 9
Main loop job finished
```

In this case it just must be run in its own thread. To understand how to do this, let's take a closer look at the structure of the Arduino program. Any C++ program should have a main function, but we don't see in Arduino. In fact, it's available but it's quite done for the user. We can find the main cpp file ourselves the easiest way to do this is to connect to the in circuit debugger to ESP 32 and start debugging if you don't have an in circuit debugger, this is not a problem we will not use it in the future. Now I will just connect through it to show the main function in order for Majapahit to work I have to add the debug tool parameter to the INI file my dibba here is coded Julian if you have any other you can specify its name in this parameter now I can start you by him you back in the Start okay let's now move the cursor over the header of this file and look at the birth to it I will leave this birth in the video attachment This is in exploiter.

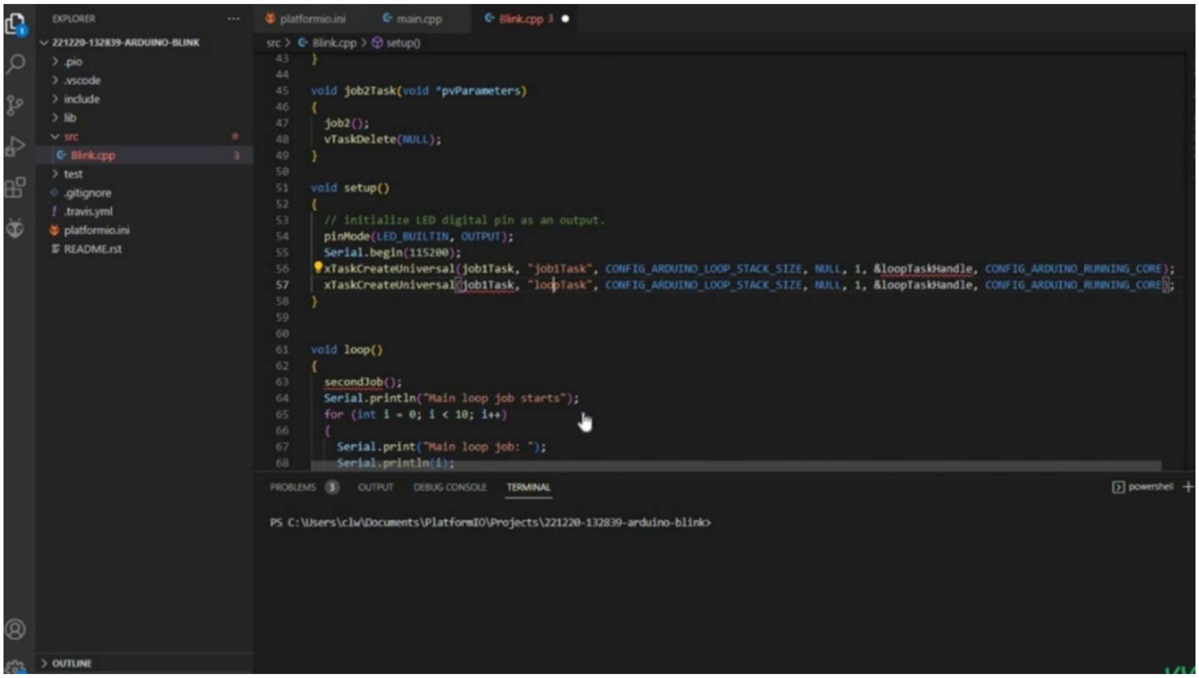


```
    setup();
    for(;;) {
        if(loopTaskWDTEnabled){
            esp_task_wdt_reset();
        }
        loop();
        if (serialEventRun) serialEventRun();
    }
}

extern "C" void app_main()
{
    loopTaskWDTEnabled = false;
    initArduino();
    xTaskCreateUniversal(loopTask, "loopTask", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 1, &loopTaskHandle, CONFIG_ARDUINO_RUNNING_CORE);
}

#endif
```

```
Name: IDLE1, State: Runni...   PAUSED
Name: main,...   PAUSED ON BREAKPOINT
  app_main@0x400d2018   C:\Use...
  main_task@0x400e1a71   Unkn...
  vPortTaskWrapper@0x40086128
```

```
[New Thread 1073465264]
[New Thread 1073463852]
[New Thread 1073467960]
[New Thread 1073456088]
[New Thread 1073445228]
[New Thread 1073454676]
[New Thread 1073457244]
[Switching to Thread 1073462440]

Temporary breakpoint
1, app_main () at C:\Users\clw\.platformio\packages\framework-arduinoespressif32\cores\esp32\main.cpp:29
```

Let's look at the code in this file. He'll system in civilization is before MIT. And then a disk is created in which they are doing a program is executed. This one creation of the disk and this disk Let's look at the this disk we see that the setup function is coded first in it and then loop is executed Infinia first, it can be seen that the Arduino program is executed is free or to task and it follows from this that we can create this task or sales when we need it. Now, I will stop debugging and we will return to the file blink cpp to create our tasks first I will copy the thread creation function this function and stop debugging I saw this line here it's set up function two times because we create two tasks and remove setup here I rename the second job to job one I simply replace it I also aged zero here to increase the delay then I create a second task of the same function and name it job two I simply copy this function we also need to create functions for the disks in which our jobs will be performed here I insert it here layer should not be the tune of these functions and if the disk is no longer needed, it must be forcibly deleted in our case inside the disk function their job function will be coded first job one and job two and then the deletion function this and this now we have to set the name of our disk functions in the x this create your new cell function which creates the disk this this job one disk copy here and paste it here here this is the name of the disk and same is here replace it to here we're going to change it to now this is handle but we don't use it now oh sorry here then I will replace the main loop code to miss such and jewelry simpler to this one my envelope job message and this is Julie and nothing more.



```
}

void job2Task(void *pvParameters)
{
  job2();
  vTaskDelete(NULL);
}

void setup()
{
  // initialize LED digital pin as an output.
  pinMode(LED_BUILTIN, OUTPUT);
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, "job1Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 1, &loopTaskHandle, CONFIG_ARDUINO_RUNNING_CORE);
  xTaskCreateUniversal(job1Task, "loopTask", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 1, &loopTaskHandle, CONFIG_ARDUINO_RUNNING_CORE);
}


void loop()
{
  secondJob();
  Serial.println("Main loop job starts");
  for (int i = 0; i < 10; i++)
  {
    Serial.print("Main loop job: ");
    Serial.println(i);
```

```
PS C:\Users\clw\Documents\PlatformIO\Projects\221220-132839-arduino-blink>
```

Also removed from the Reddit in civilization. As a result we have setup function in which the creation of two discs is the format and the main loop which also is a disc. Therefore, three discs should be form is simultaneous in this program.

EXAMPLE DUMMY CODE

In Arduino, you can achieve simple multithreading using cooperative multitasking, also known as task scheduling or task switching. This method doesn't use separate threads but instead allows different tasks to take turns executing, providing the illusion of multitasking. One common approach is to use non-blocking code and the millis() function to manage task timing.

Here's an example of a simple Arduino program that demonstrates cooperative multitasking to blink two LEDs independently:

```
// Pins for the LEDs
const int led1Pin = 2;
const int led2Pin = 3;

// Time intervals for the two tasks (in milliseconds)
const unsigned long interval1 = 1000; // 1 second
const unsigned long interval2 = 500;  // 0.5 seconds

unsigned long previousMillis1 = 0;
unsigned long previousMillis2 = 0;

bool led1State = false;
bool led2State = false;

void setup() {
  pinMode(led1Pin, OUTPUT);
  pinMode(led2Pin, OUTPUT);
}
```

```
void loop() {
  // Task 1: Blink LED 1 every 1 second
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis1 >= interval1) {
    previousMillis1 = currentMillis;
    led1State = !led1State;
    digitalWrite(led1Pin, led1State);
  }

  // Task 2: Blink LED 2 every 0.5 seconds
  if (currentMillis - previousMillis2 >= interval2) {
    previousMillis2 = currentMillis;
    led2State = !led2State;
    digitalWrite(led2Pin, led2State);
  }
}
```

In this example, we have two LEDs connected to pins 2 and 3. The loop() function continuously runs in Arduino, but we use the millis() function to manage the timing for each task independently.

The interval1 and interval2 variables define the time intervals for blinking LED 1 and LED 2, respectively. The previousMillis1 and previousMillis2 variables store the last time each LED was toggled.

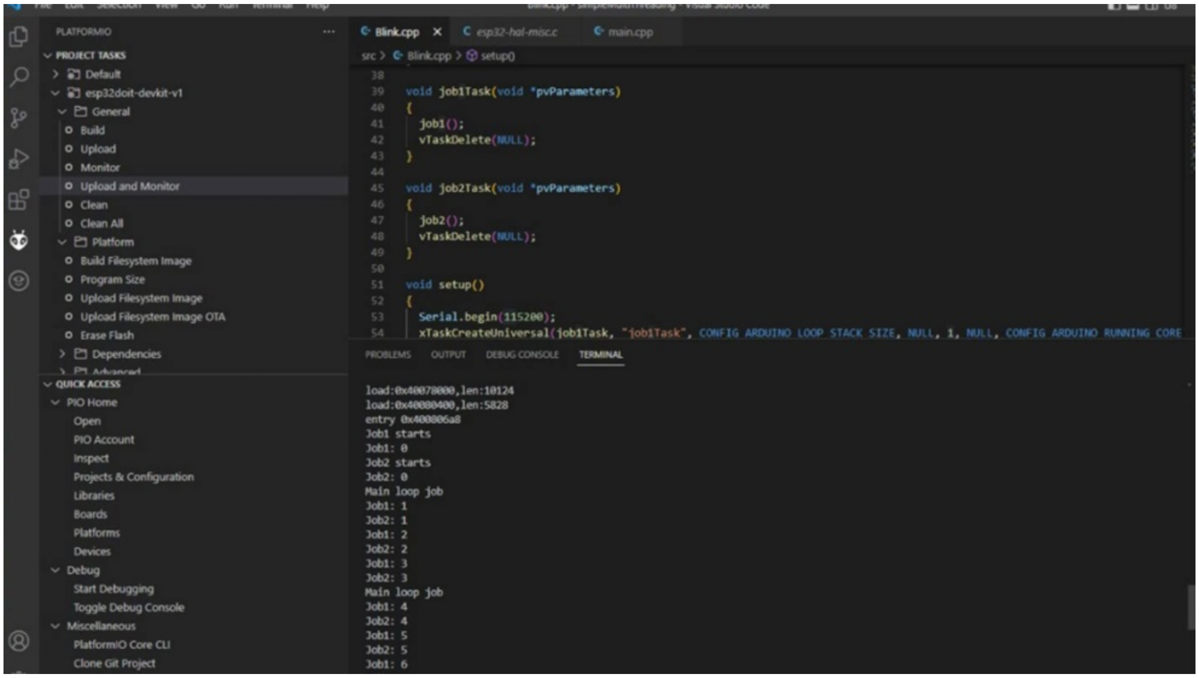
Within the loop() function, we check if enough time has passed since the last toggle for each LED. If the time interval has elapsed, we toggle the LED's state and update the previousMillis variable to the current time.

As a result, the two LEDs will blink independently, giving the impression of multitasking. Remember that this is cooperative multitasking, and it's important to write non-blocking code to allow the program to run smoothly without delays.

For more complex multithreading or true parallel execution, you would need a more powerful microcontroller that supports real-time operating systems or multiple hardware threads.

PRIORITIES OF TASKS

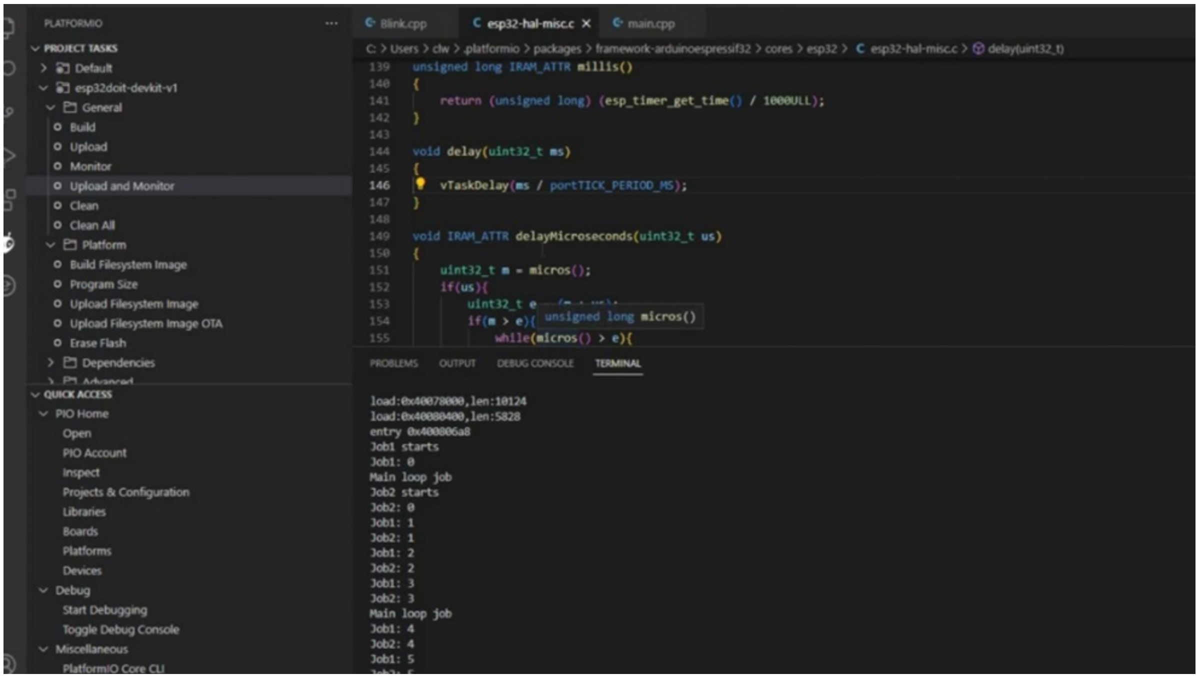
Now let's take a closer look at the tasks in this program. As mentioned in the previous project, three tasks, launch it in the program. The first step the task is doing a task. It started in the file main cpp this let's look at this parameter this is pre loaded to the disk they're loaded this video the lower the priority and FreeRTOS now let's look at the priority of the tasks you create it's the same as they are doing a desk he's also one therefore, all three running tests will be executed sequentially switching with each clock cycle or the Task Scheduler he if you look at the output of the program in the germinal we will see that all three of our tests are performed simultaneously this one job one job to job two main loop job job one job to job Huanchaco and so on.



```
void job1Task(void *pvParameters)
{
  job1();
  vTaskDelete(NULL);
}

void job2Task(void *pvParameters)
{
  job2();
  vTaskDelete(NULL);
}

void setup()
{
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, "job1Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 1, NULL, CONFIG_ARDUINO_RUNNING_CORE
```

```
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job1 starts
Job1: 0
Job2 starts
Job2: 0
Main loop job
Job1: 1
Job2: 1
Job1: 2
Job2: 2
Job1: 3
Job2: 3
Main loop job
Job1: 4
Job2: 4
Job1: 5
Job2: 5
Job1: 6
```

Now let's analyze the priority of one or the tests and see what changes I will set the the ability for job one equal to three it's one three update and monitor sorry I just connected to the USB connected to upload the monitor again just stopped USB as we can see, at first only job one is executed and then update finished their job two and are doing all the tests begin to yoke. Thus it jumps out that test COVID Correct priority completely displace tasks with a low priority. Let's in this case, tasks with a low priority low priority will not work until the task with a high priority finishes working. But what if this with low priority should also work when desks with a high priority of you can in this case they work off desk with high priority can sometimes be bootcut. To do this, there are various functions in FreeRTOS the simplest of which is a delay for the time specified in the milliseconds let's add 500 milliseconds delay to the desk oh the first job you can use are doing a delay function here upload and monitor just start to use P.



```
unsigned long IRAM_ATTR millis()
{
    return (unsigned long) (esp_timer_get_time() / 1000ULL);
}

void delay(uint32_t ms)
{
    vTaskDelay(ms / portTICK_PERIOD_MS);
}

void IRAM_ATTR delayMicroseconds(uint32_t us)
{
    uint32_t m = micros();
    if(us){
        uint32_t e = (m + us);
        if(m > e){
            while(micros() > e){
```

```
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job1 starts
Job1: 0
Main loop job
Job2 starts
Job2: 0
Job1: 1
Job2: 1
Job1: 2
Job2: 2
Job1: 3
Job2: 3
Main loop job
Job1: 4
Job2: 4
Job1: 5
```

As we can see all three tests have started working despite The fact that one of them requests a high priority first we look at the job one by Julie Logan the bulk of other tasks as I said we used to the usual of doing the delete function let's see what it goes inside to do this press the left CTRL and click on it and here we see the FreeRTOS delay function which performs the delay not through the for loop like this this delay for loop but by blocking the desk for a given number of ticks this block of hellos tasks with a lower priority to yoke now. let's replace the delays on the four loops with a delay function let it be one second delay and here let it be here three seconds uploaded monitor. I just start to use p as we can see all three tasks are executed simultaneously despite the fact that one of them has a higher priority.

EXAMPLE DUMMY CODE

In Arduino, true multitasking with priorities requires the use of a real-time operating system (RTOS) or a multitasking library. One popular Arduino RTOS library is "FreeRTOS." Unfortunately, the library is not natively supported in the Arduino IDE, but you can install it as an additional library.

Here's an example of using FreeRTOS to create tasks with priorities in Arduino:

Install the FreeRTOS library:

Open the Arduino IDE.
Go to "Sketch" -> "Include Library" -> "Manage Libraries."
Search for "FreeRTOS" and click "Install" for the "FreeRTOS by Richard Barry" library.
Write the code for creating tasks with priorities:

```
#include <Arduino_FreeRTOS.h>
#include <task.h>

// Define task priorities (higher numbers have higher
priority)
#define PRIORITY_HIGH  3
#define PRIORITY_MED   2
#define PRIORITY_LOW   1

// Task handles
TaskHandle_t highTaskHandle, mediumTaskHandle,
lowTaskHandle;

// Task functions
void taskHighPriority(void *pvParameters);
void taskMediumPriority(void *pvParameters);
void taskLowPriority(void *pvParameters);

void setup() {
  // Create tasks with different priorities
  xTaskCreate(taskHighPriority, "High", 1000, NULL,
PRIORITY_HIGH, &highTaskHandle);
  xTaskCreate(taskMediumPriority, "Medium", 1000, NULL,
PRIORITY_MED, &mediumTaskHandle);
  xTaskCreate(taskLowPriority, "Low", 1000, NULL,
PRIORITY_LOW, &lowTaskHandle);
}

void loop() {
  // Nothing to do here in this example.
}

void taskHighPriority(void *pvParameters) {
  for (;;) {
    Serial.println("High Priority Task is running!");
    vTaskDelay(1000); // Wait for 1 second
  }
}

void taskMediumPriority(void *pvParameters) {
  for (;;) {
    Serial.println("Medium Priority Task is running!");
    vTaskDelay(2000); // Wait for 2 seconds
  }
}

void taskLowPriority(void *pvParameters) {
  for (;;) {
    Serial.println("Low Priority Task is running!");
    vTaskDelay(3000); // Wait for 3 seconds
  }
}
```

In this example, we create three tasks: taskHighPriority, taskMediumPriority, and taskLowPriority. Each task has a different priority level. The higher the priority value, the higher the task's priority.

The xTaskCreate() function is used to create tasks. It takes several arguments, including the task function, task name, stack size, parameters (not used in this case), task priority, and a handle for the created task.

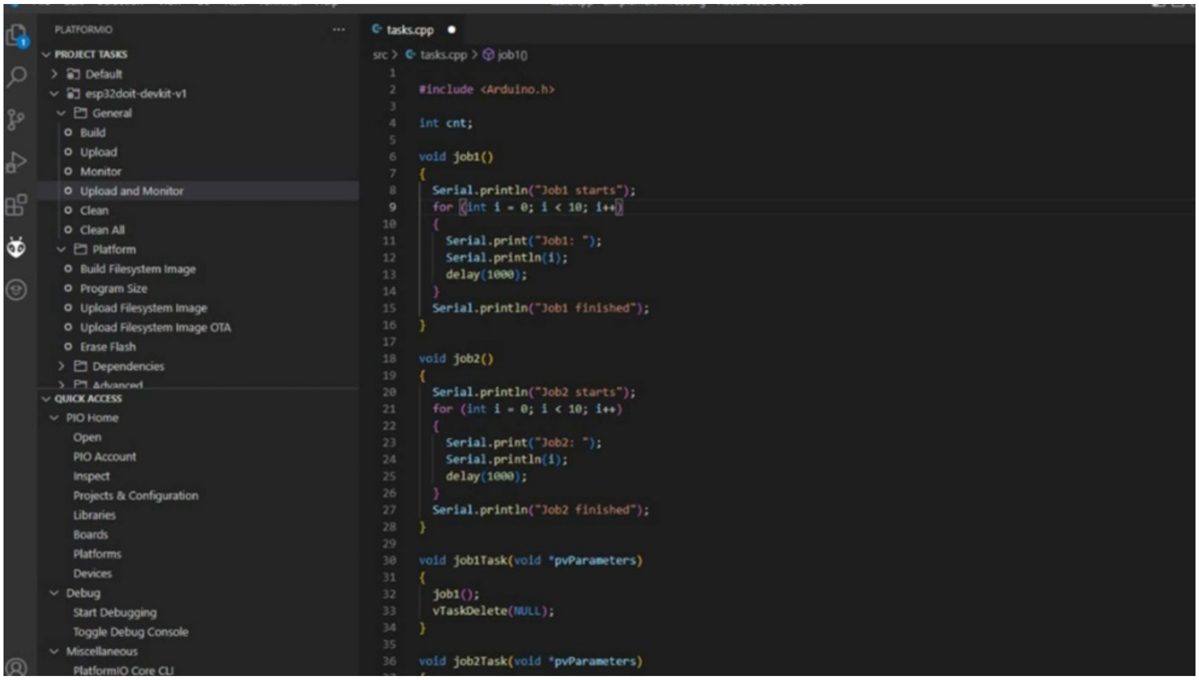
The vTaskDelay() function is used inside each task to introduce a delay. The delay is specified in ticks, and the tick rate is determined by the FreeRTOS scheduler, which is typically set to 1ms in the Arduino environment.

Remember that to observe the output of tasks, you need to have a working Serial connection and monitor the Serial output in the Arduino IDE or a serial terminal.

Please note that multitasking and task priorities require careful consideration and understanding of how the tasks interact with each other. In real-world applications, you must handle shared resources, synchronization, and potential race conditions carefully to ensure the correct behavior of the system.

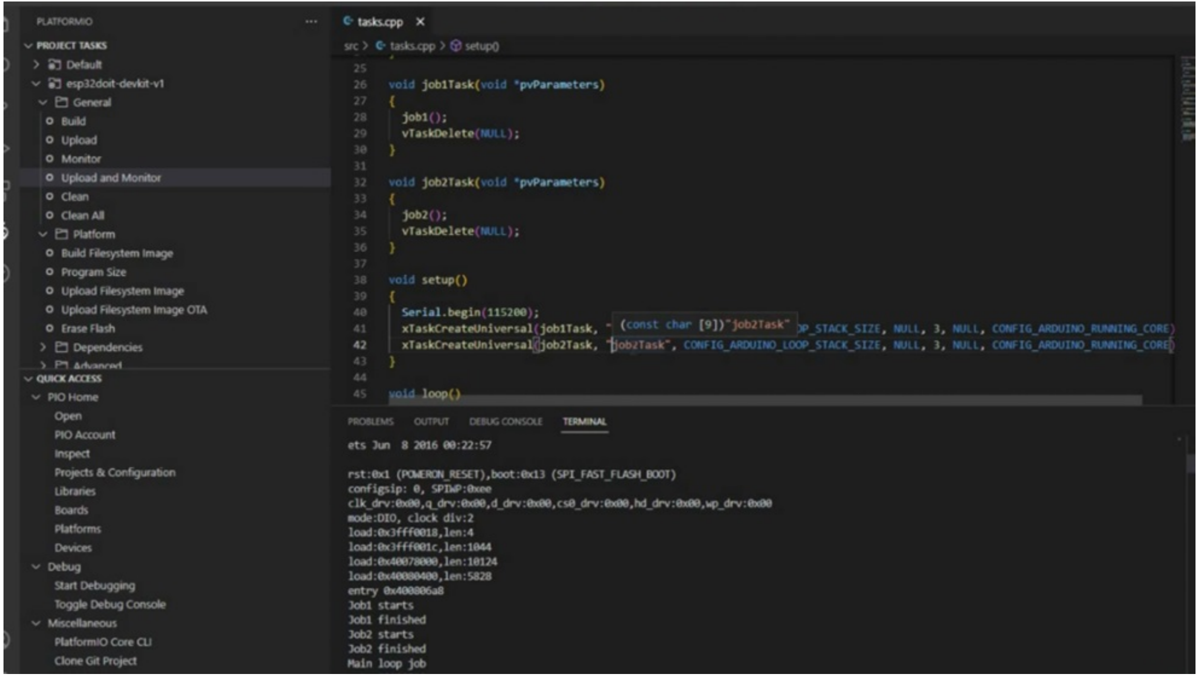
MUTEXES

Let's change shall program now let the discs in it increment the same variable but first rename file blink CPP because we no longer use the idiot let it be coded task CPP didn't name this CPP also we don't need this and who don't need to do this let's create one variable of type int and call it Cynthia that means counter this one and let our program increment this counter 10 million times in each disk deal medians and here 10 minions here seem cheap plus plus let's give this disks the same priority equal to three at the same time there are doing a disk will have priority equal to one they feel our tasks will work first and then they are doing a loop will start also they are doing a loop let's output the resulting value over the counter Cinti to the monitor then, stop the program compile the program and see how it works I just start the spin as we can see both discs did not start at the same time.



```

#include <Arduino.h>

int cnt;

void job1()
{
  Serial.println("Job1 starts");
  for (int i = 0; i < 10; i++)
  {
    Serial.print("Job1: ");
    Serial.println(i);
    delay(1000);
  }
  Serial.println("Job1 finished");
}

void job2()
{
  Serial.println("Job2 starts");
  for (int i = 0; i < 10; i++)
  {
    Serial.print("Job2: ");
    Serial.println(i);
    delay(1000);
  }
  Serial.println("Job2 finished");
}

void job1Task(void *pvParameters)
{
  job1();
  vTaskDelete(NULL);
}

void job2Task(void *pvParameters)
```

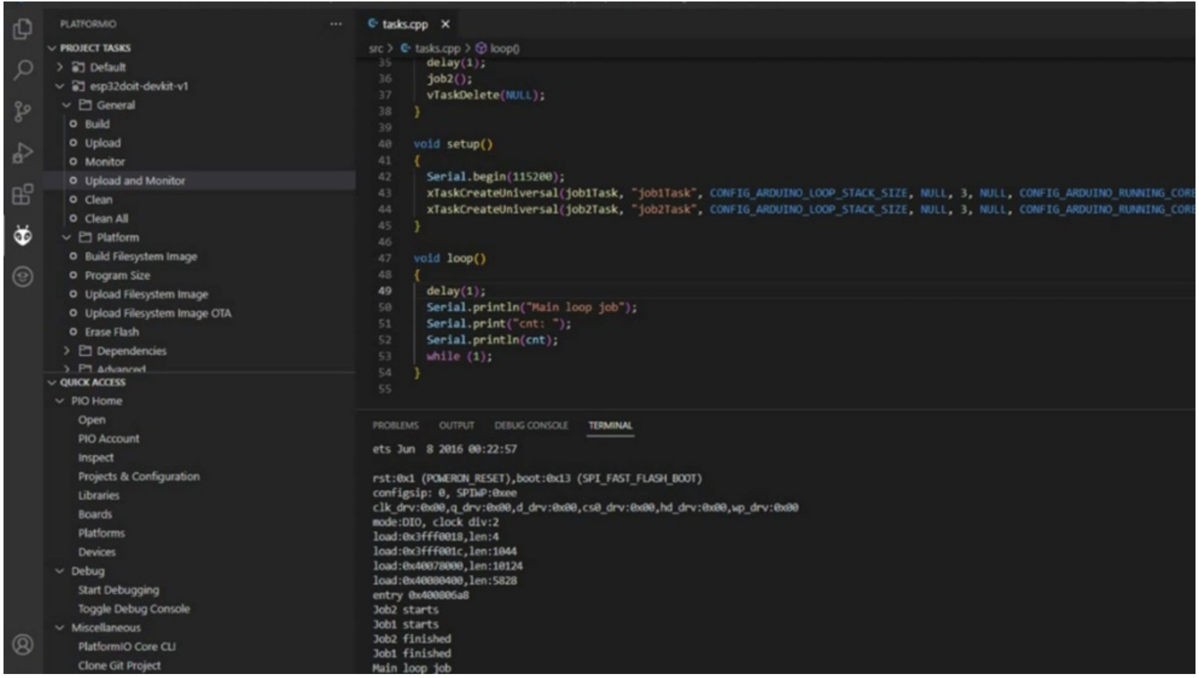
But the result of the incremental the counter is correct 20 million this one why did not just start simultaneously. Let's look at the sequence of lunching this first they are doing a setup disk starts with a low priority of one then it creates a job one disk with a factor priority of three this one this disk displays there are doing a task.



```

void job1Task(void *pvParameters)
{
  job1();
  vTaskDelete(NULL);
}

void job2Task(void *pvParameters)
{
  job2();
  vTaskDelete(NULL);
}

void setup()
{
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, " (const char [9])"job2Task" P_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_CORE)
  xTaskCreateUniversal(job2Task, "job2Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_CORE)
}

void loop()
```

```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job1 starts
Job1 finished
Job2 starts
Job2 finished
Main loop job
```

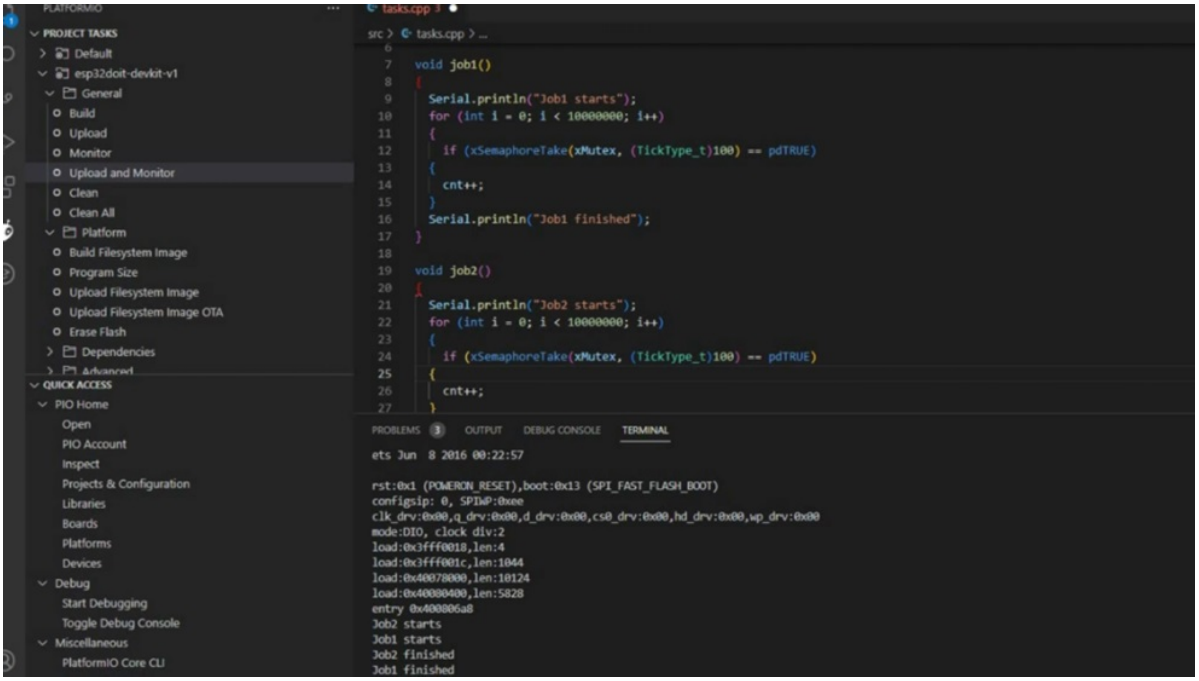
In this case, the job to disk will not be created until the job one task finishes broken. However in order for the second task to start the first one needs to be book it for a short period of time by inserting a delay of one millisecond it's beginning it's insert Julie here and in order for the tasks to start at the same time, the same delay must be inserted into the second task inserted here also in order for the compiler, not to optimize the counter. We need to put it with the volatile keyword. It's here Okay, now we can compile, upload and monitor Is 30 SP now, we see that the discs are running simultaneous starts both and finish it both but in doing a disk is started before them, which outputs the counter value to the monitor before it has changed it here therefore, we also need to insert a delay in the Arduino loop here upload and monitor is 30 SP now, our program works correctly, both discs start at the same time, then the result is displayed in the monitor on the result itself does not match the expected 20 million.



```
  delay(1);
  job2();
  vTaskDelete(NULL);
}

void setup()
{
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, "job1Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_CORE
  xTaskCreateUniversal(job2Task, "job2Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_CORE
}

void loop()
{
  delay(1);
  Serial.println("Main loop job");
  Serial.print("cnt: ");
  Serial.println(cnt);
  while (1);
}
```

```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job2 starts
Job1 starts
Job2 finished
Job1 finished
Main loop job
```

Why did this happen? This happens because both discs access the same cell in the memory at the same time they can interrupt each other in such a way that one task starts the operation but does not have time to finish because another task starts working with this variable this is the result an incorrect value is the right into the variable. To solve this problem, this need to or harness exists to this memory cell in tune. That is when one task is working with this variable, the second one should go into standby mode so that it does not conflict with the first one mutexes are used for this purpose. Let's create a mutex and discuss how it works. To define a music scene FreeRTOS the symbol for the candle D type is used let's call Oh musics X mute mute x this one mute x now you have to create it. This is done by the XML file to create musics function it should be done in the setup function in Arduino setup this creation mutex and now we can use it in our tasks. Now, when others could need to access a shared resource, in our case a memory cell it must first take the mutex and if the mutex is taken by another disk, then the current disk will be blocked until the mutex is released.



```
void job1()
{
  Serial.println("Job1 starts");
  for (int i = 0; i < 10000000; i++)
  {
    if (xSemaphoreTake(xMutex, (TickType_t)100) == pdTRUE)
  {
    cnt++;
  }
  Serial.println("Job1 finished");
}

void job2()
{
  Serial.println("Job2 starts");
  for (int i = 0; i < 10000000; i++)
  {
    if (xSemaphoreTake(xMutex, (TickType_t)100) == pdTRUE)
  {
    cnt++;
  }
```

```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job2 starts
Job1 starts
Job2 finished
Job1 finished
```

At the end of you can create the resource the disk must release the mutex taking care mutex is performed by the excema fototec which can function in Fritos to do this I insert this line here here and the hear this function we take a mutex here we also need to give it away the semaphore the key function is use it for this I inserted here and here let's look at the function of taking mutex. Two parameters are posted to it. This is the mutex itself. Enter the timeout in milliseconds. If the mutex is taken by another task, the task will be blocked at the until the mutex is released or for the time basically to eat in this parameter. Therefore in the case of successful mutex taken the function returns be true and in the case of a mutex exit in the case of timeout exit PD falls now compile upload and monitor used to start and it's working for a long time because we have mute mutex let's wait the program finishes its work with the correct content willya this 20 minutes only it will get for almost two minutes as we can see using commute axelos to share access to a resource but at the same time slows down the program.

EXAMPLE DUMMY CODE

In Arduino using FreeRTOS, Mutexes are used for mutual exclusion to prevent multiple tasks from accessing shared resources simultaneously. A Mutex acts as a lock that a task can acquire before accessing a shared resource and release once it's done. This ensures that only one task can access the shared resource at any given time.

Here's an example Arduino code demonstrating how to use Mutexes with two tasks accessing a shared variable:

```
#include <Arduino_FreeRTOS.h>
#include <task.h>

// Task handles
TaskHandle_t task1Handle, task2Handle;

// Shared variable
int sharedValue = 0;

// Mutex handle
SemaphoreHandle_t mutex;

// Task functions
void task1(void *pvParameters);
void task2(void *pvParameters);

void setup() {
  Serial.begin(9600);

  // Create a Mutex
  mutex = xSemaphoreCreateMutex();
  // Create tasks
  xTaskCreate(task1, "Task1", 1000, NULL, 1,
&task1Handle);
  xTaskCreate(task2, "Task2", 1000, NULL, 1,
&task2Handle);
}

void loop() {
  // Nothing to do here in this example.
}

void task1(void *pvParameters) {
  for (;;) {
    // Acquire the Mutex before accessing the shared
resource
    if (xSemaphoreTake(mutex, portMAX_DELAY) ==
pdTRUE) {
      // Critical section: Access the shared variable
      sharedValue++;
      Serial.print("Task1 - Shared Value: ");
      Serial.println(sharedValue);

      // Release the Mutex after accessing the shared
resource
      xSemaphoreGive(mutex);
    }

    vTaskDelay(1000); // Wait for 1 second
  }
}

void task2(void *pvParameters) {
  for (;;) {
    // Acquire the Mutex before accessing the shared
resource
```

```
    if (xSemaphoreTake(mutex, portMAX_DELAY) ==
pdTRUE) {
      // Critical section: Access the shared variable
      sharedValue--;
      Serial.print("Task2 - Shared Value: ");
      Serial.println(sharedValue);

      // Release the Mutex after accessing the shared
resource
      xSemaphoreGive(mutex);
    }

    vTaskDelay(2000); // Wait for 2 seconds
  }
}
```

In this example, we have two tasks, task1 and task2, both accessing a shared variable sharedValue. A Mutex named mutex is created using xSemaphoreCreateMutex().

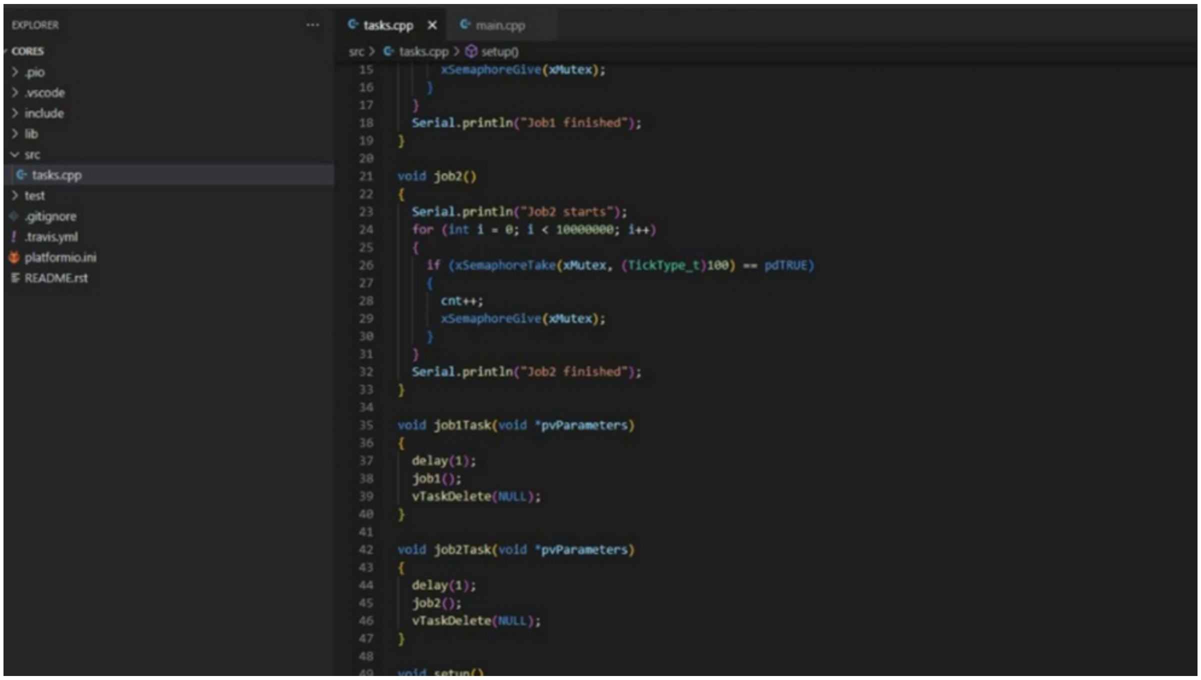
Inside each task, we use xSemaphoreTake() to acquire the Mutex before accessing the shared variable and xSemaphoreGive() to release it after accessing the variable.

The vTaskDelay() functions introduce delays to simulate task activity. Task1 increments the sharedValue by 1 every second, while Task2 decrements it by 1 every two seconds. The Serial output shows the changes to the sharedValue to demonstrate that access to the shared variable is mutually exclusive between the two tasks.

Remember that using Mutexes is essential to prevent race conditions when multiple tasks access shared resources. Be cautious about how you use Mutexes and ensure that critical sections (the parts of the code where shared resources are accessed) are as short as possible to minimize the impact on task execution.

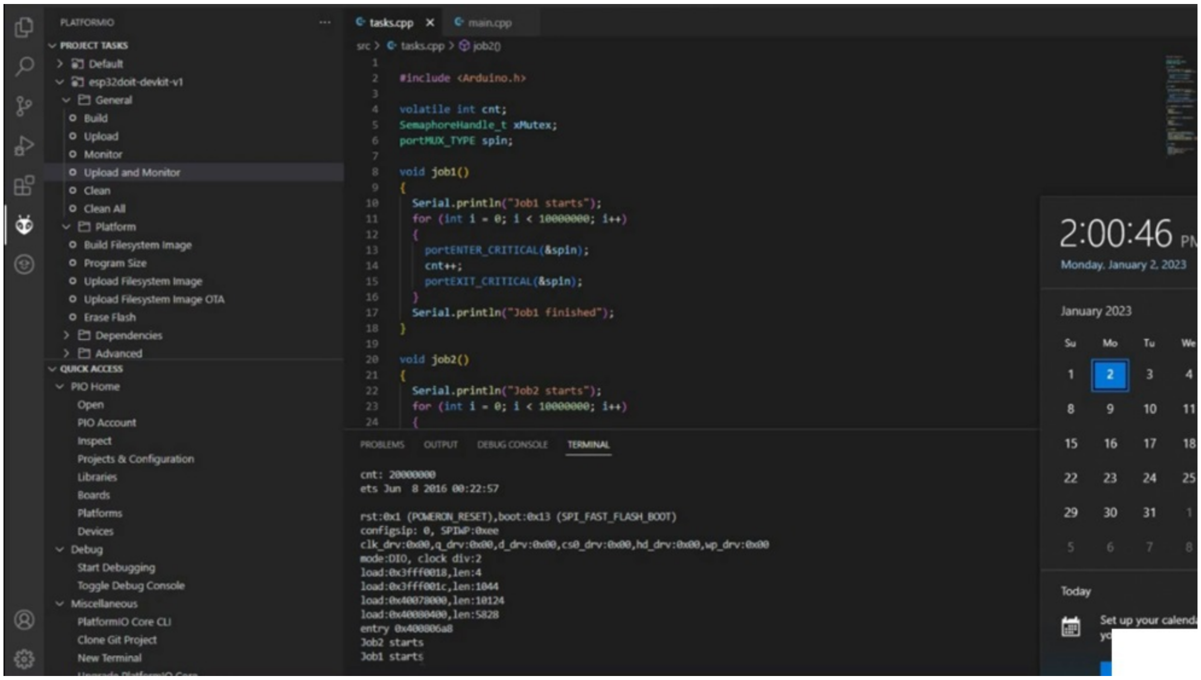
SPINLOCK, CRITICAL SECTION, MULTICORE

In the previous discuss it synchronization of disks using mutexes slows down the program when they call it frequently. This is because the process of blocking and unblocking the thread is quite, quite expensive. And if to do it often, the performance of the program will decrease. Let's now look at another synchronization method based on the critical sections and spinlocks. A critical section is a section of the SCADA for which other tasks are stupid and most hardware interrupts or disable it. I will talk about interrupts in the following projects on processor with one quarter, it's enough to disable interrupts and the task scheduler to enter the critical section, but in ESP 32 interrupts can be disabled from one quarter only on the same core. And today we will remain enable it on the other they have a spin lock is added to the entrance to the critical section spinlock is similar in functionality to a mutex it also needs to be taken and given the only difference is that when the mutex could not be taken, the task is block it and if it was not possible to take this spinlock then the blocking does not occur and the program continues to check whether it has been released.



```
        xSemaphoreGive(xMutex);
      }
    }
  Serial.println("Job1 finished");
}

void job2()
{
  Serial.println("Job2 starts");
  for (int i = 0; i < 10000000; i++)
  {
    if (xSemaphoreTake(xMutex, (TickType_t)100) == pdTRUE)
    {
      cnt++;
      xSemaphoreGive(xMutex);
    }
  }
  Serial.println("Job2 finished");
}

void job1Task(void *pvParameters)
{
  delay(1);
  job1();
  vTaskDelete(NULL);
}

void job2Task(void *pvParameters)
{
  delay(1);
  job2();
  vTaskDelete(NULL);
}

void setup()
```

Therefore, for short sections of coder with frequent access to a shared resource is preferable to use spinlock let's create let's create a spin look and replace the mutex with a critical section the port mutex type is used for the spin book I will call it to simply spin then we have to create spinlock this creation spinlock function and then we replace mutex by critical section remove this and here enter critical I'm here to critical and same for Jabo to let's compile it stop these waiting for job finishing it's finished. As you can see the program performance has increased significantly.



```cpp

#include <Arduino.h>

volatile int cnt;
SemaphoreHandle_t xMutex;
portMUX_TYPE spin;

void job1()
{
  Serial.println("Job1 starts");
  for (int i = 0; i < 10000000; i++)
  {
    portENTER_CRITICAL(&spin);
    cnt++;
    portEXIT_CRITICAL(&spin);
  }
  Serial.println("Job1 finished");
}

void job2()
{
  Serial.println("Job2 starts");
  for (int i = 0; i < 10000000; i++)
  {
```

```
cnt: 20000000
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job2 starts
Job1 starts
```

Let's approximately measure the time of its execution the time can be detected by the windows clock I just showed the ESP 32 in 40 seconds it's research it let's look at when it finished the program completed its work within 20 seconds, which is much faster than the previous version. Now let's run the one all the tasks on another quarter. Let's see which quarter they are When is running home to do this press Control and click on the last parameter in the test creation function we can see that we are doing rounds on Korea number one and we will launch the job to on Korea zero another quarter upload and monitor at the start the SP as we can see the program is already researching this happened because both of our tasks have a high priority then the system task that should resort the watchdog timer they've who our task should sometimes give time for tasks with low priority Therefore let's call delay for one millisecond 10 times in each of our task let's modify our code and insert delay for one millisecond 10 times first I'll make you 1 million instead of 10 Millions and here and I'll read one for cycle and hear delay for one millisecond and same for job two and here should be 10 Not one not 1 million this one and here now we can compile and monitor jump starting now we see that USB 32 is no longer rebooting. But the result is appeared in the monitor before the end of our tasks. So let's add a delay of 15 seconds before the output of the result here 15 seconds upload and monitor again it's working let's measure the time of its execution. I will start in 20 seconds at a start up and we are waiting for our jobs finishing. Okay, it was 14 seconds. Thus our program runs even faster on to course this because no time is wasted switching between tasks.

EXAMPLE DUMMY CODE

In Arduino, since most Arduino boards have single-core processors, there is no native support for multicore programming. However, I can demonstrate how to use spinlocks and create critical sections using interrupts to protect shared resources in a multithreading-like environment.

Please note that this approach doesn't involve true multicore execution, but it uses interrupts and spinlocks to achieve similar functionality to protect shared resources in a single-core environment.

Here's an example code for using spinlocks and critical sections to protect a shared variable between two "tasks" (interrupts) on an Arduino board:

```
// Shared variable
volatile int sharedValue = 0;

// Spinlock flag
volatile bool spinlock = false;

// Interrupt Service Routine (ISR) for "Task 1"
void task1ISR() {
  // Acquire the spinlock (busy-waiting)
  while (spinlock)
    ;

  // Enter critical section
  spinlock = true;
  sharedValue++;
  spinlock = false;
  // Exit critical section
}

// Interrupt Service Routine (ISR) for "Task 2"
void task2ISR() {
  // Acquire the spinlock (busy-waiting)
  while (spinlock)
    ;

  // Enter critical section
  spinlock = true;
  sharedValue--;
  spinlock = false;
  // Exit critical section
}

void setup() {
  // Attach the interrupt ISRs to specific pins
  attachInterrupt(digitalPinToInterrupt(2), task1ISR,
CHANGE);
  attachInterrupt(digitalPinToInterrupt(3), task2ISR,
CHANGE);

  Serial.begin(9600);
}

void loop() {
  // Display the shared value
  Serial.print("Shared Value: ");
  Serial.println(sharedValue);
  delay(1000); // Delay to reduce Serial output rate
}
```

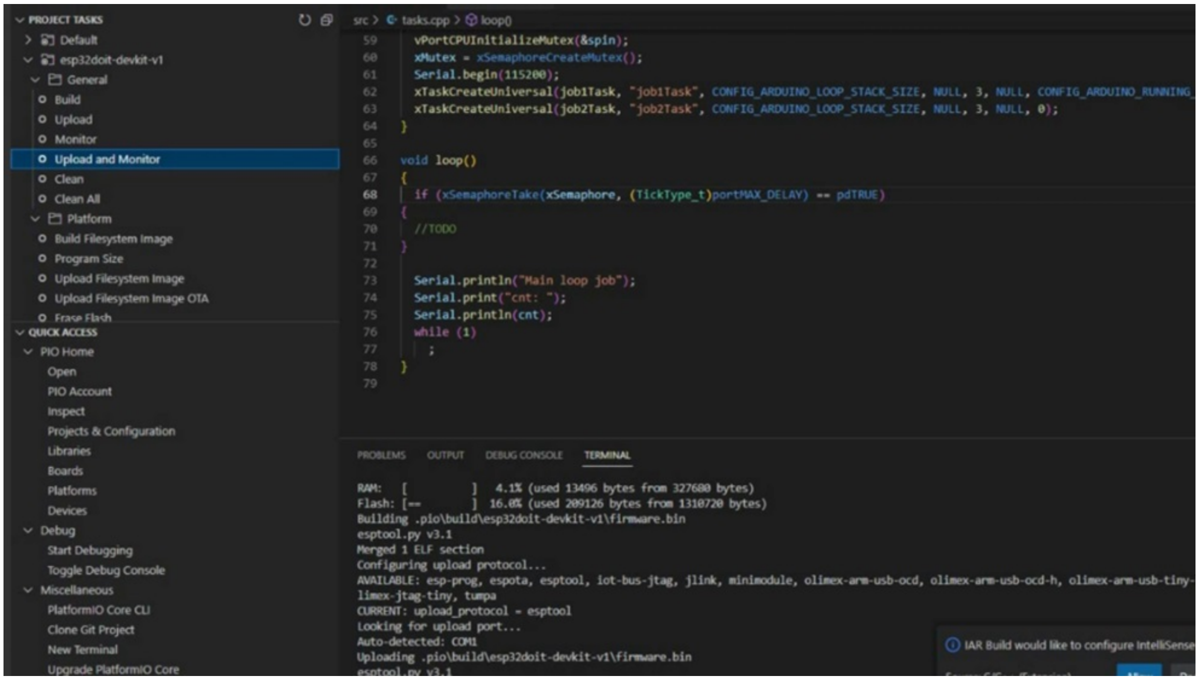
In this example, we create two "tasks" (interrupts) named task1ISR and task2ISR. These "tasks" are executed whenever there is a change on pins 2 and 3, respectively.

To protect the shared variable sharedValue, we use a simple spinlock mechanism. The spinlock is a boolean variable that acts as a flag. When a task enters a critical section, it sets spinlock to true, indicating that it is in a critical section. If the other task tries to enter a critical section while the spinlock is true, it waits (busy-waiting) until the spinlock becomes false. Once the first task exits the critical section by setting spinlock back to false, the second task can proceed with its critical section.

Again, note that this is not true multicore execution, but it demonstrates a simple way to protect shared resources in a single-core environment using interrupts and spinlocks. For true multicore systems, you would need a microcontroller or processor with multiple cores and a proper multicore programming environment.

SEMAPHORES AND QUEUES

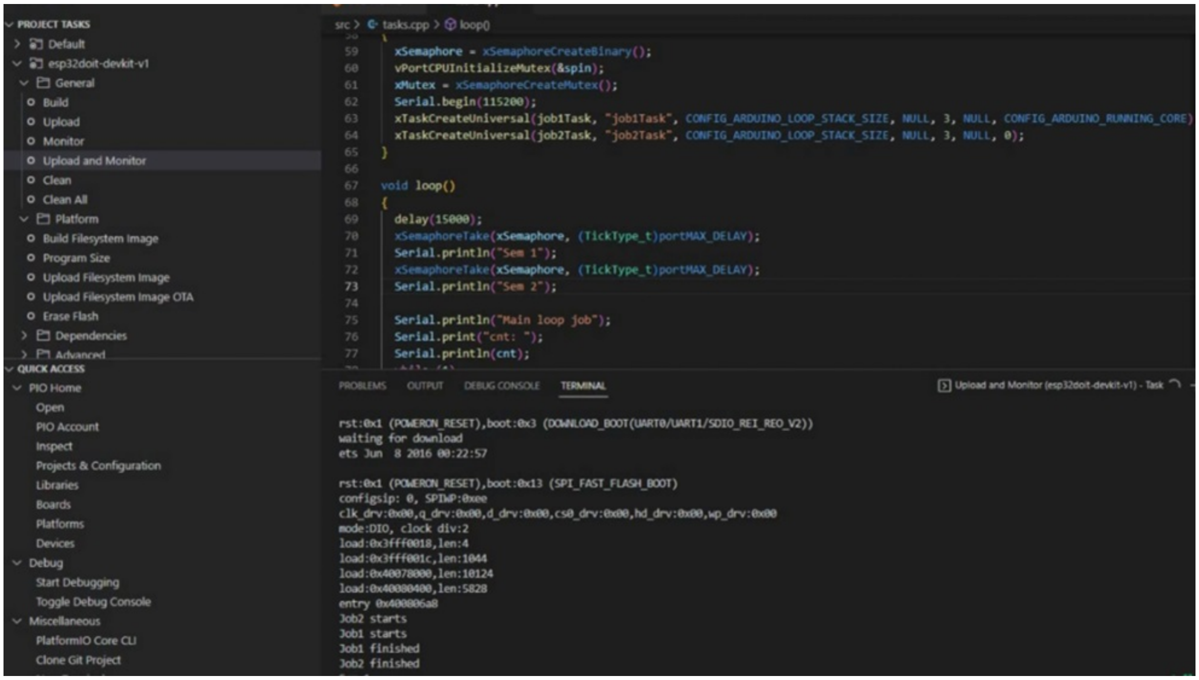
We use the delay of 15 seconds to output the counter video after the end of our tasks, but is there a way to know exactly when our desks will finish broken? Yes, for this we can use a semaphore a semaphore is similar to a mutex but unlike a mutex semaphore does not need to be given away therefore, with the help of semaphore one task can signal another about the occurrence of an event let's look at how it works let's it the simple photo program is defined by the symbol for candle type this one semaphore then we can create it in setup function now, we can use it let's send it at the end of the job one task here Next, we will wait for the semaphore to appear instead of delay or 15 seconds here this timeout in milliseconds if a semaphore is accepted the function that returns be true if the timeout has expired then be false if the timeout is not needed, then it can be replaced it with the maximum possible video of both mugs Julie this one now we can compile and test sorry USB is disconnected to start again start this way to finish his job as we can see a variant of this one is simple for the walls to receive it at the same time this tool has not finished its work yet.



```
  vPortCPUInitializeMutex(&spin);
  xMutex = xSemaphoreCreateMutex();
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, "job1Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_
  xTaskCreateUniversal(job2Task, "job2Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, 0);
}

void loop()
{
  if (xSemaphoreTake(xSemaphore, (TickType_t)portMAX_DELAY) == pdTRUE)
{
  //TODO
}

  Serial.println("Main loop job");
  Serial.print("cnt: ");
  Serial.println(cnt);
  while (1)
    ;
}
```

```
RAM:   [          ]   4.1% (used 13496 bytes from 327680 bytes)
Flash: [==        ]  16.0% (used 209126 bytes from 1310720 bytes)
Building .pio\build\esp32doit-devkit-v1\firmware.bin
esptool.py v3.1
Merged 1 ELF section
Configuring upload protocol...
AVAILABLE: esp-prog, espota, esptool, iot-bus-jtag, jlink, minimodule, olimex-arm-usb-ocd, olimex-arm-usb-ocd-h, olimex-arm-usb-tiny-
limex-jtag-tiny, tumpa
CURRENT: upload_protocol = esptool
Looking for upload port...
Auto-detected: COM1
Uploading .pio\build\esp32doit-devkit-v1\firmware.bin
esptool.py v3.1
```

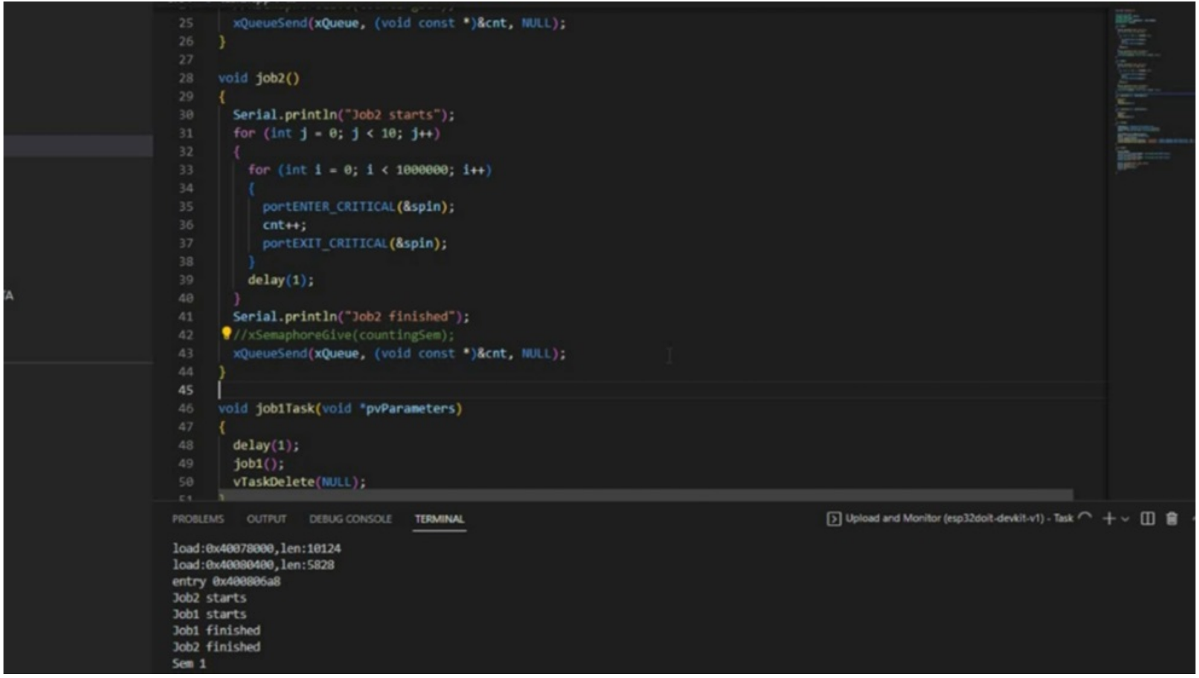
So, the counter value is not yet equal to 20 million let's it semaphore sent in to the second task we can simply copy it from here also we have to accept this symbol for in our case there is no need to check the video returned by this function. Therefore this record can be simplified. I will copy and paste this line here we will receive both similar photos one by one. You have most of the order in In which they arrive let's upload and monitor jump started as we can see, the counter value is displayed correctly. Thus, the symbol photocopy does to synchronize the end or the bulk of both tasks running on different cores with the Arduino task. Now, let's consider the case when our tasks sent both semaphores while they are doing a task has not yet received tenure.



```
  xSemaphore = xSemaphoreCreateBinary();
  vPortCPUInitializeMutex(&spin);
  xMutex = xSemaphoreCreateMutex();
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, "job1Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_CORE)
  xTaskCreateUniversal(job2Task, "job2Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, 0);
}

void loop()
{
  delay(15000);
  xSemaphoreTake(xSemaphore, (TickType_t)portMAX_DELAY);
  Serial.println("Sem 1");
  xSemaphoreTake(xSemaphore, (TickType_t)portMAX_DELAY);
  Serial.println("Sem 2");

  Serial.println("Main loop job");
  Serial.print("cnt: ");
  Serial.println(cnt);
```

```
rst:0x1 (POWERON_RESET),boot:0x3 (DOWNLOAD_BOOT(UART0/UART1/SDIO_REI_REO_V2))
waiting for download
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job2 starts
Job1 starts
Job1 finished
Job2 finished
```

In order to test this, we can insert the delay or 15 seconds in the or during the test before receiving semaphores here and then output messages about receiving semaphores here's delay and hear messages semaphore one and semaphore to upload and monitor start waiting for drop finish it as we can see the program has received only the first semaphore and is waiting for the second one this happened because we use the binary semaphore that can only be set once, but we need to send two semaphores and we also need to accept two to do this we need to use counting semaphore let's change the binary semaphore to count to one a counting semaphore is almost the same as binary one it uses the same datatype let's define it I will call it conjunction it differs only in the function of creating semaphore when creating it we must specify the maximum number of semaphores and the initial video of is counter created here this maximum number of semaphores and zero is initial video of counter. Now, you can simply to change our binary semaphore to counting semaphore. Here here int send him here and here upload and monitor this 30s be waiting for our semaphores as we can see the semaphore is To distribute the program works correctly. Now let's imagine that together with the semaphore, we need to transmit data with some kind of yoke result for this purpose queues used, which allows us to exchange data between disks. Let's replace our semaphore with a queue type of q is q quindell Let's define a queue I will call it x queue. To create a queue, we need to specify the type of data being stored in it and the size of the queue. The queue size is the maximum amount of data of the specified type that it can store. The queue is created by the function excu create specify the queue size of 10 and the data type you insert into. Now we can transmit and receive data using this queue. Let's proceed recounted video at the end of each of our tasks. Next we'll receive it and output to the monitor the scene will be performed by the function x q sent islands souped up here and here I passed to this function name of the queue reference to our counter and timeout timeout.



```
  xQueueSend(xQueue, (void const *)&cnt, NULL);
}

void job2()
{
  Serial.println("Job2 starts");
  for (int j = 0; j < 10; j++)
  {
    for (int i = 0; i < 1000000; i++)
    {
      portENTER_CRITICAL(&spin);
      cnt++;
      portEXIT_CRITICAL(&spin);
    }
    delay(1);
  }
  Serial.println("Job2 finished");
  //xSemaphoreGive(countingSem);
  xQueueSend(xQueue, (void const *)&cnt, NULL);
}

void job1Task(void *pvParameters)
{
  delay(1);
  job1();
  vTaskDelete(NULL);
```

```
load:0x40078000,len:10124
load:0x40080400,len:5828
entry 0x400806a8
Job2 starts
Job1 starts
Job1 finished
Job2 finished
Sem 1
```

We don't need timeout it's not now we can accept the preset counter willya by the function excu receive you don't need this delay. And to view the receive here I have created a received counter variable here in which the received value will be right in the queue handler and timeout I'll also pass it to this function next we will output this video to the monitor and same for second job second job also we don't need to know this entities let's upload and monitor just start the spear we see the counter video is based from the first task because it finishes its work faster. At the same time, the counter value is not yet equal to 20 million.

EXAMPLE DUMMY CODE

In Arduino, you can use the FreeRTOS library to implement Semaphores and Queues for synchronizing tasks and communication between tasks.

Semaphore Example:
In this example, we have two tasks (task1 and task2) that access a shared resource (sharedValue). We use a binary semaphore to ensure that only one task can access the shared resource at a time.

```
#include <Arduino_FreeRTOS.h>
#include <semphr.h>

// Shared variable
volatile int sharedValue = 0;

// Semaphore handle
SemaphoreHandle_t semaphore;

// Task functions
void task1(void *pvParameters);
void task2(void *pvParameters);

void setup() {
  Serial.begin(9600);

  // Create a binary semaphore
  semaphore = xSemaphoreCreateBinary();

  // Start the tasks
  xTaskCreate(task1, "Task1", 1000, NULL, 1, NULL);
  xTaskCreate(task2, "Task2", 1000, NULL, 1, NULL);
```

```
}

void loop() {
  // Nothing to do here in this example.
}

void task1(void *pvParameters) {
  for (;;) {
    // Wait for the semaphore to be available
    if (xSemaphoreTake(semaphore, portMAX_DELAY) ==
pdTRUE) {
      // Critical section: Access the shared variable
      sharedValue++;
      Serial.print("Task1 - Shared Value: ");
      Serial.println(sharedValue);

      // Release the semaphore after accessing the shared
resource
      xSemaphoreGive(semaphore);
    }

    vTaskDelay(1000); // Wait for 1 second
  }
}

void task2(void *pvParameters) {
  for (;;) {
    // Wait for the semaphore to be available
    if (xSemaphoreTake(semaphore, portMAX_DELAY) ==
pdTRUE) {
      // Critical section: Access the shared variable
      sharedValue--;
      Serial.print("Task2 - Shared Value: ");
      Serial.println(sharedValue);
      // Release the semaphore after accessing the shared
resource
      xSemaphoreGive(semaphore);
    }

    vTaskDelay(2000); // Wait for 2 seconds
  }
}
```

Queue Example:
In this example, we have two tasks (taskSender and taskReceiver) that communicate using a queue. The taskSender sends messages (numbers) to the queue, and the taskReceiver receives and processes the messages.

```
#include <Arduino_FreeRTOS.h>
#include <queue.h>

// Queue handle
QueueHandle_t queue;

// Task functions
void taskSender(void *pvParameters);
void taskReceiver(void *pvParameters);

void setup() {
  Serial.begin(9600);

  // Create a queue with a capacity of 5 elements, each of
size sizeof(int)
  queue = xQueueCreate(5, sizeof(int));

  // Start the tasks
  xTaskCreate(taskSender, "Sender", 1000, NULL, 1, NULL);
  xTaskCreate(taskReceiver, "Receiver", 1000, NULL, 1,
NULL);
}
```

```
void loop() {
  // Nothing to do here in this example.
}

void taskSender(void *pvParameters) {
  int message = 0;
  for (;;) {
    // Send a message to the queue
    xQueueSend(queue, &message, portMAX_DELAY);
    Serial.print("Sent Message: ");
    Serial.println(message);

    message++;
    vTaskDelay(1000); // Wait for 1 second
  }
}

void taskReceiver(void *pvParameters) {
  int receivedMessage;
  for (;;) {
    // Receive a message from the queue
    if (xQueueReceive(queue, &receivedMessage,
portMAX_DELAY) == pdTRUE) {
      Serial.print("Received Message: ");
      Serial.println(receivedMessage);
    }

    vTaskDelay(2000); // Wait for 2 seconds
  }
}
```

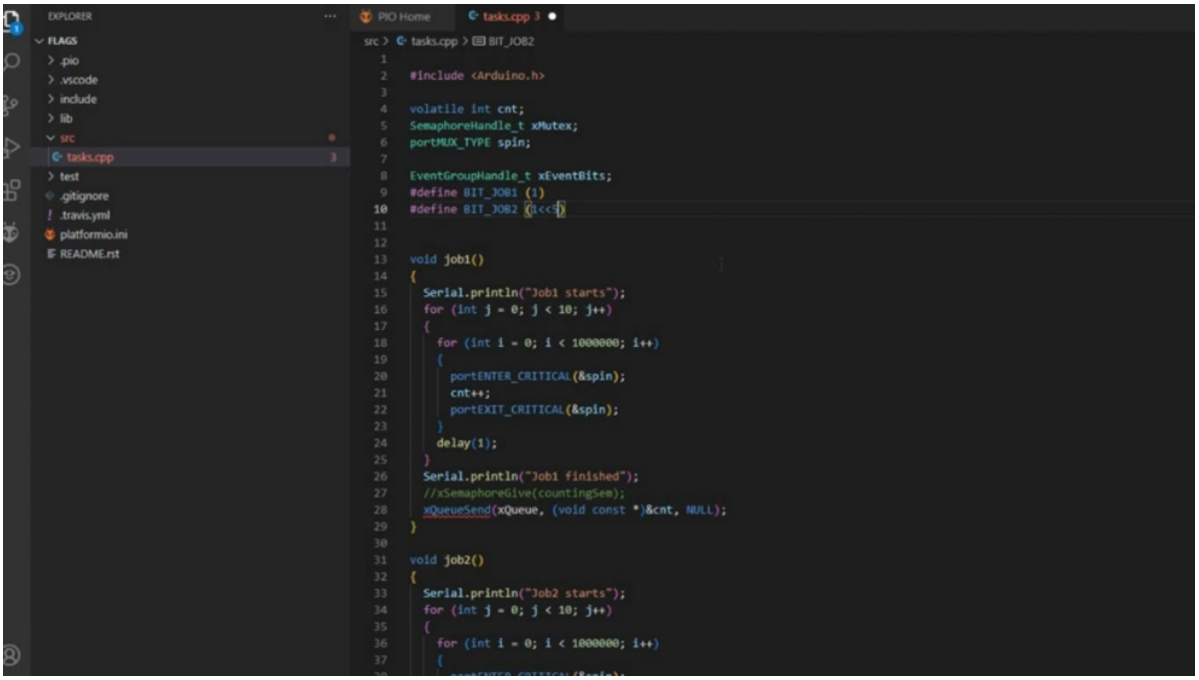
In this example, we use xQueueCreate() to create a queue capable of holding five elements of type int. The taskSender sends messages to the queue using xQueueSend(), and the taskReceiver receives messages from the queue using xQueueReceive(). The queue acts as a buffer for communication between the two tasks.

Remember to include the Arduino_FreeRTOS.h and relevant header files for queues (queue.h) and semaphores (semphr.h) to use these features.

Using Semaphores and Queues, you can create more complex systems with task synchronization and inter-task communication to efficiently manage resources and data flow in your Arduino applications.

EVENT FLAGS

If we look at synchronization between FreeRTOS disks using semaphore to send cues, there is another possibility of synchronization using flux. The flick is single bit that is set by the program when an event occurs then this flip can be triggered elsewhere in the program the flux or combine it into a group call it event group in ESP 32 There can be a 24 flux in such a group using Flex is similar to using simple photo so, use first we need to define a handle for a group of flex this group can do then we can create this group in setup now, we don't need semaphores to remove it and hear it more semaphorin Q's we also need to define the bits themselves each task will have its own bit there's a bid for job one and for job two we can assign them a new numbers let it be let it be fifth it can be any recent 24 Now, we can set this bits instead of semaphores or queues here it's beautiful job one and save for gelato now, we have to accept this bits This is done by the function x event group with bits we didn't need this all I replaced it with the function also we don't need this if you need this code you can find it in previous projects this function that returns the value of the read bits let's output it to the monitor in binary format it's been a reformat to Boolean parameters pass it to the bid weighting function.



```
#include <Arduino.h>

volatile int cnt;
SemaphoreHandle_t xMutex;
portMUX_TYPE spin;

EventGroupHandle_t xEventBits;
#define BIT_JOB1 (1)
#define BIT_JOB2 (1<<5)


void job1()
{
  Serial.println("Job1 starts");
  for (int j = 0; j < 10; j++)
  {
    for (int i = 0; i < 1000000; i++)
    {
      portENTER_CRITICAL(&spin);
      cnt++;
      portEXIT_CRITICAL(&spin);
    }
    delay(1);
  }
  Serial.println("Job1 finished");
  //xSemaphoreGive(countingSem);
  xQueueSend(xQueue, (void const *)&cnt, NULL);
}

void job2()
{
  Serial.println("Job2 starts");
  for (int j = 0; j < 10; j++)
  {
    for (int i = 0; i < 1000000; i++)
    {
      portENTER_CRITICAL(&spin);
```

If this parameter is passed equal to PD through the received bid will be received upon exceeding the function and if this is true, then the function waits for all specified beats to be set. The beats themselves are set here. There's bits also there kind of group of bits and time out policy to the function and time out. Thus in our case they are functional will not receive the receiver bits and the will wait for all bits to be set. Let's Compile and first I will uncomment this now compile and monitor its company Artis visa compile again it's legit, the CTSP and waiting for our bits as we can see in the terminal.

EXAMPLE DUMMY CODE

In Arduino using FreeRTOS, Event Flags can be used to enable tasks to wait for specific events to occur. When the event flags are set, tasks that are waiting for those flags can be unblocked and execute their respective actions. In this example, we have two tasks (task1 and task2) that wait for different event flags (EVENT_FLAG_1 and EVENT_FLAG_2) to be set by a third task (taskController).

```
#include <Arduino_FreeRTOS.h>
#include <event_groups.h>

// Event group handle
EventGroupHandle_t eventGroup;

// Event flags
const EventBits_t EVENT_FLAG_1 = (1 << 0);
const EventBits_t EVENT_FLAG_2 = (1 << 1);

// Task functions
void task1(void *pvParameters);
void task2(void *pvParameters);
void taskController(void *pvParameters);

void setup() {
  Serial.begin(9600);

  // Create the event group
  eventGroup = xEventGroupCreate();

  // Start the tasks
  xTaskCreate(task1, "Task1", 1000, NULL, 1, NULL);
  xTaskCreate(task2, "Task2", 1000, NULL, 1, NULL);
  xTaskCreate(taskController, "Controller", 1000, NULL, 1,
NULL);
}

void loop() {
  // Nothing to do here in this example.
}

void task1(void *pvParameters) {
  for (;;) {
    // Wait for EVENT_FLAG_1 to be set
    EventBits_t eventBits =
xEventGroupWaitBits(eventGroup, EVENT_FLAG_1,
pdTRUE, pdFALSE, portMAX_DELAY);

    // Check if EVENT_FLAG_1 is set
    if (eventBits & EVENT_FLAG_1) {
      // Do task1's action when EVENT_FLAG_1 is set
      Serial.println("Task1 action triggered!");
    }
  }
}

void task2(void *pvParameters) {
  for (;;) {
    // Wait for EVENT_FLAG_2 to be set
    EventBits_t eventBits =
xEventGroupWaitBits(eventGroup, EVENT_FLAG_2,
pdTRUE, pdFALSE, portMAX_DELAY);

    // Check if EVENT_FLAG_2 is set
    if (eventBits & EVENT_FLAG_2) {
      // Do task2's action when EVENT_FLAG_2 is set
      Serial.println("Task2 action triggered!");
    }
  }
}

void taskController(void *pvParameters) {
  for (;;) {
    // Simulate events being set
    // In a real application, events could be set based on specific conditions or external inputs
    vTaskDelay(3000); // Wait for 3 seconds
    xEventGroupSetBits(eventGroup, EVENT_FLAG_1);

    vTaskDelay(2000); // Wait for 2 seconds
    xEventGroupSetBits(eventGroup, EVENT_FLAG_2);

    vTaskDelay(4000); // Wait for 4 seconds
    xEventGroupSetBits(eventGroup, EVENT_FLAG_1 | EVENT_FLAG_2);
  }
}
```

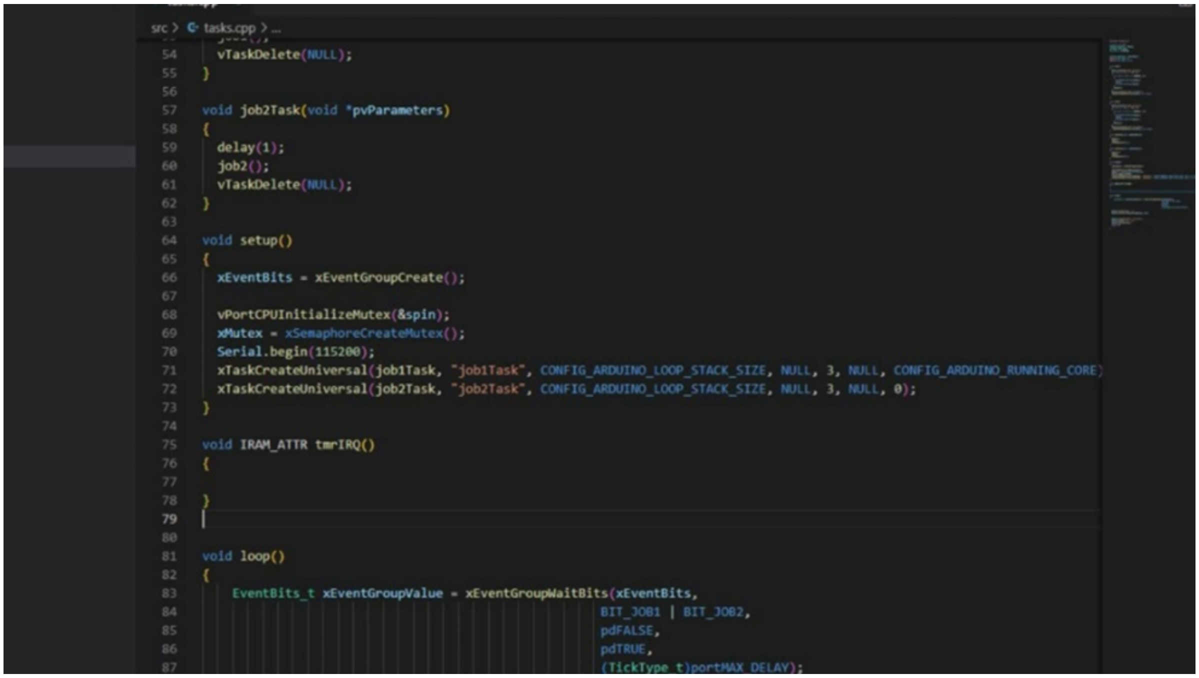
In this example, we use xEventGroupCreate() to create an event group. The task1 and task2 tasks wait for different event flags (EVENT_FLAG_1 and EVENT_FLAG_2) to be set by the taskController. When the event flags are set by the taskController, the waiting tasks (task1 and task2) are unblocked and execute their respective actions.

In taskController, we simulate the events being set at different time intervals using vTaskDelay(). In a real application, you would set the event flags based on specific conditions or external inputs.

Please ensure that you include the necessary FreeRTOS and event groups header files in your Arduino IDE for this code to work correctly.

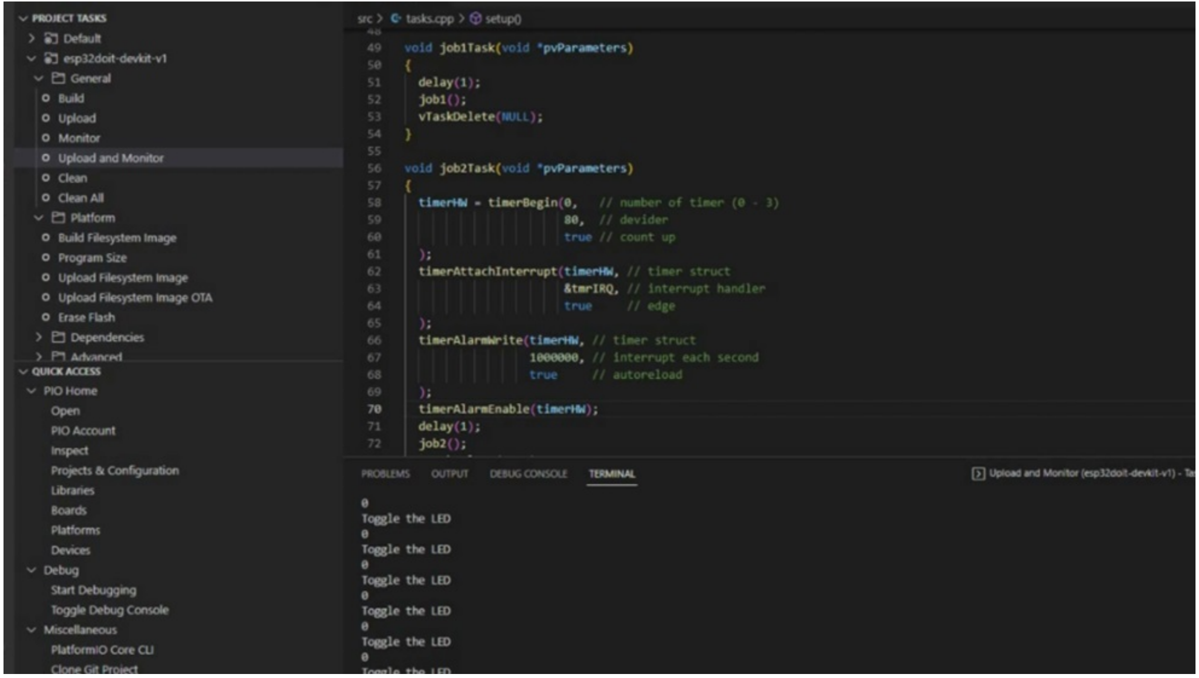
HARDWARE INTERRUPTS

Let's now look at hardware interrupts. As you know interrupts a load to suspend the main program in order to perform short function or the interrupt suspend. Also, interrupts can interrupt each other depending on the priority. Interrupts with click priority you can interrupt them with low privilege. They have free Arctos task privilege and interrupt privilege are two different systems. Any hardware interrupt with the low privilege will always interrupt the task with the high privilege.



```
  vTaskDelete(NULL);
}

void job2Task(void *pvParameters)
{
  delay(1);
  job2();
  vTaskDelete(NULL);
}

void setup()
{
  xEventBits = xEventGroupCreate();

  vPortCPUInitializeMutex(&spin);
  xMutex = xSemaphoreCreateMutex();
  Serial.begin(115200);
  xTaskCreateUniversal(job1Task, "job1Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, CONFIG_ARDUINO_RUNNING_CORE)
  xTaskCreateUniversal(job2Task, "job2Task", CONFIG_ARDUINO_LOOP_STACK_SIZE, NULL, 3, NULL, 0);
}

void IRAM_ATTR tmrIRQ()
{

}


void loop()
{
    EventBits_t xEventGroupValue = xEventGroupWaitBits(xEventBits,
                                                       BIT_JOB1 | BIT_JOB2,
                                                       pdFALSE,
                                                       pdTRUE,
                                                       (TickType_t)portMAX_DELAY);
```

Unfortunately, they are doing the functions do not allow to set the priority of interrupts therefore, we will not change it we will use the default one each ESP 32 core has its own interrupt system now, we will look at the operation of interrupts using the example of a timer let's add an interrupt from the timer to our program let the interrupts happen every second and the lid will toggle in the interrupt to define a timer use the hardware timer type is hardware timer timer let's define interrupt function for the timer now it's do nothing and I'll prototype for it and then we can create and start the timer Yeah, this functions the first function initialize the timer it sets the time that number zero divided at the direction of the count up or down then the interrupt function that we created today is attach it to the timer next the timer is set to the value to which it will count this one it's selected so that interruption goes every second and then the timer starts with the function timer alarm enable now let's it and led to the interrupt initialize it first this new civilization of the lid now we insert the toggling of the lid in the interrupt function I will also add output to the monitor message about switching traded now we can compile and test how it works. Upload and monitor they get the stolen on which call does the interrupt processing function work? We can find out by using the function export with caller ID it returns the number of the caller it's running home. Output to this number to the monitor. This explorer get caller ID Upload and monitor restart tsp we see that number of the quarter is one. At the same time, the setup task in which the timer was initialized runs on the same call. Thus the interrupt function will be run on the quarter on which the interrupt was initialized. Let's check it out and start the timer on another core. This can be done at the beginning of dusk two which runs on Korea zero I remove for these functions to beginning of dusk to hear and applaud and monitor restart USB as we can see how interrupts works now on CUDA 00 interrupt handler has a print output function to the monitor this function runs for the relatively long time so it's better not to use it in interrupts so let's create another task and transfer the functionality from the interrupt handler to this task and in the vendor itself, we will only set a flag that will be receive it in the disk I will call it lit dusk and are run it on quarter one which are the neurons and do the disk it's MQ lit disk do you find the flu for the timer let it be number two then you can sit this fluky in our interrupt function here now this flip can be accepted in the lit desk.



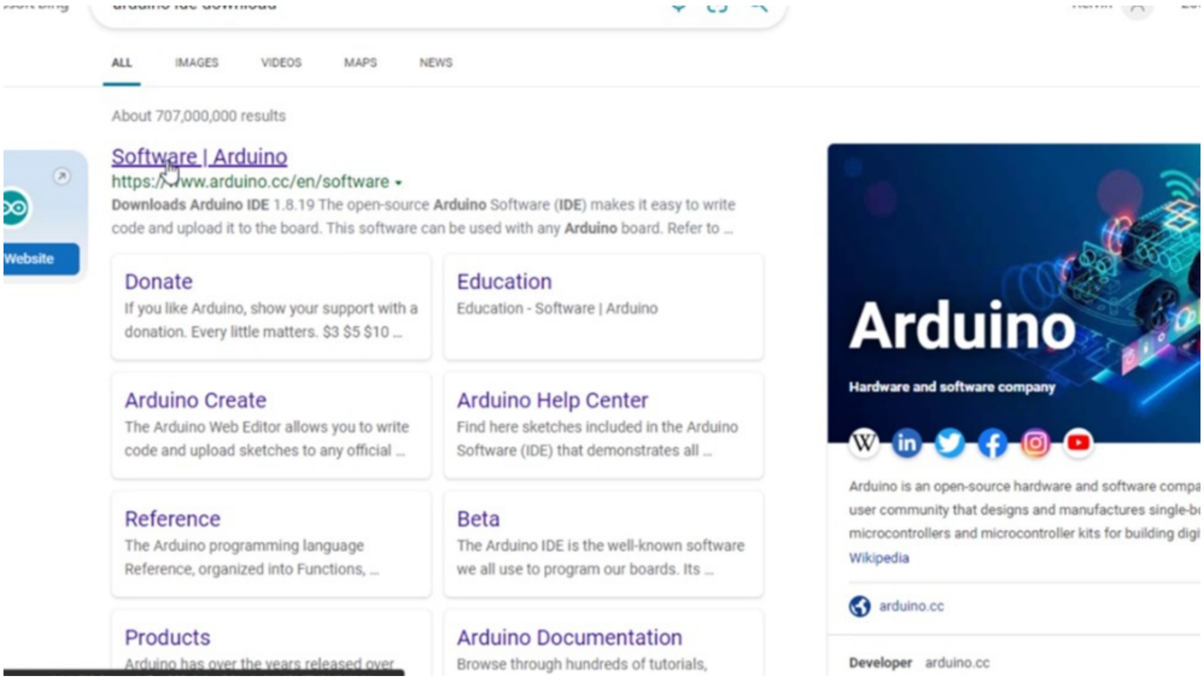
```
void job1Task(void *pvParameters)
{
  delay(1);
  job1();
  vTaskDelete(NULL);
}

void job2Task(void *pvParameters)
{
  timerHW = timerBegin(0,   // number of timer (0 - 3)
                       80,  // devider
                       true // count up
  );
  timerAttachInterrupt(timerHW, // timer struct
                       &tmrIRQ, // interrupt handler
                       true     // edge
  );
  timerAlarmWrite(timerHW, // timer struct
                  1000000, // interrupt each second
                  true     // autoreload
  );
  timerAlarmEnable(timerHW);
  delay(1);
  job2();
```

```
0
Toggle the LED
0
Toggle the LED
0
Toggle the LED
0
Toggle the LED
0
Toggle the LED
0
Toggle the LED
```

Here note that this parameter is set to be true. So they will be reset when exiting this function can exit into this now we can transfer functions from the interrupt handler to the led desk. This all functions and now we can upload and monitor as we can see the task is Iranian hard coded one. Also all our tasks continue to be performed without interfering with each other Let's restart ESP 32 and take a look at the look at it started Thus, we have created the program running simultaneously on two processor cores and performing several tasks now let's look at the fluoxetine function that we call it from the interrupt this one we can see that it's different from the one we go in the disk I will copy and place these two functions side by side so that the differences visible. Firstly, layer function call it from the interrupt because the same name only from a Sir This has been added to it. It should be borne in mind that all similar FreeRTOS functions have the phrase from ICER in the name therefore, only from a set of functions should be coded in the interrupt. Secondly, from ISIL function has another parameter. This is designed to quickly switch to the task with the highest priority at the end of the interrupt. This parameter is optional I recommend not using it and passing now. Two functions instead.

GETTING STARTED

For the software requirements for these projects, we'll be using the Arduino IDE with the ESP 32 board manager who need an Internet connectivity and we'll be using for those of us that will be using ESP tends to call the physical body or work with optional. Opie is also a good substitute for online simulation. So in this service we will be using will be to simulate it online. To go ahead, regardless that we're using house you should also install the ESP 32 port manager. Now for those that will be using the ESP testing physical or to programming to install the Arduino IDE for those of us are using the ESP 32 hardware with our browser, such Arduino IDE download so click on the first link oscilloscope opened up Arduino IDE 2.0.



About 707,000,000 results

Software | Arduino
https://www.arduino.cc/en/software
Downloads Arduino IDE 1.8.19 The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board. Refer to ...

Donate
If you like Arduino, show your support with a donation. Every little matters. $3 $5 $10 ...

Education
Education - Software | Arduino

Arduino Create
The Arduino Web Editor allows you to write code and upload sketches to any official ...

Arduino Help Center
Find here sketches included in the Arduino Software (IDE) that demonstrates all ...

Reference
The Arduino programming language Reference, organized into Functions, ...

Beta
The Arduino IDE is the well-known software we all use to program our boards. Its ...

Products
Arduino has over the years released over

Arduino Documentation
Browse through hundreds of tutorials,

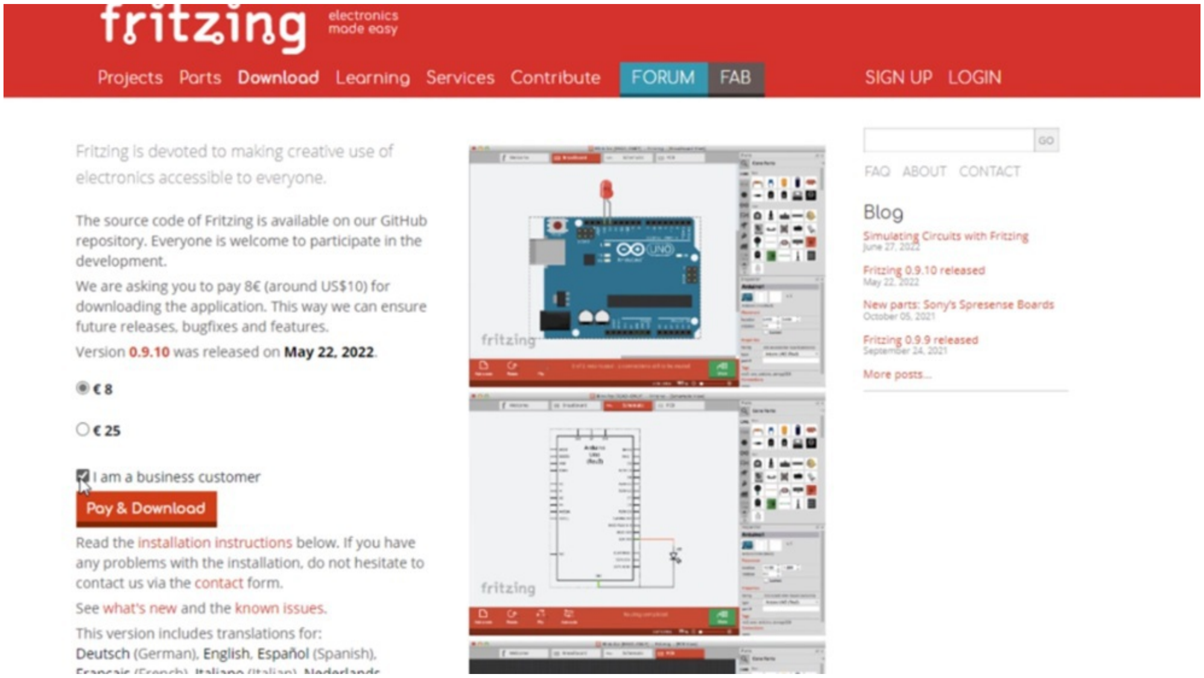
Arduino
Hardware and software company

Arduino is an open-source hardware and software compa user community that designs and manufactures single-b microcontrollers and microcontroller kits for building digi
Wikipedia

arduino.cc

Developer arduino.cc

So look for Windows installation, Windows 10 and ulcc floppies. So you can look for one that is suitable for your PC. I'll just click on them next, or you can just close that your download will begin in a moment so while it is under the other software which we won't use in this service, but it's helpful for those that will be using the physical ESP setup you need the freezing issues for drain set up diagrams for freezing them loose so we opened the first leg so freezes some websites who require you to pay before you're able to download it so this website actually has the paid version or it's so go back we'll go back and look for another website that has a free version or a trial version okay good so we can download from these websites so after the download the installation process are quite simple.



Fritzing is devoted to making creative use of electronics accessible to everyone.

The source code of Fritzing is available on our GitHub repository. Everyone is welcome to participate in the development.

We are asking you to pay 8€ (around US$10) for downloading the application. This way we can ensure future releases, bugfixes and features.

Version 0.9.10 was released on May 22, 2022.

● € 8

○ € 25

☑ I am a business customer

Pay & Download

Read the installation instructions below. If you have any problems with the installation, do not hesitate to contact us via the contact form.

See what's new and the known issues.

This version includes translations for:
Deutsch (German), English, Español (Spanish), Français (French), Italiano (Italian), Nederlands

Blog

Simulating Circuits with Fritzing
June 27, 2022

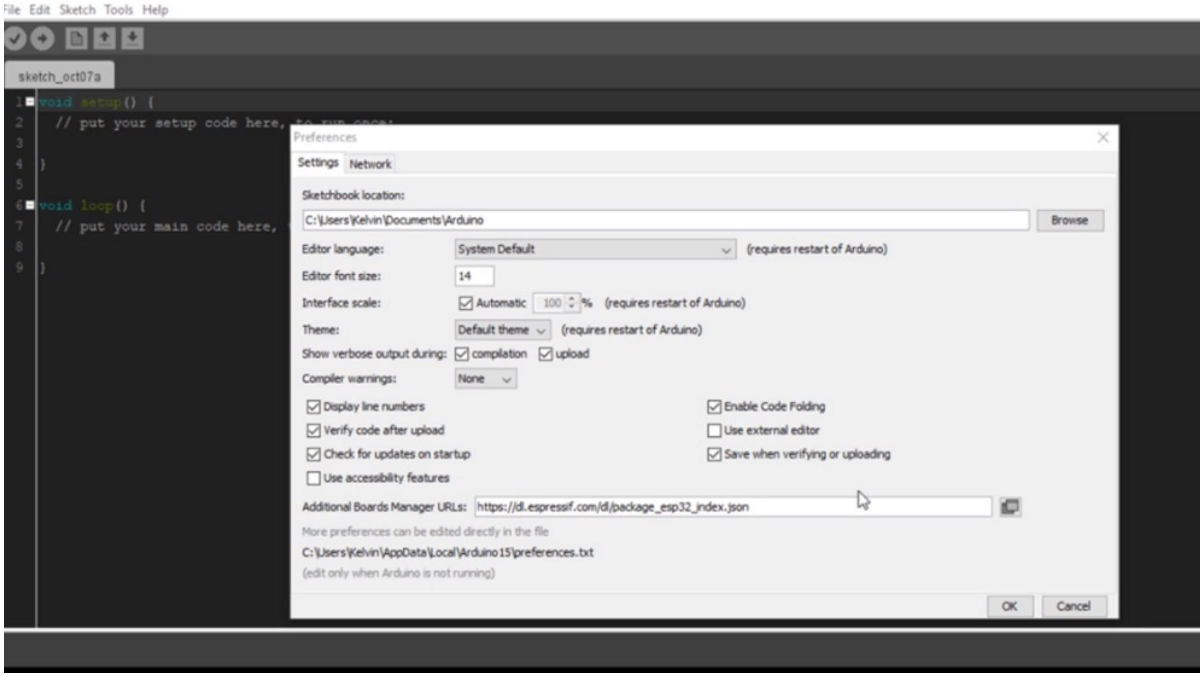
Fritzing 0.9.10 released
May 22, 2022

New parts: Sony's Spresense Boards
October 05, 2021

Fritzing 0.9.9 released
September 24, 2021

More posts...

Open the file and follow the installation processes to install the software and get started. So now in the Arduino IDE, we'll have to install the ESP board manager. So go to tools. First of all go to preferences. So once it connects your ESP board, everything will be set for you. As soon as you plug in your ESP or ESP board into your USB port.



```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here,

}
```

Sketchbook location: C:\Users\Kelvin\Documents\Arduino

Editor language: System Default (requires restart of Arduino)

Editor font size: 14

Interface scale: ☑ Automatic 100 % (requires restart of Arduino)

Theme: Default theme (requires restart of Arduino)

Show verbose output during: ☑ compilation ☑ upload

Compiler warnings: None

☑ Display line numbers

☑ Verify code after upload

☑ Check for updates on startup

☐ Use accessibility features

☑ Enable Code Folding

☐ Use external editor

☑ Save when verifying or uploading

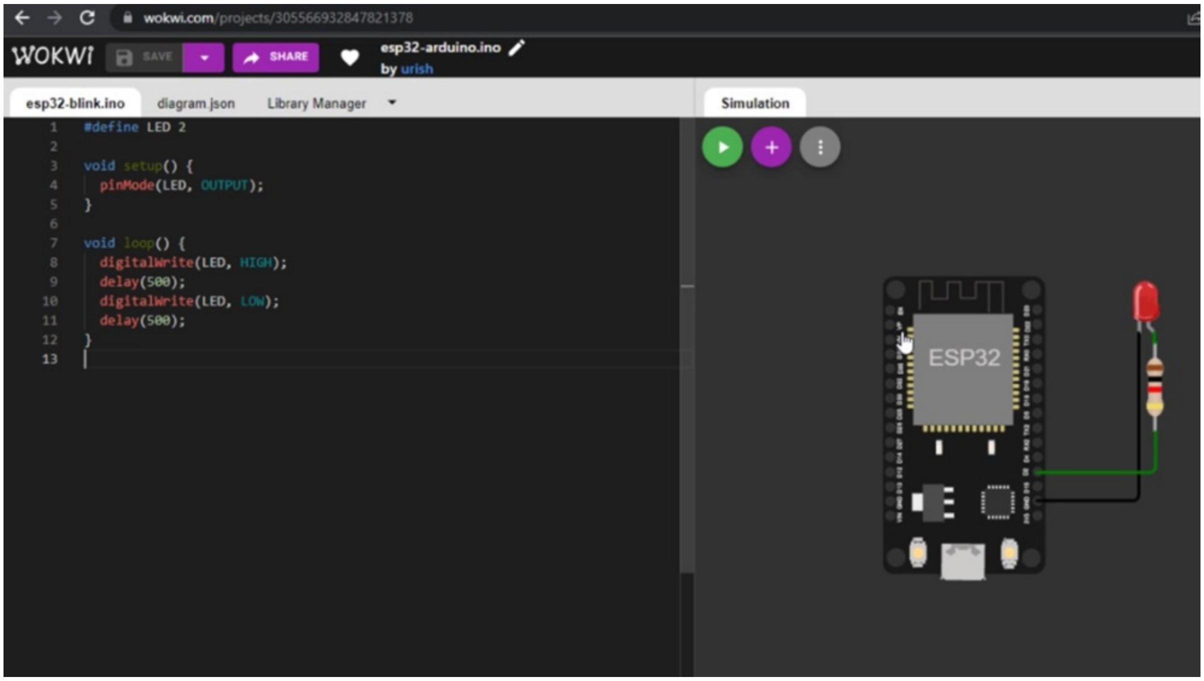
Additional Boards Manager URLs: https://dl.espressif.com/dl/package_esp32_index.json

More preferences can be edited directly in the file

C:\Users\Kelvin\AppData\Local\Arduino15\preferences.txt

(edit only when Arduino is not running)

This link is going to change to ESP and some details. So that will go to tools to get for both info keyboards manager which reports manager Okay, so search for ESP ESP 32 So this is it. We'll install it so the installation takes St. Louis have time to fast forward this video to the end of insulation so at this point the installation is complete. So after having seen how to install the ESP 32 board manager, including the Arduino IDE, let's go ahead and test our ESP can see the movement board are working with our computer PC because Boku is a software that we'll be using to simulate its online phase of using workweek for the service, let's head on to workweek.com. But here's what the website looks like, we'll click on sign in or sign up if you're a first time or you sign up. Someone has signed up in my Google account so so I mean, I mean, so you can see this project templates over here. We're just templates, you can just pick any one you want and add it to your test. So to speak ESP test, it's what you know the one with an LED.

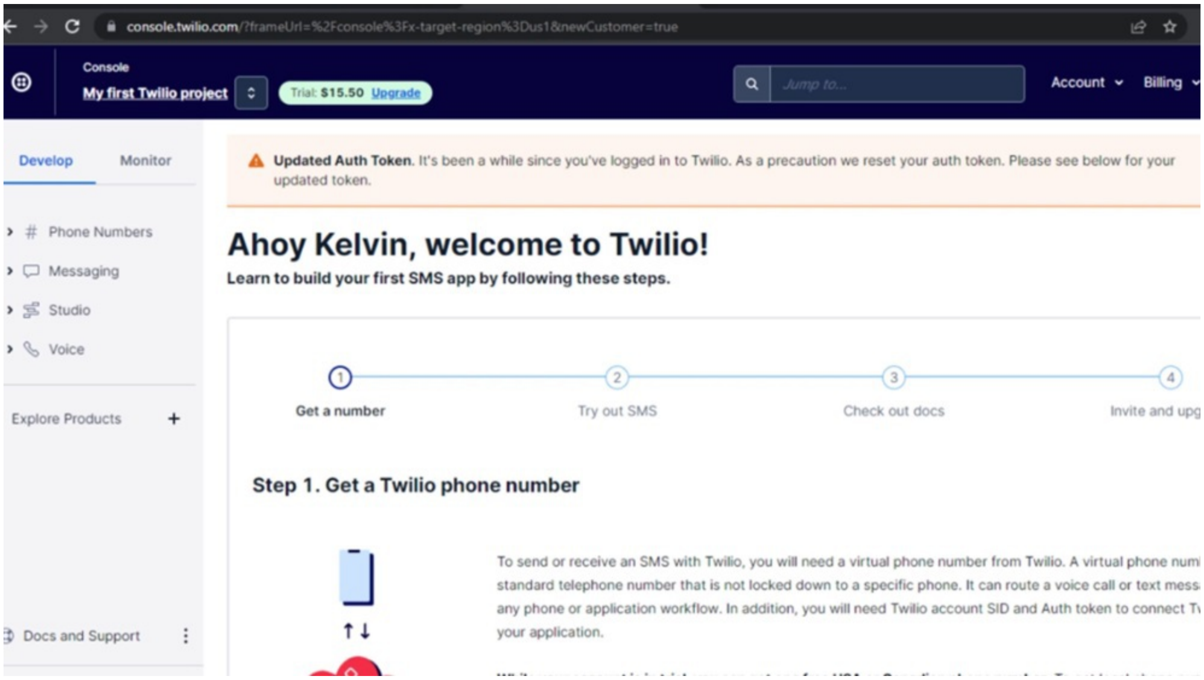


```
#define LED 2

void setup() {
  pinMode(LED, OUTPUT);
}

void loop() {
  digitalWrite(LED, HIGH);
  delay(500);
  digitalWrite(LED, LOW);
  delay(500);
}
```

So we can see on the left hand side we have the code side while on the right hand side we have the simulation but so let's simulate this and see how it works. This is just to blink an led to a blinking so now we can change the color of only different diagram dots just software can change it to yellow. That's changed recently it's still blinking. So we can change the delay we can change it from 0.5 seconds to 1000 1000 microseconds versus a second. So 1000 milliseconds or one second rather. So they simulate again. So this is what we that is all we'll be using for the simulation of this series.

TWILIO SET UP

So we have tested our ESP online simulation, it's a work piece of software. So go ahead and set up our two accounts episode we are going to set up AWS accounts we'll go to the official websites we know that so this is the websites we'll click on sign up and start building swanky concern you're going to fill out all these information’s here to start a free trial we already have an account which I'm gonna login with so I'll put my password so verify your identity so after signing up for your new accounts for when you're logging in for the first time, you have to register a phone number which you will be using so put in a phone number and verify so a code will be sent to my phone this is the code that has been sent to my phone so now I'm going to put in their code here it's 93223 and guys so meats so my Twilio account is set we'll answer a basic question here guys in the SMS regards a Latin expression with only go so get started with two so skip these we don't need all this done.



Updated Auth Token. It's been a while since you've logged in to Twilio. As a precaution we reset your auth token. Please see below for your updated token.

Ahoy Kelvin, welcome to Twilio!

Learn to build your first SMS app by following these steps.

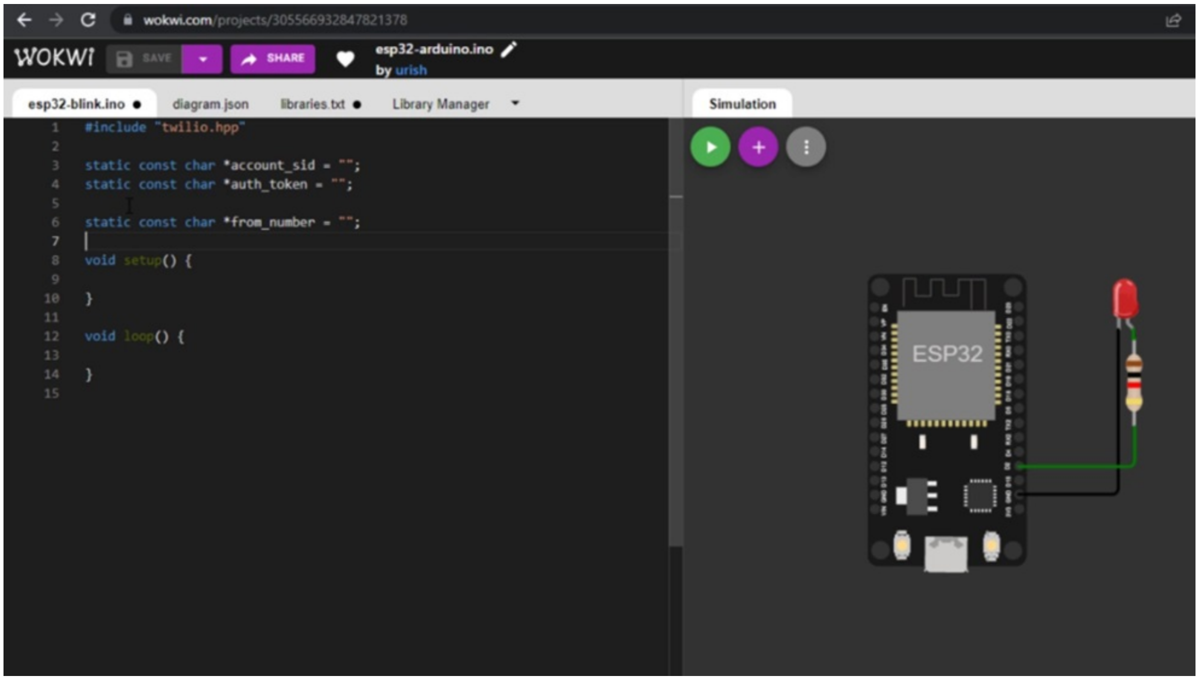
Step 1. Get a Twilio phone number

To send or receive an SMS with Twilio, you will need a virtual phone number from Twilio. A virtual phone num
standard telephone number that is not locked down to a specific phone. It can route a voice call or text mess
any phone or application workflow. In addition, you will need Twilio account SID and Auth token to connect T
your application.

So we're Intuit riilu accounts. First we'll go to phone numbers so that we can purchase a number manage by a number to search for the number that has SMS compared to these so this one has SMS we can see the send SMS so click on Buy click on Buy scroll down to where we have the buyer will buy this so we bought this now we can send SMS using this number that we bought so let's go back let's go to messaging get setup start setup. Message Service name will just give you just a very simple name and give it just with ESP alerts, ESP alerts tests and request messaging service. So this is a number who add this number, the number we purchased. So then this is our account s ID. We have to copy this out on steroids so that we don't misplace it. So open my notepad paste it in as at least I'll do my authentication to how copy open Notepad and paste so try SMS you after you get to this point where you're trying to send SMS so body of text can be anything can be hello from Hello from my ESB testing Hello from my ESP test so let's send SMS so the SMS has been sent for let me get screenshots from my phone so that's the SMS I will receive on my phone so we will send the messages a few more times and see how it's comes in in the phone. messages have been coming in into my mobile phones. So that is three live setup for this episode.

CODE SETUP FOR ESP32 USING TWILIO

So now having set up our three accounts, let's go back to the work we simulation software. Alright, the set of code for ESP using three will be writing a sample code for ESP 32 to send estimators in three loop on the Okwe software. So let's get rid of the codes in the setup and the loop. For first we need to install the three low ESP clients library then we'll come back and include that library three dot HPP Next, we declare our account Sid and the authentication token that we have in our dashboard somewhere will be blank I'm going to be blank compounds with letter so from number that is the number we purchased on the thriller software, the static constants the number we are sent into which is our number just as you registered and it will have software then the body of the message the message contents leave it blank for now come back to it later.



```
#include "twilio.hpp"

static const char *account_sid = "";
static const char *auth_token = "";

static const char *from_number = "";

void setup() {

}

void loop() {

}
```

Then we'll we'll so in the setup we initialize our chakra monitor first. Then we set up the connection for the ESP test to to connect to the internet. So on walk with simulation, it has inbuilt internet connection for the ESP 32 So Wi Fi does begin it needs no password SSID is workweek guests guest is in capital letter then six with these the ESP that's to connect to the inbuilt internet connection of the work we software so why Wi Fi is why the ESP is not connected to Wi Fi it will keep printing connecting delays over five seconds the length is gonna it's gonna print connected so in line 24 We initialize the connection between the UI or the ESP 32 and a three loop. So the LED is a circle the nest is the message function that will send the message to our phone so if the message was sent successfully, it will print a sends message successfully else Sarah does brains the response so that is for sample code. So get rid of the LED and they just need that. So account s ID will copy is from where we save this will paste will copy the authentication token and paste then the number we purchased on our Sweet Lou accounts purposes and a number were sent into then the body of the message we'll just use messages in three load and in brackets testing.

EXAMPLE DUMMY CODE

To set up the ESP32 to work with Twilio and send SMS messages, you will need an ESP32 board, an active Twilio account, and the Twilio library for Arduino. Follow these steps to get started:

STEP 1: SET UP A TWILIO ACCOUNT

If you don't have a Twilio account, sign up for one at https://www.twilio.com/. After signing up, you'll need to purchase a Twilio phone number and obtain your Account SID and Auth Token, which will be used to authenticate your requests.

STEP 2: INSTALL TWILIO ARDUINO LIBRARY

In the Arduino IDE, go to "Sketch" -> "Include Library" -> "Manage Libraries". Search for "Twilio" and click "Install" to add the Twilio library to your Arduino IDE.

STEP 3: WRITE THE ARDUINO CODE

Here's an example Arduino code to set up the ESP32 with Twilio and send an SMS message:

```
#include <WiFi.h>
#include <Twilio.h>

// Replace with your Wi-Fi credentials
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";

// Replace with your Twilio credentials
const char* accountSid = "YOUR_TWILIO_ACCOUNT_SID";
const char* authToken = "YOUR_TWILIO_AUTH_TOKEN";
const char* twilioNumber =
"YOUR_TWILIO_PHONE_NUMBER"; // Must include country
code, e.g., +1234567890

void setup() {
  Serial.begin(115200);

  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi!");

  // Set up Twilio client
  Twilio.init(accountSid, authToken);
}

void loop() {
  // Send an SMS message using Twilio
  String message = "Hello from your ESP32!";
  String recipientNumber = "+1234567890"; // Replace with the recipient's phone number (include country code)

  bool messageSent = Twilio.sendSMS(twilioNumber, recipientNumber, message);

  if (messageSent) {
    Serial.println("SMS sent successfully!");
  } else {
    Serial.println("Failed to send SMS.");
  }

  // Wait for 10 seconds before sending another message
  delay(10000);
}
```

Replace the placeholders (YOUR_WIFI_SSID, YOUR_WIFI_PASSWORD, YOUR_TWILIO_ACCOUNT_SID, YOUR_TWILIO_AUTH_TOKEN, YOUR_TWILIO_PHONE_NUMBER, and +1234567890) with your actual Wi-Fi credentials, Twilio account credentials, Twilio phone number, and the recipient's phone number.

STEP 4: UPLOAD THE CODE

Connect your ESP32 board to your computer, select the correct board and COM port in the Arduino IDE, and click the "Upload" button to upload the code to the ESP32.

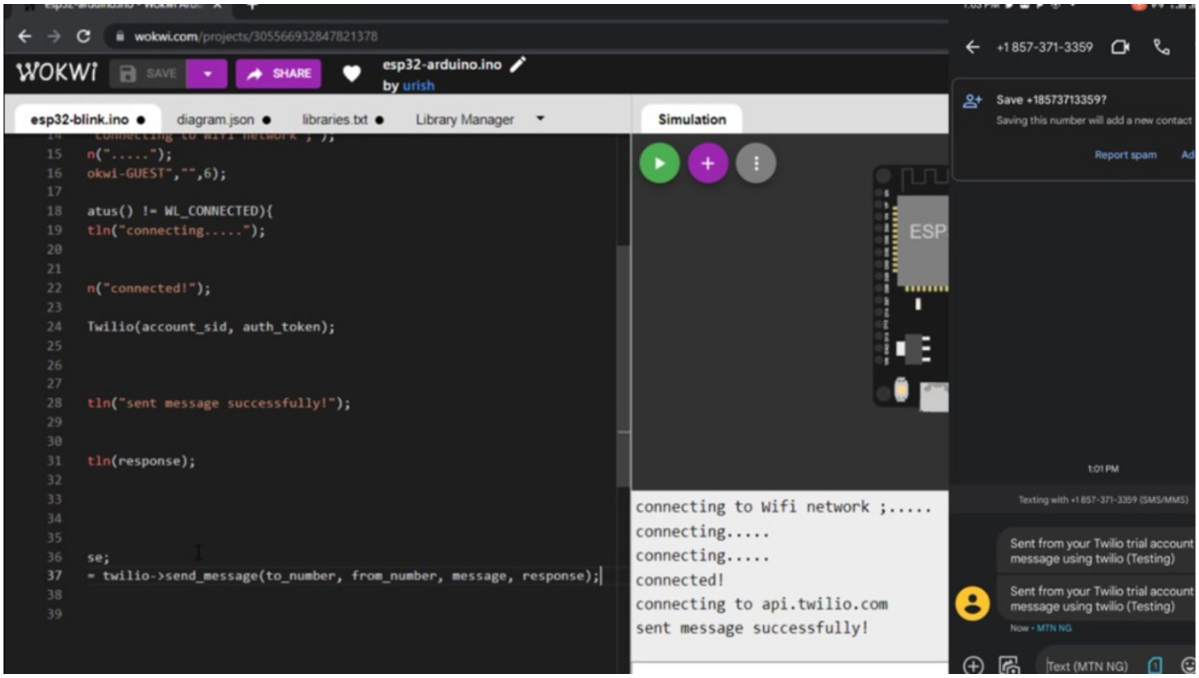
STEP 5: MONITOR SERIAL OUTPUT

After uploading the code, open the Serial Monitor in the Arduino IDE. You should see the ESP32 connecting to Wi-Fi and sending SMS messages using Twilio.

With this setup, your ESP32 will connect to Wi-Fi, initialize the Twilio client, and send an SMS message to the specified recipient number every 10 seconds. Remember that Twilio may charge you for sending SMS messages, so make sure to check their pricing details.

TESTING CODE SETUP FOR ESP32 USING TWILIO

We'll be testing the sample code we wrote in episode four. So that's my mobile phone ready set, let's simulate it that's connected to the API of Twilio to send us a message send successfully and the messages coming to our phone. So you can see the message here. And so because the function is in the setup, we have to refresh the simulation was receive another message because whatever that is in that is is in the setup Ross just was says connecting to three layers again we received SMS I can see the feedback on the Israel monitor to set message successfully.



```
n(".....");
okwi-GUEST","",6);

atus() != WL_CONNECTED){
tln("connecting.....");


n("connected!");

Twilio(account_sid, auth_token);



tln("sent message successfully!");


tln(response);




se;
= twilio->send_message(to_number, from_number, message, response);
```

```
connecting to Wifi network ;.....
connecting.....
connecting.....
connected!
connecting to api.twilio.com
sent message successfully!
```

Sent from your Twilio trial account message using twilio (Testing)

Sent from your Twilio trial account message using twilio (Testing)

So now let's move this message let's move it out from setup and move it to loop so that we don't have to restart the simulation to keep receiving the messages so just copy it we just copied from the setup condense the loop and paste it in the same thing for the success condition if your success issue give us feedback and the serial monitor then I'm gonna add a general delay so that it doesn't keep sending immediately so a delay of three seconds let's simulate. Connecting two three loops so as you receive the message. Now we're not going to refresh the simulation and the messages will keep coming in because now you put it in the loop so that is it for this episode you basically tested the ESP 32 weeks three new software.

EXAMPLE DUMMY CODE

To test the code setup for the ESP32 using Twilio, you can follow the steps mentioned in the previous response to write the Arduino code. Once you have uploaded the code to your ESP32 board, you can use the Serial Monitor in the Arduino IDE to observe the output and verify if the SMS messages are being sent successfully.

Here's the modified Arduino code with additional debugging information to help you test the setup:

```
#include <WiFi.h>
#include <Twilio.h>

// Replace with your Wi-Fi credentials
const char* ssid = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";

// Replace with your Twilio credentials
const char* accountSid = "YOUR_TWILIO_ACCOUNT_SID";
const char* authToken = "YOUR_TWILIO_AUTH_TOKEN";
const char* twilioNumber =
"YOUR_TWILIO_PHONE_NUMBER"; // Must include country
code, e.g., +1234567890

void setup() {
  Serial.begin(115200);
  delay(2000); // Give some time to open the Serial
Monitor

  Serial.println("Connecting to WiFi...");
  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }
  Serial.println("\nConnected to WiFi!");

  // Set up Twilio client
  Twilio.init(accountSid, authToken);
  Serial.println("Twilio client initialized!");
}

void loop() {
  // Send an SMS message using Twilio
  String message = "Hello from your ESP32!";
  String recipientNumber = "+1234567890"; // Replace with the recipient's phone number (include country code)

  bool messageSent = Twilio.sendSMS(twilioNumber, recipientNumber, message);

  if (messageSent) {
    Serial.println("SMS sent successfully!");
  } else {
    Serial.println("Failed to send SMS.");
  }

  Serial.println("Waiting for 10 seconds...");
  delay(10000);
}
```

With this updated code, the ESP32 will now provide more detailed information through the Serial Monitor, including the progress of Wi-Fi connection, the successful initialization of the Twilio client, and the status of SMS sending.

To test the code setup:

Connect your ESP32 board to your computer.

Open the Arduino IDE and select the correct board and COM port.

Upload the code to the ESP32.

Open the Serial Monitor in the Arduino IDE (make sure the baud rate is set to 115200).

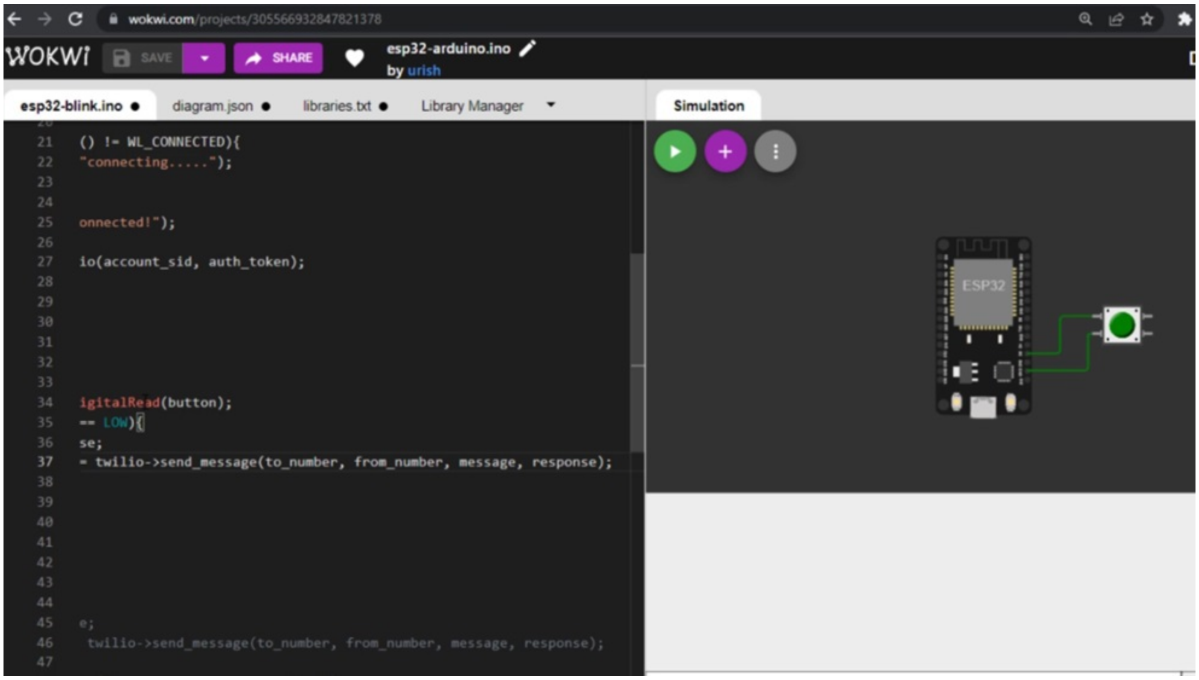
Observe the output in the Serial Monitor. It should show the connection to Wi-Fi, the initialization of the Twilio client, and messages indicating the success or failure of sending SMS.

Please ensure that you have replaced the placeholders with the correct Wi-Fi credentials, Twilio account credentials, Twilio phone number, and recipient's phone number.

Note: Remember that Twilio may charge you for sending SMS messages, so make sure to check their pricing details and use test phone numbers during the development phase to avoid unnecessary costs.

SEND SMS ON PUSH BUTTON

We are going to send the SMS by pressing a push button when the push button is pressed SMS who go through let's draw the security diagram let's set it up with what's in there. So connects the pins to digital pin two and the other one goes to the ground you can see the inputs so this is the code we used in Episode Five we'll be making some slight changes here we change the message details to be button pressed they'll come down declare the P mood for reporting as inputs Fulop, because we didn't attach any resistor so the next button is a hard to pin to then person states which is a variable we did not assign any value to it to in the loop so comment this out and then in both states is going to be digital read button but if both states is low so when it is low call the SMS function that will send SMS Audio Jungle the button state is high it does not mean it's just blank so it's only send the SMS when the button is pressed does when he's going to send the SMS so let's simulate it and see what happens and we have our mobile phone setup with connecting connected internet so now we're going to press the button to phrase a voice connection through Twilio to be able to send a message so we have received a message that is a message but there is no indicator over the side that the message was sent successfully.

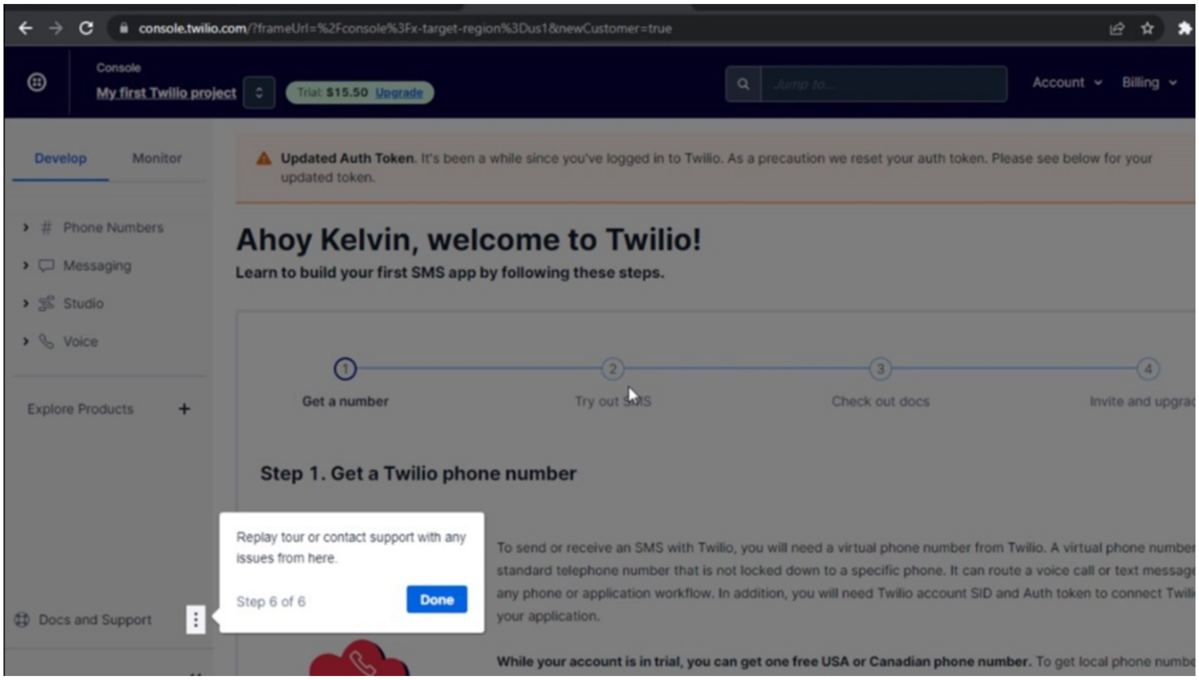


```
() != WL_CONNECTED){
"connecting.....");


onnected!");

io(account_sid, auth_token);






igitalRead(button);
== LOW){
se;
= twilio->send_message(to_number, from_number, message, response);







e;
 twilio->send_message(to_number, from_number, message, response);

```

So let's let's simulate again. So press the button now or press the button again. So in the code, we did not put a condition that if SMS was successful, that's if the message was sent successful. You should print SMS sent successfully. So we did not include that in our code. That is why when when we receive the message it wants to give us any feedback in the serial monitor. So pause the simulation. We'll go back to our code. We'll just copy the one commented there. Copy and paste it just below the SMS function so basically it'll remove the comments Okay, so let's simulate again this time around should get a feedback from the serial monitor when the SMS has been sent successfully so he's connected less press or push button right the first step is connecting to Twilio to send the SMS so we've received a message and you can see the feedback on this on this romantic sense message successfully. So if you press again, it connects again. When it sends the message we'll see the feedback on the serial monitor as well. So that is it for this episode. Send SMS on push boxing.

TWILIO SET UP

So we have tested our ESP online simulation, it's a work piece of software. So go ahead and set up our two accounts episode we are going to set up AWS accounts we'll go to the official websites we know that so this is the websites we'll click on sign up and start building swanky concern you're going to fill out all these informations here to start a free trial we already have an account which I'm gonna login with so I'll put my password so verify your identity so after signing up for your new accounts for when you're logging in for the first time, you have to register a phone number which you will be using so put in a phone number and verify so a code will be sent to my phone this is the code that has been sent to my phone so now I'm going to put in their code here it's 93223 and guys so meats so my Twilio account is set we'll answer a basic question here guys in the SMS regards a Latin expression with only go so get started with two so skip these we don't need all this done.



Updated Auth Token. It's been a while since you've logged in to Twilio. As a precaution we reset your auth token. Please see below for your updated token.

Ahoy Kelvin, welcome to Twilio!

Learn to build your first SMS app by following these steps.

Step 1. Get a Twilio phone number

Replay tour or contact support with any issues from here.

Step 6 of 6

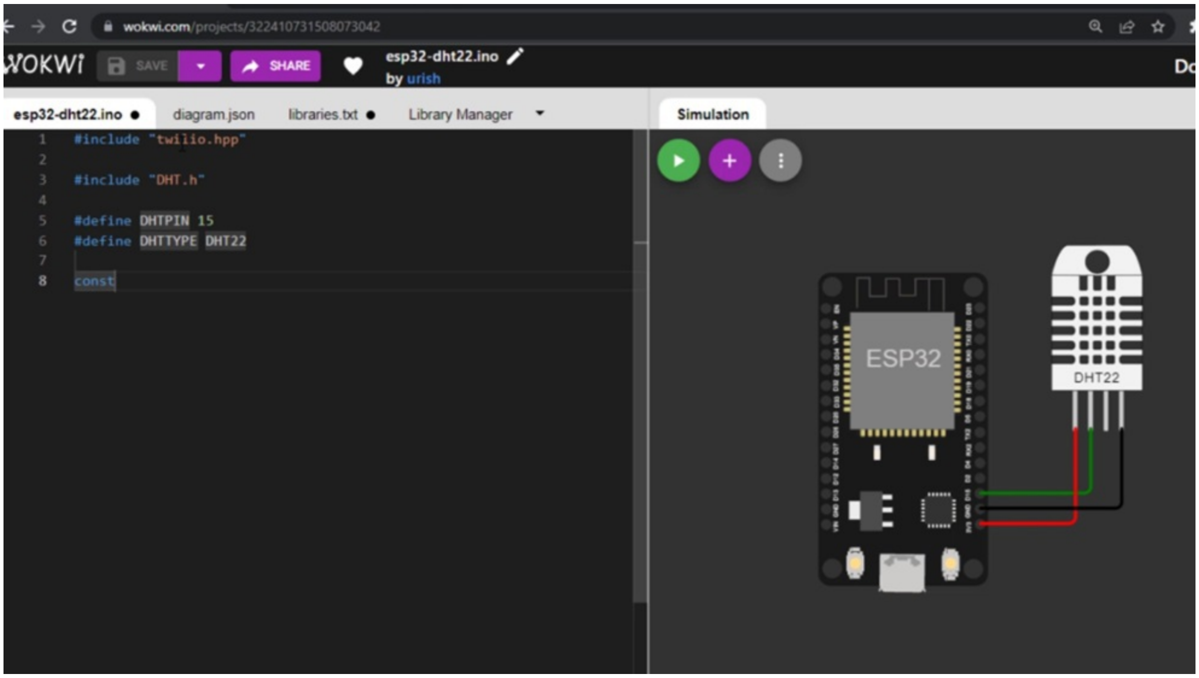
To send or receive an SMS with Twilio, you will need a virtual phone number from Twilio. A virtual phone number standard telephone number that is not locked down to a specific phone. It can route a voice call or text message any phone or application workflow. In addition, you will need Twilio account SID and Auth token to connect Twili your application.

While your account is in trial, you can get one free USA or Canadian phone number. To get local phone numbe

So we're Intuit riilu accounts. First we'll go to phone numbers so that we can purchase a number manage by a number to search for the number that has SMS compared to these so this one has SMS we can see the send SMS so click on Buy click on Buy scroll down to where we have the buyer will buy this so we bought this now we can send SMS using this number that we bought so let's go back let's go to messaging get setup start setup. Message Service name will just give you just a very simple name and give it just with ESP alerts, ESP alerts tests and request messaging service. So this is a number who add this number, the number we purchased. So then this is our account s ID. We have to copy this out on steroids so that we don't misplace it. So open my notepad paste it in as at least I'll do my authentication to how copy open Notepad and paste so try SMS you after you get to this point where you're trying to send SMS so body of text can be anything can be hello from Hello from my ESB testing Hello from my ESP test so let's send SMS so the SMS has been sent for let me get screenshots from my phone so that's the SMS I will receive on my phone so we will send the messages a few more times and see how it's comes in in the phone. messages have been coming in into my mobile phones. So that is three live setup for this episode.

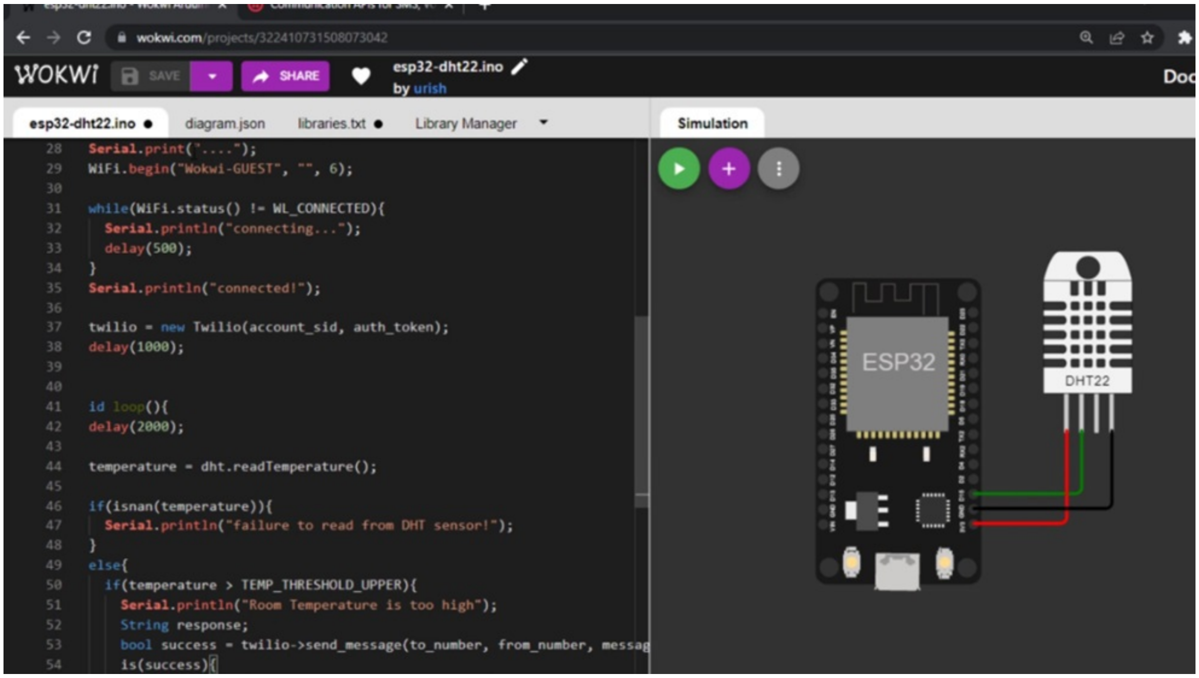
SEND SMS CONTROLLED BY DHT22

The DSC 11 temperature limit it says so we use to control the SMS they'll be sent to our mobile phones. So when the temperature rises above a certain threshold, which we are going to set, it will trigger the SMS. So let's begin we are going to play the existing code here. So that we'll be able to improve our own code.



```
#include "twilio.hpp"

#include "DHT.h"

#define DHTPIN 15
#define DHTTYPE DHT22

const
```

So let's install library flay libraries installed in three low ESP clients library first, then we'll install the DHT sensor library so let's begin the code the trigger library which this will include the the SCSS fabric in after that we'll define a DHT pin which is connected to pin 15 the DST type is DHT 22 that we'll be using then we're gonna declare the temperature threshold for the copper value which is 60 and lower value us say or okay let's leave it at that so fluids temperature that is you declare a variable will now send anybody to eat so now the account s ID which we copied from our dashboard and its remote software so be careful not to make mistake is best to copy and paste it's just type in my notes then my authentication token So next I'll declare another variable the number which I'm sending from as the number which I purchased on the to perform and put it here and then a number which will be sent to Shipman which that phone number same as on the platform. So, the message contents it can be anything you want. So also be room temperature is too high. So, for a void setup initialize the serial monitor, initialize the temperature and humidity sensor then initialize the connection. Lupu set a delay in between the release just two seconds, then temperature is equal to DST with temperature temperature was the fluid variable we declared earlier. So, if there is no temperature within which to show that Syria does prints fail to read from the HD sensor.

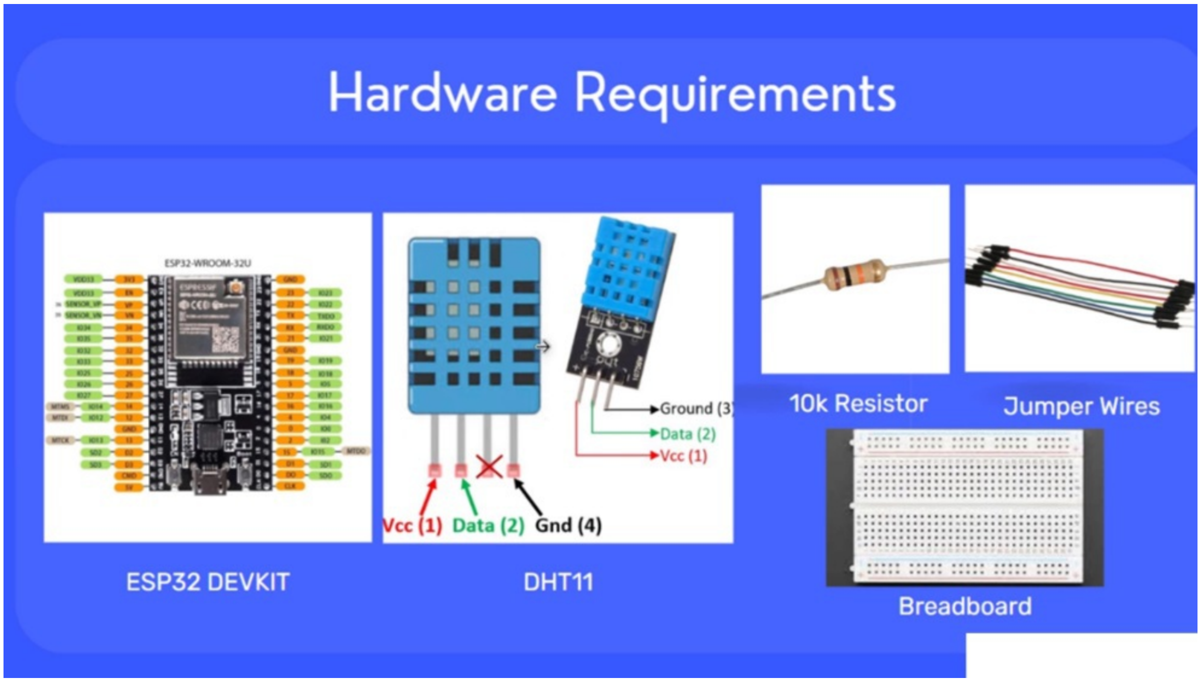


```
Serial.print("....");
WiFi.begin("Wokwi-GUEST", "", 6);

while(WiFi.status() != WL_CONNECTED){
  Serial.println("connecting...");
  delay(500);
}
Serial.println("connected!");

twilio = new Twilio(account_sid, auth_token);
delay(1000);


id loop(){
delay(2000);

temperature = dht.readTemperature();

if(isnan(temperature)){
  Serial.println("failure to read from DHT sensor!");
}
else{
  if(temperature > TEMP_THRESHOLD_UPPER){
    Serial.println("Room Temperature is too high");
    String response;
    bool success = twilio->send_message(to_number, from_number, messag
    is(success){
```

So else So first if condition if the temperature is above the temperature threshold above which you set assisted Serratos brain room temperature is too high and then the Send Message function for you if temperature is lower than the temperature threshold lower Seattle's brains room temperature is warmer it was sending SMS it was print is normal so the delay for the messages will be placed at once that is it for the school next we'll set up a simulation platform alongside our member phone so my phone is open by the side and we are a simulation is coming up okay so let's simulates connecting to network before we can see the temperature is below 60 And the sprinting temperature is normal so let's increase it above 60 So what increased above 60 It initializes the SMS system is connecting to two the census this really gassy the message has come into our mobile phone. So openings room temperature is too high that's it so let's take the temperature let's take you lower so after sending those messages come back is kind of go back to room temperature is normal. So the SMS is not submitted in regards to the level or how hot or cold the room is.

HARDWARE REQUIREMENTS

We will now be looking at the hardware requirements for our LC weather station projects. The first component to need his our microcontroller IDs and we'll be using an ESP 32 dev kits. This microcontroller unit was chosen because it has Wi Fi capabilities and is easy to use this microcontroller receive data from the sensor then when it while it is connected to the internet with its Wi Fi capabilities, it sends this data online then second component we'll be using is a DHT DHT level sensor it is zeros so it's essential that reads temperature and humidity data then we have our resistor visit a thank you home resistor it's been needed for the DHT sensor to pull it off while we are connecting it's when it is connected to the ears with its two development kits.



Hardware Requirements

ESP32 DEVKIT

DHT11

10k Resistor

Jumper Wires

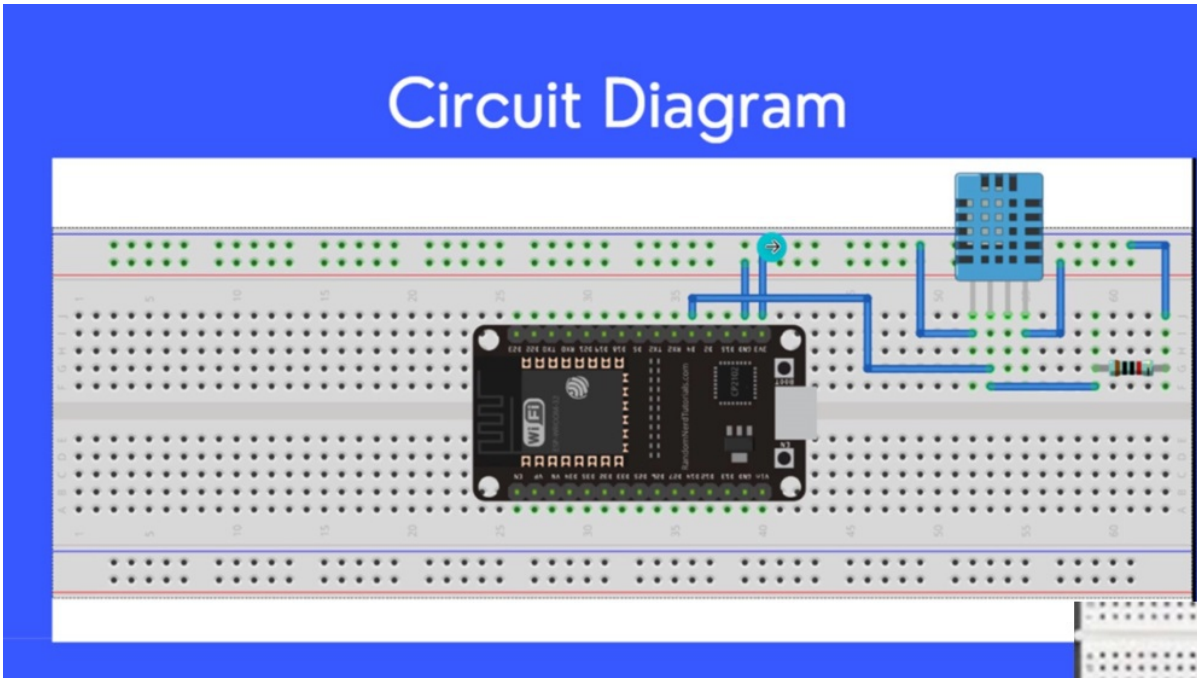
Breadboard

He was just though is a device that resists the flow of currents it should resist the flow of clients then we have our jumper wires are jumper wires connecting wires each project each electronics projects needs wires, his knees connecting wires. So that each components will be connected to each to accompany then we have our grid board our red board serves as the piece this is where we place in our components which would then connect to each other using our jumper wires on board as I chose last use connects which components each other then let those expand sheets on each components The first is our ESP 32 In this case we're using here suited to the kids C DF this delicacy is a development board that houses DSP controller alright. So, there's my controller on his own is used it has Wi Fi and Bluetooth capabilities then it is out this is it is our only development mode that makes it easier for us to use it this our development is good with the picture shown you then so what is a Mac controller at first Mr controller is a small computer that contains the processor and other peripherals such as memory and case of DSP it is to Wi Fi and Bluetooth module. So microcontroller is a tiny computer is more computer that can be used to control order components in circuits because we use controller components in the circuits. So my controller is a small computer that that's asked peripherals such as memory does Mr. Ram our room with our storage to then write program amo controller writes good, then we send it to my controller, then this micro controller serves as the breed of our own electronic circuits as renovar SATs, which he then uses the pins, these pins to communicate with the outside world. The first pain is to do with repin is that the microcontroller is powered, it's 3.3 volts, or five rules. Then we have all ground pin which we use to connect our controllers to the common ground to secrets. Then we have other pins, the GPO in there, we call them general purpose input output pins. So they can be used as input fields or as an output to communicate to the external world we have our USB ports here these years because we used to connect our controller to our computer for programming and also it can be used to pull out my controller. The main controller comes with the USB ports can also be powered through the power pins here So, our controller is the brain of the system via electronic system then we have our DHCP 11 cents. So, are these 11 cents sensor is a commonly used temperature and humidity sensor does as an imbues it with microcontroller to output values of the temperature and humidity or serial data. Alright, so what that means is that our sensual as a tiny computer inside of it, that's after sensing the temperature and the humidity of the environment, he then converts that temperature and humidity into serial data that is then transmitted from this second pin to our controller, this sensor to ask to be powered using the VCC pin. And the ground pin is a four pin device, the first pin is four pi, it has to be connected in the support 3.3 volts in the case of the SEC us then it's come it's near there needs to be connected to ground, the same ground device controller is connected, then this third pin is MC not connected, we don't use it for anything. So that's why they cross around it's this is a model these are saying so there are models that we can use essentially this model just as a vccp ad talking on the ground pain, the false pain are removed as you can see the module is created from scratch so that this model has four pins, then a beauty and pull up resistor is connected to the Moodle we'll be using a sense of our projects essentially itself, not the Moodle. So we need to we need to collect a pull up resistor. So the a pull up resistor to the data. Alright, so the next component we'll be using is our pull up resistor. As I explained earlier, a resistor is an electronic components that is used for losing currents and offers resistance in the same use. So they are used to protect our electronic components from too much parents get rid of resistors so of fuses resistance and resistance in our case we're using a fixed resistance resistor. This is in our case we're using a pin you know home is is those resistance are measured in kilo ohms resistance are measured in ohms. So in our case we're using a thank you. This though, will be used to conclude connecting origins but didn't happen. Then, one time now we're gonna visit data and then second time now we connected to power VCC.

SOFTWARE REQUIREMENTS

So I will do it on stage on projects, we can ask the software requirements The software will need for the projects. The first software is Arduino ID. Arduino is an electronics prototyping platform. They have hardware, and software products, most of the hardware of blood renew hardware and check them out. Then the software we use to program their products is called the Arduino ID that is the Arduino independent development environment. The Arduino ID can be used to program Arduino projects can also be used to program order booths, for example, our ESP 32 Dev Kit C, which we'll be using for our projects. So they are doing already computer programming, then we need a tin speak account. Since pig is an IoT analytics platform that can be used to aggregate to analyze and to visualize data. So we're using our speaker this week to aggregates data develop where from the sensor data is sent through the years for this to stay in balance. So is this peak in this case, then we are when we are getting data we visualize our data will be visualized in two fields the temperature, then the humidity will be visualized, then we'll use the to speak to speak platform to analyze our data analysis to our torch requirement only the Matoba account to create it in speaker counts. Matlab is a most product to use Matlab to program to do the analysis to speak uses MATLAB is a programming language that will allow us to learn I'll explain his level and MATLAB GUI Laguna democra work then we also need a cart platform to design our circles card with confided design. So we need a card platform to design our circuits and this case in this case we'll be using fritzing Fritzen is a card design platform it's popular for electronics prototyping and that we'll be using for our projects so let's go to the Syracuse so this is our SEC us on the SEC you can see our yes with it's so good on a bread would you answer is it's so good on a very good then can also see our sense of as boots with ES with it's too good.

The false pin on DSP this word is three V three pin that means 3.3 volts pin three V three in is the pin is even does go to 3.3 volts after we've connected our ESP this should yours we post to a computer program this we use the year reports to power is with is two three with 3.3 volts to the output from distributary pain and other reasons power our sense of right so, the three victory pain has been connects a jump power from three to 18 to an ozone power and a result our connection on the grid after connecting our three victory pins to this these points are connected horizontally this first horizontally connected together all these points on the first horizontally line are connected together as in these the this next point is connected to allow zone power points the points we connected our derivative in the five sections on the right hand side then we have our ground pin secondaries on our ground plane is common ground.

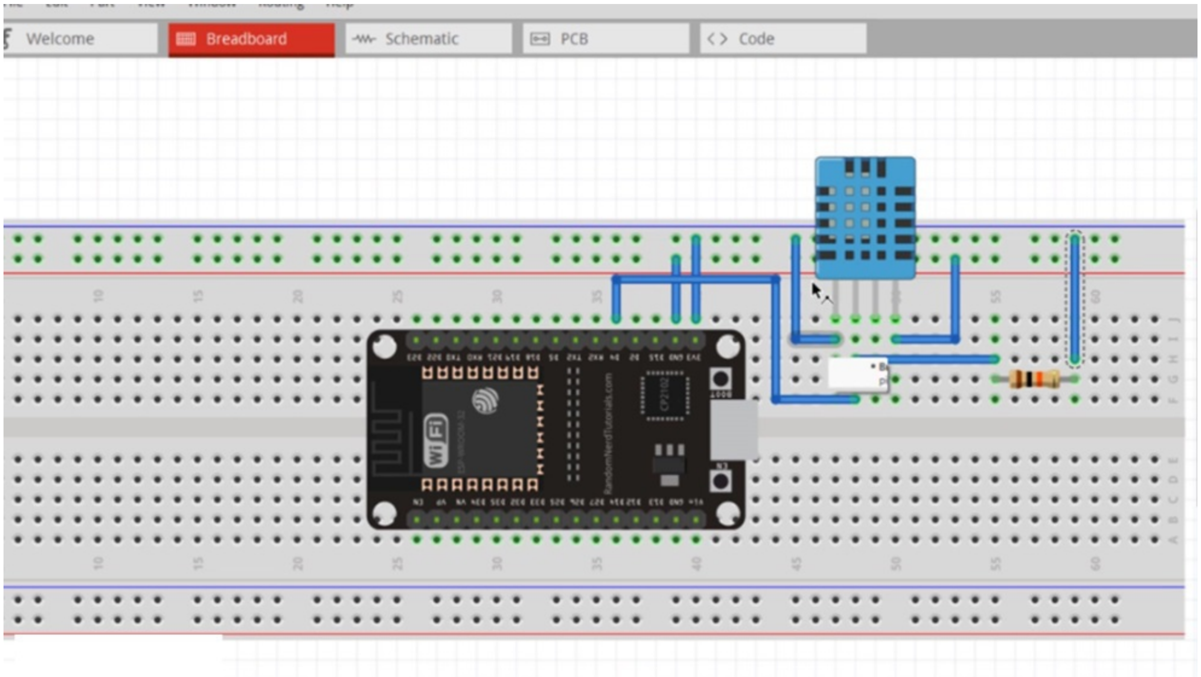


Circuit Diagram

We need this for our electronic circuits. Then it is connected to other points on this on second horizontal line. Then we need to connect our sense of ESP Moodle or ESP would so as you can see data that was pin two on the sensor is connected is connected to pin default on the ESP board. And we need to also power our sensor sensor is powered by connection is the three V three, a 3.3 V three line, then we also need to add it also is connected to the ground. That is what is on second connect, it's what connects the false pain of the sensor to the second line. And the second thing, as we've said is going to be as the data is collected as a default on the ESP word. You also need to pulito what pooling of means is we will connect the resistor to the to that pin, then connect the resistor the second time now load resistor to our power supply in this case is three V three is 3.3 volts. Alright, as you can see, the resistor is connected to the second pin on the center. So one tanagers there's going to be one second terminal is connected to the three V three line. So how does our projects work? This is a simple block diagram of our projects. We have our dataset so we have DSP data to have this peak is so sends data to the US with this sort of temperature in the 30 years within which reads it's then the ESP 32. After reading this data, send it through the environments to arts and speak. Route. We program our Sousa saying today that autistic arthritis has been connected to the after this week as been as received our data since big then visualizes the data. And then we use our MATLAB code, which will drive to predict temperature and humidity to speak with our data and visualize it then can do analysis on the data can analyze our data, which is very, very helpful.

CIRCUIT DESIGN

We'll be designing our 64 years weather station today using freezing. So the first thing you need to do is to download freezing onto solids from online from the official website. After installing, you'll see this page then we'll need to create a new sketch to create a new sketch you go to files, then you click on new it takes some time to load then would have this this is a breadboard that already placed on the sketch. So next thing on is to do so our components only one of the first components we need is our our microcontroller unit, we need our control lines we need us and so on. So to be components you come here. So as you can see, the first component you have here is a 220 Ohm resistor and click on it, you can then add the properties of resistor after placing it on the breadboard. You place it on your breadboard then come down here and add provides is a would need a thank loom register. So we can click Type 10 G, U turn that sounds nice. Thank you. All right, and you notice that the color bands on the resistor have changed about us is that color boundaries have changed. Also need our ESP did so good. So click here to search for components. And you can go around here to see other components that we need. Alright, so you see Arduino classes platform components, as it seeks to do components in the component freezing as well the components that we're going to use for our projects will need an ESP 32 Because that's what we'll be using for our projects. Freezing originally does not have an ESP this word. So what you need to do is to go online type ESP tends to freedom you then you download a zip file or dot sav so when you've done that you will come to paths then go to add to the my pots. Then when it gets to my pots I already have some components added to my path and Camilla you right click then can you click on imports. When you click on Import you can add watch details as it be called out as a PS a visit could be any extension then you add it to your parts. Alright so I already have my ESP 32 added I also have my DC level noted so I'll click on this on the ESP 32 and drag to devote Alright, so this is my ESP test. So the next thing I'll do is to add my sensor so the words you can rotate is by clicking right click on it then routines is one easy degrees so that it's Pattana will want us to complete our sensor like this more for ease of visibility remember this is the original version then please work alright so we're gonna look at our connection have said well this will create stakeholder groups. The first thing is would connect our three V three pin a power PowerPoint so as you can see, the green line indicates that all these points are connected to each other is where any increase eldest points are connected to each other. So we can shoot our free ESP tends to work Elizabeth Ford, delete this and reconnect it one more time. This is much more data. All right, then we can connect our ground pin our ground pin to the second or isn't our life As you can see, alright, so the next one needs to do is connect the data pin does data pins to data pin pins to or it is to default on our ESP test word alright so we can do this then click again so we can connect we can connect in de tu su so connects up in default to into of our DHD level as you can see you so to make it look better with an admin points to do that you let's click on the wire so I've been points so do this god I'll gain points here and again points because pain points so this looks much more data than before. So I ended up in the for to the data set so the next thing you would need to do is connect the sensor to the park in old cells or to 3.3 volts so to do that would connect this click on that points then connect is too hard to revive through sight of admin points to how I ask who do the saves in foreground then after doing this, we need to connect our resistor from data pin one I'm not going to get to tapping in second time now we're connected to the 3.3 rooms. So to do that how to do this click on that let's click and connect is this then let's clicking Connect is to this point perfect.

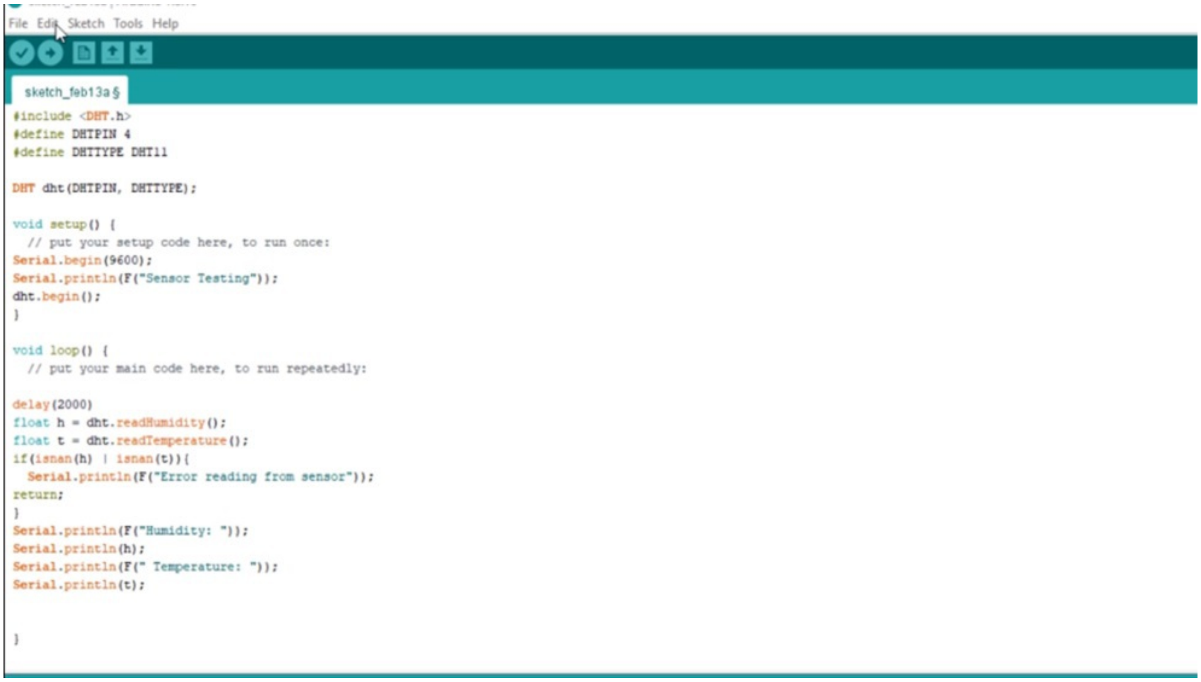


So this is our sake us let's go through the SEC is one more time you have is with us to the which will be part using a USB pin USB cord to the reports then we have our D for having default which is connected to the data pane of the sensor. You have the first peanut sensor being connected to 3.3 volts then we have ground pin connectors to the ground line and the ground line is also connected to the ground pin over ESP this the 3.3 rolls is also connected to be three point the 339 of our safeties and also remember this is our attempt to restore how we connected the data to the hub and second panel hasn't connected to 3.3 volts Klein as you can see so this is our sake use for projects we can save your project and name it whatever you want while new rates I will do it as station press control as their name it's IOC weather station then stay save your projects.

CODING 1

Now we programming for the IOC Weather Station project. So the first thing you need to do is install your Arduino ID which I have done here as you can see and then the next thing you need to do is to install some libraries that will be needed so to install I will just go to tools then click on Manage Libraries are up for Manage Libraries in a poor poor poor show. Isn't it take some time to loot is it sort of this the list of installed divers? Yeah, I think sometimes, but after some time it finishes loading then after I finished school in the search bar, the outs they've seen speak c c h i n g speak is the E AK. So as you can see, I already have it installed, we may not have it installed. So the install it's due over around it then you'll see uninstalled to install Ashwin is the second click on it to install it in fact sometimes install then you also need to install DHCP here this library is already installed, which is usually is so you type DHT right take some time to load after some time, you can scroll down and look for DHT sensor library. uninstalled us we already have it installed. So when we installed as we would go to write our code to take our clues your lab manager for Paul Allen start tracing code. So, the first thing you need to do is to add the first thing you need to do is to include the digits. So, to do that you type ash include the H c H SO WHAT IS SO including is this line ad ADHD DD HC leverage output. So, next thing we need to do is create constants for this. So the first constants will create is to create a cost and to type as defined then the first concern to place is DHCP. Then let's call it four as we see we connected our DHCP snooping for the data in addition to paying for our bird good all right. So this type is next on Stan to define to the DHCP level in our case raising this dilemma confused for DHCP twice so also so the next thing we need to do is to define I did suffer so we DHT in this is outputs DHCP DC diag alright so the next thing we need to do is add the structure of this DHT add to stroke filter structure. So we are creating it's with this function he added it is with GCP and Aspen for the indigency type as DC live Alright, so go to our setup now and we don't use the serial monitor so values serial dot begin would select the baud rates radar switch on cell say Then serial information that will be as 9600 Then the next thing we'll do is we can test this so let's use this central force routers so it's ESP to see who is working then we'll do Hussein auditors ready so the internet's they need us artists. Alright, so there's something nice to do nowadays. You can print something on our cereal when you throw cereal those prints in TVs 130 Those are insane. Okay, so we've gone years prints line further. So we can use if this is Paul's format sets and this is of your sending send so they this alright so the next thing we'll do is to type the HC to begin So this starts our request for the telephone so we're able to do we've set everything up so we can go to the moon to the moon and so ganglia and listening then can receive the data that says the shell Central's so put it in variables, so just a floats so yeah, taqueria treat is this small one to enforce foods let's get our units in our name is H Let's say float h is equal to DHT does reach you immediately is the function you use for this to do these things then floats see for the individual DHD those read Thanks Rachel is an average so if you want your temperature this this will give us our temperature in Celsius. We want our temperature in finites if I just type true into the as a parameter to the function and it's a you have been brutal as finites police yourself shows so we have to check if what we'll be receiving is our floats are not smooth Eva data is valid or invalid. So, to do that, we use an if statement say E is not this is not this is not a number. So, the first thing is our immunity and second is out and virtual.

So if any of them this by means all so if these are this is not a normal while you also have on wants us to print the line print the line serial when those print serial those print ln to print a line dots error reading from sensor reading from sense of then to make sure nothing of causing in Paragon our loop function so examples return here. So valued is then if our orders are valued, whoa up on demand prints our serial dot print line. So CVT I started our immunity then Congress prints in the October from since all of our immunity from JSON, you can do this then the next thing is to print the average of those prints again I'm pretty sure all right, so temperature received it as you will see nice use of space for the training and receive it as it costs. So you can't do this then we can also add a delay, so he doesn't print every single time, our sensor reads every one second, every one second on Facebook, I'll make it two seconds for younger kids. To do that you add a delay. So we'll have two seconds, that's 2000 milliseconds, which is equivalent to two seconds. Alright, so I think we are done with the code for interfacing or sensor with is with this.

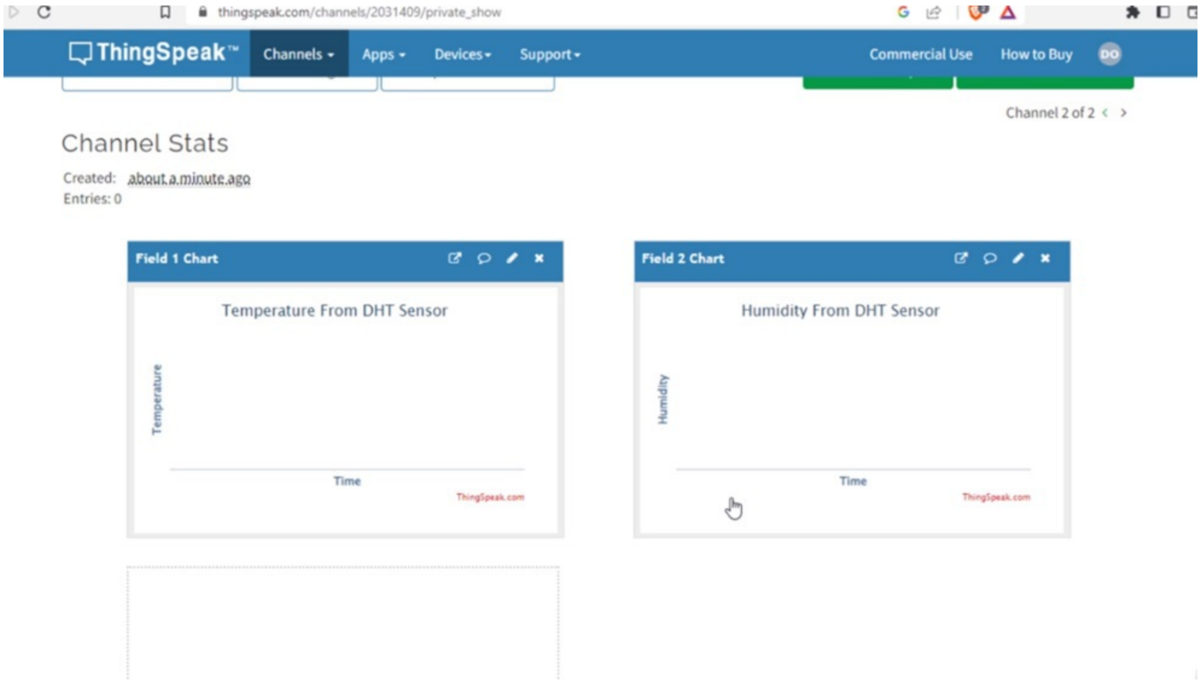


```
#include <DHT.h>
#define DHTPIN 4
#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  // put your setup code here, to run once:
Serial.begin(9600);
Serial.println(F("Sensor Testing"));
dht.begin();
}

void loop() {
  // put your main code here, to run repeatedly:

delay(2000)
float h = dht.readHumidity();
float t = dht.readTemperature();
if(isnan(h) | isnan(t)){
  Serial.println(F("Error reading from sensor"));
return;
}
Serial.println(F("Humidity: "));
Serial.println(h);
Serial.println(F(" Temperature: "));
Serial.println(t);


}
```

So we can check that here to upload the code because they need to do some verify compiles initially, there was no good you have to see its fancy with us. I'd say mine on Dancing with anyway also stem how serious this thought jagriti new food is so new, the IOC one. So how serious. So since I've said it's compiling the sketch down, so give it some time to do that there was an arrow, you should have put semicolon after delay for the value in place. So it should work now. Our code has compiled successfully. Then we need to upgrade the code size which is what you need is to connect your years before your USB cable to the years before the vs that's ESP test. I've heard renders Bucha to then select your God as angles both manager then select your board as ESP 32 ESP 32. Dave Good to see them. Alright, it's actually on let's it's ESP 32 So it's already installed. Yeah, studies and after that's you split your vote as if that's true, then click to go to two. Then choose ESP ESP 32 As your vote after doing this you come here, click on Upload and you upload your crude remote so after you're good as done has been done uploading, go to Tools, then click on ceremony and as you can see the temperature and humidity machine temperature and humidity shining alright so immunity is possible and 5g is in percentage economy this does change it.

THINGSPEAK SETUP CODING 2

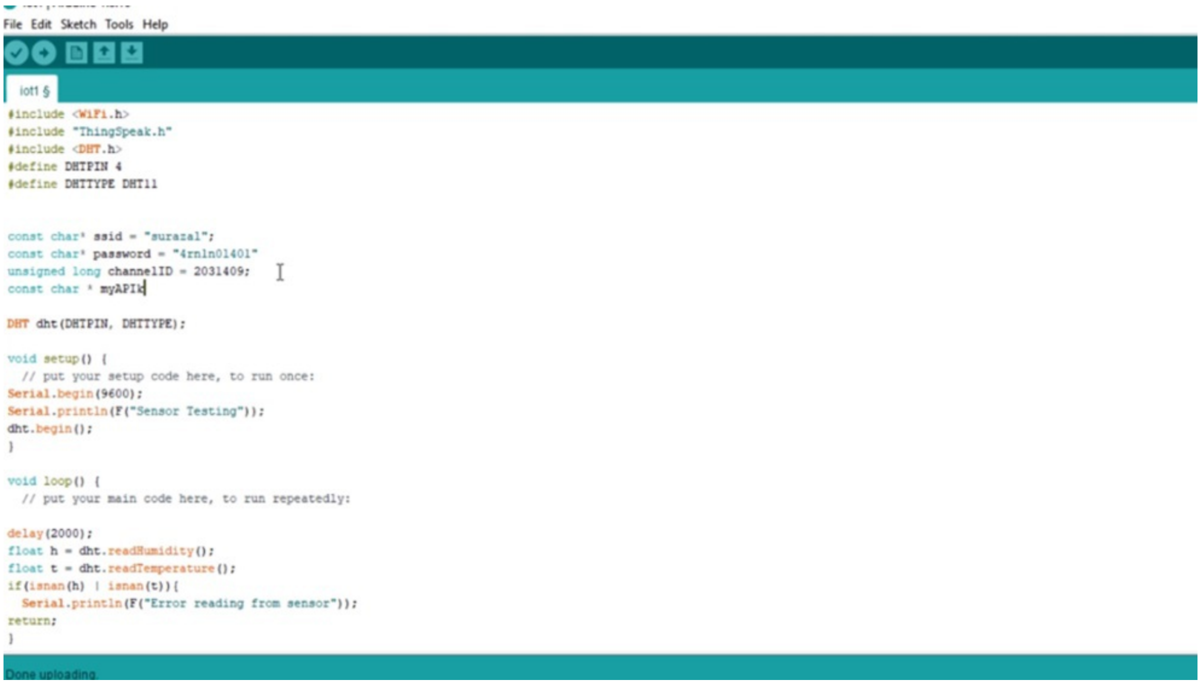
so the next thing you need to do is to open you to open your browser whose it is because then you create an account that creates an accounts and you're logged into the account you'd see this page. So what I want you to do is Hayes's page you click on new channel and you create our channel listeners I will see soon then scribe it hundreds if not Iran describe it as a you can describe it as an old see weather station with Asian then forcefield we fuse our are the data I want to upload to defend speak so the first few dozen temperature so you click on this one is one field Salafi last name is alright, so you scroll down then you can click on Manage bankers meta data is just on all sides. I can do little things, although save the channel and click on to continue. So there really are two charts. The first is temperature deserve all these immediately I was gonna see anyway, as we can add this chart once the title reaches to the air temperature from this so our x axis let's make it time for visionaries, it might be days so you can change the color to any color you want using exclude or are you alright so let's see this.



Channel Stats

Created: about a minute ago
Entries: 0

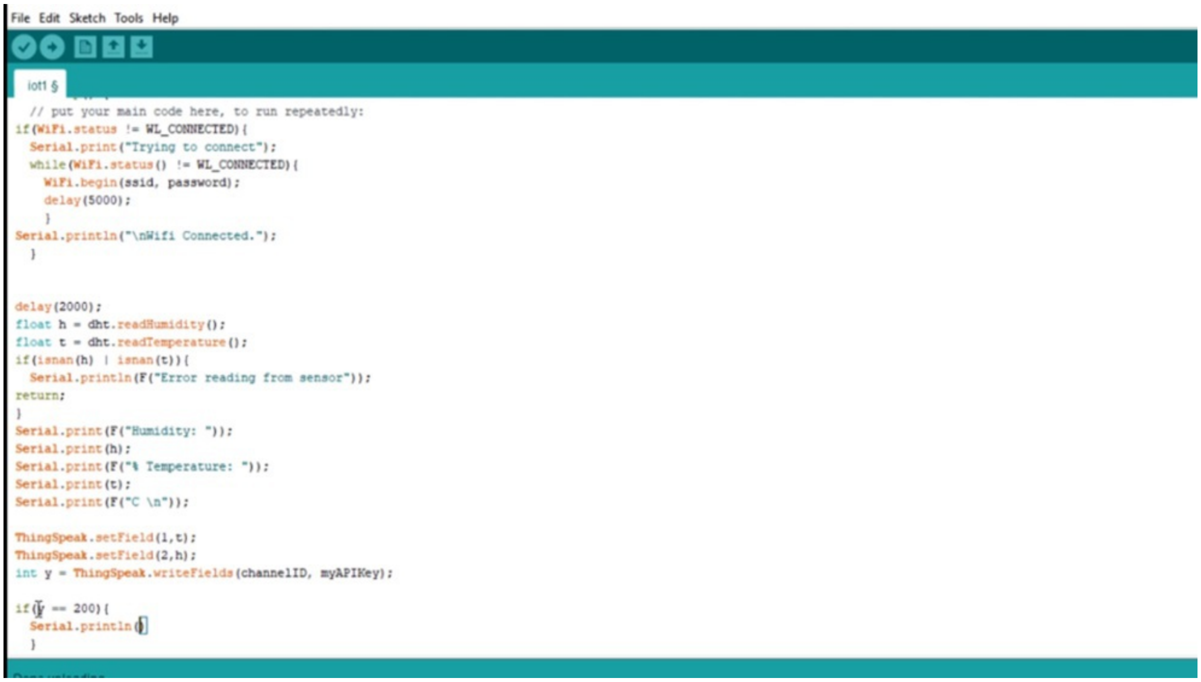
So we have temperature from this so let's do that for these to click on the pencil. icon in the industry we really see how easy this is so it's time by axis are you really gonna change people we want to change her so we have that saved who go to output Janessa. Our good is to connect our us with is the internet connected Wi Fi and send our data from ESP our data from sensor sensor readings to others. So let's do that now. So let's start writing the book. was last time we printed our house was line by line for better format so let's print them all on semi house put for the reason that Rachel as you have seen here I removed the LM LM to just be serial dot print zero just give us with our formats and makes it look a bit official initial. So what we have here is we have our outputs here. So this is the temporary Jamie's reading and temperature it looks much more better and isn't anymore we did before. So let's go to connecting our ESP this to the Wi Fi. To do that we have to add the Wi Fi that includes the Wi Fi library to our Wi Fi which included will do ash include Wi Fi who would use our brackets then say ash include Wi Fi. All right, then we also include our third speak library. Hopefully you installed it from the library manager how it seen speak the each Alright, so we've included author and speak clever. Who Janessa needs to do is to add our the parameters the SSID for the Wi Fi was connects. So in my case I'm using so Rosario su Raza doesn't remove the Wi Fi owns connect to type EA type in your Wi Fi name. So you should have a hotspot or analysts work on or a Wi Fi connection on. For use Connect Yes, because since then you would add the password so in my case is a password is a combination of numeric characters. Alright. So then the next thing is, yeah, you should add your password password to your Wi Fi. Then the next unit is our channel ID, you know, we've created a channel on Teamspeak. So it has an ID. So it has an ID so we can get the ID opposites here. So let's check the channel I used to get your channel ID scroll up, then you see your channel ID on the left hand side. Then you can copy it and paste it in your code. So let's paste it here. Alright, then the next thing we need to do is to add our API key, our writing API key. So right to the channel, you will need an API key from the websites so oh my god, everything is numbers in a mine my API key API key to get it from the salespeople tries to get is you go to API keys you click on is then this is right PPI Oh copies and used alright.



```
#include <WiFi.h>
#include "ThingSpeak.h"
#include <DHT.h>
#define DHTPIN 4
#define DHTTYPE DHT11


const char* ssid = "surazal";
const char* password = "4rnln01401"
unsigned long channelID = 2031409;
const char * myAPIk

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  // put your setup code here, to run once:
Serial.begin(9600);
Serial.println(F("Sensor Testing"));
dht.begin();
}

void loop() {
  // put your main code here, to run repeatedly:

delay(2000);
float h = dht.readHumidity();
float t = dht.readTemperature();
if(isnan(h) | isnan(t)){
  Serial.println(F("Error reading from sensor"));
return;
}
```

Done uploading.

So I've copied and I'll paste this here so next we need to create a new instance of the structural say Wi Fi clients clients Alright, so after doing those will use was set up and set the Wi Fi mode to be Wi Fi station underscore s see that as your Wi Fi mode. Then the next thing you do is you begin to speak clients did sense with clients. So speak the words begin April's clients is diving clients then we'll go to all our meal mean then why did the code supinate to the Wi Fi value because connects to the Wi Fi Alright, so to do that, all we'll do is we'll check the Wi Fi status will tell you if I suppose so that will be those tuples so if it is not connected, if it is not connected now it will be on the score connected is not connected. So what we want to happen who tries to connect with Alright, so this connects it's huge. Liz indicates there are transparency throughout those brains. Then regarding foods attempting to connect or trying to connect, trying to connect then we will wow Do Wi Fi starters he's still not connected while you still open a page would begin the Wi Fi so Wi Fi does begin so you put your SSID as we already say SSID you can type any you can find this indirectly is ID and password we already declared the variable or whoever inside the password then we'll do it for five seconds to refinance. Alright, so the next thing to do is to indicate that's connected gonna do those early serial those brain cells now you'll see Wi Fi calling then the next thing we need to do is send our data up as I was going I said I'd send data to the same speak right so to do that we would type in serial so to speak since we the sets huge then we're gonna put our first few distributors or one commodity then the second field then speak that says you are viewed as ever viewed is immediately Alright, so then all right, we'll type on these those is to is to write the fields ready sets the field so we need to write it so the channel ID which you ready says the right API key entered is my API key Alright, so since we've done this, nobody would stop it was we have to make sure our data is made sense we have to make sure that has been sent.

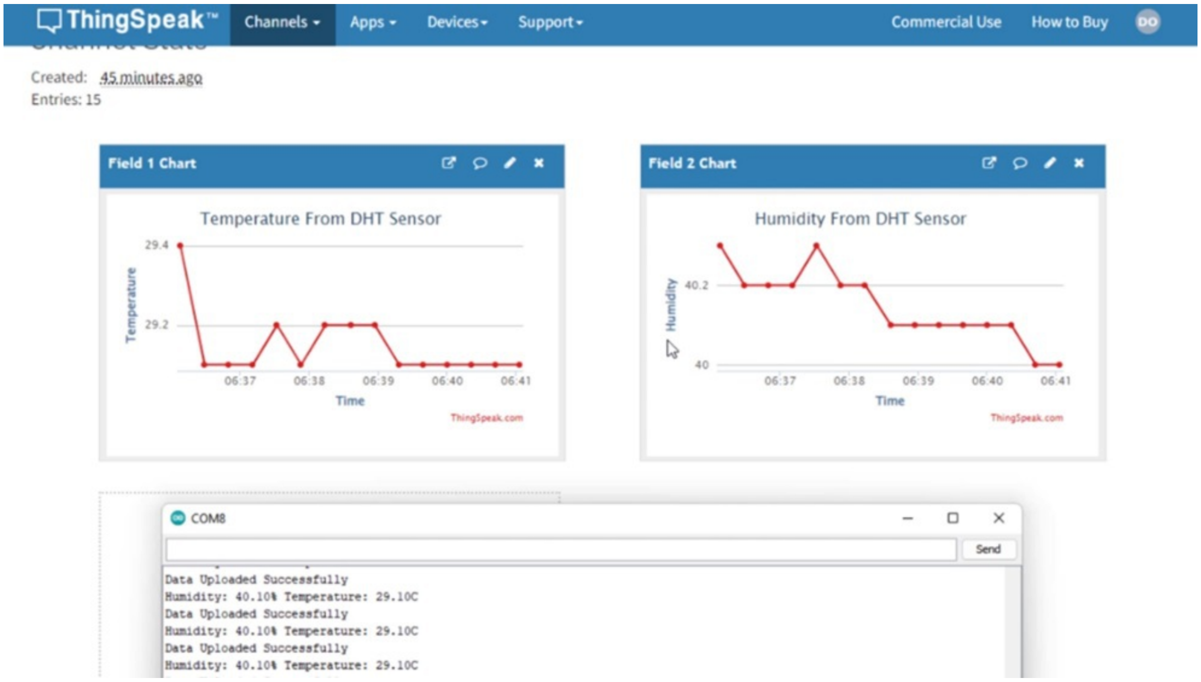


```
  // put your main code here, to run repeatedly:
if(WiFi.status != WL_CONNECTED){
  Serial.print("Trying to connect");
  while(WiFi.status() != WL_CONNECTED){
    WiFi.begin(ssid, password);
    delay(5000);
    }
Serial.println("\nWifi Connected.");
  }


delay(2000);
float h = dht.readHumidity();
float t = dht.readTemperature();
if(isnan(h) | isnan(t)){
  Serial.println(F("Error reading from sensor"));
return;
}
Serial.print(F("Humidity: "));
Serial.print(h);
Serial.print(F("% Temperature: "));
Serial.print(t);
Serial.print(F("C \n"));

ThingSpeak.setField(1,t);
ThingSpeak.setField(2,h);
int y = ThingSpeak.writeFields(channelID, myAPIKey);

if(y == 200){
  Serial.println()
  }
```

So to make sure our data is in sense this is why would give us response from speak evolve as Vincent would get honest if if we could tolerate so that was if Y equals 100 Then we should say channel has been updated or the past and successfully channel data since successfully data uploaded successfully says as fully as it did updates uploaded successfully then else if you're an arrow Good question we do the prince was Euro could our greens Austria monitor G arrow good. So program uploading data. URL good H TT E is a barracuda is here it is up arrow who then if odd do this method of adding numbers to your cereal households alright so this should work.

MATLAB CODING 3

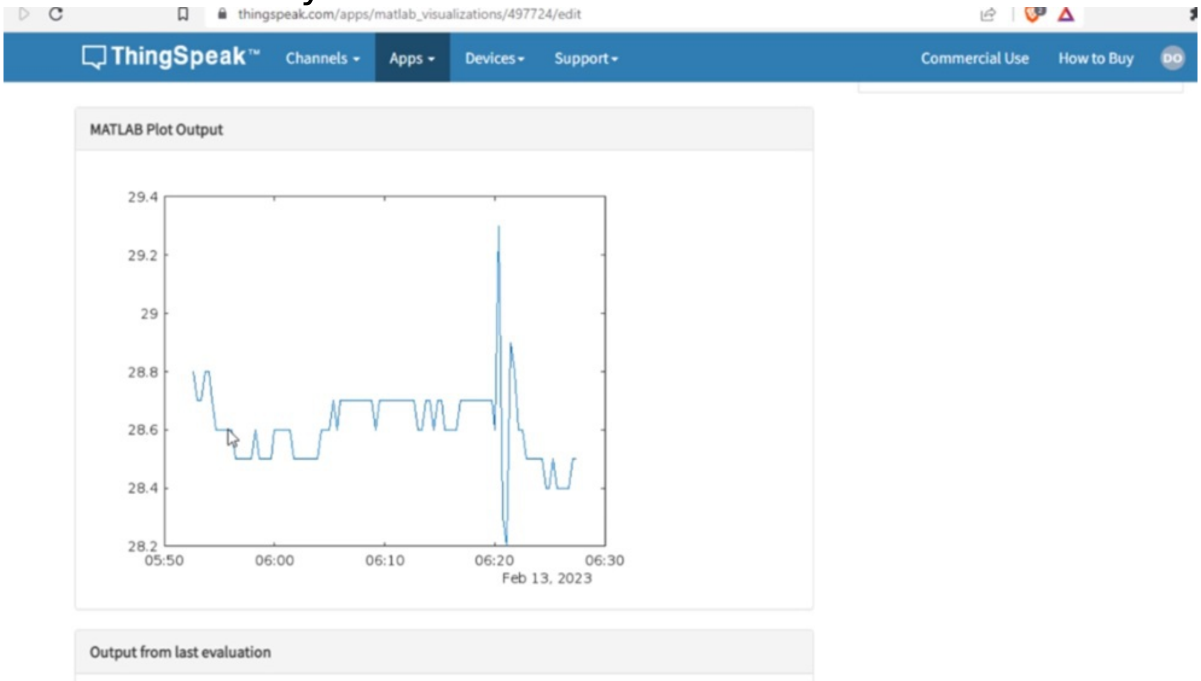
So we add an area that should output parentheses, parentheses, it should compile successfully now. So cool has compiled successfully, before we go to speak. To check it, we have to increase the delay throughout 20 seconds. Speed does not allow uploading for less than the frequency of users. So let's increase our delay to 20 seconds. So as you can see, our data has been uploaded successfully. And if you go to yours and speak, as I haven't read it out show as you see the issue or the trends, good, the next thing to do is to visualize the data. Next thing to do is to analyze the data, I use it to predict.



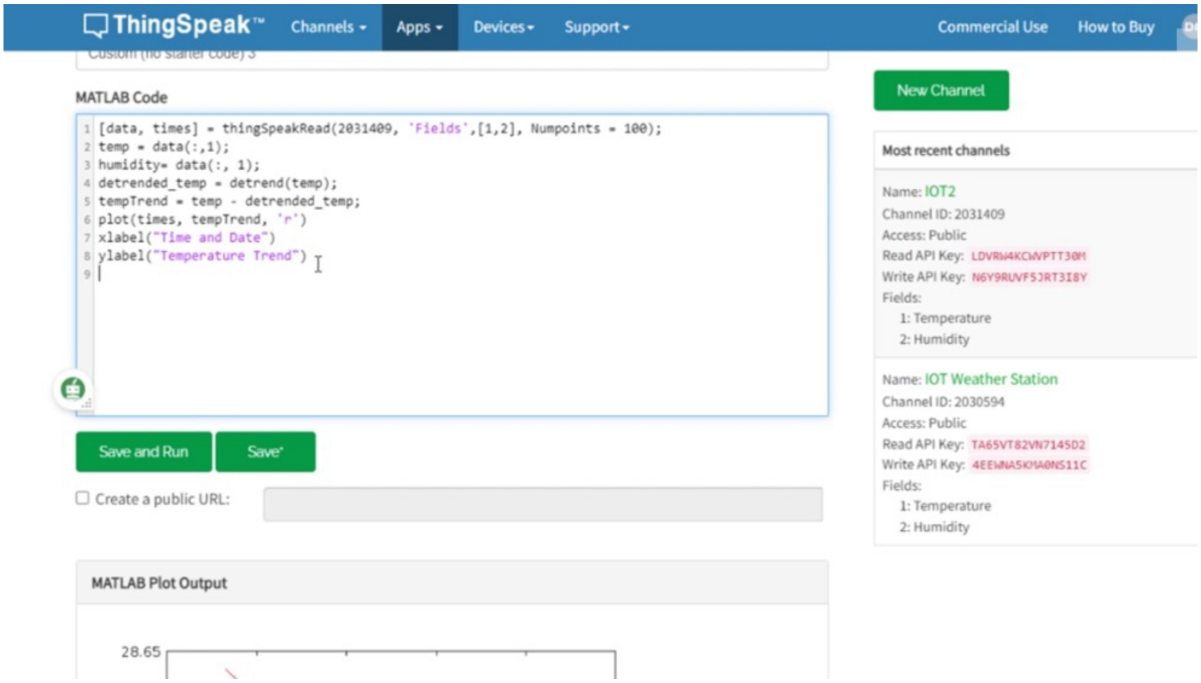
Created: 45 minutes ago
Entries: 15

```
Data Uploaded Successfully
Humidity: 40.10% Temperature: 29.10C
Data Uploaded Successfully
Humidity: 40.10% Temperature: 29.10C
Data Uploaded Successfully
Humidity: 40.10% Temperature: 29.10C
```

So to continue with our analysis, you can click on the Apps tab, then you can do math lab analysis you can do to go do a lot of things, you can click on new you. And there are some good, there are some templates here already. So let's take templates from here. He will only then work on Costco moves data so in analysis, an example is calculating and splitting average minutes. There are other examples, the high coverage, you just pick on on retreat, then all we're doing is to predict temperature. So to do that would be right now. So it's be custom news data could then click on creates and create our war eligible genomics gets the trend about average temperature is training. So that's what we're doing now, to get a chance to work every time to show them visual trends. We also do visualizations, so click on Moore's Law visualizations, you can do analysis, let's visualize our temperature trends. So the new same thing can creates Alright, so to acquire data from so via data from the thingspeak work right? So the date what you can do is to type in I think all being is being a matrix, UCLA workss walks with matrix so good data types that we have since speak races we agreed notizie the case. Alright, so since we agreed, then we have our channel number. So China was saying See, it's 140 Since we agreed this was to read the data from our this week, we have our fields then who's our force field want to access our force field and our second field then we can put number of data points. So if you notice I already acquired up to over 100 At that point so I'll pick the first Andre the PowerPoints then what we'll do is our temperature developed on separate without visual musicals. So this just gives us the false data. Then we gone with operasional sorry, not ratio, etc. Here we did see zip on Hello. So all the rules, let's then we can add the outputs. If we go there, we have to name a few, because we will draw a chain the word and purchase training. Zoo plots comm plots Aqua temperature anytime. So to do that we'll do times and then run plots if you run this would get complete click on Save on alright alright to assess some of the data, you can either make your channel private or public. So let's for ease ease of users in regards to COVID. So if you make a channel to my channel, then my channel is how click on it's then public view. Click on blog posts, then you can go to share, then make it public share channel view. So when you share channel view with everyone is we don't need to add an API keys are good. So back to the podium for the apps. So yeah, since we made our channel, probably, you should run successfully.



So click on Save our own and our code has run successfully. As you can see, it plots a lot in temperature trends over time. So this is deployed over the visual trends over time. So what wants to do is this is the plot of the temperature itself, we won't find a pattern we'll find the trend. So to do that, we'll add some codes. Alright, so we deleted lines of code for class and let's continue now. So the next one we'll do is detrended. We'll type in detrended templates call is the training plus view so once the train our temperature was the trend is then after the trend in is we'll get the trend itself who said you would find the difference between the D trended on on the original temperature so yeah 10 Minus add seven columns after you're good alright, so after does become plots then trended then trend now against time see all the visual trends over time. So this is to add call your calling is rated as the RMS so you can name it you can give x Livio calibers our x axis is time and it's then our y axis Wi Fi near is temperature then temperature change in our case, roof against the polls All right, so let's see. So as you can see, we have the temperature trend or the visual trends over time. So you can see outputs here. These are temperature trends over time. You can also do this for our humidity. So let's do that. Regardless of LEDs and leaders remember to add fields is ready make it easy for you to go on.



```
[data, times] = thingSpeakRead(2031409, 'Fields',[1,2], Numpoints = 100);
temp = data(:,1);
humidity= data(:, 1);
detrended_temp = detrend(temp);
tempTrend = temp - detrended_temp;
plot(times, tempTrend, 'r')
xlabel("Time and Date")
ylabel("Temperature Trend")
```

□ Create a public URL:

Most recent channels

Name: IOT2
Channel ID: 2031409
Access: Public
Read API Key: LDVRW4KCWVPTT30M
Write API Key: N6Y9RUVF5JRT3I8Y
Fields:
1: Temperature
2: Humidity

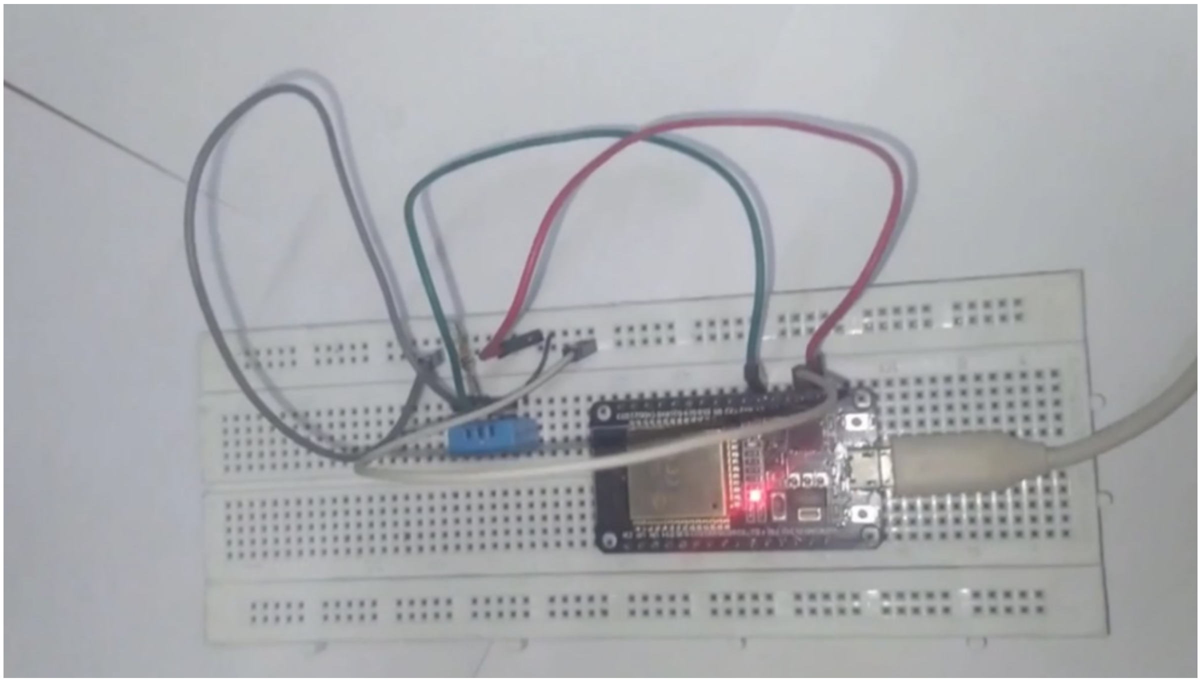
Name: IOT Weather Station
Channel ID: 2030594
Access: Public
Read API Key: TA65VT82VN7145D2
Write API Key: 4EEWNA5KMA0NS11C
Fields:
1: Temperature
2: Humidity

So you should have figure two down here without two fingers Alright, so All humidity just changes to share you need it see it UMB trended on so changes humidity again some of us call it on trains let's call it on train DVDs the trend changes three releases for humanity data alright. So yeah virgin trends is hydrolyze runs before we run we need to add so close threads will make it this far so plots then this is several sub plots can only display one thing was a time.

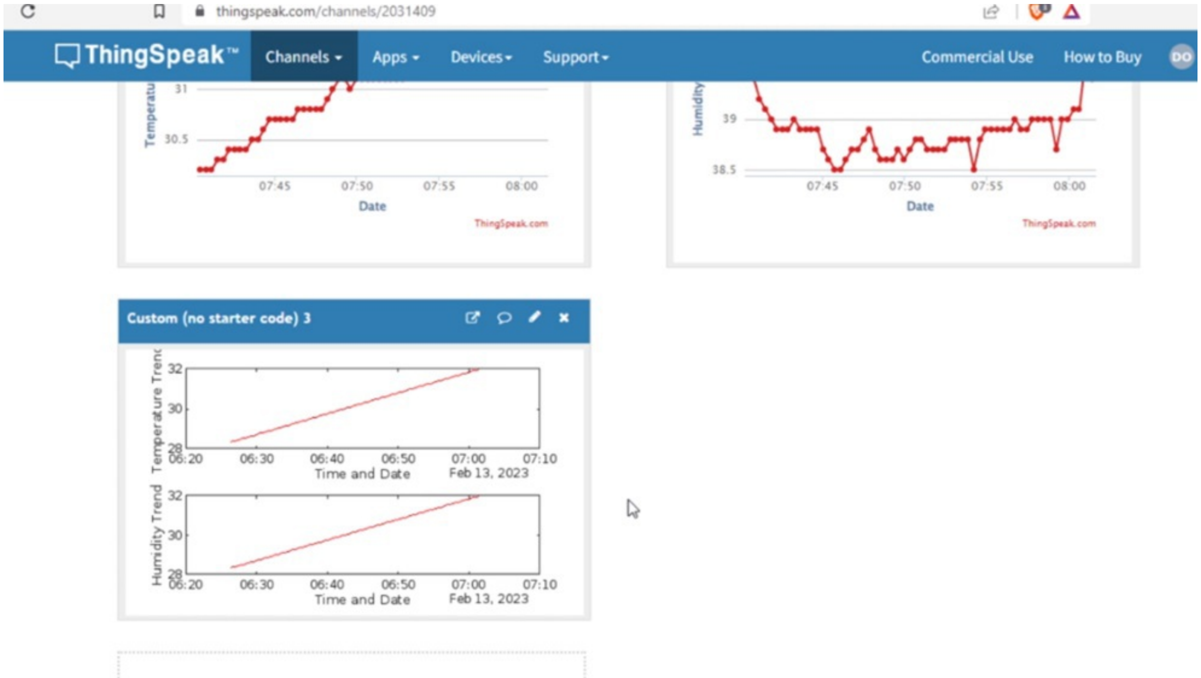
So others change this to unity trend if your unity straining to this will be our immunity train alright so this is our this is EVC love these these are huge trends is our temperature is changing.

PROJECT DEMONSTRATION

Shows our ESP microcontrollers panic density breadboard as well as sensor. And as you can see the jumper wires which connect the controller to the sensor, as well as the tent loom resistor that's going into itself.



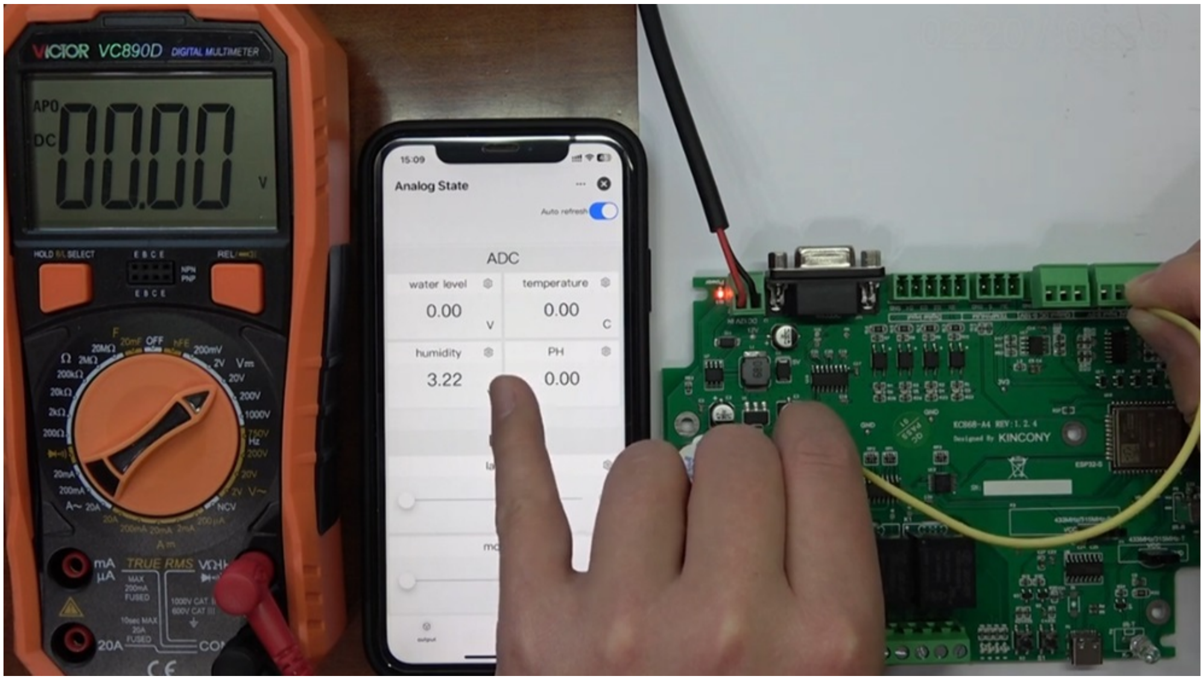
So a little reminder from the video we can also see the USB cable connected. Here's my controller from the laptop computer. This is my go to ice bath with the USB cable monitors with the USB port on board. If you project in, reads the sensor data and sends it over to your Wi Fi to our AdSense account. This is order speak account. This is the channel we are using for our project this January for project. As you can see, we have our temperature and our UV by which we looked for three entries. So another thing you can do is make a public preview to assess your temperature and humidity data and so in other words, your public view you should have made public give us an opportunity to share it you can share and click on Share channel with everyone. I'll make it public the way you make a public view I can use this link here.



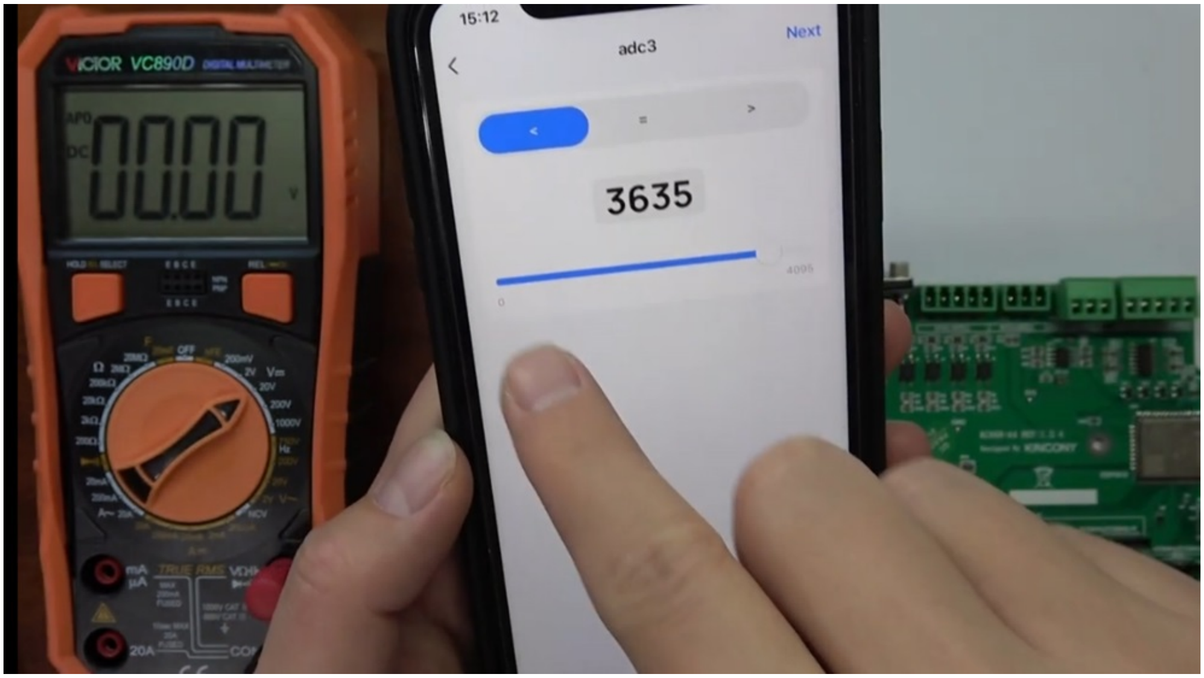
You just add speak your channels then your channel you add it then you choose our Shane field on that data virtual field switches are you ready to add our generation will be the area so by clicking the issues, temperature IMEC trends to use these to show the retirees the trend you can check it by device as your ESP test. Io this episode is running from anywhere in the world but check your data live I can see the trend.

KC868-A4 MONITOR ANALOG SENSOR CREATE AUTOMATION

I will show you how to use this KCS firmware with our ESP 32 port so that you can create the automation on the Twitter application. And as the first let's look at how to use the analog input and the DC output you can see here that is our info pod actually, you can use the KCS framework in any our ESP 32 pod, this application does have output input that is the anak so that we can monitor and control disability remotely by internet. So you can see here I can add a time and a tariff and this relay that to remote country and this is the input the input state you can see if you have channel this is digital input and check out this video to input one. So that that will be green, you can see here and that is a certain page that is analog, you can see here that the ADC and the DC DC that is 4025 Watt input and the sound channel for four to 20 mili amp pair so that according to your pod as a hardware results, so we can test it, such as at least a three and a four that ended up being put through and now we'll input the four that will convert input zero to five what you can see here that channels we see channel four. So, here on the PCB board that have three votes, you can see here we can test it actually you will connect to your sensor analog sensor and this point, just a former test, you can see here that about us we fought. So, we can let this part I can test it for Ace Ace Ray and then again produce ray. So, you can see if I shot for this way you can see here the wattage is minus 3.2 Because hills on difference is correct.



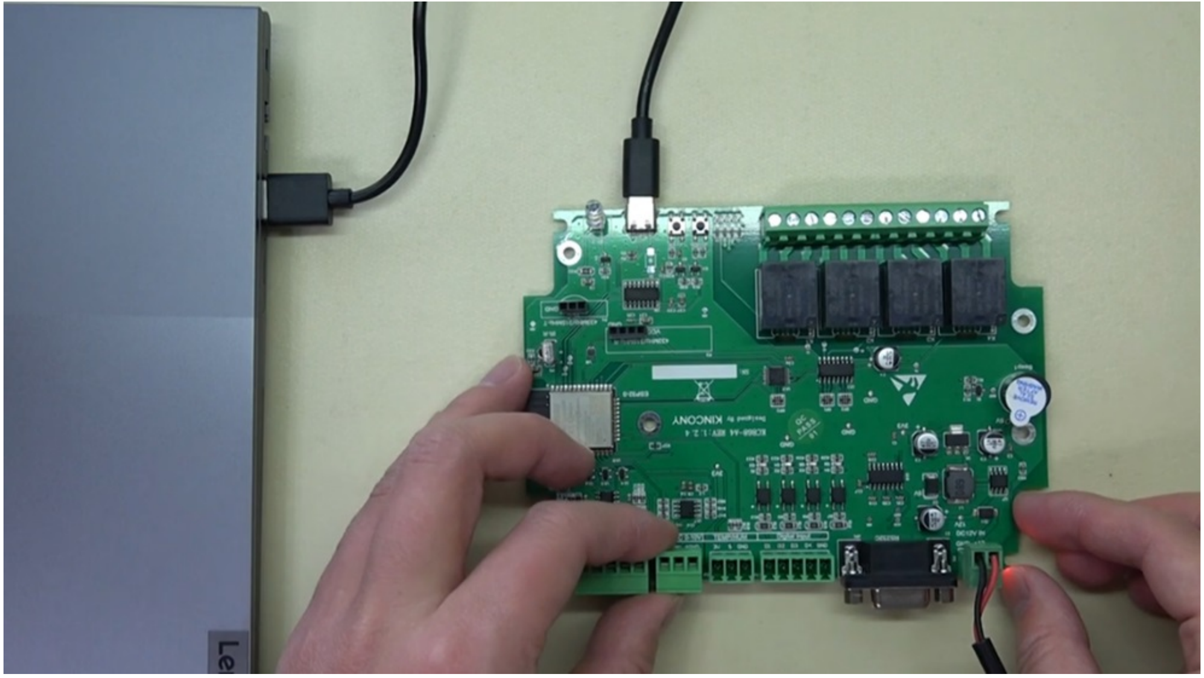
So, this is her mind the wattage remote if I release my cable, you can see the voltage will be changed to there. So, this can remote monitor the analog input and here you can see that the AC that she told to analog output that PSA is for the long term what So, this have to channel THC that help just this one at day one and day two. So, you can see I can use this multimeter I can test it you can see here you can see I have tested this digital analog output to channel one. So that you can see here, I can change this crossbar. And you can see the VOD changed, this is 65% that is the voltage if I change to 100 100% Almost 10 volt. So this can according to your finger you can change the different voltage output usually this can use for the dimmer and use for the Moto speed change. So they say channel one and channel two years thing so you can very easily to change this different output. Okay, let's look at how to create the automation according to this analog input, before we have tested the digital input trigger this digital output. So this time you can see we can click and since since mode, you can quit since mode and one device state is changed. You can see here I have used a four bar you can choose a four bar, this time you can see I can choose this ADC three, this channel so we have a analog input. And here you can see that can input the number that's the max is for solid and 95.



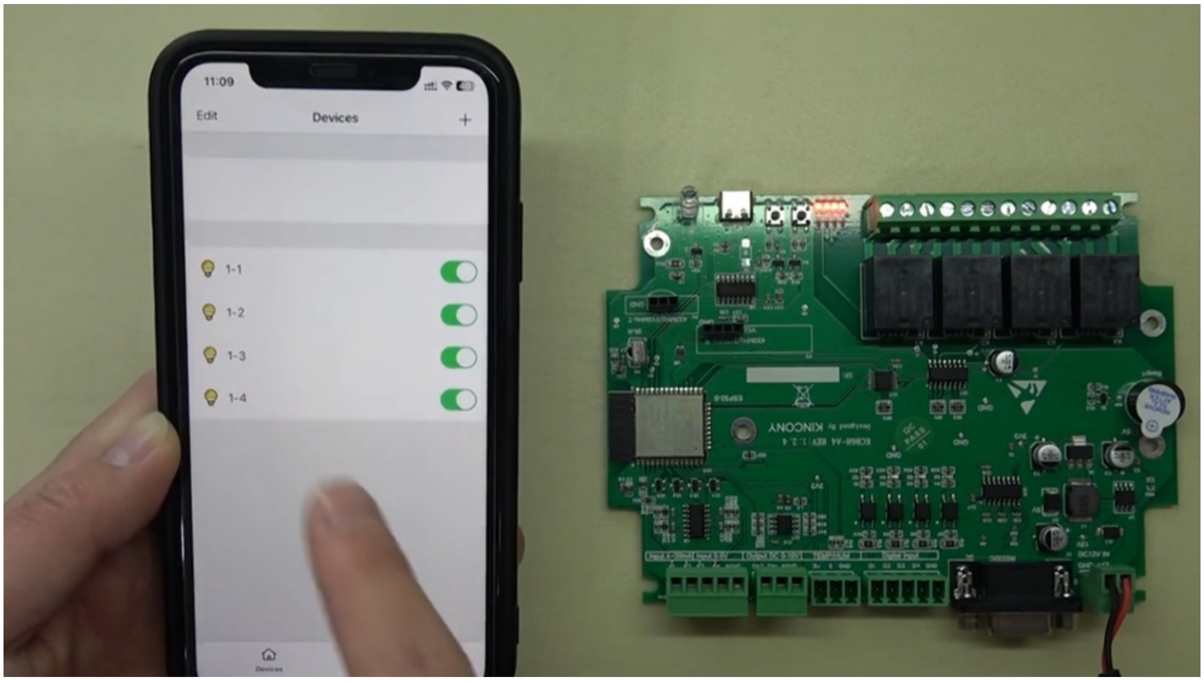
So you can change this percent that means the wrench the zero know what means the yellow and this is for sun and a 95 that means five watch. So if you want make assess mode, when the watch which then half of the fibers such as three. So what I almost can change to this half of this present and quit To that, that means go to the end of this the EAC and this is small and this is large. So you can click Next and is the action just as you can see here, I can edit the test round the device. So such that I can turn on this a4 board, really one that really one is the light. So I can click that the light is out. So I can click Save. And next and then we can press this safe and enable this one. And here you can see if the analog display is good then this number and then the relay will be okay, let's test it. So let's look at this monitor and connect with these three what okay, you can see when the what it actually and this this really. So that is for automation. When this if the analog input the value is high or low or equal, so that you can continue this really output of photo editing off. So this is for automation and set by this to your application.

KC868-A4 WITH TUYA APP AND KBOX WITHOUT INTERNET

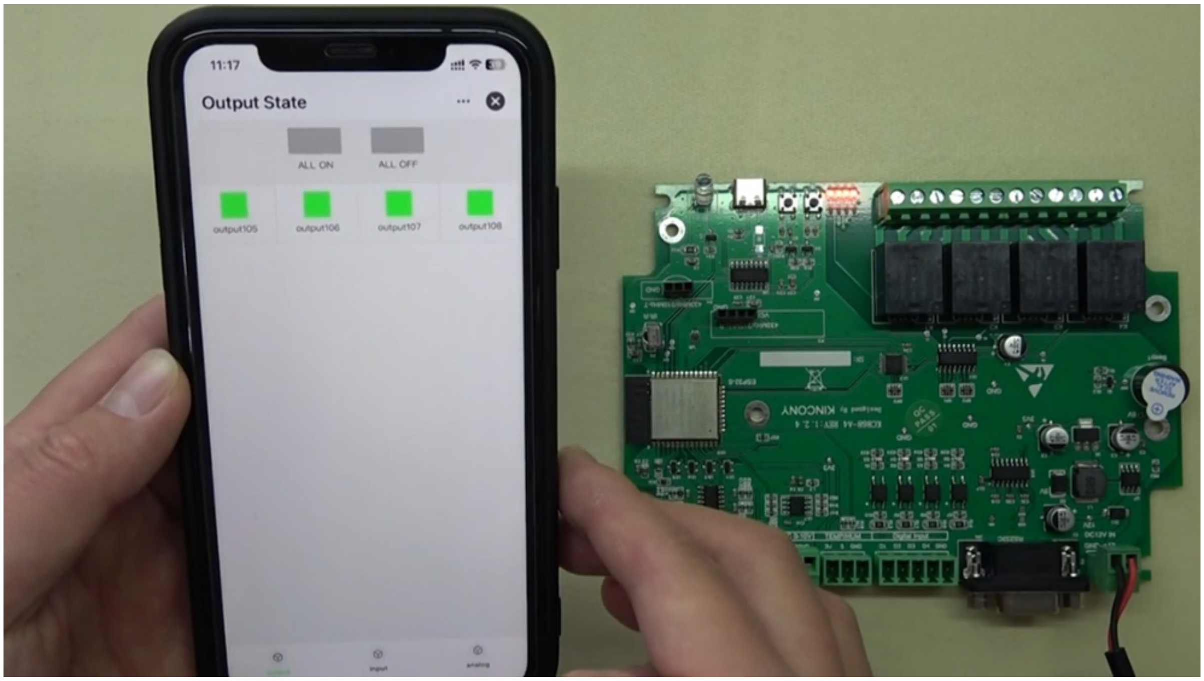
I will show you that this case 868 A four board so that can use this to your application and the key box can work at the same time because you can use this KCS new firmware downloader to this ESP 32 module so that you can use this to your application for Internet remote country and use this key box in local network by Wi Fi. So that without internet these two application can work at the same time. So this will download the new framework will let you bet you a football become very powerful. And in the last video we have detailed externed as a KCS firmware and this time I will take this a football for example just use the Wi Fi how to use the Wi Fi connected to your load and how to use this Wi Fi AP mode. Use the key box directly. Okay, let's begin and here you can see that is KCS that we have already made the video help expand it this time you can download this Casey at KCS just by this ESP download the tour downloader for this link and you can download the the firmware being file from Khinkali form. So you can see here this you can go any form you can download all ESP 32 Badiou from this webpage and now I have used a four part so I downloaded this firmware if you use another mod of the case eight six ACRS BOD you just downloaded the corresponding firmware that be okay okay after download you can see here the download a tour just we can click and very fast to download a tool ESP 32 You can see here we can choose the ESP 32 Because our a football use the ESP 32 module and here you can see you can select use this icon select this a4 thing file and write began the arrow address and now you can let your board you can see here you can connect with a USB cable is your type C USB cable and connect to this might compute the USB port.



So you can see here I can just connect with it. And here you can see this power for 12 Watt a for use by the 12 Watt now I can use this power hour and you can see the radiology is now okay now we began to download the firmware for this board. So you can click this component you can see that the auto detect comes through and click Start that's very easy began to download Okay, now you can see they're still finished, we can close this window and Close. And now you can let it rip power out you can power off and power out and you can remove your USB type C cable because not needed to use again. Okay after our report, you can see this computer Wi Fi signal, you can see the displayed that a four that is for each mode, we can click and press Connect then your computer will connect to this a football directory and use this IP address just a 4.1 this is fixed enter and you will log in by the management web page and enter the username and the password is admin admin so that you can lodging so now we have logged in as a full board and the first time you can see the board because it's new. So this is you need to set the UTC. So you can see here because I have not connected with the Internet. So that is not updated the data and attack from the internet data. So you can see here the system you can first re set to yo tch and press theme and the part will restart okay, we can see this port cool. If you want to use the key box, you can see here we just needed to enable the TCP server because the key box here to work with the TCP server and there's a local port you can press any port now I have pressed this 4196 And you can press save and press ok And you can see the monitor and the TCP server is working okay now, I will show you how to use this mobile phone and connect to this AP mode the Wi Fi model directly. So, without load that can use this key box application control this really directly. So, you can see I can use the Wi Fi and connect to this a4 pod. So, if nobody is no password, because mainly used for config and now I can click key books and the press ENTER device and his IP address that you fixed 4.1 and the the part we have set as 4196 and the DC relay and the channel is four channel.



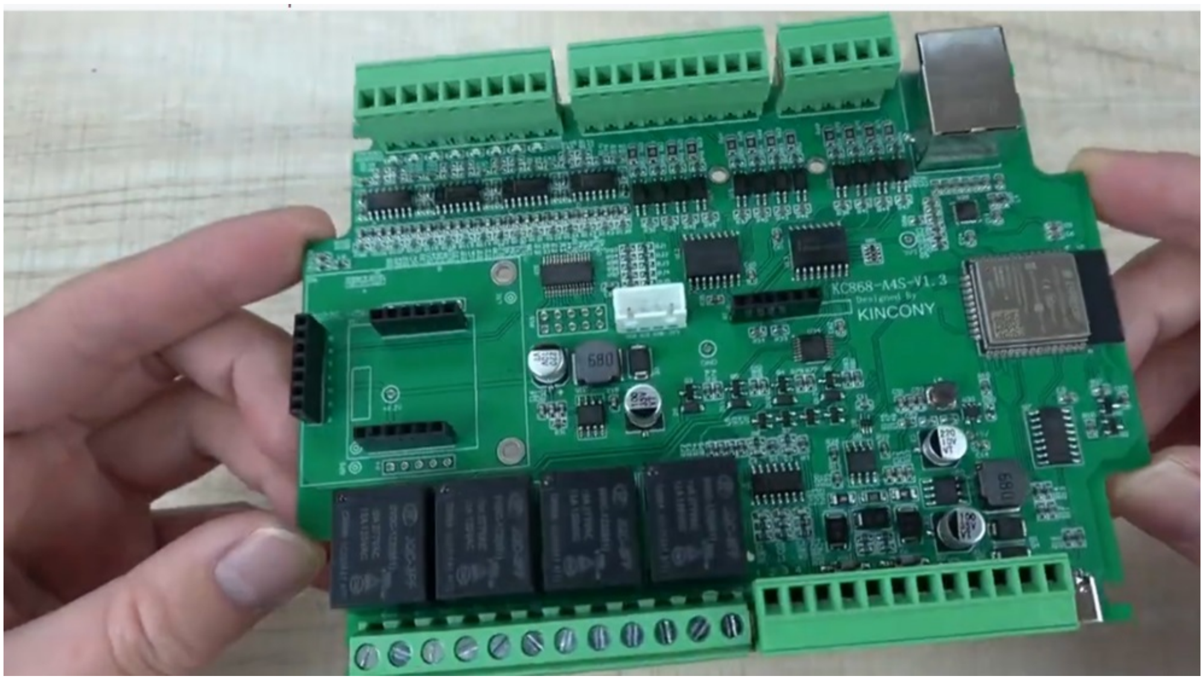
So, you can press this safe enough here you can see that have Mr for channel, you can see here I can to really one relate to the reasonable gray for So, this time, I can use this key box and control display on and off directly. So, you can also Renee, press edit you want to light the light and you can choose this icon for different title for them and in the Clip mode or touch mode. If you use this clinic mode you can see I can turn on and off and hold on you can edit if you want to change another work mode, this change click to touch mode this touch mode so that you can press down you can see the icon has changed and this is a click and it is a touch just a touch release touch release. So this can contain this cotton model. So we can see touch and remove that will very fast. So this is use the Wi Fi and AP mode are connected to your keybox directory. Okay, next we will look at how to let this Wi Fi module connect to the internet and connect to your Lord. Okay, we can go to the network and here we can set to sta mode so that we can use internet because I will show you how to the application that we have a location we'll use the cursor. So this is my load and click Save save and wait found that will be restart okay because the AP mode is in infective. So this time we can let my computer and connect on my road and is there I can open this scan device Tor you can see here I have used the Wi Fi of the computer this my computer network card and computer IP address click on the Start monitor pot and as you can see here that will list by now to the A football by Wi Fi at this IP address. So now I should use this IP address to lodging and here the username and the password lodging and now you can see this is I've seen SGIP that have showed and because a football no Ethernet so this sashings not at this point of time you can see here that is our update to this data anytime from the internet because I have connected to the router so that have the internet have set to this time. Okay, now we can see here that the generally have the TCP server we have already running but at this time we should change the key box to this IP address. So you can see here if you use the key box now you can see not working because IP has changed. So I needed to delete, delete and edit device input IP. This is a new IP of this part and here's a part also for 196 and model four channel and press save. And here you can see I can continue this and again. So this helps let your mobile phone connect to louder and this is connected louder. So this can work with K box by the st mode. Okay, next I will show you how to use this to your application that is smartlife application Let's begin. If you want to use a two year application for a full part also is very, we just click these two sections that you need to enable the two year apart. This time, you should buy this two year lessons, you can see here that 12 lessons from King County, so almost very cheap price. And you can use this lessons always, no time limit, no date limit will always can be used. If you have interesting you can contact with us, we will still use this lessons to be an essence. So you just copy and paste this product ID and the device ID and the device secret. So you can see here, because I'm in China, the ID is in China. So the lessons I use by the China if you have you in your repo, or in your USA, so we can sell you different lessons for different wrench. So you can see here, I just copy and paste. And here is the URL and just a safe restart. Wait for a moment. As you can see, this QR code is autogenerate for two application. So now let's use this mobile phone. Because this is a test bar last video, we have quit. If you want add another bar, you just press this add and add a device and scan this QR code. And now you can see I can use my mobile phone to scan this QR code. Okay, you can see that a full board have added successfully. So now you can see this bar a full board, click down and you can see will open this page. And you can see here, I can click that really 100 This Folly is now this time, I can use this by Wi Fi or by 4g Switch of 5g so that can remote control Bad Internet, I can click off. And this is Oh.



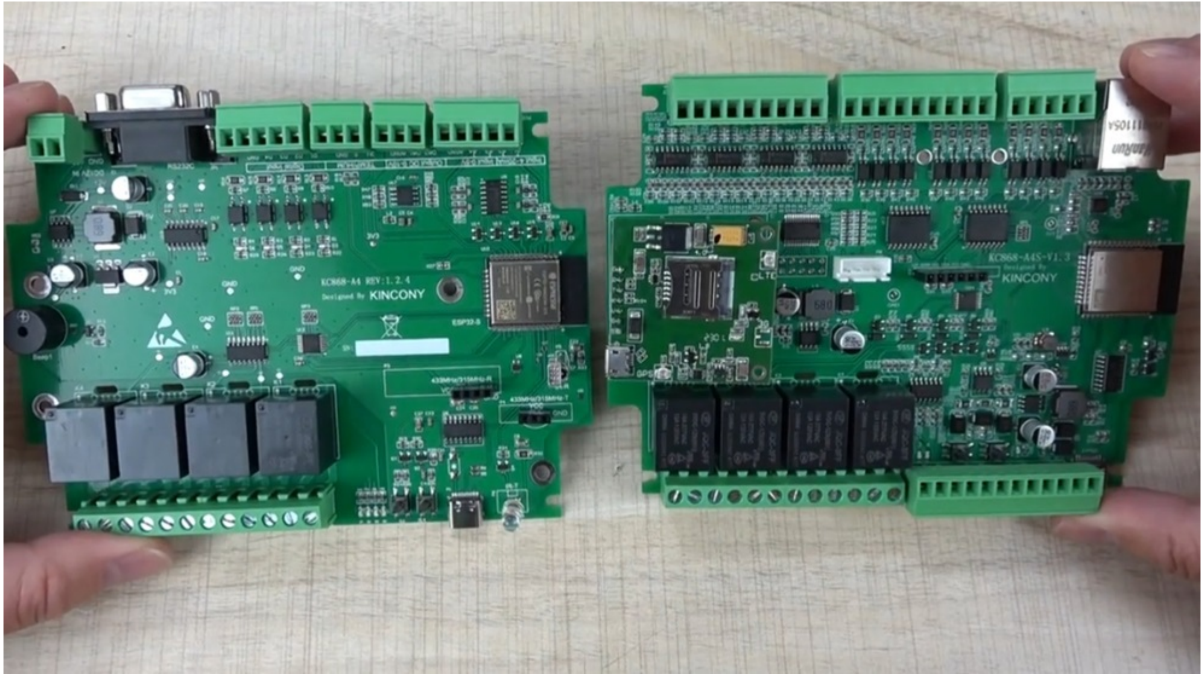
And also you can see the input a state if you're short this input a foreground that will be turned green. So you can see I can use this cable and this is ground this a Foursquare and this is input a one. So you can see that so I can become green. And this input the way you're tangled. So this is the input. And here you can see that have the analog analog input and the THC that is output a zero to 10 vote. So you can see if I change this prosper that the TC will output a zero to 10 what that can use for Teema or change Motospeed. And also this is a second channel of the dgac you can use remotely and also you can click or you can set the DC name and the max value and the minimum value and is the unit you can see I can input to this present. See, and you can see that it changed and also you can change our channel of the analog input. So you can click this icon changing the name to the Mini and the max and the unit so that according to your sensor, so you can change our settings and also you can change this input input a name so you just can hold down you can change this name just such as a nice answer. I can click see. So you can see here that we will change the name and also if you want to change this Loli name so that click output and the hotel that can change. So output a name such as a light, light and press C and you can see the light is changing the name so you can rename it and and it will be easy to remember which really is which name. So you can I can use this to your foldable country that by the Internet if internet is broken so I can change us switch to my K box you can see the state heat update fee for and change total. Yeah, you can say state update and close. Close close. Close. Close. And here you can see In such a key box update. So this can use in local network without internet and this tuya can work remotely. So you can use two application at the same time so that you can go to go home or go out of your home go to office on the way So, anytime you can use your mobile phone contract is a verb work.

KC868-A4S NEW ESP32 HOME AUTOMATION BOARD RELEASED

I will show you our new product and this will card case h six eight a four asbach and this part will have as many hardware resource fist on this ODE a four part and this old a four part have released by two years ago. So mainly used by the Wi Fi and is awareness the four channel relay output. But this time we have designed this new part the new part also have four channel relay but how many digital input that healthwell digital input and we'll also have the team output analog to tell what about each analog output and eight channel PWM output. But for the communication part, we have added the Ethernet and is for it five supported by this part also were supported the GSM for same kind of use. So, 2g and 4g or 3g or can use by this part and this part also will compact him with ESP home use for the home assistant also you can write to the Arduino code Tammela to the ESP 32.



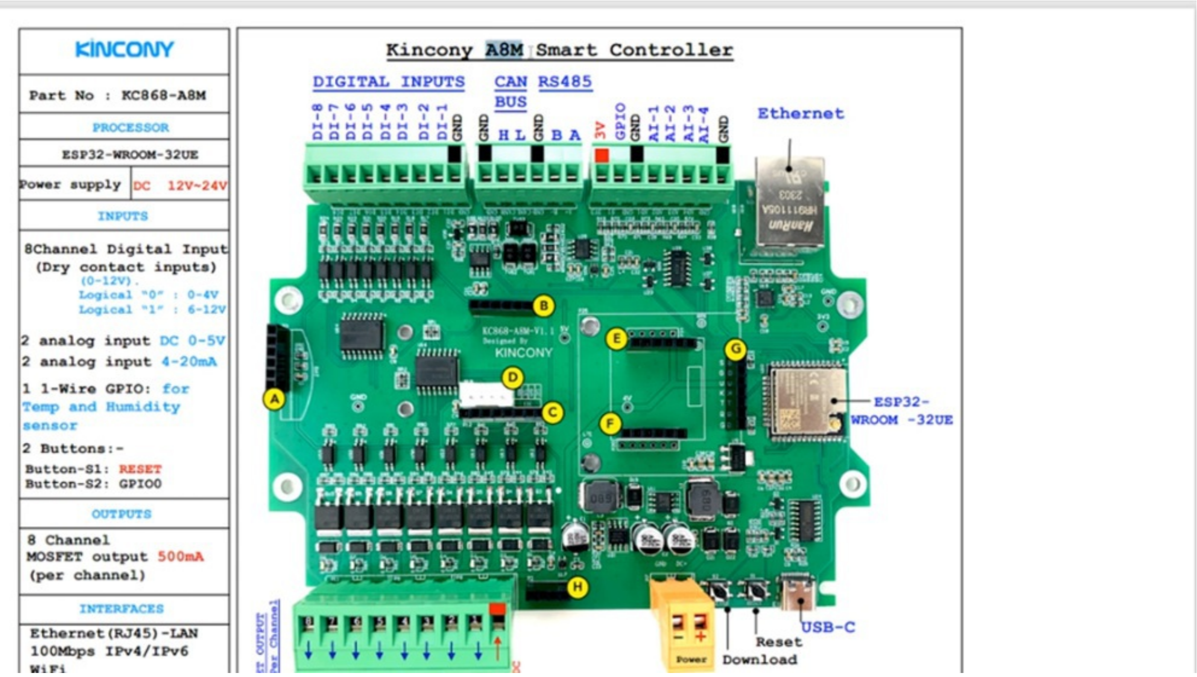
And then let's look at this board hardware details. Let's look at this as a first part you can see this is the PCB part you see in front of the side and this is back backside you can see the PCB details Okay, let's introduce you this is the ESP 32 module and this is Ethernet and this is digital input. So, that will support to have what logical oh by the dry contact signal input here and is actually the analog output you can use for Timur that is eight channel of zero to 10 Watt output analog and this is a chip that extends the analog for 16 channel output. So, this is eight channel for the anode to 10 watt and here is the PWM output extender just controlled by this chip that is I squared C chip have connect with this ESP 32 module and this is I squared C extend so you can extend it for I squared C temperature sensor or filament sensor or many different sensor and here we can see that is really fortunate fortunately that that Max is a part time pair for large current and a normal open normal cross and account for our channel. And here you can see that is the analog input and is for it fi and the some buttons for reset and for user defined you can define this GPIO zero so yes you can program with this case function by yourself. And how you can see that is a socket that for RTC module that ETS service recently won you can see here when we installed this RTC module just as this way ts 3231 that in high quality RTC chip. So this can install the PI this cup just this way and here you can see this socket you can see here you can install it this is 2g module, this is GSM module that you use for Kochi that year Sim 800 GSM module, but I think some countries have stopped us a 2g signal. So, you can also replace with this module and with this one, this is sin 7600 that is for 4g. So that will be speed will be fast communication with the Internet. If you want to use this module any you can connect with this target and this also can use a cop just just as this way you this bother to fix because that's how we hope that hey, we're hooked on a PCB. So you can install this 4g module and this module also will support this GPS function. So these have to antenna this forward GSM and this is for GPS. So, you can connect with this two antenna as they say for GSM and this is a GPS antenna, so that for GPS date sent to internet for this solution. So, this is an a4 s part and you can see here that is a four part.



So, this one I have released two years ago. So many people like this part about this multifunction and his part also you can see that is the same size almost a CSS. Berta the hardware resource will be many many many new results then this older person and as you can see, that is a pressing issue and installed on the thing so, you can install another thing we'll just use this plastic a shoe and then we can open it you can see I can put to this a4 spot on this part and close this this one. So, you can see here you can see the back that is for communication and the for the interface and the instant on the general for this case. So, I think this is different pick a different from this a four pack. Can this one I think you can make many different IoT project use by this a for a spark.

KC868-A8M ESP32 CAN BUS HOME AUTOMATION BOARD FOR HOME ASSISTANT

I will show you our new product that in case it's 688 M part. So you can see this part is smart and this part is an eight channel MOSFET output. You can see also used by the ESP 32 works that have the IPX socket so that you can extend your Wi Fi antenna and the weekend and this part also is a part of the ESP home using home assistant and you can read the code of your ID not IDE so this video I will show you the hardware details and how it works in home assistant. Okay, let's look at the first that is our ESP 32 module and this is the Ethernet and Ethernet chip. And here we can see that it ended up being input and the one one where device you can connect with the TS 18 B 20 temperature sensor or DHT warmer or DHT tool temperature sensor and a family sensor. And yes that is is void fi interface so that you can use the long distance for communication. And in this time, the important we have we added a campus in this eight channel version pod. So let's look at this Tiguan of eight m pod.



KINCONY

Part No : KC868-A8M

PROCESSOR

ESP32-WROOM-32UE

Power supply DC 12V~24V

INPUTS

8Channel Digital Input
(Dry contact inputs)
(0-12V).
Logical "0" : 0-4V
Logical "1" : 6-12V

2 analog input DC 0-5V
2 analog input 4-20mA
1 1-Wire GPIO: for Temp and Humidity sensor
2 Buttons:-
Button-S1: RESET
Button-S2: GPIO0

OUTPUTS

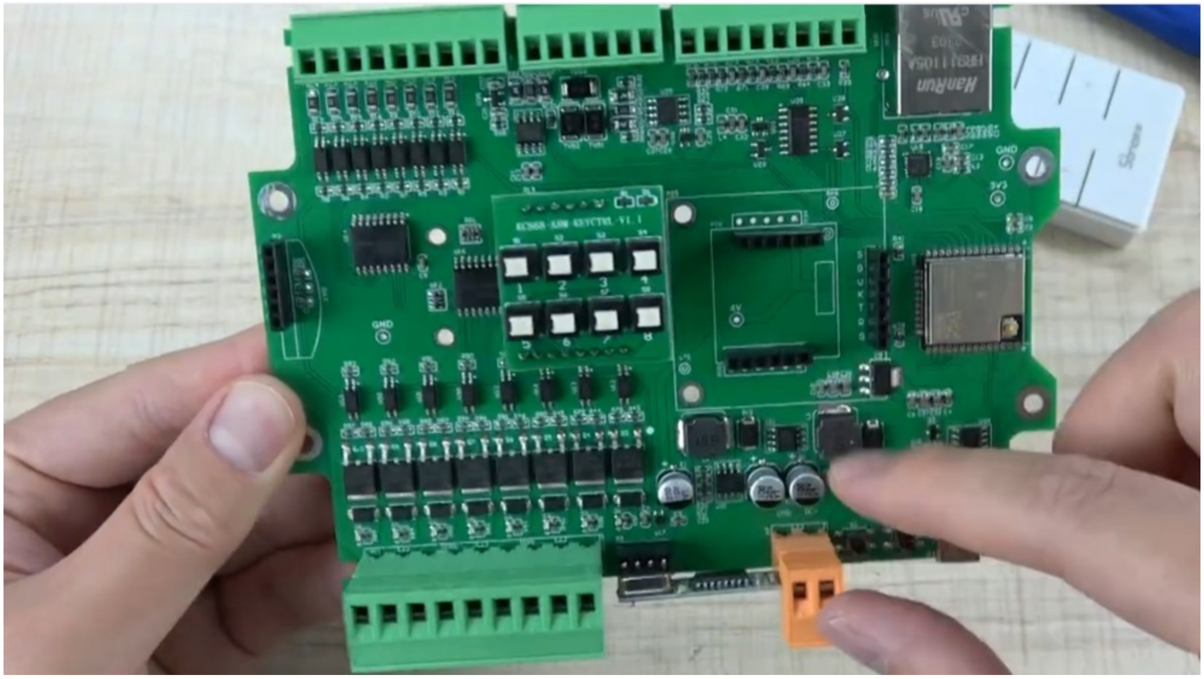
8 Channel
MOSFET output 500mA
(per channel)

INTERFACES

Ethernet(RJ45)-LAN
100Mbps IPv4/IPv6
WiFi

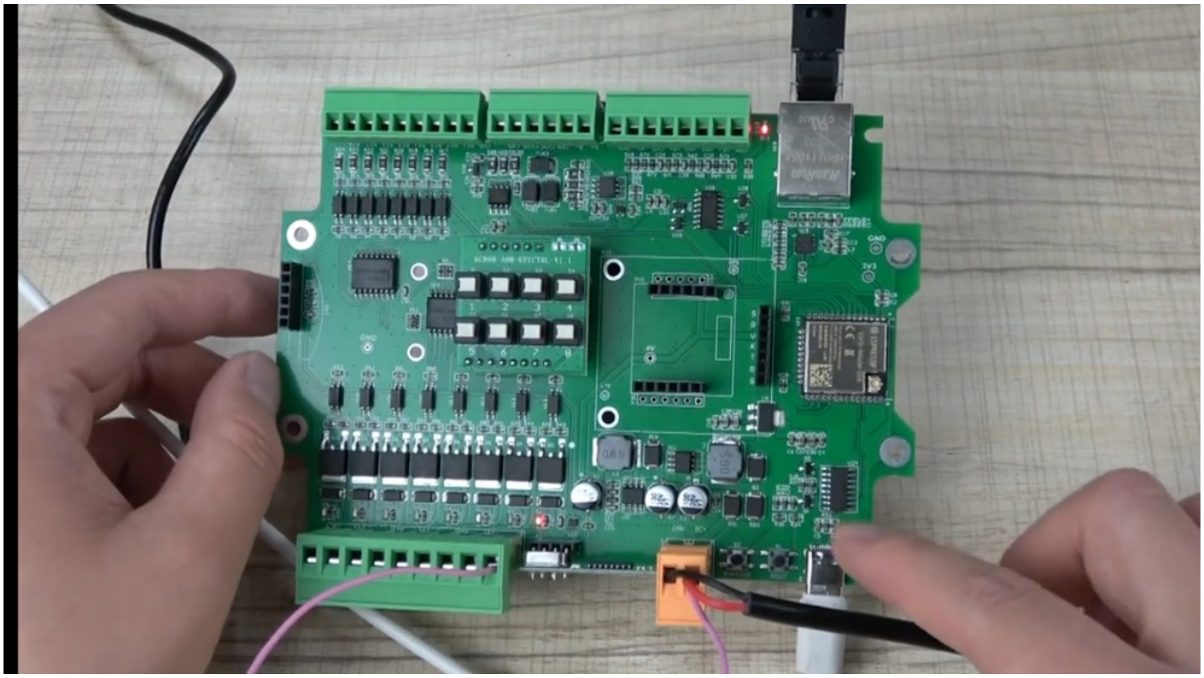
Kincony A8M Smart Controller

So you can see here that we have the ice for it fi but we have added this campus not h and L so you can connect it to your cam system. And this is input digital input. And this is MOSFET output an error MOSFET that support Max 10 M pairs so it's very large. So, this is eight channel and this is V that connect with the power supply. You can see here that it were what are 24 Watt input to this core just this whole so that can output 12 watt or 24 watt and this is the power supply. This is orange with power supply and to button for research and afford to shower zero and this is a USB for download the firmware. And you can see also how many socket the socket socket and a socket and the socket. So, if you want to use the first 50 megahertz you want to use this remote, so that you can install this RF receiver module this socket. So you can see here that will be easy to install it and this you also have a socket you can see here that is deferred installed on the PCB board. Because that for many countries this HDMI output physically so without Ethernet and without Wi Fi and without software that can country into directory Later I will show you how it works. And then remove this small keyboard you can see that I scuzzy interface so you can extend your I scuzzy device such as a temperature sensor or a sound device just by this socket. And we start back to it and this RTC module, just use this one RTC module, ts 32 and 31 So, you can just install it here you install the battery that will receive the crock system. And you can see here that in 4g module or 2g module, and this is a 2g module, as you can see is sin 800 2g module, if you want to use this module is cheap and you just install the clubbing it you want use a 4g module and you can connect with this socket. So you can see I have another part you can see this part and this part you can see you can fix the RTC module by this school and this metal body so that will be very stable. Because this how this whole You can also install it as this way because I need to save the time I don't operate to this step and this is a 4g module that is sin 7600 and the install the SIM card and install also can fixed by the school and metal metal body. So you can see here that have to hoe so you can install the 4g module on this part. This is another our new part that is Ace 1332 M that is next video I will show you how to use this part. This also is our new product. And you can see I can remove it. So that's this part how many interface for different hardware used. Okay.

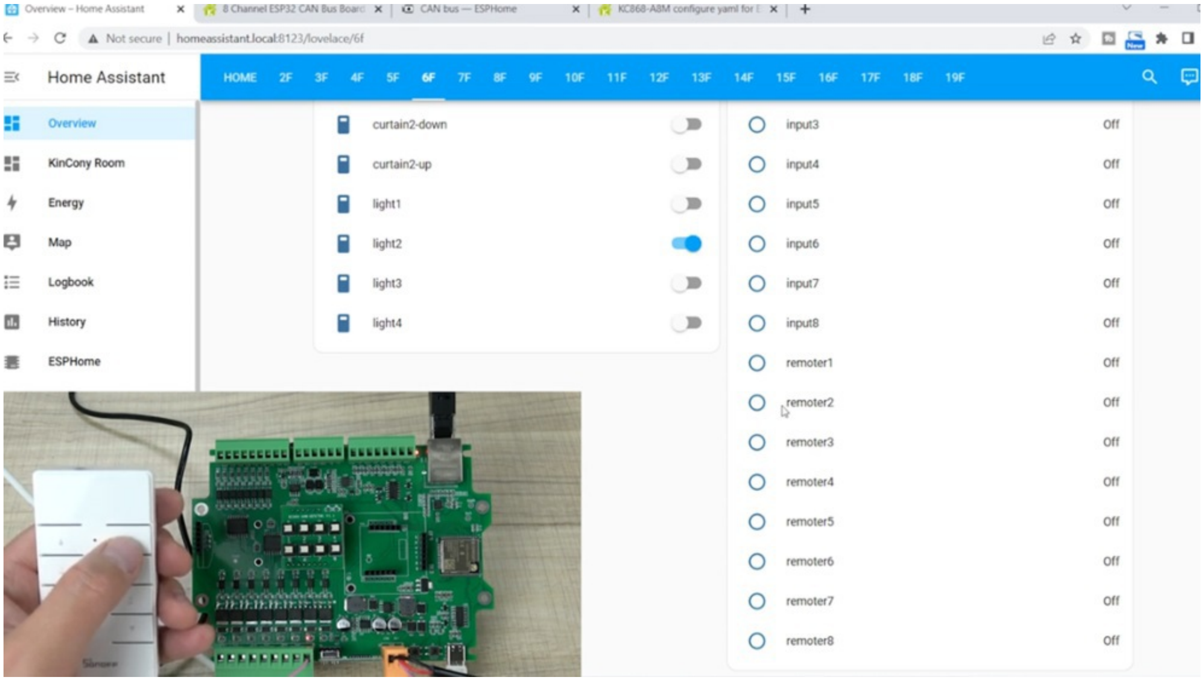


You can see here That is some details you can see this Tiguan and I have said that have supported the CAN bus. So we have surprised the CAN bus driver and the PC software monitor and the back and Arduino library for can receive and send so you can see this I didn't receive demo source code camp receiver and this is a project for can sender so that you can get back by yourself and also you can see that in the ken PC software for the pack can pass the communication send and receive okay this you can dip back by yourself if you want to use this next let's look at this how to install and integrate with pod in home assistant let's connect with an ethernet cable and connect with a socket the socket also is removable. So I need to use the power this in my power line that input for 12 Watt for test. So you can just plug it in and this is 12 Watt I will connect with this V because this is the power supply for this eight channel MOSFET so I can fix okay, this is fixed. So if you want this 12 Watt that will output 12 Watt if you have connected with nine for 20 For work, that's the most fit will output a 24 vote it's up to you and connect with the USB came with my Raspberry Pi okay now we can let it a power I can act with

12 What you can see the red LD is this is our and it's a pod you can work but I have not downloaded the firmware for ESP 32 works you can see I can press this button one and you can see this early in indicates that means the first channel output that began work so this channel is physically contained in this HDMI output.



So you can see I can press this button and the first 5678 So you can see that the terminal this time is output 12 What I can close it okay, that you turn off Okay, let's look at the home assistant. You can see our phone we have sound restores have upload, just you can see here that config Yaman file for home assistant by the ESP home. So you just copy and paste all config at here we have enabled the input and output and also enable the if receiver. So you can see here if you can click ESP home and we have C our eight m part that is an MRI because I have already copied and paste it here and the you just click install and then choose the USB cable to install this time I click back and this I have already installed home assistant by the ESP home. So I can click here and we can see that it output you can see I can click this channel to turn around turn around turn around. So this can be controlled output so that off and this eight channel and I have linked the input without put you can see her eight chin of the input this digital input word you can see the remote also I have learned eight Chen of this part this is some of remote but you can use any for surfing because remote. So you can see if I shot for this channel one input one this ground and this is one input one. So you can see this is become now because it's triggered. At the same time you can see the light one is out. If I remove it, you can see the input is off. If I shot again, you can see I shot again you can see that because this is not a good contact. The light is off. And the input state is so this can be tribal the output I have set the ESP home. Okay, I'll remove it So you can see our code, the ESP home, and eight m bird. And you can see here we can find the important one, you can see this code, because if I'm press just the input is pressed. So this switch toggling according to the last one. So you can set this input a one trigger like two, or three, or at four, just a change in the number. So this can set automation for the input and Chacos output a code, okay, this is for digital input. And the next thing you can see that we have also sound remote one and the remote two. And until remote eight, that is the code I have to code by this remote, so you can see if I can change to the dashboard. And you can see here this remote, I can press this button to you can see this channel to output the target. So you can see what I press you can see here, the light two is our of our of just changed the state.

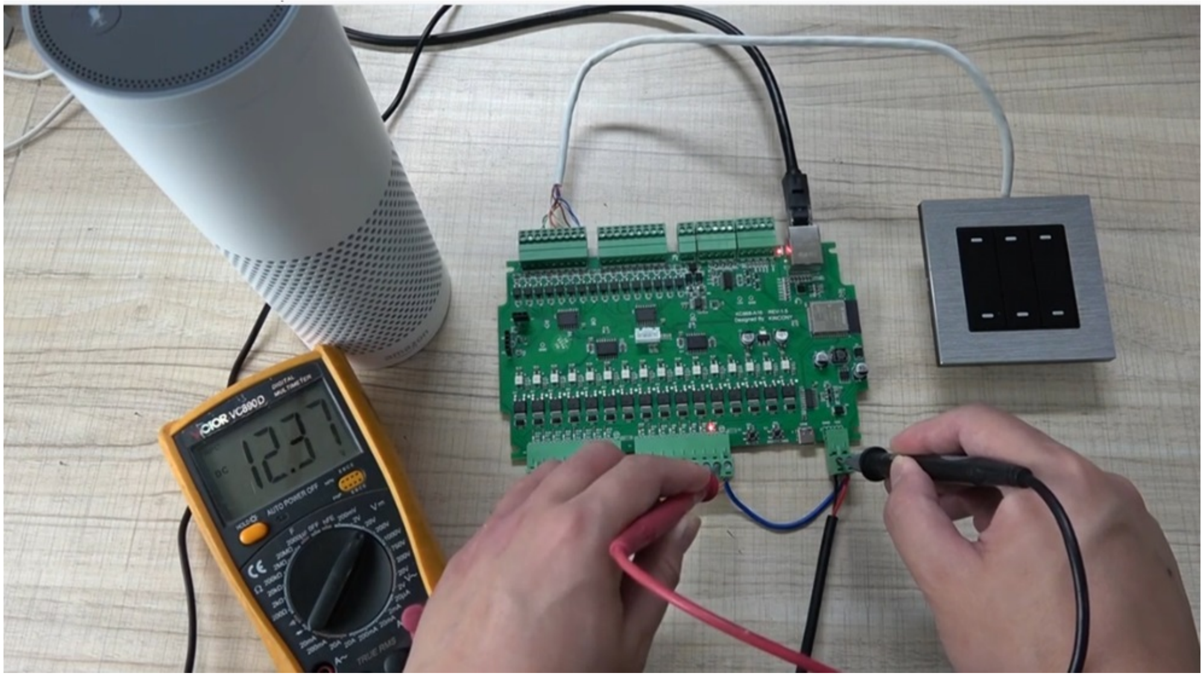

Overview – Home Assistant
8 Channel ESP32 CAN Bus Board
CAN bus — ESPHome
KC868-A8M configure yaml for E
Not secure | homeassistant.local:8123/lovelace/6f
Home Assistant
HOME 2F 3F 4F 5F 6F 7F 8F 9F 10F 11F 12F 13F 14F 15F 16F 17F 18F 19F
Overview
KinCony Room
Energy
Map
Logbook
History
ESPHome
curtain2-down
curtain2-up
light1
light2
light3
light4

input3 Off
input4 Off
input5 Off
input6 Off
input7 Off
input8 Off
remoter1 Off
remoter2 Off
remoter3 Off
remoter4 Off
remoter5 Off
remoter6 Off
remoter7 Off
remoter8 Off

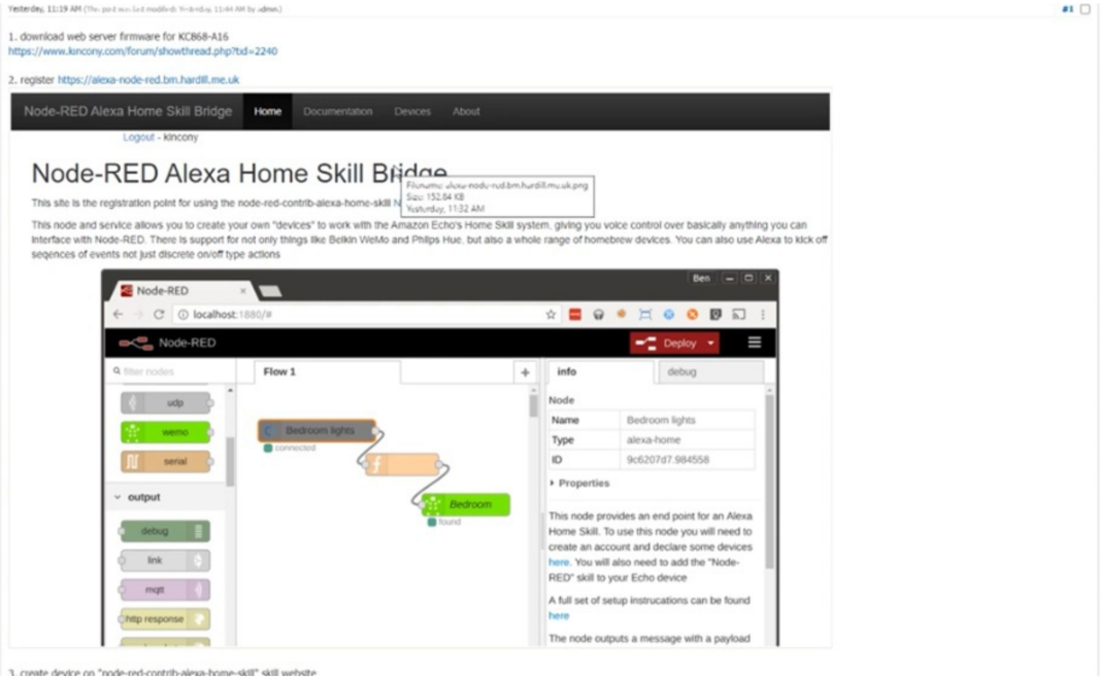
And also you can see the state it here will be changed to our so if I hold down this button, you can see this is run LD I'm not released my finger. So you can see that will be always I release my fingers that will be off. So this also can use for the if a sensor or for some PIR sensor, motion sensor or smoke sensor. So you can see I can click the lights three and light four. And here you can see every channel will country by my remote. And this turned off. So you can see here that eight channel key code for every button I have the code that you can see the logo output by the ESP home, just so you can see here. Just so you can see here, when you press this button that will output in logo window if you want use the campus you can see here, the ESP home also supported the campus, we have used the hardware solution that's the ESP home support the chip, you can see here that the campus chip we have according to this diagram and designs is powered by the campus system. Now you can see I can use the home assistant to turn on the output of the output. But this according to the network. So if you have connect disconnect with your network, the pod will can't work with a home assistant you can see it can't work. So, this time you can use this menu country but so this also can use directory to use without any software that is for the last step if you software embed hardware is bad or your Raspberry Pi is bad. So you can use this button and the country in this output directory if I connect with a network cable again, so you can see the home assistant will refresh and now you can see I can turn out and turn off again it can work recover we also did on this board for our practical show you can see here this is our show you can see this is a box is a box also is supported in standard thin layer. So you can see that will be installed on a thin row to your power distribution box and we just can put this box this one and then we can install this cover this and fix this score. And you can see this a function and this is the back. So this will be easy to install. And also if you want we can print your logo and print a sound test on the box. So that is very easy and it will be more beautiful.

KC868-A16 INTEGRATE TO NODE-RED FOR ALEXA VOICE CONTROL BY HTTP

I will show you how to use Alexa Voice country in this a 16 part and use the words which countries are a 16 board with a digital input and a country this output. So, you can see here we can use Alexa turn on light one okay you can see here the LD that is become read Alexa turn off red one okay okay you can see this earlier and also you can use this button okay I have used a six comparator you can count 123456 So, this also can many country prize this was which and we can see this hardware that is a 16 part you can see here that your kids ages six to eight is 16 that have data input and this digital output with this no really on the board this just most fit on the part. So you can see here is a PCB board and then we have connect with my switch you can see the switch is dry contact switch. So I have used this one and connect with this just this digital input I have used a six channel for six button you can see here I use this one you can see here that is a Casey is K calm Let's kick out the blue have connected with this terminal there is k 1k 2k 3k 4k Six on Turkey six so that connector with this terminal and this is dry contact Eric K one okay to connect with a key card that will toggle the output so you can see we're not like this okay, this is for the digital input and because it's most fit output, so you need to connect with this feeling this to v you can see here we need to connect with 12 What this 12 What were what so this terminal and this terminal in the connect with 12 what this whole together so you can see here I have connected with together because when we turn the output to turn off so that will output different voltage if you have connected with where watch why waste country bad luck Alexa that turn ON light one so that will output 12 watts. So you can see here we can test it. Let's look at test it from this ground and this is channel one.



Now you can see the multimeter is zero volts Alexa, turn our light one okay and you can see the light one is on the channel when the multimeter is 12 volt Alexa, turn off light one Okay, okay, you can see when it's turned off so that almost is zero what. So, this terminal you can connect with your contact or you can connect with extended relay for country large current load okay, this is the hardware of this book. When the next step we will use this software downloaded to this board and I will tell you how to integrate this board to know the read so that we can use Alexa speak very easily. Okay, let's look at this topic I have uploaded is a 16 integrals to know the red AXA how to step by step. So the first step we need to download is a web server firmware of the a 16 part. So we need to download it to this ESP 32 Because you can click this one that will open this web page. And the you just download this zip file and this zip file you can just unzip you will find this to file. This is original source code. And this is from wrapping file. You just use this being felt installed for your ESP 32 directly. So you can use this ESP 30 to download a tour like this and just a feeling this pass of your file because I have copied this firmware on this pass. So you just import it to this pass and this address must feel as this 10 solid and then now you just can click out and connect with a USB USB cable because I have already download. So you can connect with a USB cable and choose the compound and click Start. When you start, download the firmware is complete, and you will begin to read power out just the power of the Power Hour. Oh, so you can press this reset button. And now you can see Roger in your load this in my rod, we can log in and see this field list, we can see the device here, this device is my board, we can find out this IP address of this ESP 32. So that we just can use this IP address and opening the web part. So you can see here, I can use this IP address just to enter. So we open this web page that the web server from this part, so you can see I can click Oh, you can see this tunnel, the channel one and this channel two. So now I can use the webpage or country Edward channel so that you can also say this URL. And this is our so this time, we can use the web server from this a 16 part, this is our first step. And as a next step, we need to register a username and the password for the Alexa Knoedler the skill webpage.



Yesterday, 11:19 AM (This post was last modified: Yesterday, 11:44 AM by admin.)

1. download web server firmware for KC868-A16
https://www.kincony.com/forum/showthread.php?tid=2240

2. register https://alexa-node-red.bm.hardill.me.uk

Node-RED Alexa Home Skill Bridge

This site is the registration point for using the node-red-contrib-alexa-home-skill N

This node and service allows you to create your own "devices" to work with the Amazon Echo's Home Skill system, giving you voice control over basically anything you can interface with Node-RED. There is support for not only things like Belkin WeMo and Philips Hue, but also a whole range of homebrew devices. You can also use Alexa to kick off seqences of events not just discrete on/off type actions

Node	
Name	Bedroom lights
Type	alexa-home
ID	9c6207d7.984558

▸ Properties

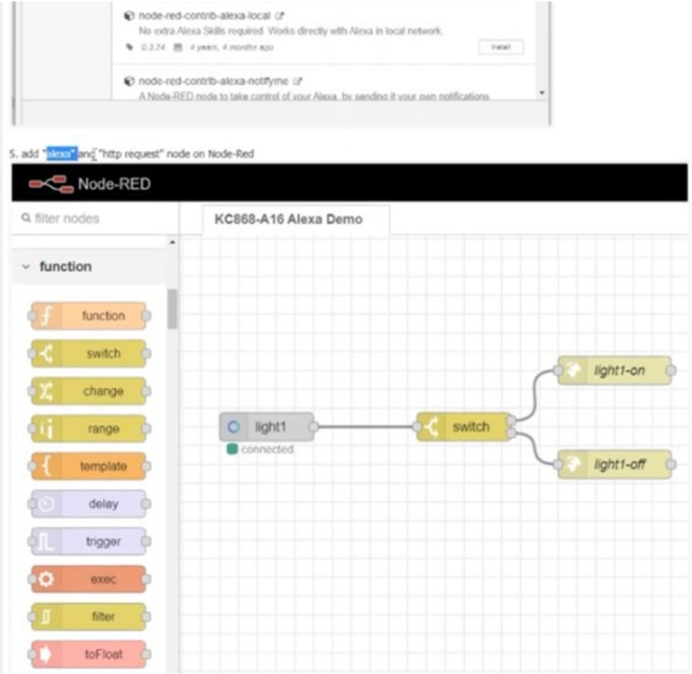
This node provides an end point for an Alexa Home Skill. To use this node you will need to create an account and declare some devices here. You will also need to add the "Node-RED" skill to your Echo device

A full set of setup instructions can be found here

The node outputs a message with a payload

3. create device on "node-red-contrib-alexa-home-skill" skill website

So that will be very easy, we can just click this web page. And here you can just readjust and use your username and password and the email because I have already registered so I just click lodging I use my username and the password and adjust the clicker lodging. And when the first time you just click this device and this will be empty because no LM device, I have created a tool. So I can create it when you first time open this you will see it will say this window. So, we have 16 channel output so we can create the 16 channel device. So you can add a device and you name it right one description us I can input right one this actually just for our and RF maybe such sound device for gamma, but this time just for the MOSFET and this type will choose the switch that will list on the Alexa application. So I can click OK. This is number one first device and I can also input the second device that card read to yes also and the switch type. So this led to and Lapsley we will quit 16 lacked okay, this is the last one night 16 Switch and Okay. Okay, we have quit 16 output of the switch. Now the step two is complete. This is complete. And and the three steps we also is complete this register and this is great device. And now go to the force. The step four, we just open the Node read and install the Alexa property. So I can open this node red. And this is a node i have created before that have showed an Alexa to voice cache. I can delete all as the first time you know the word empty no anything. So you just need to install and the manager and install just input Alexa. That is use this one this skill no don't read Alexa home skill I have installed. So you just click Install like this button like this button. That will install online after Install Complete and then we can see the next step. And then you will see step five. That's just to place Alexa and HTTP request on the node read.



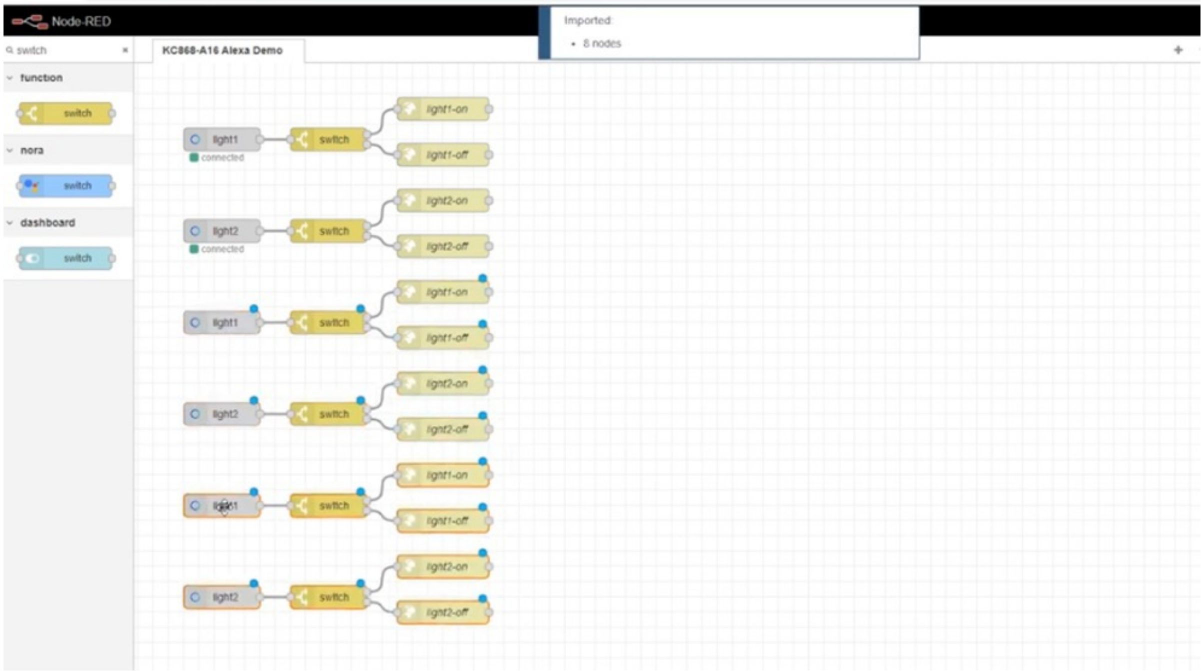
node-red-contrib-alexa-local
No extra Alexa Skills required. Works directly with Alexa in local network.
0.3.24 4 years, 4 months ago

node-red-contrib-alexa-notifyme
A Node-RED node to take control of your Alexa, by sending it your own notifications

5. add "alexa" and "http request" node on Node-Red

So we can see here. I can input Alexa, you can see this Alexa note. If you're not input this words, you can also scores this down, and you will see this new Alexa, note and adjust the drought. And you can click, double click and configure what is actually the King County I have used, you just you can add a new configuration. So I just click Edit. This time, you can input your username and your password, this username and the password just from this website. This website, just this username and the password. This the username and the password and the password just to install the input here. So you can click Update. And this time, you can see that we'll update the device name or from this web page Rajak so you can see the 16 device have update at here that is 16 device. So you just click this number one, right one, this device number one, I can choose this light one and click down. And then you can click Deploy. So you can see this is connected. So this node have connected to this web page, the Cardo server successfully. And then next, we can please start the back, you will see what we have received when you turn on the light. Okay, we have dry this debug node to the node rat. And the next step when you're to use our mobile phone. This node will add later, we just use the mobile phone. Because we need to add this Alexa skill. You can see here is my mobile phone, you can use iOS iPhone, or you can use Google and your phone. So you can see here we can click more. And this is a skill and a skill, we can input this node red, we can just input node red. So here you can see this node red with just a click. And you just enable the scale because I have already enabled it. Now I only issue this disabled. So you need to press enable. Because like this. If I disable it, you just choose enable to use that will link with your Amazon Alexa account. So you'll need to input the username and the password that we have. We're just king Kony and this in my password, input as this password and press this button. And you can see as Alexa is linked, in a crease, close. Okay, we can go on the next step is discover device, just this the last one, we can click this just camera device that will need about 45 seconds, we'll just wait for a moment the speak is such and you can see here many devices ready to use. And you can see here that the light one Alexa application. So you can see I can click this light one now. So you will see the output will be true. And when I click the off, so that will be first. So that word note output is a different shoe and first according to my mobile phone, when country R and the country have. So if you want the Alexa note and the country in this physical light and output, we just needed to use HTTP. Okay, here you can see I can use the HTTP request. And the request we can double click and you can see here, that URL so you can see this web servers. When I click this button, that will turn out right one, just use this URL so I can copy and paste it here. That means name that is light one out. So I can click it down, I just can't use another HTTP request, you can see I can double click, I can also go to here, I can turn off this off, so you can see the URL is off, I can copy this one, and paste it here, this URL is for off. So this is a light one off. So this to note is send a command to this part. Now I just use a switch node, use this switch node and the connector with this light one, Alexa node, and I can double click the first way is a true so it's true that will go to this way and add is false. So we'll go to this second way. So click so, you can see these two note I can connect to this one that is for turn ON light one and this is for turn off right one. So you can see here I can deploy and this time you can see I can use my mobile phone because I have used the voice country's and Alexa will always speak automatically. So I can use this one you can see here I can click Turn now you can see here the light is on and the click the light is off okay now we can test the ways I speak you enable this speak Alexa, turn our light one okay you can see here the light one is now this feedback the true and false Alexa, turn off let one okay, you can see here that it turned off. So, this is very easy just to use this this fi almost three notes that we can delete this one not needed to use and deploy and this is for light one so we can just copy and paste this node or for other 15 channels of output. So we can move it and here we can just a copy and paste you can see here I can double click this time I choose light two. So this light two and the switch not needs to change and this one you just needed to change this this one means the first channel and this number two and this is to that tweet and also this like tweet off so I can go to off and play Okay, let's test it Alexa, turn on led to okay you can see the

LED. So, this node is working Alexa, turn off led to okay you can see here the left with off.



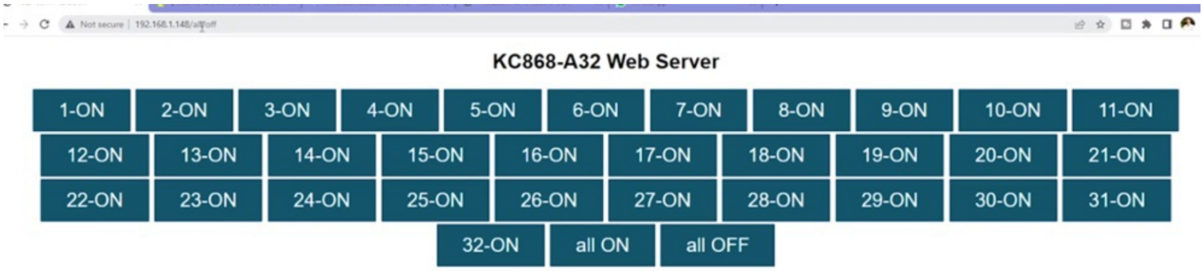
Imported:

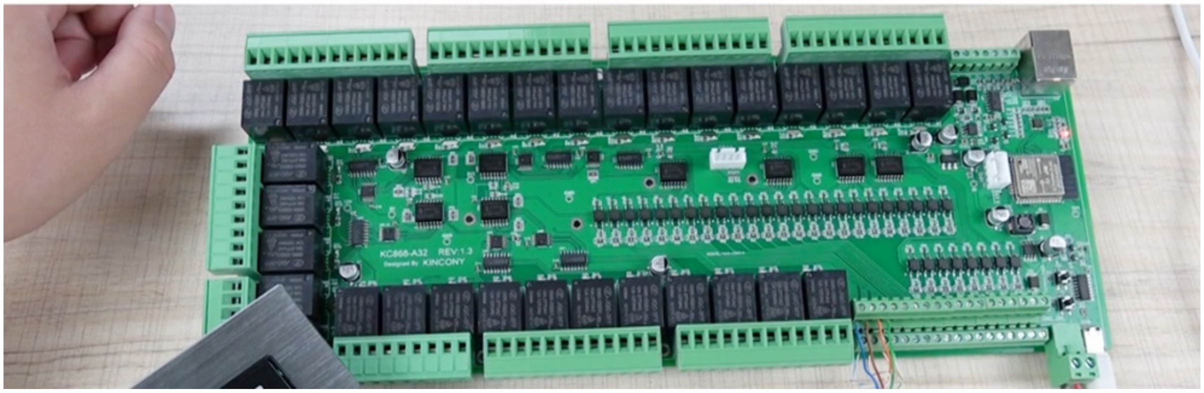
- 8 nodes

So, you can just add this way a copy and paste the four Totally 16 Channel. So you can copy this four channel and this is 123456 and we can copy and paste at here that is 12 and four channel okay. So you can see I can predict this one that you spray that is four and 16 Okay and we just did this and number three the split off okay See see the last 116? Okay, now you can just click Deploy. Now you can use these Alexa to country or 16 channel outputs that have all connected together. So this is how to use the Alexa to control this option output. I will upload this follow up to our form so you just go directly to input and import it not needed to change so much note that it really takes time.

KC868-A32 ESP32 WEB SERVER DEMO WITH SWITCH BUTTONS

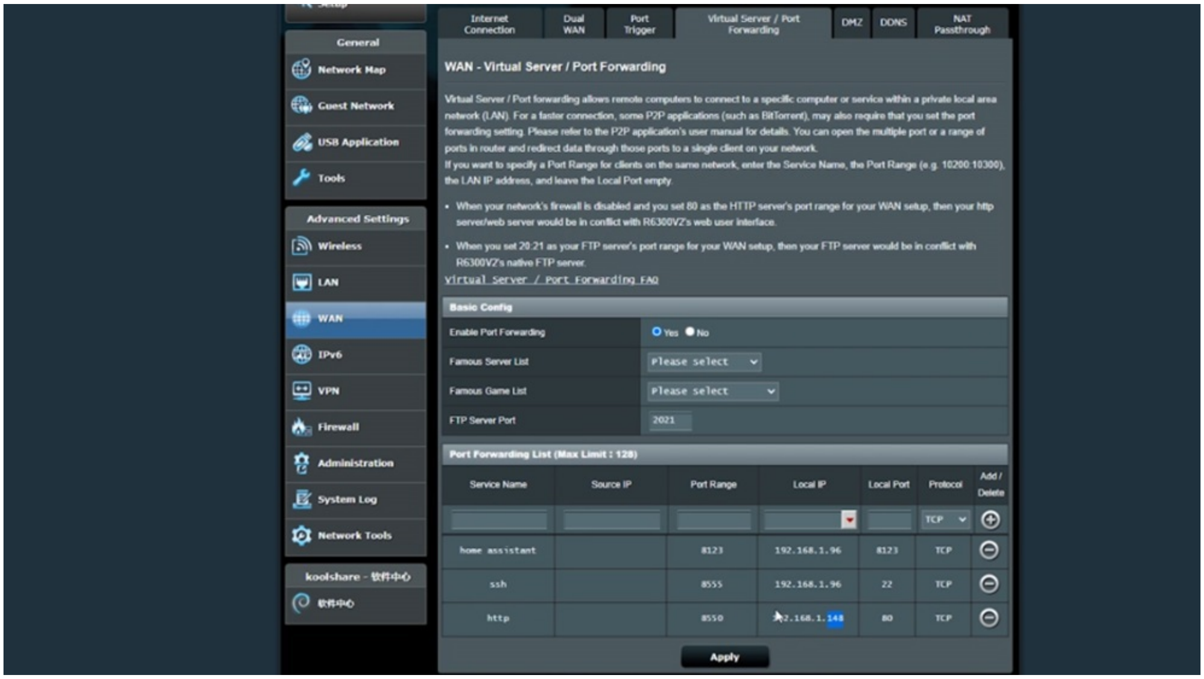
I will show you a solution that is we have a server that focus h six phase 32 This is our ISO 32 part this solution mainly used led this board work as a server, this time you will not help the internet you will not have the car server, but this pod just the home surf is a server. So, that can work in local network, but also you can use the internet and remote country by the DDNS or by the port forward function after you load. Let's look at how it works and its board I have used this switch I have connect with this momentary switch with this digital input of the eSATA two pod and this switch just every channel I will pretend will have one LAN and this is a PC is common that I have connected with this port. So you can see I can use the angular 123456 that I can use this momentary switch. So you can use launch switch just according to Arduino code, because this part is open source, the schematic is open source and the Arduino source code is open source I have already uploaded to our form. So you can download and change it for different switch panel and for different switch type. So you can see I can turn off this is by manually country by this but right here you can see is our webpage on the computer, or you can use the buttons to turn editing off, such as this really one, you can see I can tell this really is and it is I can click again that it took off kind of the Channel Tool, I can turn at a turn off. So I can turn off just like this. So our relay is on. And I can click off. So that's already off. So this is controlled by the web page.


KC868-A32 Web Server



So just use this IP address. So you can see you can use the HTTP command, integrate with a certain part of the hardware. So you can see if you want to turn on turn on a new one, you can see I can press this button at a time to just copy, you can see I can use this URL, if I want to turn off, I just change it to AF so you can copy this URL, and you can see I can paste that here paste and and you can see the link is off. So I just change to oh and enter. So the relay will be out. So this is sent by the URL. And we can see this code a URL in the Arduino code. So, their support code how to turn it off you can see that you know way in off the channel two is on the channel one is on so you can just change this color. So that will be on and off countries that relay and here as you can see is so on and this is our So, this is using HTTP command.

So you will use a pi the PC software also you can use your mobile phone webinar you just use any piece of software on your mobile phone you can see I can open with Barbara and input the IP address as the computers you can see, I just imported this one. So I can use to another one. As this is a real one you can see here to our new one this will be when you turn off and turn off the channel too. And also you can see I can tell and this is off. So this is the prime mobile form. And you can see this pattern also can out suitable you can see I can change the different revolution or different webinar size that will be out change this display that so this is country by the mobile phone and the PC in the local network. But another way if you want to let this part remote country and by 4g or remote a country outside of your home, so there is a way that is used apart for the fashion of your load. So you can see this in my load. I have Largent you just needed to go to the When, and that's how our port forward function such as you can see here I have at this port HTTP service, you can add this HTTP servers port forward a function to your load, so that the local network is at the ATA is used for the HTTP for the web page, and we can put map to this part, this part is just the US not used part that for internet. So, that will be used this eight port to this port 85 and 50 that according to this IP address, this 148 is our port you can see as this is our pod IP address. So, this time you can see I can use this when I pee, you can see here this is the way I presented for Internet IP.



WAN - Virtual Server / Port Forwarding

Virtual Server / Port forwarding allows remote computers to connect to a specific computer or service within a private local area network (LAN). For a faster connection, some P2P applications (such as BitTorrent), may also require that you set the port forwarding setting. Please refer to the P2P application's user manual for details. You can open the multiple port or a range of ports in router and redirect data through those ports to a single client on your network.
If you want to specify a Port Range for clients on the same network, enter the Service Name, the Port Range (e.g. 10200:10300), the LAN IP address, and leave the Local Port empty.

- When your network's firewall is disabled and you set 80 as the HTTP server's port range for your WAN setup, then your http server/web server would be in conflict with R6300V2's web user interface.
- When you set 20:21 as your FTP server's port range for your WAN setup, then your FTP server would be in conflict with R6300V2's native FTP server.

Virtual Server / Port Forwarding FAQ

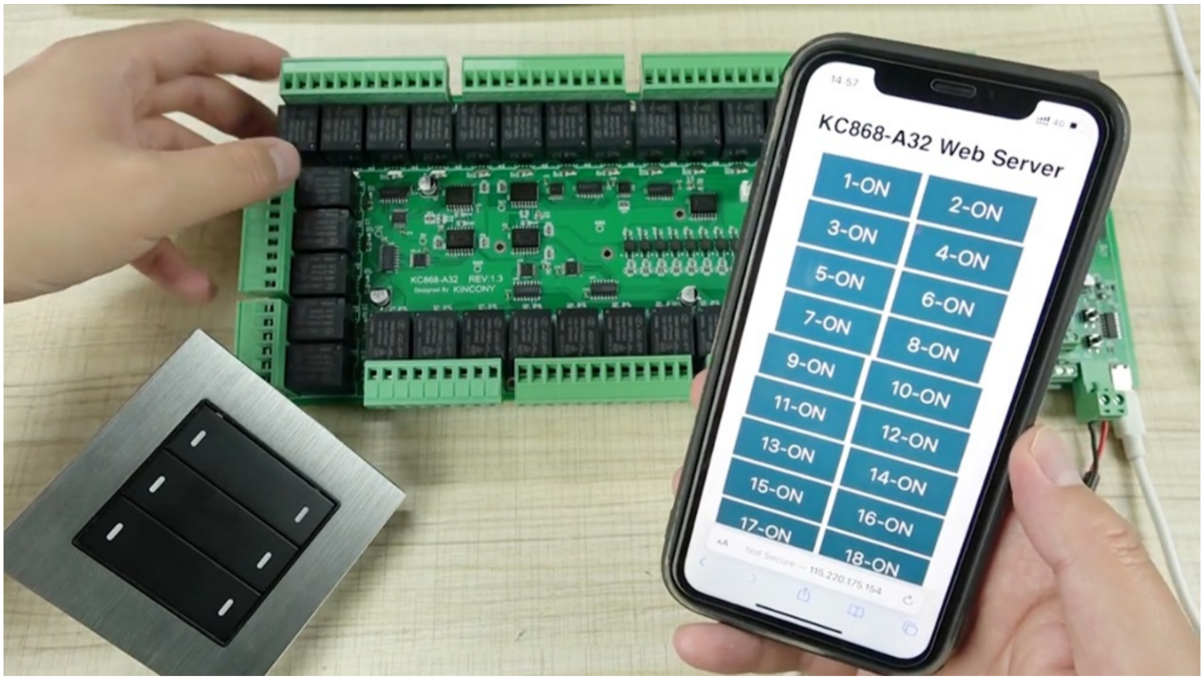
Basic Config

Enable Port Forwarding	● Yes ○ No
Famous Server List	Please select
Famous Game List	Please select
FTP Server Port	2021

Port Forwarding List (Max Limit : 128)

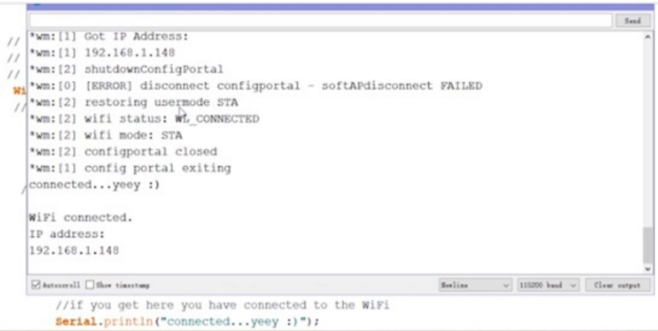
Service Name	Source IP	Port Range	Local IP	Local Port	Protocol	Add / Delete
					TCP	⊕
home assistant		8123	192.168.1.96	8123	TCP	⊖
ssh		8555	192.168.1.96	22	TCP	⊖
http		8550	192.168.1.148	80	TCP	⊖

So you can see I can copy this IP address and paste it here and add is a port and enter or you can see I can also visit this web page or for hour and a half this just working in local network, but also you can see I can use my iPhone this iPhone I can change to 4g this is used by the Wi Fi now I can disable the Wi Fi because I have used the 4g Now you can see here your 4g. So I can open the web banner and input this way I've seen that one one 5.2 To Zero Point this part and you can see enter so, you can see here this is the page and this is by the forges.



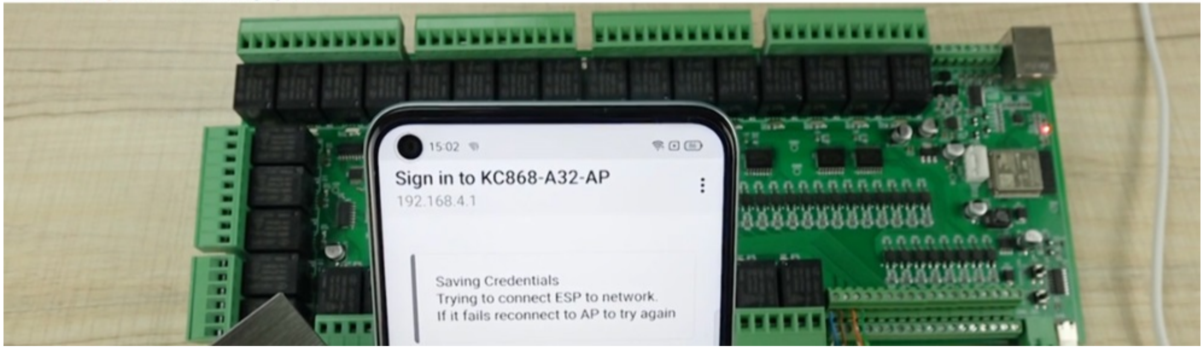
So, I can also you can see I can cut out this really is and to relate to that is controlled by the internet. But this part is a walk has a server I can use all this is on and off this country by the forger and this time also you can use this fan page you can see I can use this wall switch and also you can use this and this one at the same time. And here you can see I have two mobile phone this is by the Wi Fi you can see here that is Wi Fi and this is 4g this a 4g so you can see I can really one off this unfortunate and off. So these two mobile phones are using local network and used by the Internet and work at the same time. So this is for the remote country. Okay, let's look at this is used for a 32 part. Also use this code you can change a little and use for a 16 and a 16 s or a eight, a six and a four part that is widely used by the ESP 32. And if you have interesting you can download our source code, I have uploaded our source code at here, you just can read our form. And you will see the source code you can download. And the way you download you just open this when you download it you just open this Arduino code and you will find here that is our source code. This you need to install for us to this is I squared C chip the library for

Arduino and this Wi Fi manage that is the library you need to install. Just install the library and the managing library. So you can just input Wi Fi manage and you will searched and this library and also you can input a message and this time I will show you how to set the Wi Fi SSID and the password. You can see here the code actually you're not needed to use this set up SSID and password in your Arduino code because we will use this mobile phone to configure the Wi Fi. This code I think you can you need to delete it or comment. Because you can see here we have used the Wi Fi manage so that you can use the web page. I have used this Wi Fi Manager, you can use this to configure the Wi Fi by the APM or the first. That's the first you will see this. So the two h and the US you will connect it and to configure the Wi Fi SSID in the password. So I have downloaded this code we can see the ceremony it and the hot hot coffee is the Wi Fi Wi Fi. This pattern we have defined for the clear the Wi Fi information. So is it actually the Wi Fi reset. So you can see we can for demo. The first time you have to use you just click this button. And you can see is that is clear the Wi Fi and enable the Wi Fi AP mode that will have the SSID for this AP mode. So you can use your mobile phone at firstly configure this h 32 port Wi Fi you can see here and click the Wi Fi you will find that is 32 AP that is signal for this ESP 32. So you can click this one. And we can see this is a web page you have see displayed so that will configure the Wi Fi easily by the web page. So you can configuration for the Wi Fi and then we'll scan the signal of your office or your family or your house you can say there are many Wi Fi signal have scanned I will use this can Connie is my Wi Fi. So we can pick up this SSID and the you just need to input the password okay, you can click Show password or natural password and just click save and easy which configures the ESP to Wi Fi and set to sta mode for the ESP 32 And you can see here the sulfite output that have conducted the year that is successfully and you can see the mobile phone will disappear and connect to this company.



```
*wm:[1] Got IP Address:
*wm:[1] 192.168.1.148
*wm:[2] shutdownConfigPortal
*wm:[0] [ERROR] disconnect configportal - softAPdisconnect FAILED
*wm:[2] restoring usermode STA
*wm:[2] wifi status: WL_CONNECTED
*wm:[2] wifi mode: STA
*wm:[2] configportal closed
*wm:[1] config portal exiting
connected...yeey :)

WiFi connected.
IP address:
192.168.1.148
```

```
    //if you get here you have connected to the WiFi
    Serial.println("connected...yeey :)");
```



Sign in to KC868-A32-AP
192.168.4.1

Saving Credentials
Trying to connect ESP to network.
If it fails reconnect to AP to try again

And here you can see that is the IP address have displayed. So we can use this IP address copy and paste that to your web bar. So you can see I can paste it to here and enter you can see this webpage and you can and are and also you can see this font list import and also you can go to the web round and you can see here is view the customer list. And you will find because I have many device you will find that this ESP 32 Is this IP address. So this have found in the in your Lord, just use this IP address and the logic in the web part or Racz in your mobile phone. So that will be easily to configure the Wi Fi and the country in this relay output and easily for integrated part of basic HTTP commands for the server part software or hardware device.

KC868-AI SENSOR MONITOR DHCP LAN WEB SERVER DEMO CODE FOR ARDUINO IDE

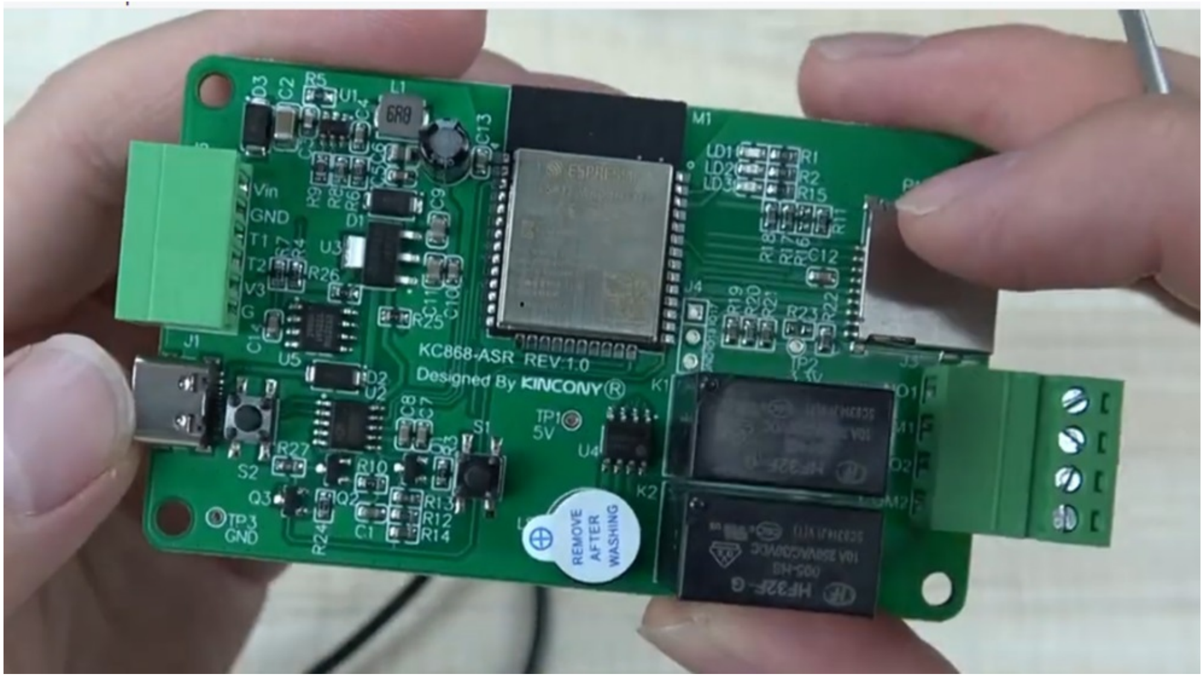
I will show you our solution for the web service you can see here is our before video where we show the lamb web service with a 16 s pod that is the following output, but this time, I will show you another solution that is for input. So, we will use this case each six eight API the input pod for is a web service that you can monitor as a state input the state on the web page or by your mobile phone or by your PC software. So, you can see here you know source code we have uploaded to our form you can download after you download this code and the USC This is our source code and this is being found that is from where you can just add it directly to us and you can you can not write any code just download it to our ESP 32 module, how to download the this freeware to this module, you can see our before video help details. Now, I will show you the results after we downloaded the freeware and you can see I can open web part. So this is a Web Part that have 1234 to 48 channel input. So you can see here you the 1234 24 and it is a 25 on 248. So that is for the digital input and this line this terminal and the way your sensor maps support when sound meter enough for you to use. So you can see what I shoot for this last year ground shot for this channel one because this is

Channel One number one input I shoot for this and you can see this a turn red, so that the sensor your child and if I release my hand this not shot so it will recover to green so this is not triggered. So you can see I can trigger the number two input and you can see the number two and on cue this is 48 this 48 I have triggered so that will turn red. So this car monitor the different sensor if you have project for many sensor that will be very easy to also to support HTTP command. So this is by the PC software we can manage. And you can see here the WebPart I can change the size of the web browser that will auto size are for the window. So this is a 48 channel and it is about PC software. And maybe you will ask me How about its work in the mobile form. Here you can see this is our NGO form. You can see here how many input part that 48 channel input apart I can put that here and this in my iPhone, the iPhone also have 48 channel so you can see I can put up here is that how this to mobile phone and maybe you will have more mobile phone stream about four or four mobile phone is no problem.

And now you can see I can shut for this I shut for the channel one input one you can see red and the red and the PC software you read. So they will feed back the state at the same time for our device. And I shot for this this terminal. You can see I shot while this is turned red. And you can see how to turn red. And I released you can see it's recovered. So I can shoot for this 48 It's a 48 this is at last this last one that red. And this website will auto click an auto refresh. So and you can change this, this I have set the three seconds to refresh the webpage. So if you want to change the refresh interview time, so you can just open this additional source code and find out how you can setting as this command that is two means two seconds every two seconds to refresh the webpage. So you can change it to 123 or 0.5 that will be very fast refresh. So I think this is useful for your project. It's up to you what time the interview you have to send. Okay, this you can see our code, just download the front of this thing and you can change it and the use for your project. This is monitoring priser network cable that to the four Ethernet that will be stable and for monitoring all sensor states.

KC868-ASR ESP32 SD CARD DS3231 RTC 1-WIRE SENSOR BOARD

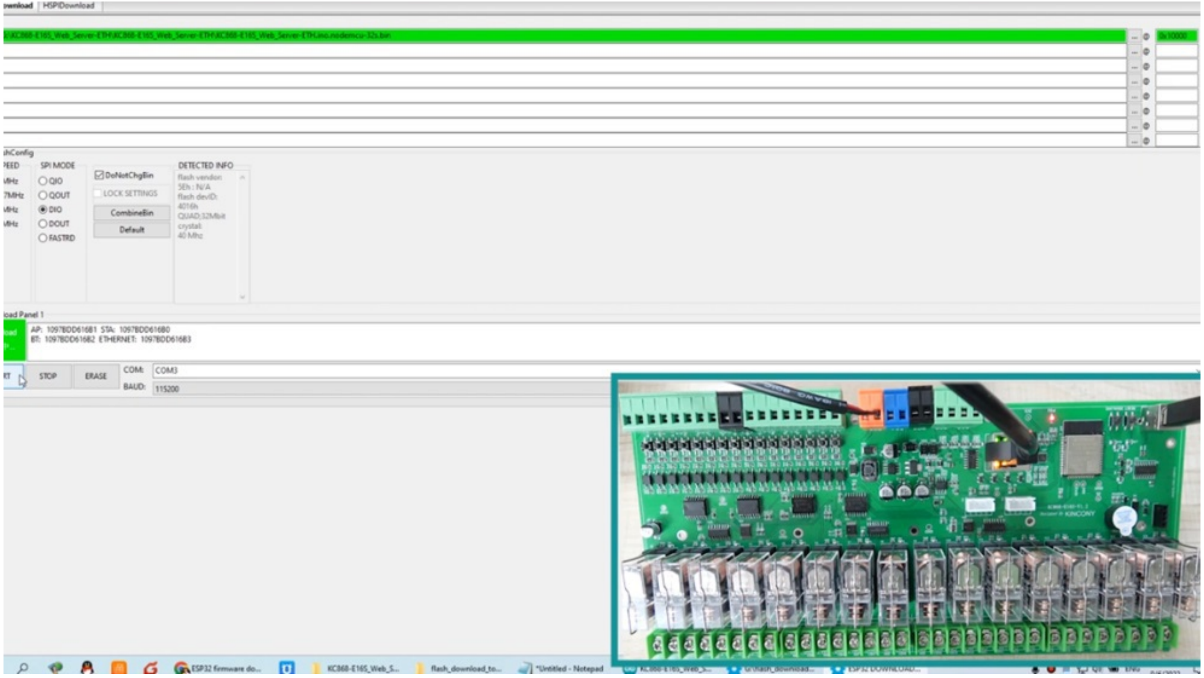
I will show you our new product. This is a very small PCB board we caught case 868 ASR because we have installed this SD card on the board and also we have designed our RTC chip high quality cheap on this part. And in this folder will have two channel relay and two channels of digital input we use for when were sensors such as temperature sensor and the hammer the sensor so this part will save this sensor date to the SD card. And also you can see here the battery that is for Save the clock. So you can use this part easily to save this temperature sensor and the humidity sensor date and the to the TXT file or to the Excel file. Then you can put this SD card to your compute and eluded the date and the leader the file to your SD card so that you can list it and assure you the history of the census state and also this folder with the date made for ESP home so that you can integrate with your home assistant. So let's look at this hardware details. Okay, let's look at this as awkward. You can see this is in front of a PCB and this is the back let's look at this details that ESP 32 module. And here we can see that we have V power supply, power and ground and it's a T one and T two this T one and T two you can connect with a temperature sensor Yes 18 B 20 Oh this you can see here temperature and humidity sensor the HT one one or the h2 two o th two one just use this one and connect with this socket the socket as we removal so you can easily know have the school just hold down. So you just insert the cable and Akua let's look at this part.



And you can see here that is sway vote and ground and this is for USB for download and these two buttons one button is for ESP 32 reset and one button is GPIO zero with ESP 32 So you can use your code for your own functions. And this is speak and this is a really two channel relay you can see here so really we have just Tampere has a max of load this the terminal of the relay output that have the calm and a normal open a calm and normal open also removal you can easily connect with one and this is a microSD card. So you can insert the SD card to this socket to save your sensor data or save your history data. And you can see here the pantry the pantry is save the time clock because you can see here we have added this high quality RTC chip that DS 3231 chip for the clock that is very good for the time schedule functions. And for your project to write any additional code we have design support and a set for this plastic issue. So you can see here and this box also can support installed on your DIN rail. Just look at this way you can see I can input to this chord and close this chord and you will see this is a four sensor and these are four USB type C and this for relay output the terminal and this you can see that the SD card with the insert SD card or get out to the SD card. And this you can print your own logo or own test on this on this practical issue. So you can see just a Nexus this position you can install to your power distribution box. Okay, this is case 868 ASR port, and hope you make your own home automation project.

KC868-E16S ETHERNET LAN WEB SERVER DEMO CODE FOR ARDUINO IDE

I will show you this 60 s pod and work with the Ethernet cable. And for the web server that uses Azure to dB command, you can see I can use this word partner to turn out and turn off the Array, just a click button for and for off. So this can also turn on all that for all days on and off, and check to know every different relay. So that will be easy to integrate with a certain part of software. So this time, we have used the internet last video we have used the Wi Fi, but the Ethernet will be very stable. And also this part will support the menu control button, you can see here I can click and our bar is a button. So that I can connect with this switch panel. The switch panel will try contact signal just to connect with this terminal and to this terminal. And I have used this button on the part just for testing. So if the panel is broken, or the part is broken, or the internet is broken, so you can also use this button to turn on and turn off. Okay, this is a function of the Webster but how to use this freeware. You can see here we can upload as source code and as a freeware on the form of the 60s. So you can download this zip file and unzip this file you will find and this is for where and this is called I think you just locked in it to use just use this ping. So also you can download this ESP cert here to download the Tor. So this can download from our website and this file. And after you download, you will see this package, you can just click and choose this chip type ESP 32 and press OK. And here is the first learn, you can import your past of your freeware just you can see here, your input to the past of the freeware and the must input the address at this number. So as you can see here, we have not this address and this purse. And now you're just connected with a USB cable with your PC, I can connect with it. And you can see which components you have used, you can see the Device Manager in Windows. And here we can see I have used comes through. So I just click and choose a comes through and just click Start.



KC868-E16S_Web_Server-ETH\KC868-E16S_Web_Server-ETH\KC868-E16S_Web_Server-ETH.ino.nodemcu-32s.bin @ 0x10000

SPI MODE: ○ QIO ○ QOUT ● DIO ○ DOUT ○ FASTRD

☑ DoNotChgBin
□ LOCK SETTINGS

DETECTED INFO
flash vendor:
5Eh : N/A
flash devID:
4016h
QUAD;32Mbit
crystal:
40 Mhz

AP: 1097BDD616B1 STA: 1097BDD616B0
BT: 1097BDD616B2 ETHERNET: 1097BDD616B3

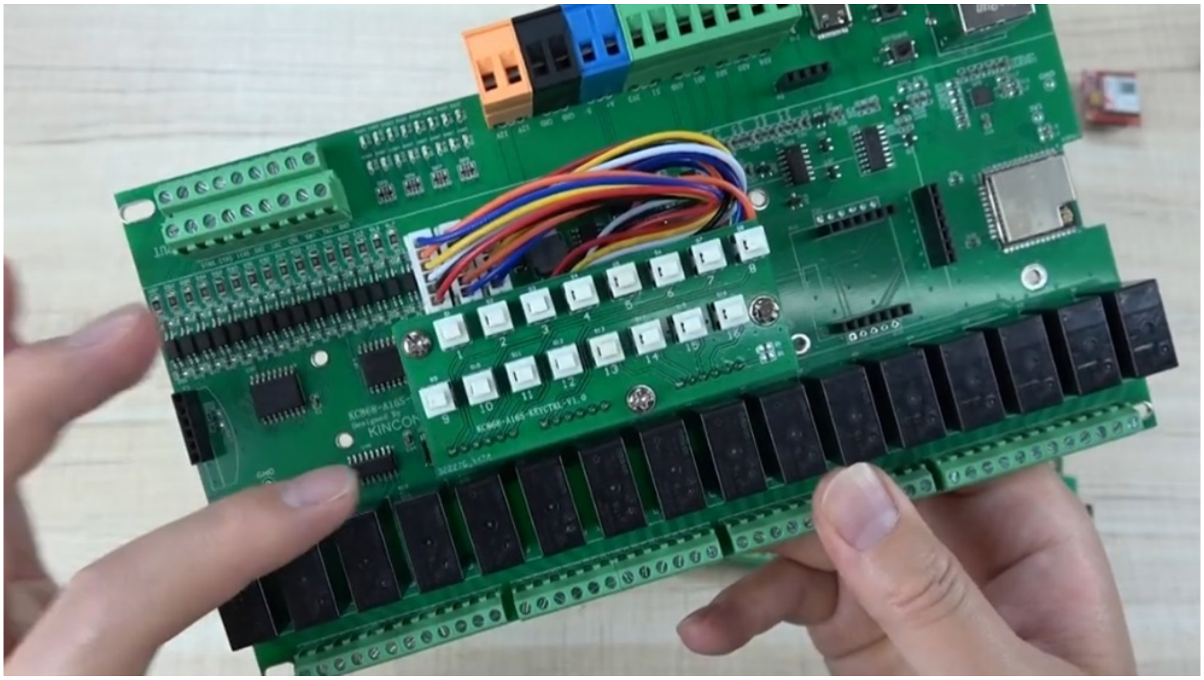
COM: COM3
BAUD: 115200

Center will began to download. So just three step, the first step input, this path of your freeware and the second input address. And the third, just choose the comport and then just click the start. Then we began to download after the complete download you just needed to power or press this reset button. So let's wait for a moment. Okay, now you can see it's finished. When the finished you can remove this USB cable and press this reset button here reset button you can press and you can see the network cable LD will be blank.

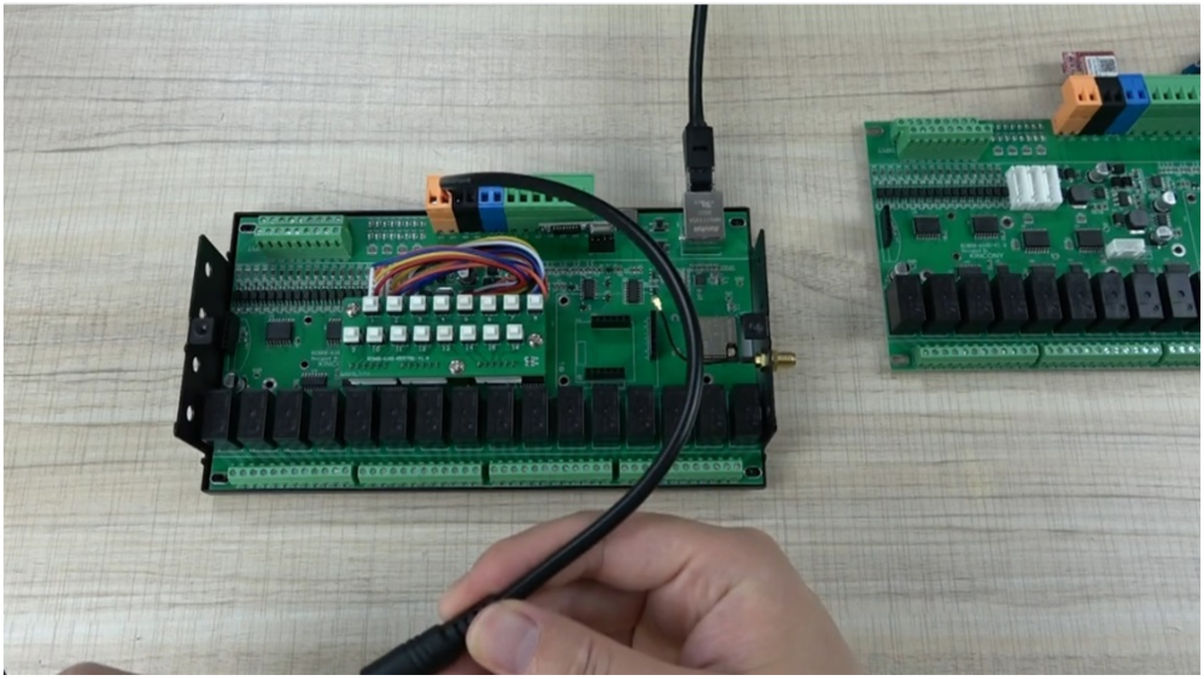
And the waveform won't because now we have configured this part of the network with a DHCP. So you can check your route. And you can check this we have used this IP address that have a WT 32, eth 01 That is our 60s Park. So you're just a few this IP address to the one partner and like this. And enter then you will see this terminal. And you can just click and for AF and here you can see this, our Arduino code is another file and record you can see here that have the code, the default DHCP if you want to set the static types, so you can see you just copied and pasted this code just for your IP address for your gateway for your DNS. So you can just copy this code and paste it here, just copy and paste it here. And the last just uncomment you can see here, you just let this command enable. So you can see here have a note, if you have static IP address, you just let this command enable. So if you want to do the DHCP, you just uncomment so that will not use escapes. So this will very easy just choose a static or DHCP. Okay, this is for networks that and maybe if you were asking me if the network is broken, whether the command can be controlled by this menu control button, so you can see here, I can click on and off. But this time, I have removed this network cable, the cable is broke, the network is broke. So you can see I can also turn turn off this button, different button. What are and so this is work in locally for the manufacturing button. So that will be 50 for the part. And if you have connected with the table again, so you can use the web part again. So this is for local network country and for the network country for the ESP 32 for the East 60s bug that can use the HTTP command for the web server.

NEW KC868-A16S ESP32 HOME AUTOMATION RELAY BOARD FOR HOME ASSISTANT

I will show you our new product. This case 868 a 16 s part. You can see this part we have designed by the ESP 32 module and it also will support to the home assistant by the ESP home. So let's look at the hardware details and the second I will show you how to work in home assistant by the ESP home. And in this part you can see that help 16 channel relay and digital input 16 channel and as for it fi and power supply and analog input and there's a GPS device such as the GS 18 B 20 temperature sensor and as you can see here is the type one sound details there are many interface the pod and our relay that will have common and normal open and normal clothes and also support the fluids module and the tuition module used by the GPS Oh GPS. So you can see here let's look at the details the ESP 32 module and the Ethernet port and the reset and the GPIO zero and the USB type C and this is I have module if you want to use a note you can see you can use this and plug into the socket and so that you can use this remote wireless remote was actually megahertz remote and we'll remove it and this early indicate that for relay output and this is a nice classy interface extend and this is the RTC module you can see that TS cert history and a 31 This is an RTC module. So if you're using it you just plug it into this socket and here you can see that have some interface because we have designed the menu control button and you can see this is for 4g module and this is for the Twitch module that is thing 800 for Twitch module you will just plug it into this this socket so that you can use this module and install this sim card at this target Okay, let's look at this PCB board. So that you can understand easily and you can see that we have designed this menu controller button you can see here this the 16 channel menu control button that can control this 16 channel layout directly. So if the home assistant bed or the software is bad oh the ESP chip is bad. So you can use this country this relate directly lit I will show you how it works in the demo video.



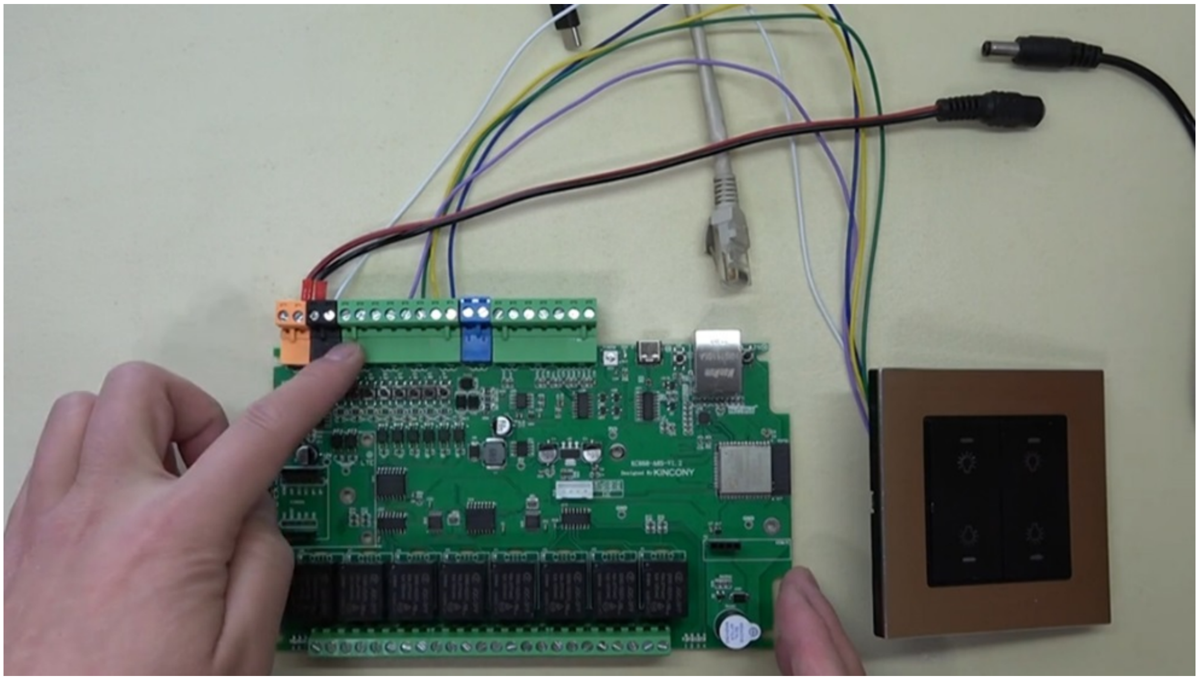
Because this is a physical and work with the relay without any software and the framework, we have used the term parody you can see that is a high quality homerun really that Max is 10 pair. So, that will be enough for you for many ePrints to use. Okay, this is a bar and if we install the forged model on it, you can see this part this part we have installed the 4g module and install the SIM card at the socket and in this menu control button and this the RTC module. So you can see that have fixed the paisa school. So that will be very stable and this bottom part and this the 4g that 4g part. So, you can see that is installed as a four module when two three on the PCB part and this is the back back of the PCB, this is in front and as if you want install this bar to our box, this time we have designed us this black box so that you can see if I installed this part PC back to this box you can see here this metal box and spark installed on the DIN rail. So you can see that will be easily as part of installing your power distributed tissue box. So you can see that I can close this cover. Just select this you can see this is appearance and the UK is the antenna according to your place because there are many holes for your install the antenna you can change by yourself which heart we use. And if I installed this remote Listen I have received module so you can I can plug in so that oh work with your remote. Now I will show you how it works in home assistance we can open this box and connect with the Ethernet cable I just use the Ethernet so that will be stable. You can see here this time I'm not used the Wi Fi so I have not connected when you use the Wi Fi you can connect with this socket to this IPX socket I just afford save the time and show you the demo and the Ulis DC 12 watt power supply and this socket also removable so I just connected with this power.



You can see I power and power LD is now the ESP 32 is no firmware no software, but you can see I can press this button for when you want you can see the LD really one and you can see the number two straight for so I can tell tuna tuna tuna tuna Tom manually manually country by this button, I can turn off. So this can work without any firmware without any software just contribute directly. Okay, let's begin use the home assistant. And I will download the firmware by the ESP home and connect with a USB cable with Raspberry Pi. Let's look at how to configure the home assistant. You can see in our phone, there are a 16 s there are many results just you can see here, the Configure Yama file for ESP home. So you're just to copy and paste that will be very easy. Because we have created the home assistant config file you just copy and just copy and paste and you can see here in our home assistant I have created a testbed. So I have created a device the ESP home and a 60s and edit this just a copy and paste I have already pasted it here. So I can press save and install and by the USB, click this and buy this USB and wait for mount and then the firmware will download to this USB and to the ESP 32 module. Okay, now is complete, you can see the logo file and connect the starting the Ethernet and the finish successfully. So I can click stop and close it. And the USC is a 16 westward that change to online state. And now I can change to the dashboard. So the follow up will help create a 60s power output and input. So you can see I can click an x ray one you can see the red LED is now and you can see I can turn on the second and the third and close close. And I can also edit it and let it have this on and off but so I can click see. And you can see this button and I can click so you can see already 16 channel is now and that 16 channel is our and it's the input stage we can use this for sensor or for switch you can say this the input one that's the default trade off what I use my twice and the shirt for this input one and ground you can see here I shot that is out and released that is our so this you can use for any dry contact sensor for this part. And you can see that in fact you can see how many input that input one and input two one and two and until 15 and 16 and so, so that you can use many switch button of your sensor and you can see if I shut for the ground and the second channel and you can see that input two input two is will be out. So this is used for the input state and this is a for output OK, so now you can see if I remove this network cable so that you can see the home assistant dashboard. Not work can't work. But this time if I press this button, it also can work. So this can use for emergency for many countries and also I can remove this USB cable and you can see that can work. And if I connect with this is another cable again and you can see the home assistant is refreshed so that you can see I can click on and off and off again. So this can work again. But remove this cable and this can't work but or this button it work perfectly. No problem. So this is for the emergency use when when certain is bad or the system is bad.

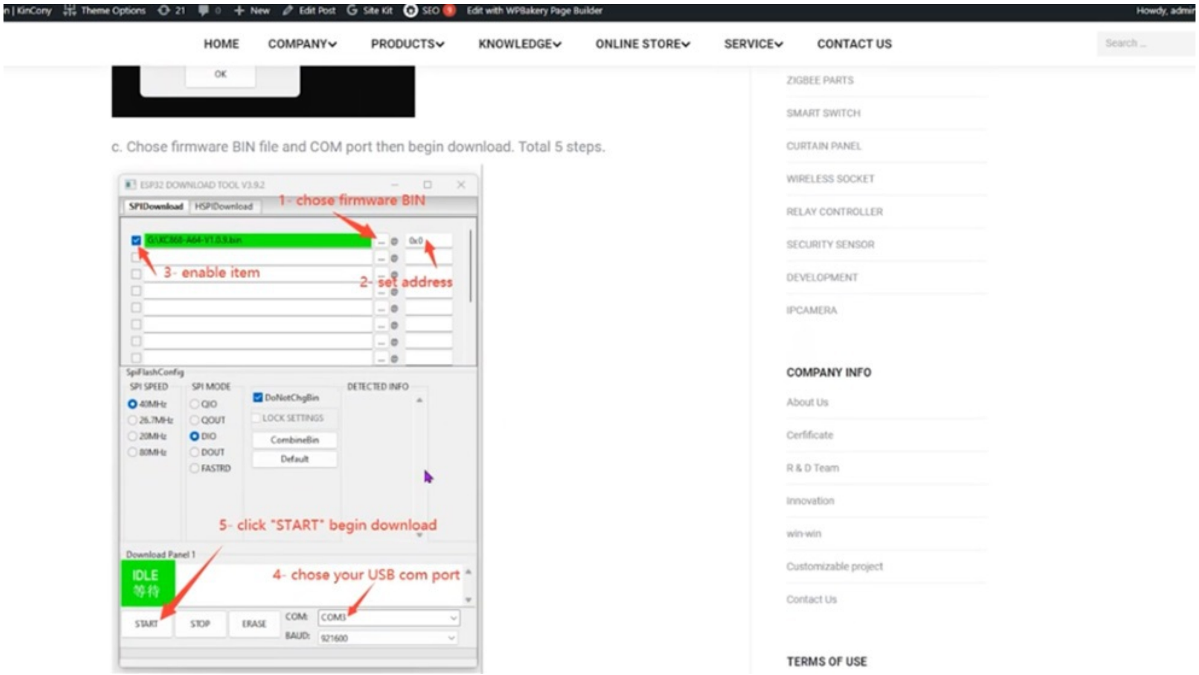
KCS_ FIRMWARE FOR KINCONY ESP32 BOARD DETAILED EXPLANATION

I will show you how to use this KCS firmware that is the newest Khinkali firmware for ESP 32 part. So this is very powerful and that's the first we will look at what's the KCS under the case as you can see the website that means the King County super because the firmware that development back in County in 2010 this way, so this is the new year stuff we're in Wales. So that can let your ESP 32 pod is very powerful. So that you can see here have many different pages for different functions. Now we will look at this firmware that will almost support Khinkali ESP 32 board you can see here that is a four A 688 s and a 16 and e 16. Yes and Acer 32 and a 64 and a 128 actually almost always ESP 32 part was the part this case the s firmware and we will continue to make that update after the KCS firmware such as our new board are based on the ESP that will be integrated to the KCS firmware Okay, let's look at this user guide step by step. So that you will see how is it powerful and here you can see we have used this it as board for example actually you can see such as a four and a 64 and AI and many many ESP 32 offer the King County Board will support KCS and this time we just take this for example because this is a typical part that is how the digital inputs and the analog input and have the relay outputs and you can see this board I have connect with this voluntary switch. So I will show you how to let this whole switch panel work in different work mode.



And at first you have this ESP 32 port you can connect with this USB cable and with your compute, this is my computer so we need to download the firmware firstly so you can see I can connect to this USB type C cable and connect to yet and as a part you can even use the Ethernet or use the Wi Fi this time this Ethernet and the Wi Fi can work at the same time so that you powerful function and at Later I will show you how it's powerful. Now I connect with the Ethernet cable and the USB type C and this is for 12 watt power supply. You can see I can connect with this power supply 12 watt and early however it rarely is on so let's begin with the software to download for the firmware to the ESP 32 module firstly Okay, let's look at this step by step you can see the first you need to download the firmware to the ESP 32 Park. So you can use this ESP 32 Download a tour just at this tour. So you can download it after your download you will see this file and this file where you will need to use and the firmware you can download the form that can come in form. Here is the company form. You can see here that is a KCS form we are system. So you can click and actually we will upload the firmware ping file to here now is ready to which the New

Year holiday finished so that we will upload it to it. And here you can see this protocol, some HTTP command Modbus command and MQ add command and the string command. So this, this different protocol will support in KCS. The next I will show you how to enable this protocol. And here you can see that we have the ESP 32.



c. Chose firmware BIN file and COM port then begin download. Total 5 steps.

1- chose firmware BIN

2- set address

3- enable item

4- chose your USB com port

5- click "START" begin download

☑ D:\KC868-A64-V1.0.9.bin 0x0

SPI SPEED: ● 40MHz ○ 26.7MHz ○ 20MHz ○ 80MHz

SPI MODE: ○ QIO ○ QOUT ● DIO ○ DOUT ○ FASTRD

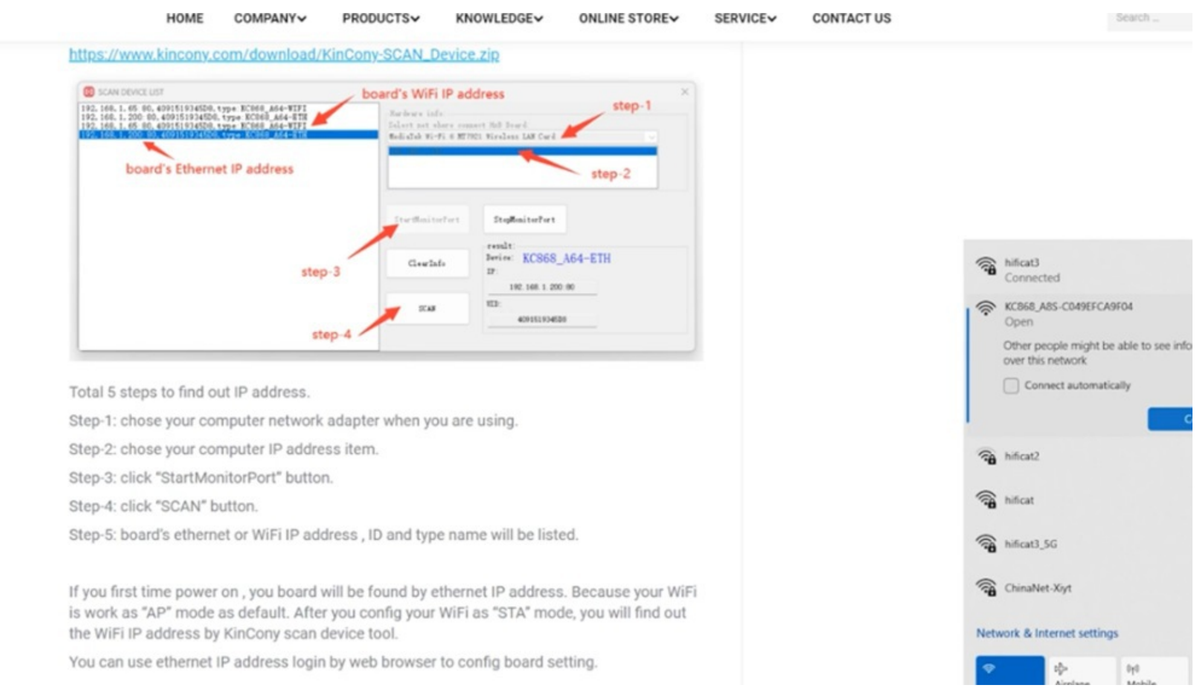
☑ DoNotChgBin

□ LOCK SETTINGS

COM: COM3

BAUD: 921600

Download the tool, so we can click this one. And this chip type will choose the ESP 32 Oh, our partner will use ESP 32 and click OK. So you can see this window and you just need to choose this ping file as this eight s bar because I have used this bar and this ping file for testing. So you just choose this. Select this file and To adjust you can input the ad just to begin with. And it is a default of positive birth. And here you can see that in the component this component because we have connected USB that how to detect it comes through and this is speed download the speed I choose this went that fast. So you just click Start that will be began to download. It's very easy to download the firmware wait for a moment and just after you download finished you need to re power out of your pod. So okay, this is complete and we just close finished and the crust we can go on. So now you just need to let this part power off and power on just a reset or you can press this reset button or your thing. So you can see now the part have restart okay after this board Power Hour, so we can use this King County scan device store to find out the Ethernet IP address so you can download this file I have already downloaded just this well we can click open it and now you can see okay I have the compute the tablet I have used the Ethernet and the Wi Fi because my computer have used the Wi Fi connect to the load. So I choose the Wi Fi network card. If you use the Ethernet come off the computer. So you can choose the Ethernet because usually the tablet, the tool network card and choose after to this network card, click this IP address this IP address is your computer's not the parts. So you can click this one and click this button start part monitor part. So we can just scan click a scan. So this you can see this one point 14 Is your partner's IP address. And you can see here the IP address and this is a UID that is the ID of the ESP 32 So you can see we just do this five steps you will find out to your pod the Ethernet types they say eth that means the Ethernet if you have set the Wi Fi because the first time the Wi Fi is walk as AP mode so you will see this Wi Fi signal that has our H s pod. This is a UID and this is the model of the pod. You can use Wi Fi if you can't get to this part just the way the AP mode if you set to sta mode in future, so you can see you will find this is Ethernet IP and the Wi Fi apps at the same time at this window so later I will show you how to set the STA mod of the Wi Fi we can go on that we have C HC mod so if you use this AP mod if you want to use the Wi Fi without Ethernet cable so you can use this 4.1 and this to logic webpage.



https://www.kincony.com/download/KinCony-SCAN_Device.zip

```
192.168.1.65:80,409151934SD8,type:KC868_A64-WIFI
192.168.1.200:80,409151934SD8,type:KC868_A64-ETH
192.168.1.65:80,409151934SD8,type:KC868_A64-WIFI
192.168.1.200:80,409151934SD8,type:KC868_A64-ETH
```

Select net where connect NoB Board
MediaTek Wi-Fi 6 MT7921 Wireless LAN Card

result:
Device: KC868_A64-ETH
IP: 192.168.1.200:80
UID: 409151934SD8

Total 5 steps to find out IP address.

Step-1: chose your computer network adapter when you are using.

Step-2: chose your computer IP address item.

Step-3: click "StartMonitorPort" button.

Step-4: click "SCAN" button.

Step-5: board's ethernet or WiFi IP address , ID and type name will be listed.

If you first time power on , you board will be found by ethernet IP address. Because your WiFi is work as "AP" mode as default. After you config your WiFi as "STA" mode, you will find out the WiFi IP address by KinCony scan device tool.

You can use ethernet IP address login by web browser to config board setting.

hificat3
Connected

KC868_A8S-C049EFCA9F04
Open
Other people might be able to see info
over this network
□ Connect automatically

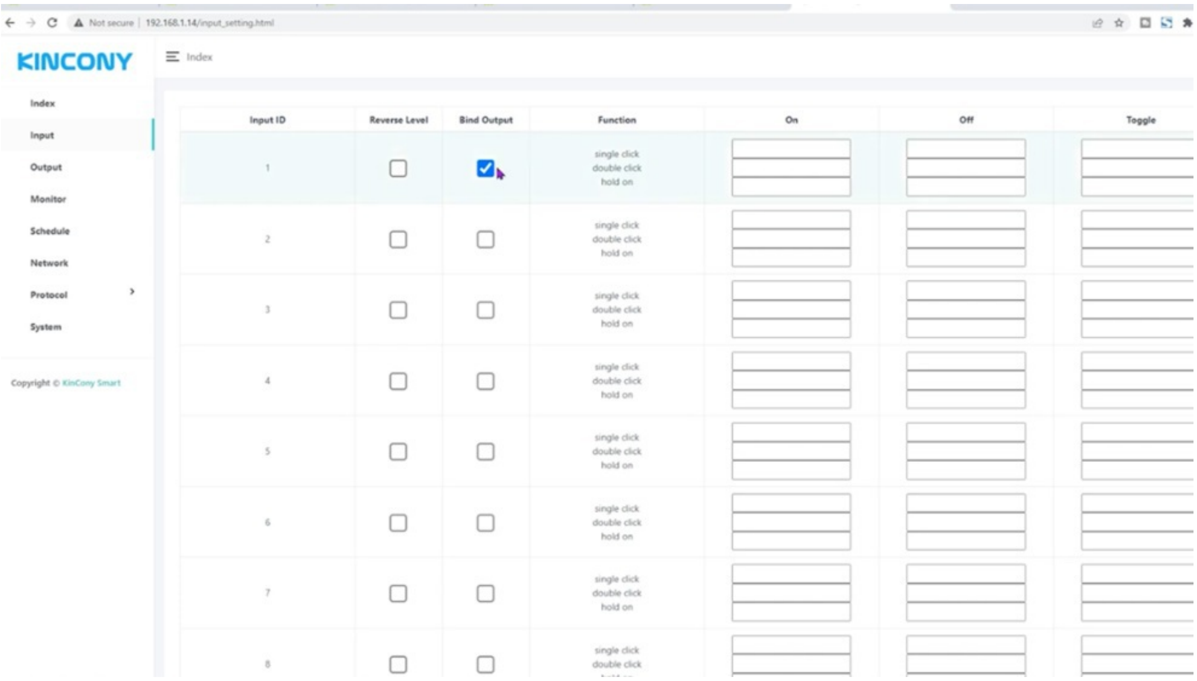
hificat2

hificat

hificat3_5G

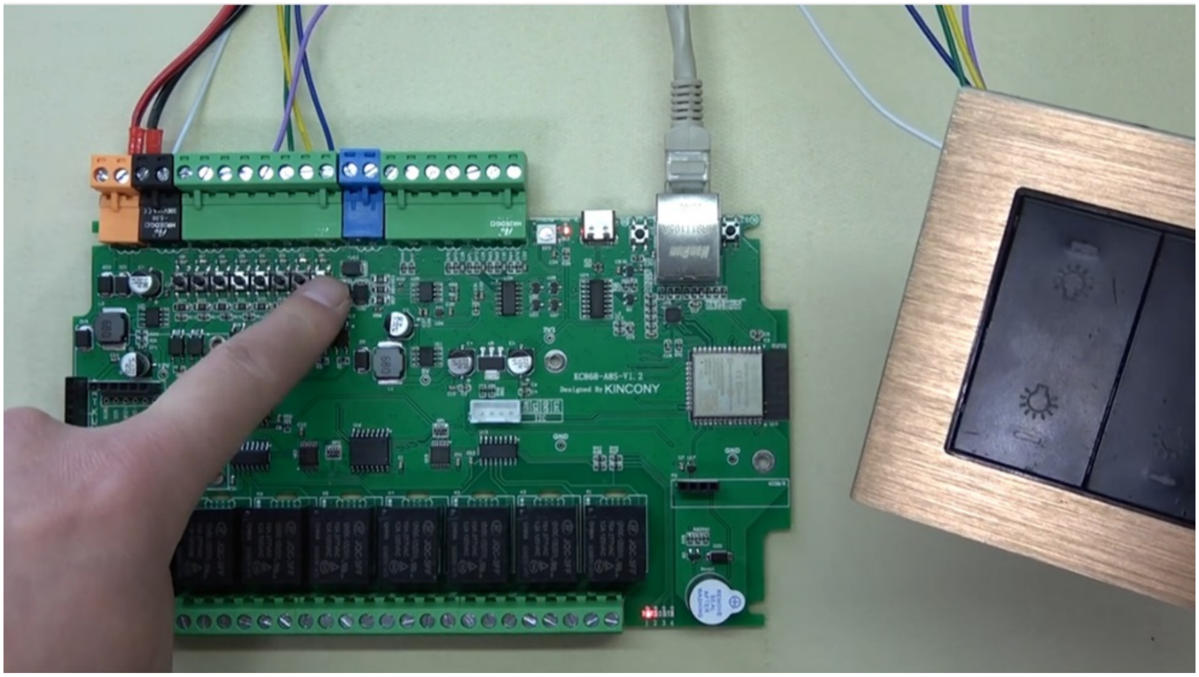
ChinaNet-Xiyt

And now I have used the Ethernet because it's energy stable. I use this IP address lodging web part so you can see Yeah, I input and you will see as this is a web page for management, you can input the username and the password the default or is admin admin. So this is username and this password also use other means so click lodging and you will see this window this is the homepage is homepage you can see the port module that you have used a test pod and this is after where from where was done Watson in future we will update it so you can compare this versus whether you're new or old and this will appear the date that means the freeware with what time and make by the King County company. So this is very new and insecure January and 18. And this is the remember that is ESP 32 it and this time, the border style, the border style. You can see here because the time do you need to set Firstly, in China, I mean China so and the two sets the UTC that go to this system. And you can see here you can send to your own UTC timezone. Because in China you add eight. So, I can save this firstly, you can see here when I refresh, I go to this homepage that terrible is UTC at eight. So, you can see now this date and this time as soon as the computers. So this have alt update to the date time from the internet. And you can see here that that is saying okay, this is LAN IP that is scanned by the SR, this LAN IP you can see here that the MP and the net mask and gateway that auto select this Wi Fi IP if you have set the STA so you can see here the network because the default is Wi Fi is AP mode. So you can sell to sta that will connect to your own load. And I press save and press OK. And this time I refresh this web bar you can see this Wi Fi IP you can find that this Wi Fi IPs. So if I have used this scan device, I can clear and the list can click this button and you will see that we'll have two different IP address and this is the full IP address and this is for Ethernet. So that eight s Wi Fi and this is a it has eth Ethernet. So, you can also use if you have close this window you can use this IP address to lodging this web web page. So you can see I can enter this IP address and you can see this also can open this webpage right because I want to the speed fast I use the Ethernet so I can change to Ethernet and the larger again okay, we can log in by the Ethernet and this is the first homepage that have the system settings you can see here that is 10. And now let's look at the next notch input this input is very powerful. So, you can see because this eight s part you can see the Tiguan that as well as part have digital input and digital output or is eight channel this is eight to digital output and this is eight digital input and this unfortunate analog input. So, you can see here in our input webpage that have deferred is eight channels of digital input. So, if you use a four bar, so, you will see that the four channel after input, if you use a 64 board that you will see the 64 channels of the digital input at this table.

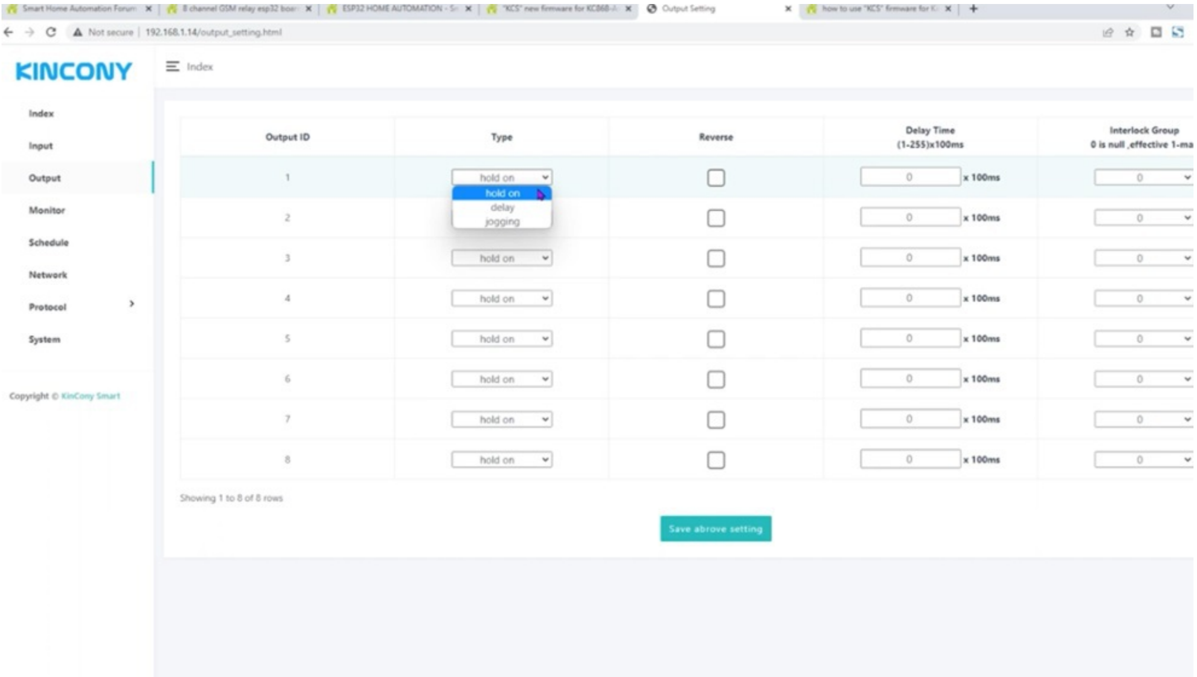

Not secure | 192.168.1.14/input_setting.html
KINCONY
Index
Index
Input
Output
Monitor
Schedule
Network
Protocol
System
Copyright © KinCony Smart

Input ID	Reverse Level	Bind Output	Function	On	Off	Toggle
1	☐	☑	single click double click hold on			
2	☐	☐	single click double click hold on			
3	☐	☐	single click double click hold on			
4	☐	☐	single click double click hold on			
5	☐	☐	single click double click hold on			
6	☐	☐	single click double click hold on			
7	☐	☐	single click double click hold on			
8	☐	☐	single click double click			

So that is very powerful for this table. Okay, let's look how to use this table. And you can see here that how many input and some options you can see if you have checked this box that means input one and brand output just the relationship the input with output if you uncheck this box, so, this input this input will not travel this output a Bert will use the MQ TT protocol and the TCP protocol to input the triangle that will send the message to the MKT debug or TCP server or TCP connect. So, this time, if you want the input and she can see here that is three command actions for arm and four off and tagging if you wants the input one and toggling this relay one just as the output one. So, we can input at clear. So, if you want the input A two and A toggling with this relate to so that we can we can this referring to two so you can see here is a digital input I have connect with these four buttons 1234 So, this is what channel one to Channel Four. And also you can see here an eight button for my test. So that can from one to eight. So you can see this is four one, when I press one this LD is triggered. So that can indicate when to sweet for and is this cable that support Max one kilo meet when southern the Meet distance so this is very long distance So, if you want to test you can also press this button 12345678. But this time when I press this button, and the input is triggered, and it's very easy nursing and nursing work because we have not set the logical process bottom one and the wishfully two and a half. So if you want the input a one triangle, really one, so you can see this, I just click the input, triangle output, and toggling really one and this input to toggle it to, so I can press this sync button. Okay, wait for a moment, that will reboot. This time, we can remove this USB cable, because we have already downloaded this firmware. So you can so you can remove it. And we can let this board large. So you can see I can this input A one, click the similar one, that Tagi. Press again, really when you the off and off. So you can also press this button, this ruler one or the one really well. And this is number two, in project two. You can see here that it's really too late to enter, but I have not set this way and so you can see this nothing work. Okay, now you can see how to use this input.



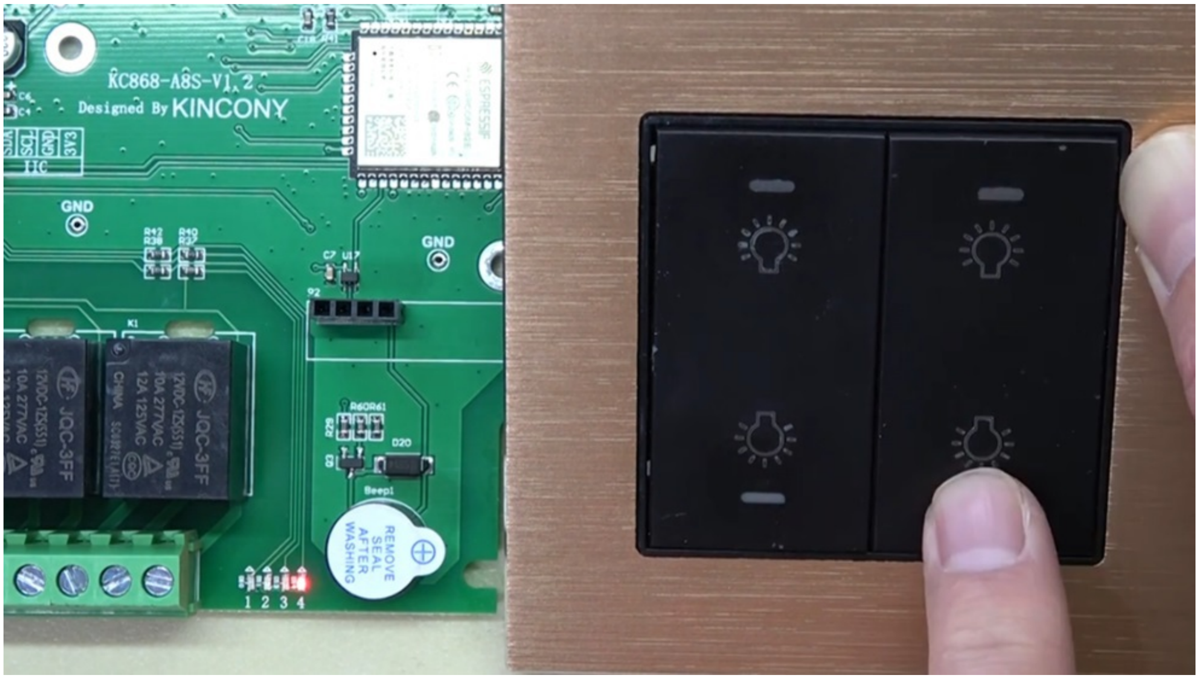
And with a one click to toggle this relay one and two. So this is a four One Click Work Mode. Because this function you can see here, Cap support one button for one click, single click, double click and hold down. So if you want use this button one, and also for double click, just you can see here nine just the one click for an autograph, but the double click the no function. So if you I want to get this button and work for double click, so you can see for example, I can let double click, I can turn relay one to relay eight. So you can improve this, when to eight channel that tab click will be out. And also you can use this space 12345678 These two different the input away. So if you want, let it's easy, so you can just input 128 For this format. Okay, we can see, just click save this button and after reboot. And now let's test it. If I one click that will be toggling to do one. If I double click, you can see the eight really are now this is Fortuno indicate and it is unfortunately indicate So, so this I can double click because it's already is so is also one to eight relay. So I can use this for to turn off all relays for hoedown work mode. So you can see I can input one, this time I can use another format to input this different number of channel 12345678 That Hoda will turn out better turn off our relay. So I can click Save. One click four and four are just talking display and double click you can see already though, it's generally now and the hotel that way we are off. So now you can see this button one that helps relay functions for toggling back and forth on and off. So just set by this button one so see see example for how to use this one button for different work mode and for different functions. So I think you can define our button by yourself. Okay, let's go out this time we'll say this reverse level options that what means so you can see I can enable this just I can disable this and only use this for turn on release one. You can see just test this function of number one input just this time I have not enabled this. So you can see I can save. Pay attention to my finger. You can see here, I help press this button up See my fingers have not released this panel. So, that is four down four down triggers this input. So, you can see I have not released my hand. So this LD is that for input. So, this triangle effect here that is a press action is down, up to down up to down that is for this button I release my hand okay, now we will test it. So this time I have click this one and we can see let's turn off the relay firstly, just did it slowly one turn off firstly, you can see here, this input that will reverse level what is this means and what's different and this time you can see my finger or pay attention to my finger and you can see here I help press this button you can see I hold on this button and this red LED is our but my thing I have not released it with this panel and the relay is not work. So I will ready to release my finger. So you can see here I raise my finger and the relay is on. So this time that reverse means the effective of the button that is up that up. So the down to up that effective for this travel of the input. So this is a function of this reverse level options that is which effect him if I have check this box that is up to turn that up to down if I check this box that down to up. So this is a different of this effective. Okay, let's go on we can look at this online guide. And the next is output function. You can see this in output first three different work mode that is hoedown delay and jogging. So let's look at this one. And this time we have clicked this output web page. So you can see because these eight s bar have H Chen output so that have list eight channels. So if you have used another part so that will list all output items on this page.


Smart Home Automation Forum
8 channel GSM relay esp32 boa
ESP32 HOME AUTOMATION - S
"KCS" new firmware for KC868-A
Output Setting
how to use "KCS" firmware for K
Not secure | 192.168.1.14/output_setting.html
KINCONY
Index
Index
Input
Output
Monitor
Schedule
Network
Protocol
System
Copyright © KinCony Smart
hold on
delay
jogging
Save abrove setting

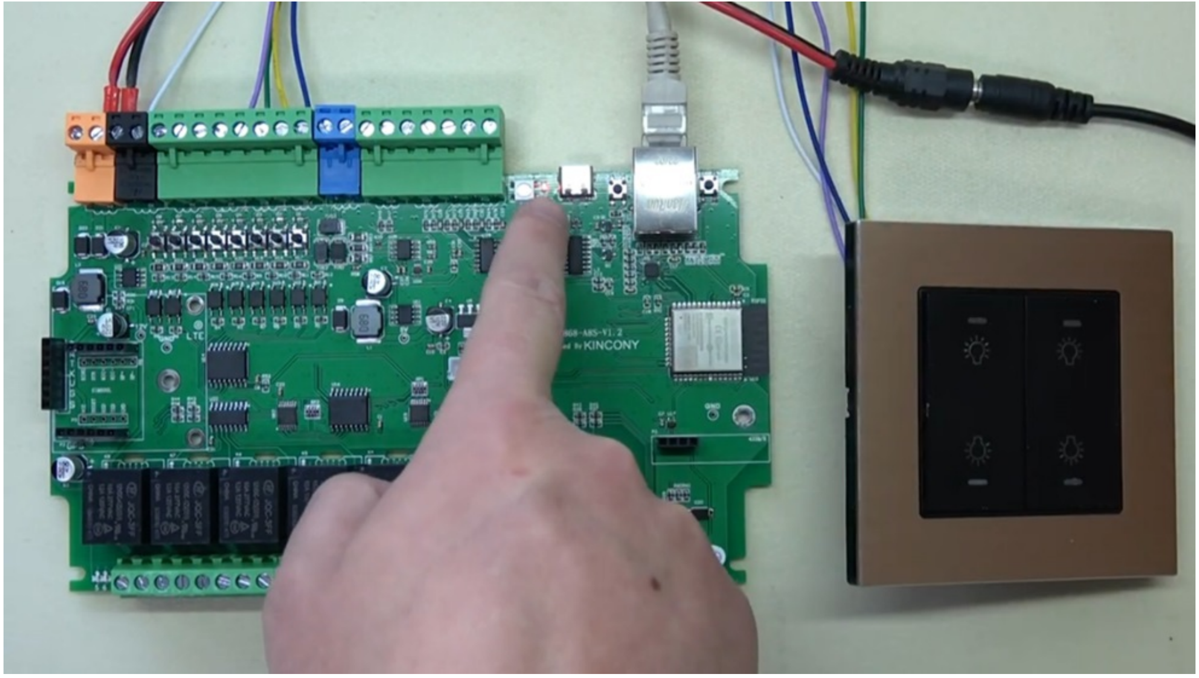
Output ID	Type	Reverse	Delay Time (1-255)x100ms	Interlock Group 0 is null ,effective 1-ma
1	hold on	□	0 x 100ms	0
2		□	0 x 100ms	0
3	hold on	□	0 x 100ms	0
4	hold on	□	0 x 100ms	0
5	hold on	□	0 x 100ms	0
6	hold on	□	0 x 100ms	0
7	hold on	□	0 x 100ms	0
8	hold on	□	0 x 100ms	0

Showing 1 to 8 of 8 rows

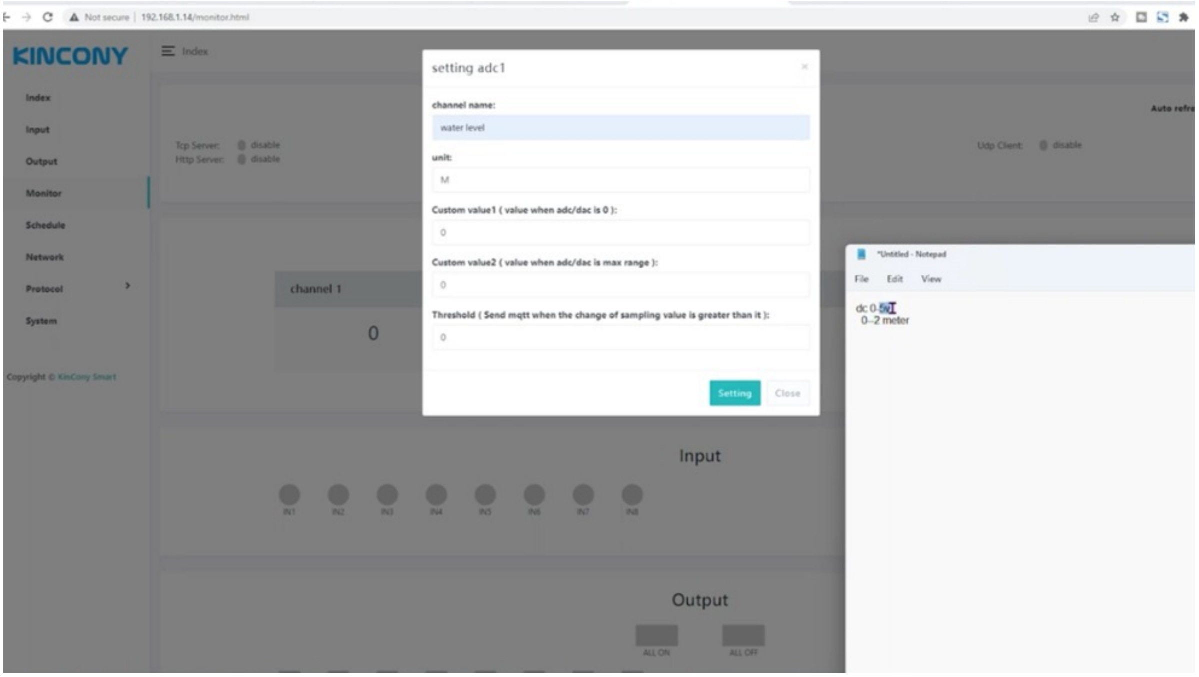
So you can see that deferred is hoedown and is a What's the holdout means that's the holdout means if you turn around this will be one that is always so if I can set for the delay and let this you can see here we can also test with this input to one and important two that I can set to that is really, really tall that target only one and Okay, well let's test the way this relay to the switch to. So you can see here because the deferred outflow tool is set to this hoedown. So I have press this button that is helped pass this button that is out. So that will always we can turn off the little one just I press this button to turn out really to that really to always never stop. But if I have choose this one that actually that just have an auto or function. So you can see here I have a K input the delay time for this delay action. If I press 20 that means two seconds. I press this see Sorry, I needed to input because we tested this channel to input a tool. So okay, input 24 seconds. Save setting. Okay, let's turn off this. Firstly, turn off this output the first day and look at this important tool. So you can see this button too. I can press this button and a one two, just the way to two seconds that will be off. So, you can see I can press again one two, that is our so this four out of just that means that delay. So, you can set any channel after array which can use this delay function sometimes for the safety program. So you sometimes the loader needed to out of so you can just input the delay and the delay time Okay, let's look at the next that judging What's this means we can set to judging and this can change to zero also number two of the input. So, you can press save and press OK Which one what you can see here that input to my finger and the hood release hold release. So this usually use for the cutter moto so you can use for oppo and for Tao. So, you can see here I can just who now will be out and release will be off right now. So off of that country Moto is very useful. So this is for jogging. Okay, let's look at say this next this function that means reverse. So, this input you can see have the reverse function that is for reverse swishy affect you and output we have also designed this reverse options for this relay output because you can see when I turn around turn this button and re will be brought sometimes because we have the calm normal open a normal clothes, usually we have used the normal open and it's come with a load birth sometimes you will need to use the novel clothes. So that needed to reverse your state of the relay. So, you can see if I have chin check this reverse function and press the annual report. So you can see the effort is Oh and this time I press this button too you can see here a button is pressed, but the relay is off. So the state is changed its reverse state. So, this time you can use a calm and normal close to your Lord you can say ah and that changes the state change the state okay this is a reverse function the next you can see that the interlock so that you can set you have eight channel so you can set to this one that means these two output ways the interlock relationship and you can see here that number two this interlock. So, if you have eight channel relay, your Mac's have this four groups of the interlock if you have 16 channel arrays, so you Max how eight interact groups. So you can see here I can change those array and the changes so these three groups will affect you. As you can see I can press this button actually I have used this for channel of the switch so we can test the way the input a 1234 and also will react to the input easy to test with just use this one just 12344 input target one target two targets we're talking for. So this weekend, click save. So you can see I can turn overlay one is turned off really one to one lower again but this time I have not turned off everyone but I can want to turn on relay to you can see I turn relate to relate to it. But the lonely one will be out of firstly. So you can see when w two is out. But this time I want to turn on really one pay attention to this LD you can see here. I totally want this really to we'll need tongue off first. So this one and a two is one to one to this to LD and to this tool relays will not work at the same time. So that will protect your Moto. Will not let these two cables or hell connect to your power. So you can see and this is buttons way and important for your for release me and re for you can see here. I can tell now release me And really for also is always the interlock relationship.



So you can turn off turn off relay for now off of place way, but just can't let this tool really work at the same time. So, these two button have come have worked with the relationship of the interact in this way and the full have inter interact relationship so this you can use for the Qatar moto for app and this photon and then maybe another moto for up another moto for down. So all these have set the interact by your software. So you can easily to set the relationship if you want to set this one no interlock if you want release one and release three is interlocked so you just read this to number as the thing is okay. So if you want the eight and the two ways, interlock, so you can set to this one you can say for just a single number that will be with the Interact mode. So this will be easy. Okay, let's mocha are set to 00 No means no interlock is effective. So the default is set to download. So without any interact mode. So that effort usually will be used the hoedown so we can save this settings. Okay, before we have set the input, that is for single click, double click and Hoda Now I will tell you how to set the speed actually adjust to the speed of the double click and the hoedown. You can see this system, you can see this function, that's a double click tan if you see at the SMA 50. Maybe you have set this to one seconds, that means one solid, Ms that one seconds. So these two items, you can set the double click 10 and hold on 10 According to yourself. Okay, let's look at this sections, you can see that the NTP server that is the time server on the internet, so our time when the party have the power and connect to the internet, that will auto update this time to your part. So you're just the default set to this time. Cardoso is okay. And this, this is the username and the password of your lodging webpage, just as this web page you can see the default domain and you can change the password just this password. So you can change your login password. And this function is I have to tell you first that the clip output after restart is that means use for the power fetch the power feature, the real estate will remember when the restarts So, you can see if I have enabled these functions, you can see I have really one, but this time the power is off the really one because it's the party has no power, but this time I connect to this power again the power have the power again. So you can see the release will be out to recover before the power Fisher. So this can be turned out automatically that other one if you have disable this function, you can say I disable this function and press save and restart you can see the part and restart that will be a as the default function as a default state is off. So you can see here I can turn this relay one, but this time if I power off if your house is powered off and the house is power coming in coming again. So you can see here I have connected with this power.



However the law works this time that really won't will not out. So the deferred just Oh is off. So this is the options of the rest, keep out output after restart, just recover function. Okay, let's look at the next we have seen this output all options on this web page. The next is very important just dashboard so this is a monitor. So you can see here that to monitor. The monitor you can see this a different port which have connect to color server or you client have Cloud Connect to is apart because I have all have disable this protocol next time we enable this you can say here is the protocol or disable the disabled. So I will enable this that will be can monitor which have enabled and which have connected with a car service successfully. So, this is where you have enabled this protocol to use Okay, let's look at the next the first day and here you can see that analog to digital the actually the ADC and because some part will have ADC and some board without ADC just according to this hardware resource. And this part eight as you can see here, how ADC, so that you can see just this a one two a four analog input, and here you can see this input the input a digital input, you can see eight digital input you can see when I press this button, you can see here that digital input will be and so, this you can see I have processed input file and the input of six the input of seven and the input eight. So, this can monitor the input of state digital input or state and how you can see that is output the output you can see the state now that really 123 is out. So you can see this is going I can also use this mouse to country by the web page on the compute. So you can say here when town to mountain mountain mountain. And this is for and as you can see, this will be already will be off and is listed for and this output. Okay, let's look at this details of the ADC. That is you can set click this to set a wrench because if you have connected to what levels is, so you can input the name change the name to what lab and the unit you may be set to the meat meat or we can use this now to meet the customer value one and customer matter tool is what means because you have used analog sensor, if you have used a TC, they are able to fine watch. So that's the wrench you can find out your sensor, maybe your sensor is for the low is zero and the true to meet.



setting adc1

channel name: water level

unit: M

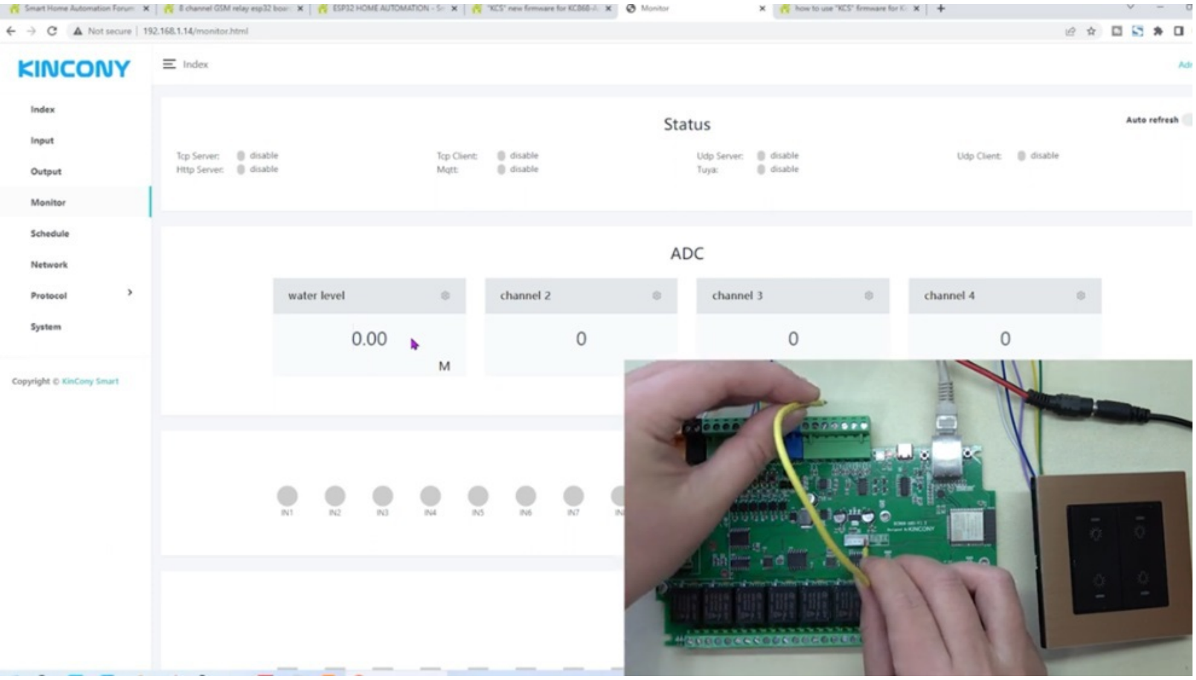
Custom value1 (value when adc/dac is 0): 0

Custom value2 (value when adc/dac is max range): 0

Threshold (Send mqtt when the change of sampling value is greater than it): 0

```
dc 0-5v
 0-2 meter
```

So this you can see this is the lowest lowest or most is zero. So this is the arrow and this is the value to that convert to the physical unit. So this five watch because the user don't know what the five watch. So, the max that means to meet is a sensor Luxor reg. So you can input to and listen to that hold that is the difference value changed without sends the message to TCP or to MKTG such that we can sell to the Aleppo four 0.1 meat that just has changed over the the lab we have changed. So that will also send a message to MKT oh by the TCP. So, we can impress this setting and you can see the name has changed and the Unity has changed this time if you have connect with what So, I can shoot for this flight test because these three points with what you can see here I should I get this power from this and for this channel channel one you can see that in my computer screen that 1.2 Sorry, I needed to make a good contact.



Smart Home Automation Forum
8 channel GSM relay esp32 boar
ESP32 HOME AUTOMATION - Sr
"KCS" new firmware for KC868-A
Monitor
how to use "KCS" firmware for K
Not secure | 192.168.1.14/monitor.html
KINCONY
Index
Index
Input
Output
Monitor
Schedule
Network
Protocol
System
Copyright © KinCony Smart
Auto refresh
IN1
IN2
IN3
IN4
IN5
IN6
IN7

Status

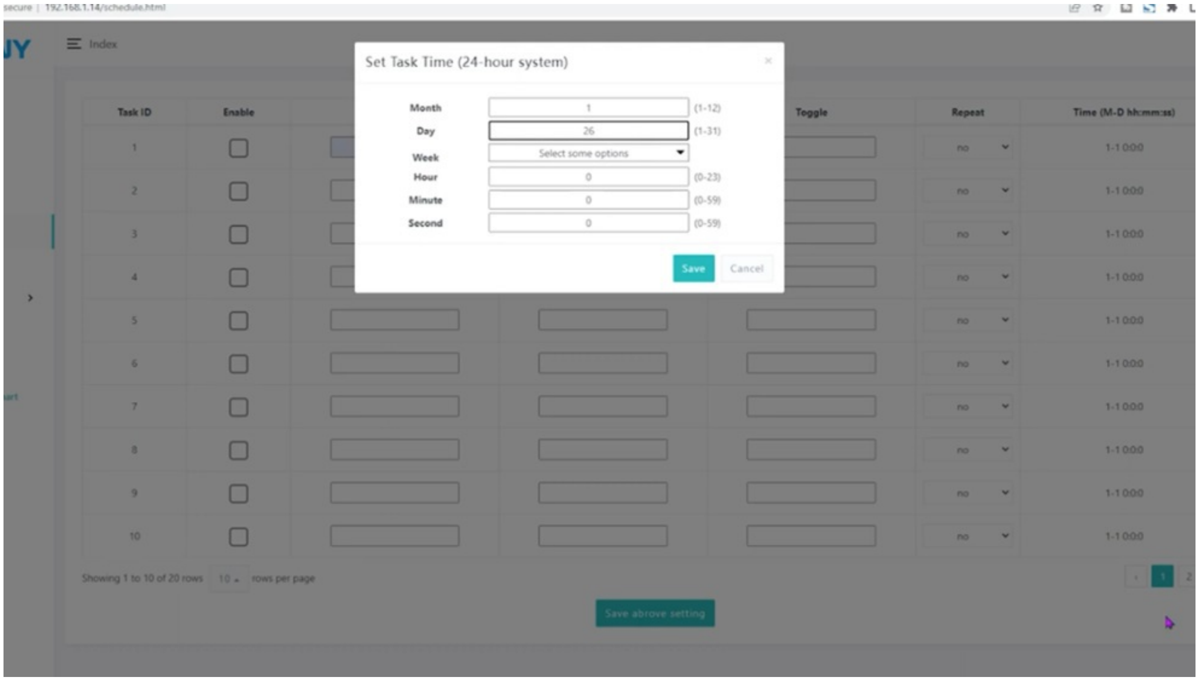
Tcp Server: disable
Http Server: disable
Tcp Client: disable
Mqtt: disable
Udp Server: disable
Tuya: disable
Udp Client: disable

ADC

water level	channel 2	channel 3	channel 4
0.00 M	0	0	0

So one point to meet this actually is of what three points three. What. So these have converged to the meet unit. That is the auto refresh function. So that you can enable or disable if you enable this auto refresh with a web page where every five seconds will refresh this value. So if I have not set to the to meet, I said to the watch, just I want to say the analog important watch. That is zero to five what so you can see here, I click setting as you need to change to what and this time I can also connect with this one lot. So you can see here and then I got one and connect to this swing point this way you can see here that is almost the switchboard of this NLRB input one. Okay, this is how to set this analog. So this bar have to four channel analog. So you can see here arrow icon you can send to the different member and a different vein wrench and a different unit for this different channel. And it is a digital input, you can also click, double click this test, you can see here double click, you can set to such as a model of the input state, you can also change to the maybe this gas sensor, or maybe this door sensor. So you can change the name just a click, double click and double click. So you can change this input a state's name.

So you can see here, I can just for tribal, and tribal, you were easy to remember and save which sends a triangle. And this is the fourth output for the delay. So you can also double click, so you can see these are up, maybe this is a cut up and maybe down and maybe this will light okay, you can change. So you can see I can just turn off turn off, so you can see the state will be changed and also is turned on and off. Just according to yourself. So this is a four output input an ADC. So if you burn we'll have the T AC functions that for dimmer, so you will register this T AC options on this web page. So this is a monitor to monitor the state with cursor and is a different political and different analog and digital input and output state. So this is the dashboard of this part. Okay, let's look at the next. This is the schedule. The schedule that's also easy to set up just you can see, you can quit your scheduled task that is for enable or disable, such as you can let this talk in a new one. This just added before how to set the channel if you want to attend well already or turn off already, you can just set by this format if you weren't already, such as 8128 Channel, and the weather needs to repeat, repeated by me miles every week, every day, every house our minutes. If you don't want to repeat just click this No. If you want repeat you can according to minutes, according to day, according to our is, it's up to you. So you can click this you can set a mask date if you don't want to set just just to keep it when you're on the floor every week. Every day you can set the week after day after day that we can test it okay let's test the schedule function that we can create our task. So, you can see here we can let the live one tool relay aid will be so that we can use a timer if you want to repeat you can choose this item by minute or by the hour or by the day.



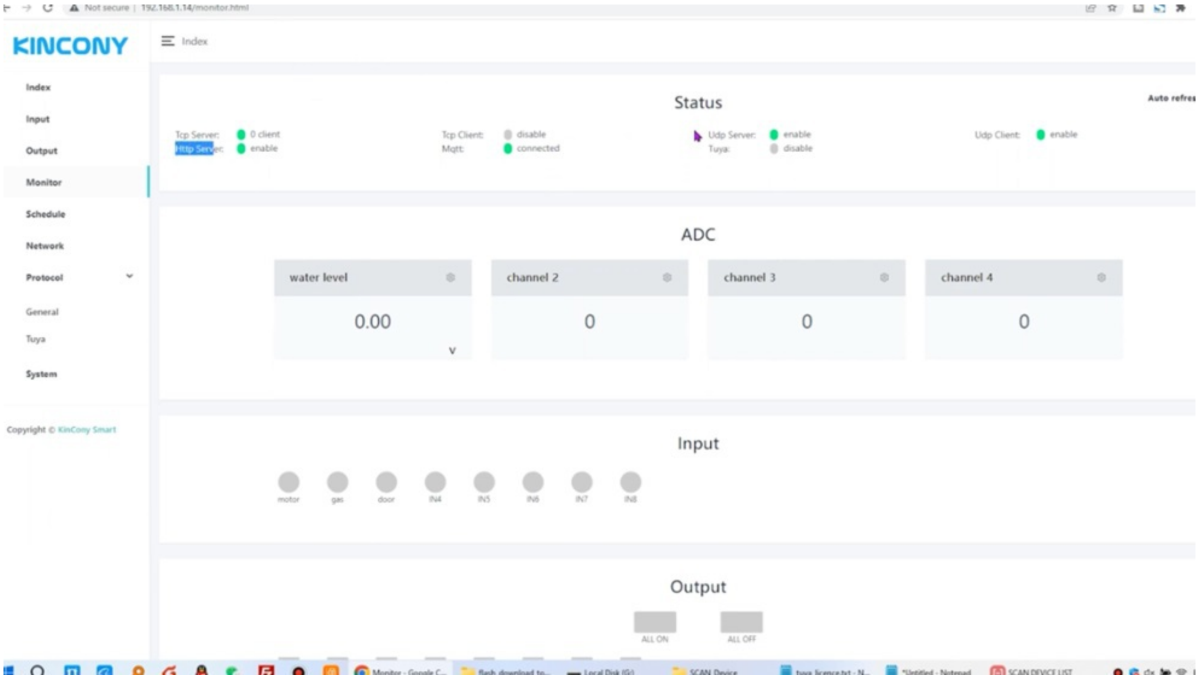
Set Task Time (24-hour system)

Month	1	(1-12)
Day	26	(1-31)
Week	Select some options	
Hour	0	(0-23)
Minute	0	(0-59)
Second	0	(0-59)

Task ID	Enable			Toggle	Repeat	Time (M-D hh:mm:ss)
1	□				no	1-1 0:0:0
2	□				no	1-1 0:0:0
3	□				no	1-1 0:0:0
4	□				no	1-1 0:0:0
5	□				no	1-1 0:0:0
6	□				no	1-1 0:0:0
7	□				no	1-1 0:0:0
8	□				no	1-1 0:0:0
9	□				no	1-1 0:0:0
10	□				no	1-1 0:0:0

Showing 1 to 10 of 20 rows 10 rows per page

The first we can test it the first time mouse January and 26 Because this is today's date and for every day for every day and the hour. So you can see here I can enable this way and save setting okay you can see this is 1pm 1pm that he generally so this task is running if you want every day, this task this timer is running so you can just choose this day repeat by day or you will need every hour minutes arrived will be running tasks you can choose to this hour, so it's up to you. Now I can disable this one and save and we will test go out So you can see this bar report. And then let's look at the next than network. And in the network as you can see, you can choose the LAN the Ethernet cable by the DHCP or via static. If you choose a static you can input any settings by yourself, the default will use the DHCP if you want you the DHCP you need to make sure your Lord have enabled the DHCP function and here is the Wi Fi, the Wi Fi you can let it enable because I want to test faster so I disable the Wi Fi and you can set the AP mode or the STA mode if you use the STA mode so that the Wi Fi part will connect to your load. And here you can see if I have enabled this for sta that saved so that I can use a Wi Fi and Ethernet at the same time. So you can see index you will see this Wi Fi as T IP address and this Ethernet IP address you will have two different IP address so that as a part if have connect to the internet, the part of the network cable you disconnect or broken so we'll switch to the Wi Fi to connect to the internet the corner server such as the MK TT if you have to use the Ethernet cable firstly because the Ethernet cable will be stable. But sometimes if the network cable is broken, so that will also change to Wi Fi. So that makes sure your partner will always connect to the internet callosum. So this is different functions for the ACR responder. Let's look at again. So this is for land and this is why the Wi Fi and here we can see the protocol This is a powerful page. So you can see the ESP 32 part use the KC s firmware that will support MQ TT HTTP server and the TCP server and TCP client and the UDP so and the UDP client and is four to five. So that will have different ports. If you have used this and KDT I think you can integrate it to the home assistant and then you use enable this HTTP server I think you can either to integrate or lock sound system and this TCP server you can use by the key box applications that can use in local network without internet and it is a UDP that is for integrate to the third party software and this is what five that's running by the Modbus protocol. So you can which put call you want to use you just enable enable enable so that you can just enable like this I can use this one and the TCP client just disabled firstly and here you can see I can connect to the NK TT so the TCP server when you to create to this part and press it now we can go to this minute and you will see all these have monitored a state that is somebody running and a new client and that TCP server is running and MQ dt is connected to Microsoft and here the UDP is enabled and you can say this to the next client next protocol is very important I will introduce to you and how you can see all the state and what the HTTP and MQ TT and the TCP stream you can see in our form that is a port actually delivering protocol and what about protocol and mkd the protocol and stream protocol so that you can all use this port in this webpage just enable this they can work together running at the same time.



Status

Tcp Server: 0 client
Http Server: enable
Tcp Client: disable
Mqtt: connected
Udp Server: enable
Tuya: disable
Udp Client: enable

ADC

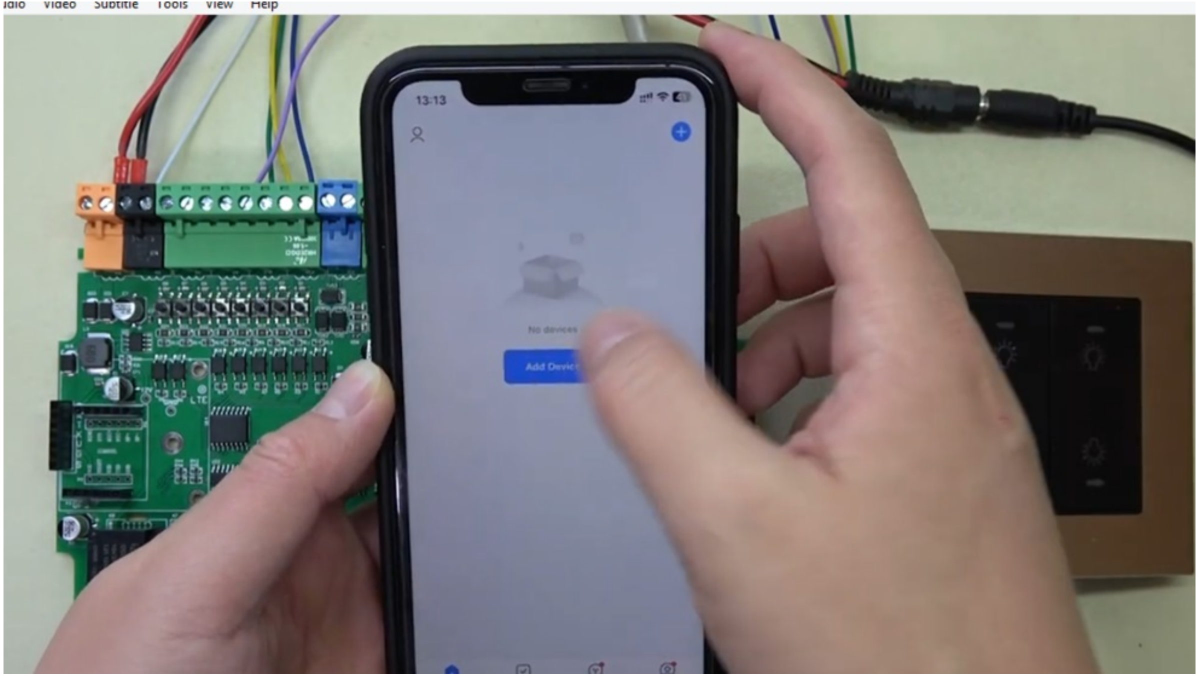
water level	channel 2	channel 3	channel 4
0.00 v	0	0	0

Input

motor, gas, door, IN4, IN5, IN6, IN7, IN8

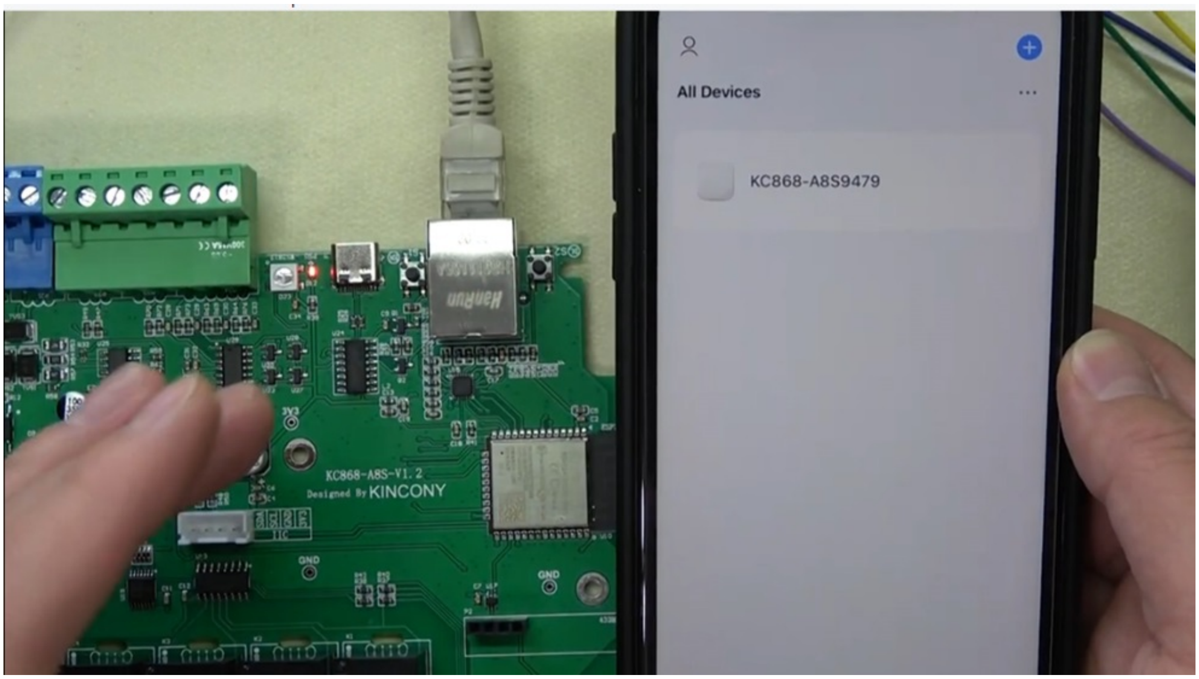
Output

So this bar you can use to your application and use a home assistant and use the key box at the same time not needed to change which you use by Wi Fi which by reason at all can use running by the Ethernet cable. Okay, let's test with this key box. So the key box we need to enable the TCP server that part and this IP address is very easy so I can show you how to do the cable and you can see here the my eyes phone you can see this key box because cable seeing using local network without Internet just click and input the IP address of this as part that is and there's a part 4196 And this is really the Tapi relay and this is eight channel relay and this eight channel of relay output. So, we can press Save, and you can see that we have the list eight channel, you can see here I can tonally one relate to this wherever the 4678 Okay, you can see that will be very easy to integrate to the key box and you can use the key books in local network just by this Wi Fi. So, this will work without internet, okay, this is how to use a key box to local network country this really okay, let's look at to the next the next you can see the most importantly the two year system and the way let the Tweah system will be very easy to use. So, you just enable this to your options. And this needed to be our lessons you can pay upfront we are all buyer from the King County because the lessons we need to pay money to the Tamiya so, after your pain connect with the King County, we will give you this three hour lessons that three or four items you just copy and paste it's very easy. So, you can see because this lessons I have used in China maybe you have using western US or you is in your group and using India. So, you just buy from us tell you tell us which ranch you want to use we will give you our lessons. So that will be speed will be fast. So now I tested with this China. So you just paste and product ID and this device ID and device secret and gain ID. So, you can see copy and paste these items press this safe okay, you can see this will be auto generated or QR codes that will be added to to your application show you how to use this to your application this is smart life.



And to add this device, you can just click Add Device and scan this QR code to your computer screen a QR code and you can see here I have scanned that have successfully added the eight keys each 688 As part I can click this down and here you can see this will be seen this page you can see I can tell only one willing to potentially really for you can see here that into application can use by the Wi Fi or you will pass 3g 4g 5g for remote country by the Internet and this in turn off and this is so you can see the state is update and this is you can see here I can use this click this to only one the better switch and this will be one will be update and I can press this to laugh and guitar okay as this formulae output in the two application, you can see here that have the input a state you can see here 1234 is nothing because we have connect with this button use for switch country this relay output if you want to use this input as a sensor, so you just unlink your input without. So you can see the 5678 you can see the state is monitored by the to application remotely because the input 5678 that have not linked with output. You can see here that is the input channel. You can see here the four channel have planned output, and this five to eight that is our link. So you will see this five to eight that can monitor the state. And this is already used for the input switch for the output, so we're not update the state to the, to the application. So these are different, what's the difference between this link and link the output and with the input, okay, let's look at the analog This is ADC, you can see that as thing as the software on the PC, you can set as actually the name and the many value added the max value of the sensor and the unit such as you can see, okay, input what level and the meaning is clo meet then there's a max is to meet the unit I can input and you can see I can press this see if you can see is the water level and this unit is MIT. So, you can rename and the setting for air which you know can see I can shut get three voltage for the analog input and one for test, I should voice what you can see here that is a

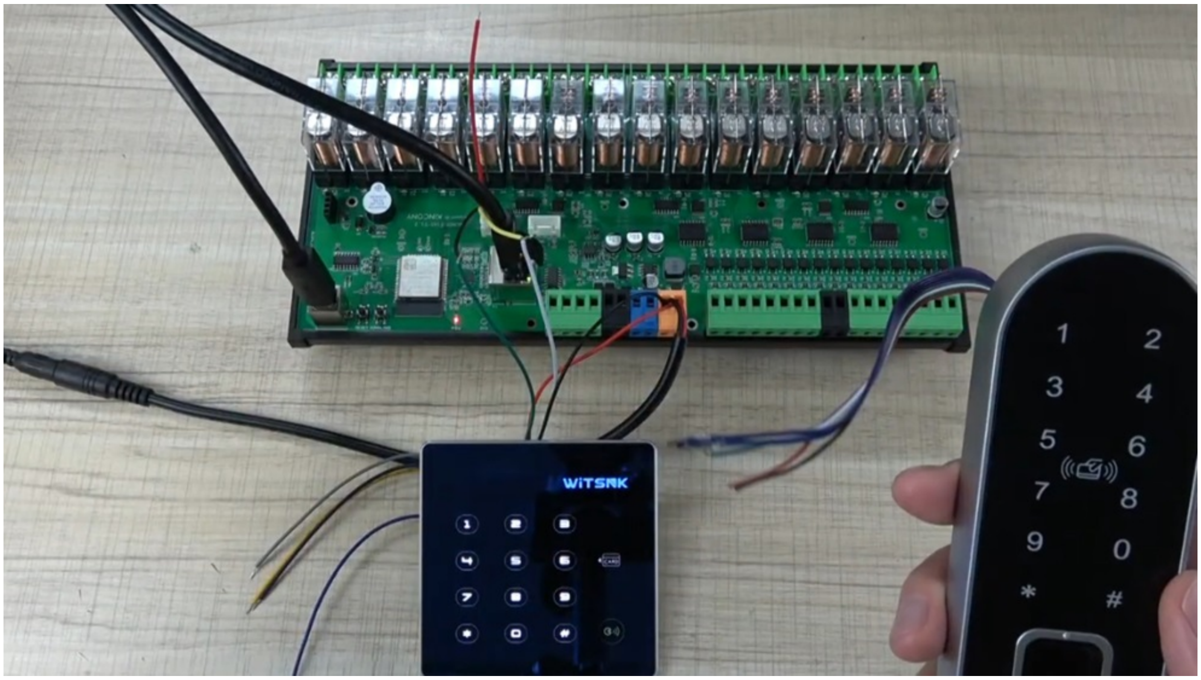
1.2 meet. So, this helped connect converse with a meet unit of the water level sensor. So, this is for remote monitoring this analog input and I have already enabled the auto refresh. So this every five seconds will refresh the state. If you have disabled this auto refresh and the unit a tool like this to refresh I suggest you enable this auto refresh. So, this is a four tier application that you can edit to the aid as part actually ever eight ESP 32 module pod downloaded KCS firmware you can add your part to this to your application.



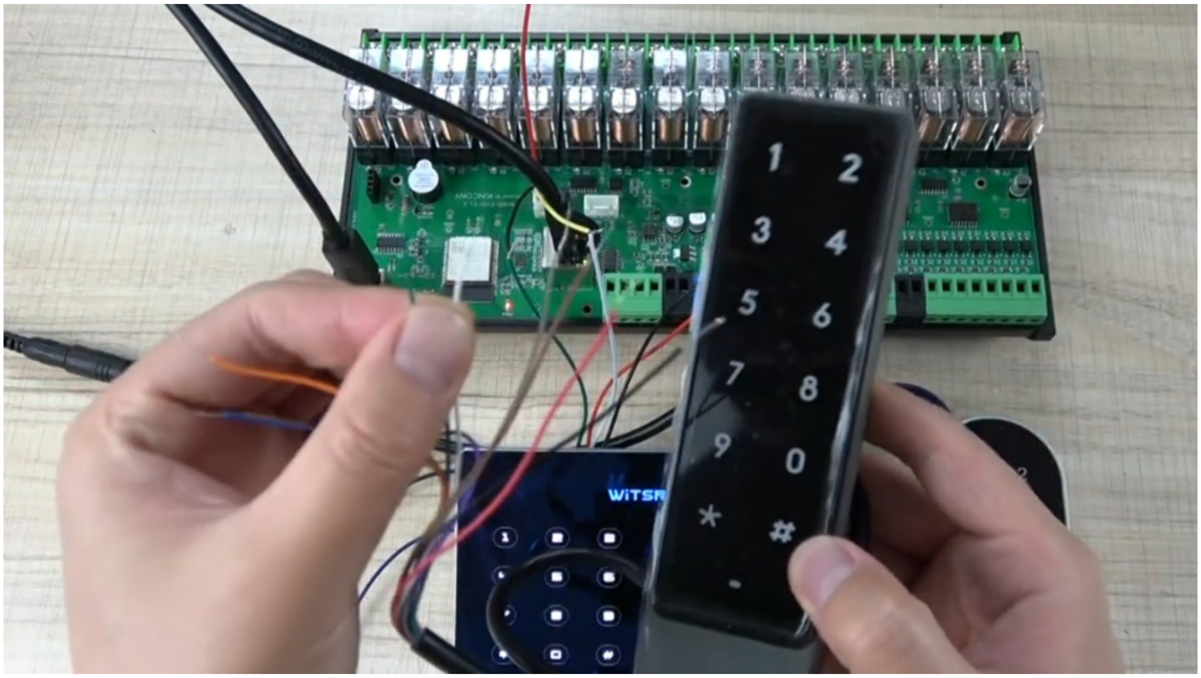
So that you can use this to application for the sins mod you can create which which sensor or which state to change to so that you can travel any tuya output a device. So, you can see such as the input one is our and then you can add a test run device because only I have this part it has if you have many Tweah products, so that will just add here. So you can use then come out automatic country in different to atomize. So, this is very useful for home automation. So, this is how to let this aid as part of connect to this to here by this Ethernet cable. So you can see I can use this Connect for the internet for remote country. I can also use this key box you can see as it's a cable box, I can tell to round to round to now, I have turned on a four channel relay fortunately, and now I can open this to your application you can see this state update I can turn off totally this is totally oh I can also turn now you can see the key box also update. So these two different application will update the state of the day I can turn off only one billing is so you can see these two here that will really if I have already you can see here two are already and you can see the QuickBooks that already is so this is it update Okay, at last I will show you if the network cable is broken, so that I can remove this network cable because I have added this IP address by the key books on this network, isn't it a couple of so now this key box can't use but you can see here that you have to your application you can see I can turn off you can see also can use because the board a4 Board have changed to Wi Fi module. So that will connect with the Wi Fi module with clear crossover. So you can see I can also use to edit but this time if you have connected the Ethernet cable again so you can see the LD is building. So this time we switch to the Ethernet cable now You can also use this turn off, turn off, turn off and turn off. So, this can auto change the land by the Wi Fi or by the Ethernet just switch which LAN is available and go to which communication with this LAN and we use this Ethernet cable firstly, an Ethernet cable bad thing, we'll switch to the Wi Fi okay, this is how to use the Twitter application and the K box application. And as a next step, we can look at this last System Web page, the system webpage, I have introduced you this sections and this is based at Port just reboot this part and this is recommit to blockchain that you will be clear all this information have said about the input output and Wi Fi. Just look how to manufacturing and this will be enable this eight s for the AP mode because now you can not say this AP because the Wi Fi sta if you want it become s AP disappear this AP address, AP hotspot, you can hold down this button, you will you will have a function button that you connect with ESP GPIO zero, just the hoedown 10 seconds apart or we will recover factory recover or you can just connect this February cover is seen. So this is how to use this KCS firmware. So it's very powerful. I think. If you have any questions, you can leave your message or you can have some problems, you can feed back to our form of the KCS freeware system.

ADD WIEGAND ACCESS CONTROL SYSTEM TO HOME ASSISTANT

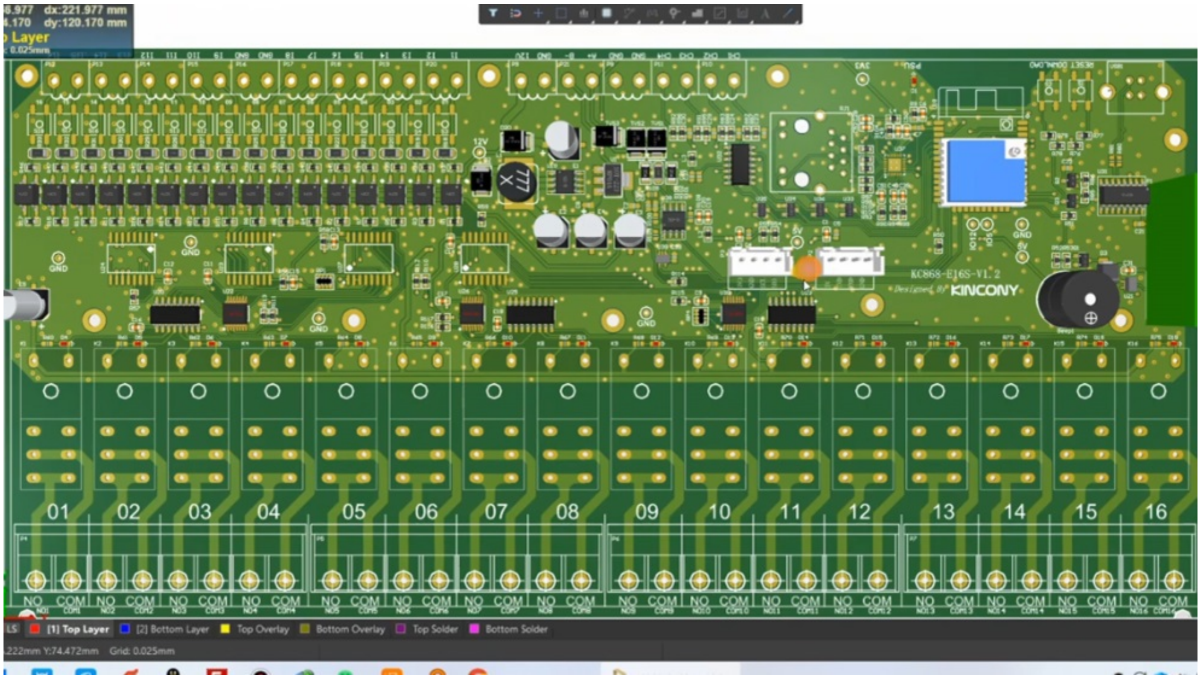
I will show you how to add this access control panel to home assistant by the ESP home. So that you can use this important number and contrast issue relay output such as you can see, I can input 101 And so, this really will be out and when there are no one, enter this the first one leave we off, if I input one zero to enter, the secondary will be on and when zero to enter, the secondary layer will be off. So, this time I will show you how to step by step and use this access control panel. And as you can see, this is the version of our inputting the number and you can see here the other device, so that you can see you can use the fingerprint, so that we can use the fingerprint to country differential relay.



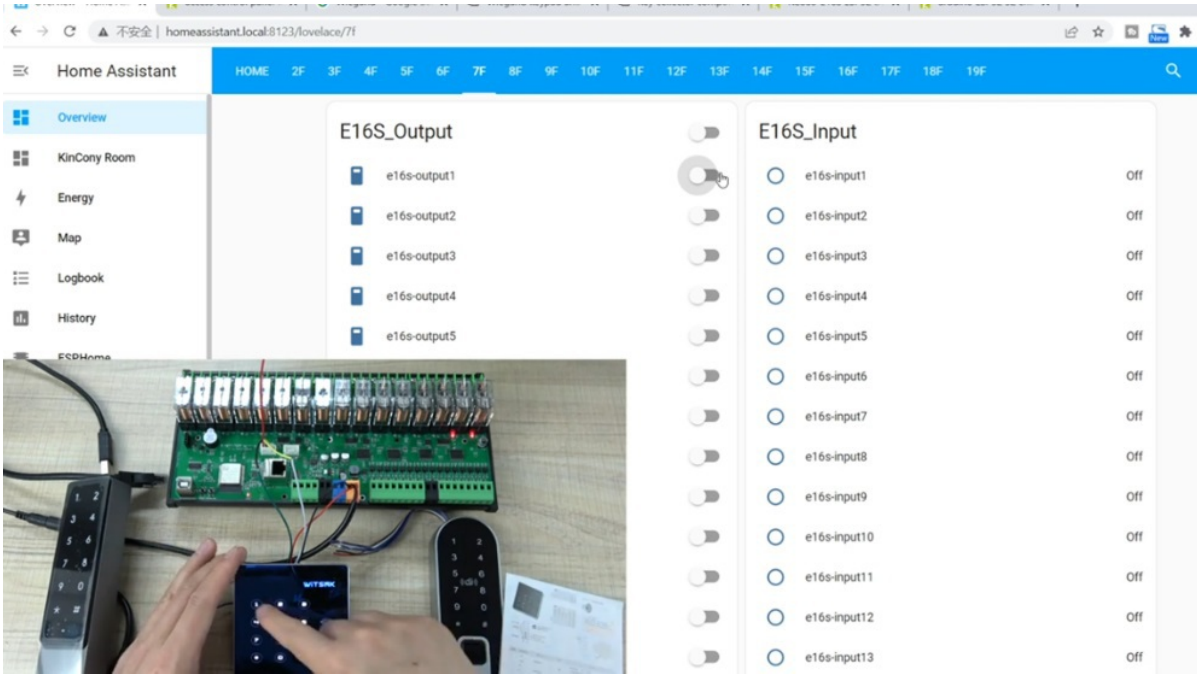
And here you can see that is another one that only by the password, and also you can use the IC card. Okay, let's look at the hardware details how to integrate it and this panel, we have used the weekend protocol. So, you can see here that is very important, you need to understand what is the weekend. So, this is the interface you can see the search from the Google that help very popular have used this I have a long history that from the 1980s so this can easily use by the access control system. So you can see there are almost two one. So, in your ESP home, you can see that also can support the Weygand pic, keypad and the leader. And this is one photo and like this one, and we can see this the images and almost the weekend, you can see that have one t zero and T one. So, you can see that t one and t zero these two points. So, you can see that in D zero and that D one. So every panel you can see there not so much once and also you can see that's how so much worse work, we only need to use to one justice to one the Queen and is the white. So that when you press the key and the user IC card and use this fingerprint, that will output to the date are found these two wives. So, we just needed to decoder this one off put a signal so that we can country the different really adjust according to the different data output.



So, you can see this have three kinds of access panel the first one second one and the third one, there are many user menu, but you can see here this one is this one, you can see here the lock and connect with this, this just a relay board this is 60s part because they have extender to chow I can use so you can see that why one at a wh zero and the connect with this board, only this to date. And also you can see this panel, this panel user manual just to this one, and let's look at you can see this user manual also you can see that t zero and D one that is yellow is occurring and the T one is white, and also you need a 212 watt for the power and the ground. So, the totally you just need a full one just a 12 watt ground and an E zero and Q one and this is a third this one is this one, you can see the menu and here you can see also you the queen and is white, so that 41 and the T clo so the T one and T zero that is the standard interface of the weekend. So, this is either to confer your access panel held this two weekend wire and so, that will be very easy connect to the ESP 32. Now I have used this e 16 s bar, but actually you can use any of ESP 32 word. So, you can see this e 16 bar because this time I have used this they have extended the report. So that how are two GPIO pins I can use I also have integrate with a 32 bar. So you can see here in our form, I have integrated this a 32 bar or found disk to In cables from this kinds of access panel and it is canned so under the ESP home config file you can download because you can see here our PCB design, this e 16 s so we can see this already.



And you can see here just I use these two pins and if you use the AC 32 And we can see the 3d mode and also you can see here is a socket that has the ground x and the TX and a five watch. Actually we only use the X and a TX just these two pins are for ESP 32. So you can use any ESP 32 module just to choose to GPIOs connect where your access control panel is work well. Okay, let's look at you can see that in the home assistant, I have integrated this part to home assistant you can see I can turn on and turn off and on and off because I have set the interlocked mode so some channels will not work at the same time. And now you can see I can use this access panel and the time of year one and a turn off for the one if I remove this USB cable and remove this network cable, you can see that you disconnect the Home assistant now you can't work but you can see I can also use the access panel. So this kind of controller one and 102 that you can train read to. So this is working locally without the internet without network without Wi Fi. Just it connect to it directory so that can work well if without any network.



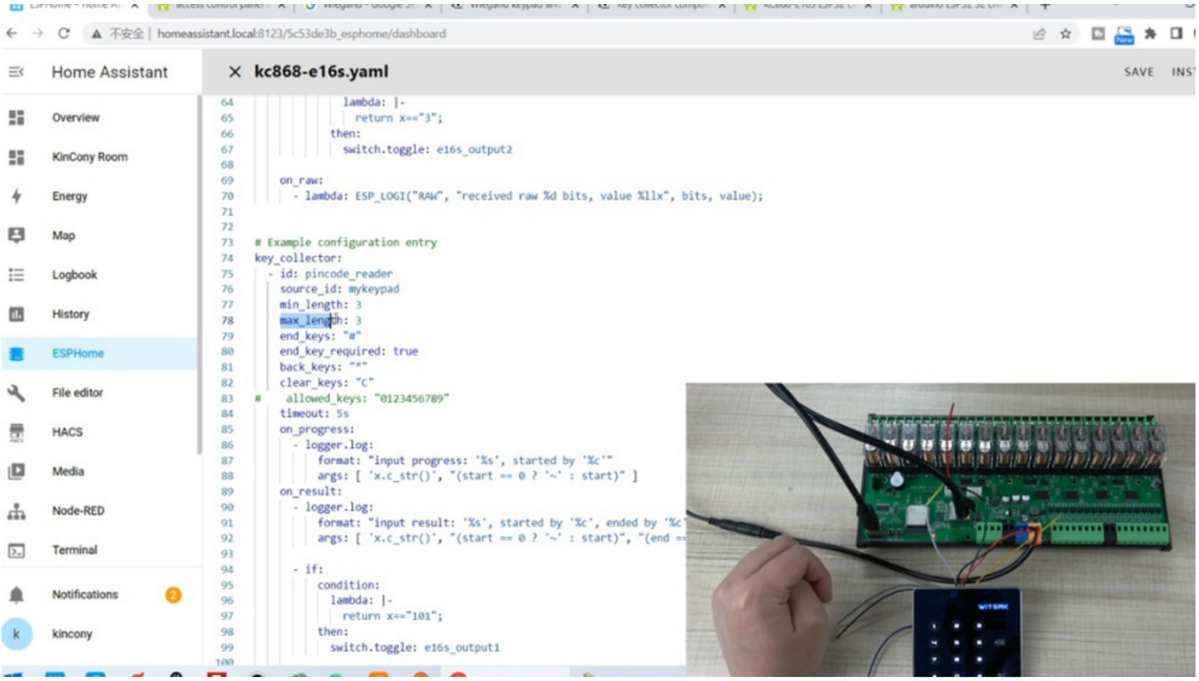
E16S_Output

e16s-output1
e16s-output2
e16s-output3
e16s-output4
e16s-output5

E16S_Input

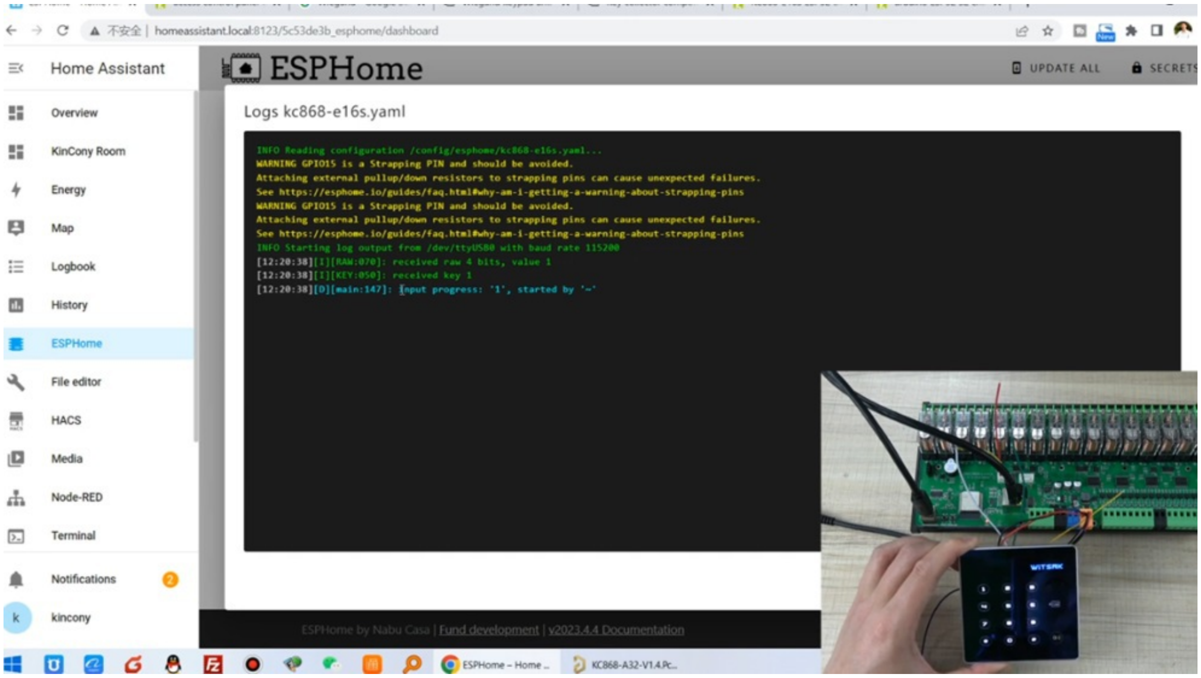
e16s-input1	Off
e16s-input2	Off
e16s-input3	Off
e16s-input4	Off
e16s-input5	Off
e16s-input6	Off
e16s-input7	Off
e16s-input8	Off
e16s-input9	Off
e16s-input10	Off
e16s-input11	Off
e16s-input12	Off
e16s-input13	Off

So I can turn off this you can set by the ESP home which number and which finger you can control this panel. So now let's look at the hardware. I have hot why? This is the pin define one to 12 watt power supply and ground and that is yellow. And the one so you can see I just use this cable the red have connected with this 12 watt and is a Breck ground. This is the ground because I have connected with the power together and these two lines D one and D zero. I have connected to this. This terminal. You can see here these two cables. I have connected with this to shower pins. And this is power not needed to use. And first is ground not needed to use. So you can see I just use this middle two teams at here. This fire what not used and this grant did not use just these two pins. Okay now I will connect to this network cable and the USB cable and show you how to work with the home assistant. And how to set in a home assistant. We can see here that ESP home we can and the keypad, you're just a copy and paste is very easy. So that you can define the T one and T zero. This two piece. This T one is used by the Wagan not ESP 32 D D one and D zero. So this D one and D zero is for this weekend put and we just copy this cord and a to your ESP home. So you can see here I can go to the ESP home. And that is e 60. S bar this bar is online. So I can click Edit. And you can see here that I have copy and paste edit here. I had defined as this is the ID just chose by yourself and we see GPIO defined for this ESP 32 Two pins and this is output the key have pressed and the same to the x and this is the talk that have used my finger Later I will show you why it works. And this time, we can also add this code that is for ki collect. So you can see that is a key correct component of the ESP home so that because l can I press a button or press R key that will consistent with number. So you can see here. I also copy and paste this code and into my ESP home. Just at this sections So this is very easy to change. So you can see here, the Id just the pin lead to find by yourself. And as a source ID, my keypad, because we have created a keypad, you can see here just on my keypad, and this is the me lens, just three number because I have input a 101.



```yaml
              lambda: |-
                return x=="3";
            then:
              switch.toggle: e16s_output2

    on_raw:
      - lambda: ESP_LOGI("RAW", "received raw %d bits, value %llx", bits, value);


# Example configuration entry
key_collector:
  - id: pincode_reader
    source_id: mykeypad
    min_length: 3
    max_length: 3
    end_keys: "#"
    end_key_required: true
    back_keys: "*"
    clear_keys: "C"
#    allowed_keys: "0123456789"
    timeout: 5s
    on_progress:
      - logger.log:
          format: "input progress: '%s', started by '%c'"
          args: [ 'x.c_str()', "(start == 0 ? '~' : start)" ]
    on_result:
      - logger.log:
          format: "input result: '%s', started by '%c', ended by '%c
          args: [ 'x.c_str()', "(start == 0 ? '~' : start)", "(end ==

      - if:
          condition:
            lambda: |-
              return x=="101";
          then:
            switch.toggle: e16s_output1
```

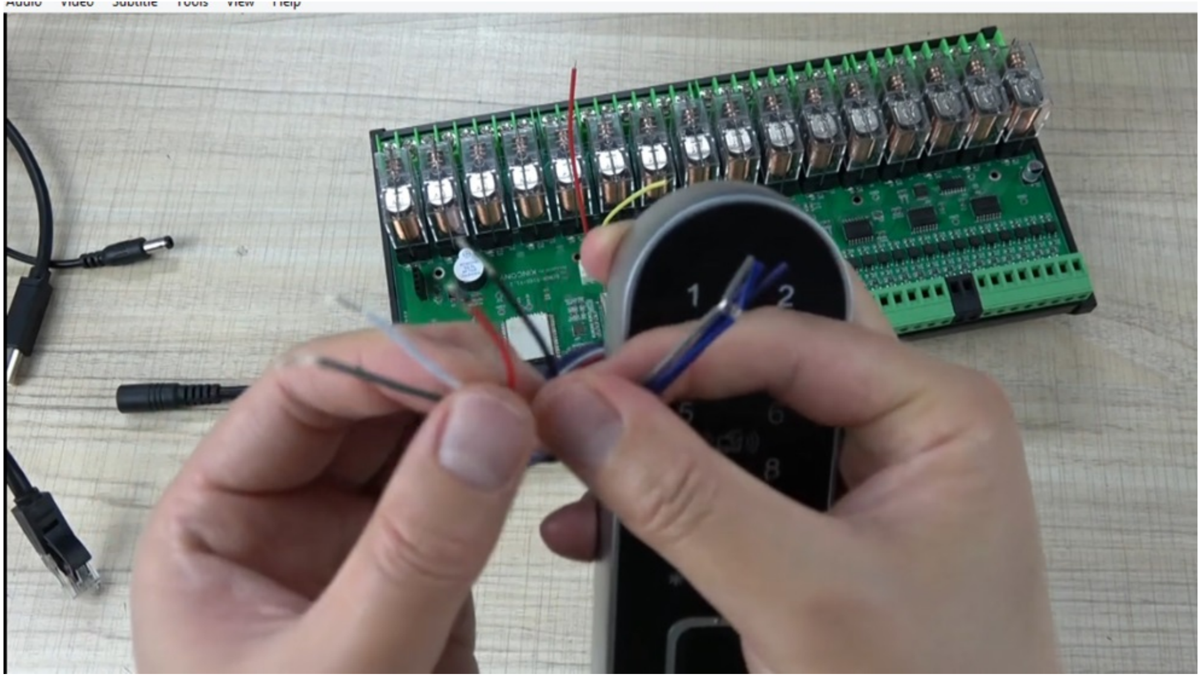
So that means the three and the max also yesterday, so I only accept your input a number, slip it, and is the End key that this, this button, and this one, you can see here and exactly this one for end, if you want us by this way you can change. And this is true, because we need the End key. And this is the back, maybe sometimes you have input and while you can back, Backspace, Backspace and use this, this button and this is for clear for I have not had this see, if you've had have this see you can sit at this step sections. And this turns out if you input a button, but the time out that five seconds or you can see it large or smaller. And I'm pressed process that is important, this will be feedback. And this is input in prayer progress that mainly just this means in progress and if you input this buttons, that is end, so that will be a result, this is the result and for the x, so we just make a condition for the x. So if x is 101, I will toggling the switch that is 16 s output a one because I have quit, you can see here I can create a switch that is E 16 output a one just as this ID or put a one. So if I haven't returned the x that is 102. So we can type in the switch that is thing as are the two. So if you want change to 202 and that can change anyway, so you just can change this number. So, that will be very easy. If you see one there are three that continuously, so you just change by this way. So this is up to you. And this because I have slipped that you can see that is ribbit, if you can change to the four, the four bit like this, the four so you'll maybe you can set when the other the other one that is continually one. So this is input phone number, and to turn and turn off the relay one. So now I can change to the switch so that I can save the time and only input screen number. Okay, you can see this is a core. So that is very easy. But the important is how to monitor the log output. So you can see, I can click See and click Install and click by the USB cable, because I have already installed so I can cancel it, I just click to monitor the logo found you can see here, I can click logos by the USB cable and click now you can see here this is logo output you can see I press one button well you can see that ESP Home Hub code that is one you can see this input progress because I have not imported the end button.



Logs kc868-e16s.yaml

```
INFO Reading configuration /config/esphome/kc868-e16s.yaml...
WARNING GPIO15 is a Strapping PIN and should be avoided.
Attaching external pullup/down resistors to strapping pins can cause unexpected failures.
See https://esphome.io/guides/faq.html#why-am-i-getting-a-warning-about-strapping-pins
WARNING GPIO15 is a Strapping PIN and should be avoided.
Attaching external pullup/down resistors to strapping pins can cause unexpected failures.
See https://esphome.io/guides/faq.html#why-am-i-getting-a-warning-about-strapping-pins
INFO Starting log output from /dev/ttyUSB0 with baud rate 115200
[12:20:38][I][RAW:070]: received raw 4 bits, value 1
[12:20:38][I][KEY:050]: received key 1
[12:20:38][D][main:147]: Input progress: '1', started by '~'
```

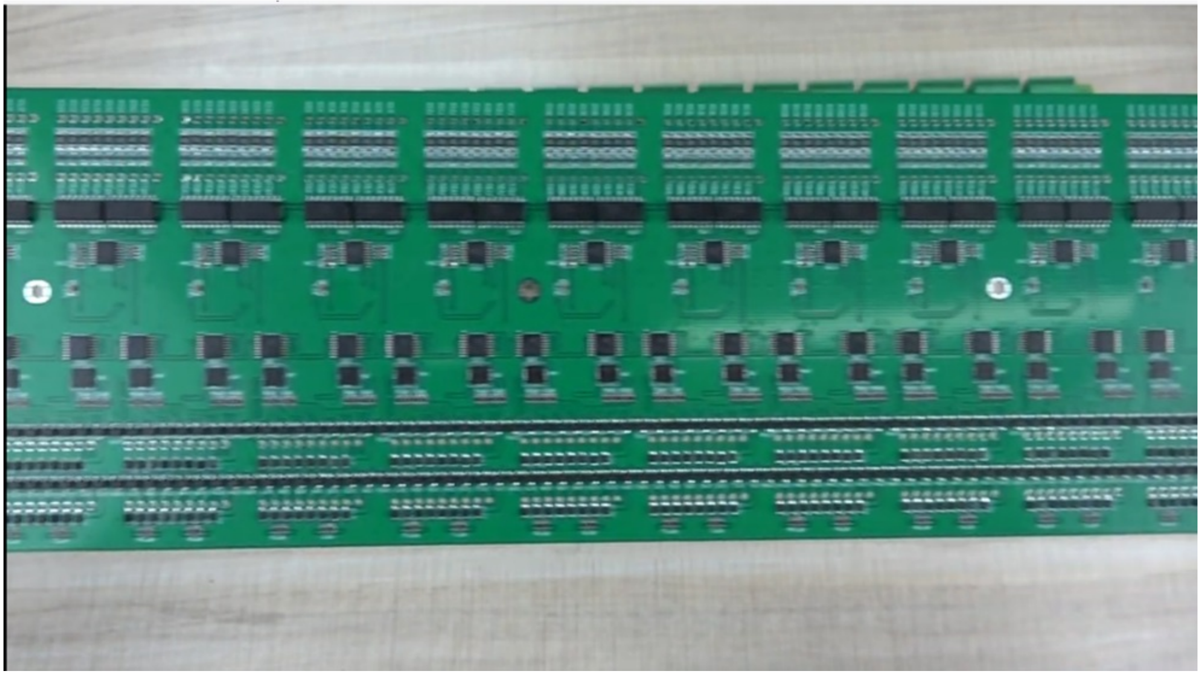
So if I can input slip button 101 And as you can see that means that result the input is a result I have input a 101 because we have a start and we have end of this but so this helped the code as a whole number. So that is 16 as part you can see the output is arm because it's talking it's so if I can press a one, zero to end and you can see that you have the code the input the result is a 102 so that we can toggle that really to so you can see as this relates to the target. So if I can input When to not have the end, you can see that means in progress, that means not have the end button, just you can use for style speak, when you press this button, maybe have a music or maybe have a speak just to have a voice. So if you have press this three button when we are those three and so that means the result because we have defined the three members of the panel. So you can see this help received a W that is a code for the weekend. And this is received the key you can see I can press the one that's the key is a one. And if I can press this button too, you can see that in the receive tool. And if I can press this now, so you can see that is press nine. So this can decode error patterns or information output to the ESP home. And the ESP can can quit the automation. So you can see here because we can decode and key on tag Aw, the IW is the XO code of the buttons. The tag I will show you that used by the finger and IC card I will show you have the later so this key also is very useful. And when you decode this key and target and IW so you can quit home automation in ESP home because this have an action so you can do anything you want like a two hour relay 1230 this button and turn off relay 123 Just it's up to you. So yes, this is the code and quit the automation. And this is how to work in home assistant because of the weekend people keypad components. So this is where we're useful. Okay, let's look at another key path. Now I will show you how to work with a fingerprint. So that way we use four we can use a different finger and the content is different really. Okay, let's power off and disconnect. Okay now I have removed this cable found this part you can see here now I will show you and connect to this panel.



And this is this one and also we will use this just for wise so you can see that is for power. And is actually the foreground the black. And this is for date according and the white just lists for whites to use. Okay we have prepared to this panel. And you can also see the logo file output and by the USB cable. And now you can see that here if I can press this finger and you can see that have output to the information that means the target this tagging tool because I have already reduced with this two finger this the finger tool and the fingers re I have logist so you can quit many fingers so that you can create a time finger to your panel no problem. And this is my sort of finger you can see here if I can press this figure and you can see that have received the target value yesterday. So I have prayed to the to automation, that the finger tool, we have country in this relay and the finger three, I can control this way. So you can see if I can press this finger that will speak thank you and turn off this relay and this finger. You can see here that this relay is off so I can toggling this tool relay by finger. So this time you can see the output that is target. So you can see the ESP home. You can see here at this time that is output from this tagging value. So we can just compare this x we can go to the ESP home and edit you can see here this is our tag automation. So you can also use this code you can see this decoder just pista from the ESP home and you can just quit your full condition and if the X is two In this ID I have preset from my fingerprint to access country let it out too. And if output to that will be toggling this is acting as pod relay one if I can get the x is read just as this finger so that will Toby that e6 thing as part relates to so this can according to the different finger ID this the finger ID and the country it differently so if I needed to edit the first the fourth finger I can show with you how to work it I just needed to see your access control system because this how our code how to edit a finger so this is how to add the finger so that's the ministry password firstly 1234 This administrates now I can add a finger new finger press the finger this this one please press finger again again please press finger again okay, returns the number okay this number just a tiny number. So I can quit four and enter I just heard successfully is press the finger okay you can see this finger have reduced you can see already registered you can just press the finger okay you can see now okay, except because I don't want to register my finger. So, I just can press this yes yes except measurement. Now you can see I can click this the ESP home and local output and by the USB cable and now you can see I will use this finger I have just new. So you can see here and you can see that Hal received the target is four. So this the number four I have reduced to this access country and let this input is for because I have quit so you can quit this file the fifth finger that you can get the tech become file and this becomes six and seven and eight 910 So you can quit tagging from one to 10 so you can do some automation found different relays so that the first day you can relay one and the second you can relate to another 30 you can raise three and just up to you you can define different finger for different assess mode. So just according to this target, so you can see here the target so you can quit if I can let the release Lee and country by my first finger so you can see I just can copy and paste this if condition copy and paste it's very easy just to paste that here under the condition that is a four because my new fingers for and choose this screen entrepreneurs free so you can see I can see and install by the USB cable wait for moment okay, now you can see one second you're connected successfully. Now we can test it with my first finger that have new array just so you can see. Yeah, can you can see that will speak thank you because I have set the chinos array is auto off. So you can see I can add a turn off. This is automatically turned off because you can see here that the release rate that have auto off for one second, so that I can use this finger for output applies. So that will be out of after one second. So this is how to use the finger to control this differently. And they use this panel and the input to the password or input to the number to control the array so you can use any access control panel just use the Wiegand protocol integrator Hamas isn't the full automation very okay,

ESP32 BOARD_ KC868-A256 512 GPIOS FOR HOME ASSISTANT

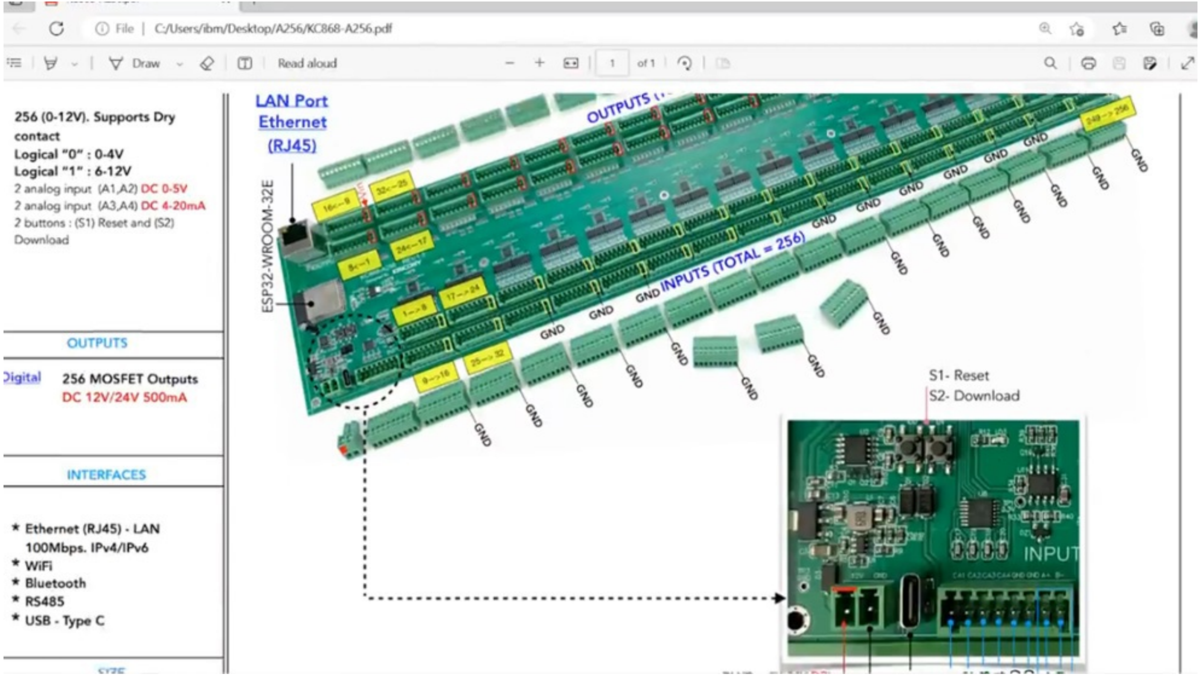
I will show you our new product, maybe you will think I'm very crazy, because we have released this part, this is case 8688 256. So that very big part but it also made by ESP 32. So you can see this part will totally help 512 jous just the for 256 digital input 256 for digital output. So you can see this in my hand and this bird so less almost as my leg and this bar we have designed about four months have very long time because we're the hardware design and I'm testing with this part and the six for the ESP home technical support. He helped helped me add this product to the ESP homes a new nest of verson so that now you can use this bird and integrate with a home assistant to buy the ESP home. Because before it was done the ESP home NEMA support is so much cheaper house by the homeless for home assistant and and now he helped me and it works this solution the hardware solution so that you can use this bar with ESP home very easily. Okay, let's look at this part hardware details Okay, let's look at this a 256 that is made by the ESP 32. So, we can see this part is very big that have plastic issue you can see this part is Pastificio installed on your king rail. So, you can use this thing well, you can see here if you want installed into your power distribution box, you can install by this license way Okay, let's look at the some hardware details you can see this is the Ethernet and this the ESP 32 module and S one and S two Parklane that is for reset ESP 32 and the GPIO zero for defined function by yourself because the ESP 32 support to us by your Arduino source code or you can download the ESP home for home assistant and this is the power supply the power supply is the part where what is part 24 What it's up to you and this is a USB type C. So, now you can use this type C cable and be ready to connect with this USB C socket. So, that will be easy to connect with your computer. And here we can see how many many many different terminal and this you can see that is input this terminal is input and this is output. So, you can see how to use this input and output because this socket isn't removable. So, you can just remove this target. So, you can either connect with socket when you are one buy this school and plug it in to this socket that will be easy to connect with a why and then let's look at this pin define and how to use this bog I can now I can show you abscissa be part the PCB part is not complete just to not soldier this terminal without this terminal. So that we can see this pin defined easily.



So you can see here that is the PCB part and this is the back you can see so much components many many components. So you can see and this is the input input range and also have support long distance use the drag contact sensor or drag content momentary switch because we have some so many chip for the input. And here you can see that is output the output we have used the MOSFET there are so many MOSFET output. So you can see here our channels have one MOSFET so that totally how 256 MOSFET out okay, let's look at this pin different details. And you can see here because we use a PCB partner we will say this pin to find easily not Have a four channel analog input actually two channel is for DC zero to five volts and the two channel is for four to 20 milliamp pair. So, that can support different sensor and here the A and B that is for is for it fine for long distance for the multipath you can write your Arduino code for Modbus Communication with this part and this is the input you can see the input that have 1234588 channel input you will see how background so, this just use this input a shot with a quad that means triangle input a one if you want to track with the input of two you can short with ground oh you can connect with a 12 watt to this input that means the logical zero so, short will count that means logical one and connect with a 12 What or not connect anyway that means logical zero. So, you can say this is for each channel and this is for an eight channel you can see that is 492 16 channel digital input. So, this is the ground. So, you can see this is 7234 input ground and 25 to 32 and ground. So, actually on Tuesdays you can see here you will see that 256 And also have background. So, every channel LH channel we have a ground for commonly use. So, you just a shot for the ground for triangle the input. So, this is how to use the input and this is the back of the input. And let's look at this art. So, you can see this output the output you can see also you the 12345678 So, you can see that here we'll be watching and here you can see the nine to 16 channel output that also have watching that for the power supply for the MOSFET and you can see our each channel however in every channel each channel have 42 v. So, while we have designed this for because if you want this eight channel output 12 Watt, so you can connect with this whole with a 12 watt, but if you want this eight channel output a 24 word so you can connect with this whole with the 24 Watt so that you can use eight channel output a 12 watt and an eight channel output 24 word that will work for different odd for the watch. So you can see every channel have that independently this whole so you can connect with this whole who is a different watch. So that these eight channel will output to the watch according to this whole. So you can let this out of the 12 What and this output is conditional what are these out for the 24 What are these out for the 2012 What so just According this whole this whole this whole this whole so I think you can use this part output a different watch for your different load. Sometimes maybe you have connect to extend the lonely or sometimes you have connect song value or songs sometimes you will connect different power for part of the load. So you will be flexible and output to the different Butch with the different most feet out. Okay, this is how to say this output terminal and you can see that is the most fit. I will most fit that we have designed use fat and pear enough for you to use. So this part will be easily to install to home assistant by the ESP home.

ESP32_ 512 GPIOS WORK IN HOME ASSISTANT

I will show you how to integrate sis largest ESP 32 part for home assistant by ESP home you can see this is case 868 A 256 This part is very large and there are many different digital input and a digital output will try to challenge the limits of the home assistance Okay in this video I will show you the first step we will look at the hardware details and the second I will show you how to integrate and set by the ESP home. And now let's look at this is a part because this part I have removed this socket because I have already connected the wine I just plug it in the socket will be easily and this is a PCB board we have designed you can see that how two different sides there are so many different components and the chips and the resistance and many different chips on this part two to layer all have this power parts. So this is very big. And this part also you can see that the thin blue that have plastic button you can see here we can install this thing just just an axis so that they can fixed on your power distribution box. Let's look at this is a webpage about the aid 226 You can see here that on our web page there are many different chip shells. So that you can see this will be you can see the clearly so you can see here we can get to the large data data output this is oil output 256 channel and this input input this the Ground Round Eight channel will have a ground so this is V for our eight channel so that you can let different output terminal output a different watch so that you can use the 12 watt or 24 watt and this is for the power supply.



256 (0-12V). Supports Dry contact
Logical "0" : 0-4V
Logical "1" : 6-12V
2 analog input (A1,A2) DC 0-5V
2 analog input (A3,A4) DC 4-20mA
2 buttons : (S1) Reset and (S2) Download

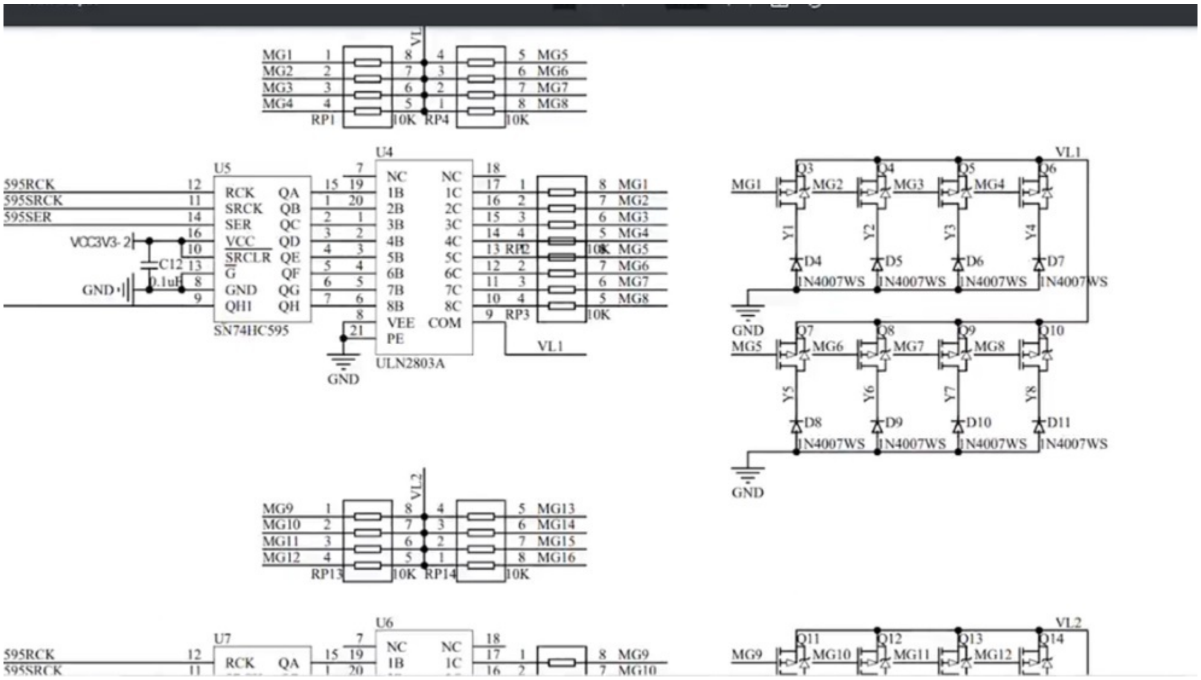
OUTPUTS

Digital 256 MOSFET Outputs DC 12V/24V 500mA

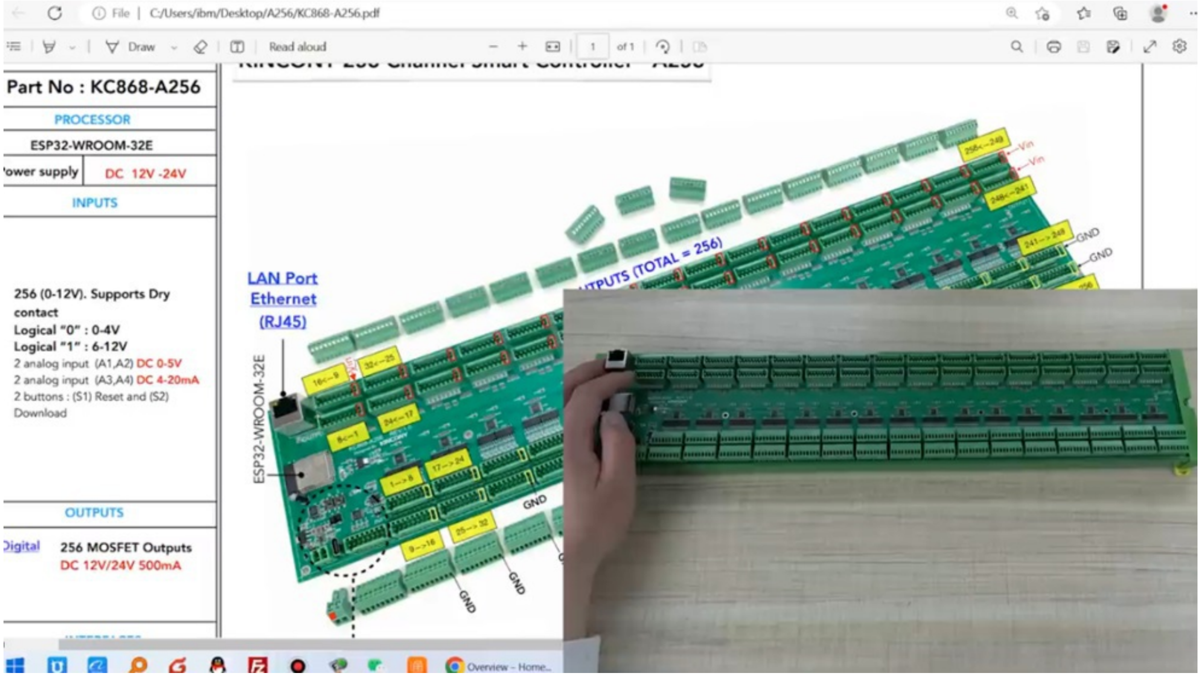
INTERFACES

- Ethernet (RJ45) - LAN 100Mbps. IPv4/IPv6
- WiFi
- Bluetooth
- RS485
- USB - Type C

So that's the part will be very powerful and so many different input and output and also there are some different terminals you can see and then I've input and the fortunate and is for it fi interface for communication and this is power supply okay this is a hardware terminal details and we can also see this PCB board we have designed you can see here that the PCB also is very large and we can see in 3d version. So you can see there are many chips just extended the GPL for the ESP 32 And you can see the Tiguan schematic, you can see the service and document and the you will see all our open source bird you can check this a 256 This attack one and then many many Tiguan that use this segment for hc 165 chip for just for the input.



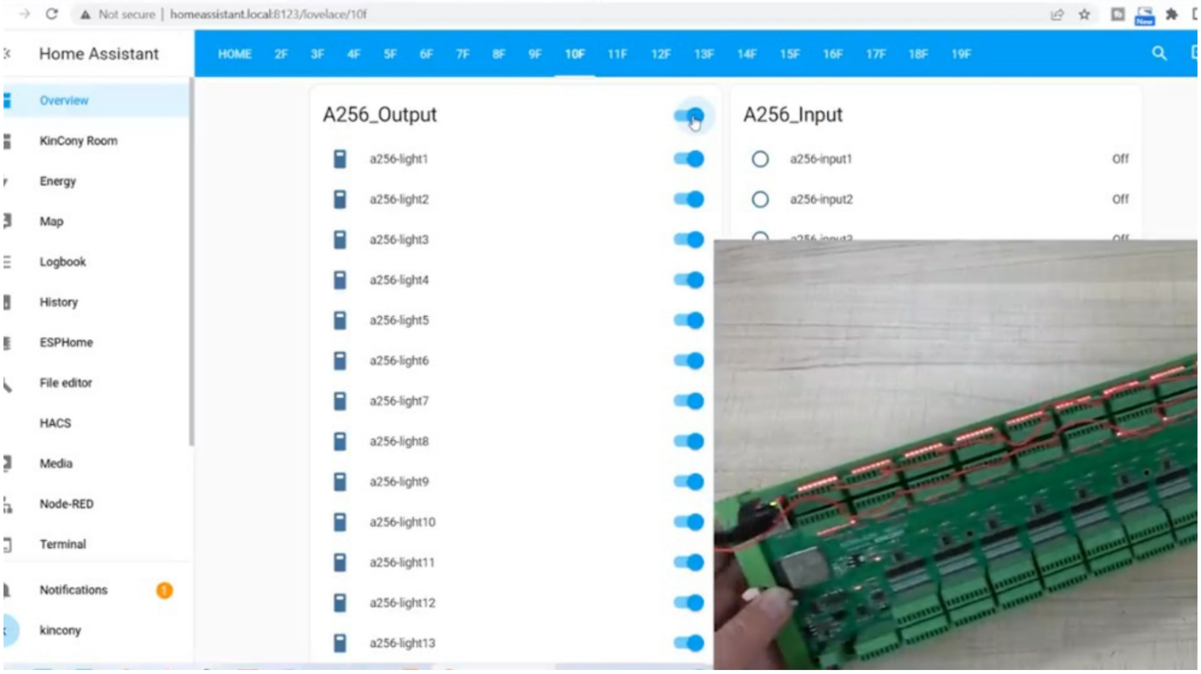
This is a four input. So we are held connected with many chips to extend this input just like this. And here you can see that is seven four hc 595 These chips that's used for extended output. So the output stays most fit output and we can see as they say Tiguan for ESP 32 Okay, now this time we will look at this home assistant dashboard you can see we will have 256 channel output, there are so many many output and input parts. And we have already posted ESP comm Yama file on our form. You can see here, you just you can download this file, we can click and there's so many config file to underline. Let's look at the first thing we need to plug in the terminal.



File | C:/Users/ibm/Desktop/A256/KC868-A256.pdf

Part No : KC868-A256	
PROCESSOR	
ESP32-WROOM-32E	
ower supply	DC 12V -24V
INPUTS	
256 (0-12V). Supports Dry contact Logical "0" : 0-4V Logical "1" : 6-12V 2 analog input (A1,A2) DC 0-5V 2 analog input (A3,A4) DC 4-20mA 2 buttons : (S1) Reset and (S2) Download	
OUTPUTS	
Digital	256 MOSFET Outputs DC 12V/24V 500mA

This is for output because there are many V so we need to connect it together and maybe you have different watch so you can connect with a different power supply and you can see here I have let's see So much terminal work connected together. So, you can see this is the first row and this is the second this the second law in this law I have used this smart cable just to you this cable. So, this small cable that will be connected easily just use this cable and our terminal you can see because this terminal is V So, I have the first terminal and connect to the second terminal V and the third V So, that will be many many many connected together that are just for testing so, you can see this have totally 1-234-567-8910 1112 1314 1516 So, that will help certainly to channel terminals so what can you use the multimeter data just make sure that how connect so, the first minute and then the last time I'll make sure I have connected together our Code of Conduct now that's probably you can see this is the first one you can see I can just having the second okay you can see now the first row that you connect Oh V is connected together and this is a second row okay, this is the last one but the last one we need to connect with this power and from this 12 What do you can see I have used this cable that is for QC 12 What Okay, now we're can plugging the last one. Okay, see, I can probably now you can see the RV can connect together with 12 watts. And now we can connect with this network cable. And the let me power that you can see. And you can see the LED under the network cable. So earlier the blink. Okay, let's go to the home assistant. You can see this dashboard, this terminal and you can see that will become available so that you can see so many different let's test with number one switch. You can see the early indicates and then number two and number three. Okay, let's test with the Oh, you can see I can click this one that many different switch is on.



A256_Output

a256-light1
a256-light2
a256-light3
a256-light4
a256-light5
a256-light6
a256-light7
a256-light8
a256-light9
a256-light10
a256-light11
a256-light12
a256-light13

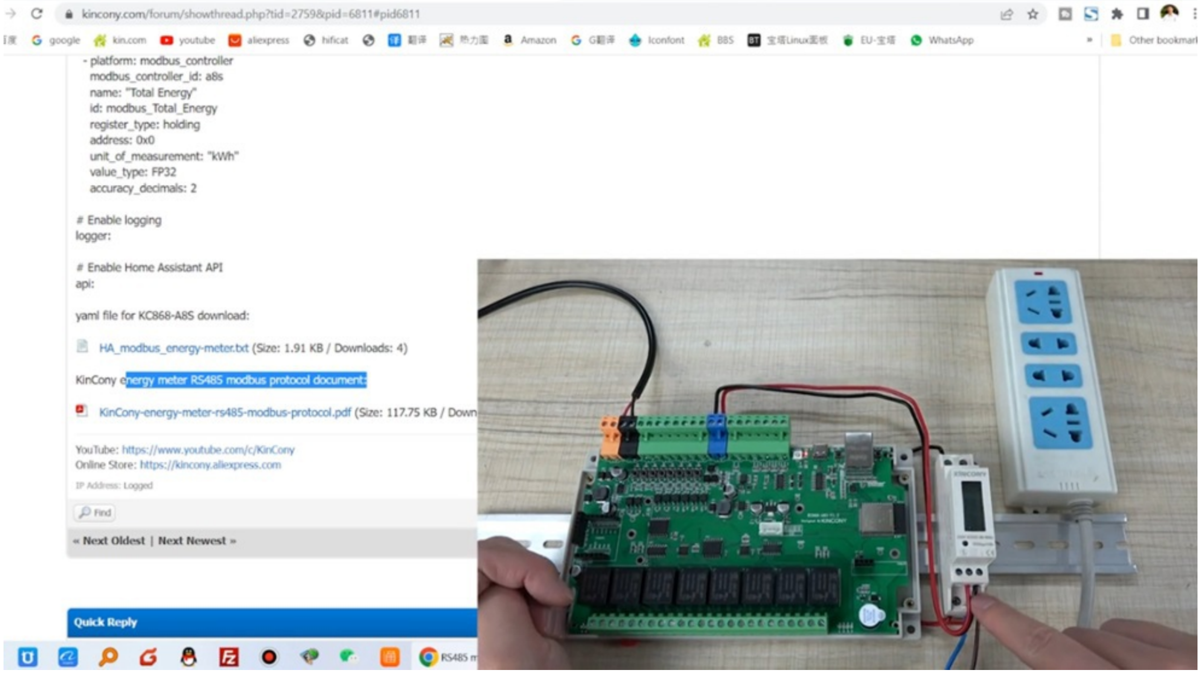
A256_Input

○ a256-input1 Off
○ a256-input2 Off

So you can see also the home assistant dashboard. You can see oh yeah, if I turn off you can see I turn off and you can see the home assistant update update update very fast. So you can see this I can change with different switch for an NF and hot set by this one. We can go to this ESP home and quick tap a 60 and quit a 256 part you can see I have already quit this part. You can see the edit as this code you just find from our website, just this one and I have opened it just to copy and paste. You can see here just copy and paste that you can see so much command to the last command. You can see almost four solids command for this part, input and output. And after you're saved you're just You can install and download the Paris USB cable I have already download so I just tested the result for you click overview and dashboard and let's let's test the input. So, you can see there are so many input digital part for your can use. So, you can see when I shut the input one because that needed to shut for the digital input with ground so you can see the Tiguan this ground this ground is grown with ground one. So you can see I can shoot for the digital input one this part is the Quran and this is a digital input one. So you can see in my home assistant dashboard you can see that is so this is the first one I have trouble I released my why that will be off and this is channel two. You can see I shot I shot for Channel Two that and release it off. Okay, let's test the 256 the last channel so we can see this part. This is the last one you can see as this is also grant and this is 256 channel you can be assured for this it will be on and another channel so you can see that will be so our this channel can be detected in home assistant.

ENERGY METER ADD TO ESPHOME BY RS485 MODBUS FOR HOME ASSISTANT

I will show you how to integrate this energy meter to home assistant by ESP home. You can see here that is our eight s bar so that you can see that is as for it fi interface so you can connect with energy meet that the protocol is Modbus so that we're easy to extend the device by us is for it by Modbus protocol. And the you can see here that is our part. So this eight s pod have a copy of ies 485 Actually you can use any Brandis product not only can Coneys just to use by ESP 32 Oh ESP 8266 so that you can use the ESP home, we will look at this dashboard of the ESP home. So, you can see this home assistant dashboard today when we use this dashboard that can monitor the voltage and the current and the power and the frequence and the total power consumed consumer. So, you can see here that is our hardware device. And actually you can use any Modbus energy meter not only can Coneys but if you want to choose the company's power meter, so, you can see our official Express online store you can see many different power meter, so that you have one face or three face large current or small current is according to your load. So, we can see this is an ESP home Yamo file, I have posted our form. So, you can just copy and paste that is the TXT file you can download LDC only Demeter put that in motor bus protocol, you can see you can download the file this file, this is our meter and you can see this is the the pin define that is a one and two and the three and four just to leave line input and leave that output and then the neutral input and neutral and output and you can see that is a and b is 485 and connect with this 11 and 12 pins.



kincony.com/forum/showthread.php?tid=2759&pid=6811#pid6811

```
  - platform: modbus_controller
    modbus_controller_id: a8s
    name: "Total Energy"
    id: modbus_Total_Energy
    register_type: holding
    address: 0x0
    unit_of_measurement: "kWh"
    value_type: FP32
    accuracy_decimals: 2

# Enable logging
logger:

# Enable Home Assistant API
api:
```

yaml file for KC868-A8S download:

HA_modbus_energy-meter.txt (Size: 1.91 KB / Downloads: 4)

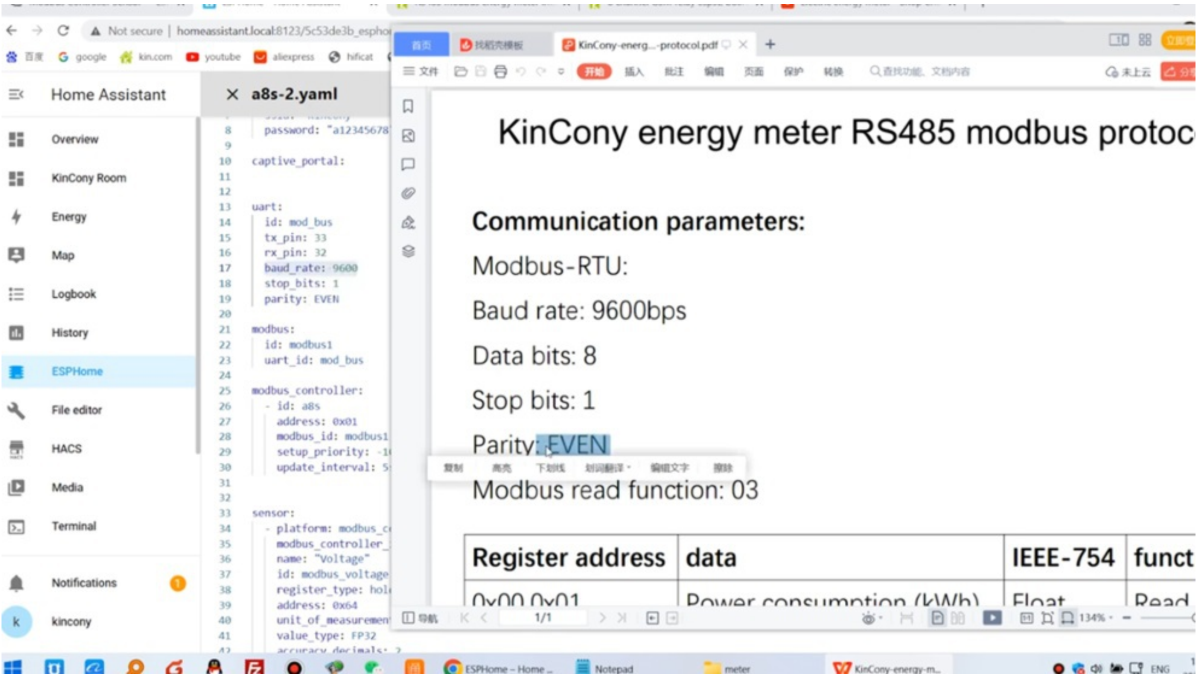
KinCony energy meter RS485 modbus protocol document:

KinCony-energy-meter-rs485-modbus-protocol.pdf (Size: 117.75 KB / Down

YouTube: https://www.youtube.com/c/KinCony
Online Store: https://kincony.aliexpress.com

IP Address: Logged

So, I have used this L and Newsha incoming and the for this part and Corsola is energy meat and to this power socket and this is for communication. The hardware Dagwon is very easy. Now let's look at how to integrate it by the ESP home. And you can see as this is our protocol I have download just like this, because if you have an energy meter you also will have this document or file in your cellar and also you can see this is a file configuration file. Okay, let's begin go to the home assistant and this ESP home and we need to create a new device and continue under the name we can choose to HS this this is a second one because I have already integrated the first one and this will use the ESP 32 This will can click Skip then you will see this is it as part of the second part we can click it and because I have already created so, you can just copy Ctrl R select all and copy and paste it here. So, you can see just a little code the pardoning and this is the Wi Fi SSID and the password you can also use the PI the Ethernet I just show you guys this Wi Fi actually will suggest to you the Ethernet and these are important is began this sentence. So, the first you need to define are you at that use by this CI X team and X pin. So that is according to your ESP module. And the benefit benefit, what you do is just say your document on your energy meet, you can see here, King Kunis when which is nice 600. So you just feel this nice 600 And the stop bit and you can see the stop bit is one. So this is one and the reason apparently, and this is even so you can just copy to this one. So this help define your server part.



a8s-2.yaml

```
  password: "a12345678

captive_portal:


uart:
  id: mod_bus
  tx_pin: 33
  rx_pin: 32
  baud_rate: 9600
  stop_bits: 1
  parity: EVEN

modbus:
  id: modbus1
  uart_id: mod_bus

modbus_controller:
  - id: a8s
    address: 0x01
    modbus_id: modbus1
    setup_priority: -1
    update_interval: 5


sensor:
  - platform: modbus_c
    modbus_controller_
    name: "Voltage"
    id: modbus_voltage
    register_type: hol
    address: 0x64
    unit_of_measuremen
    value_type: FP32
    accuracy_decimals: 2
```

KinCony energy meter RS485 modbus protoc

Communication parameters:

Modbus-RTU:

Baud rate: 9600bps

Data bits: 8

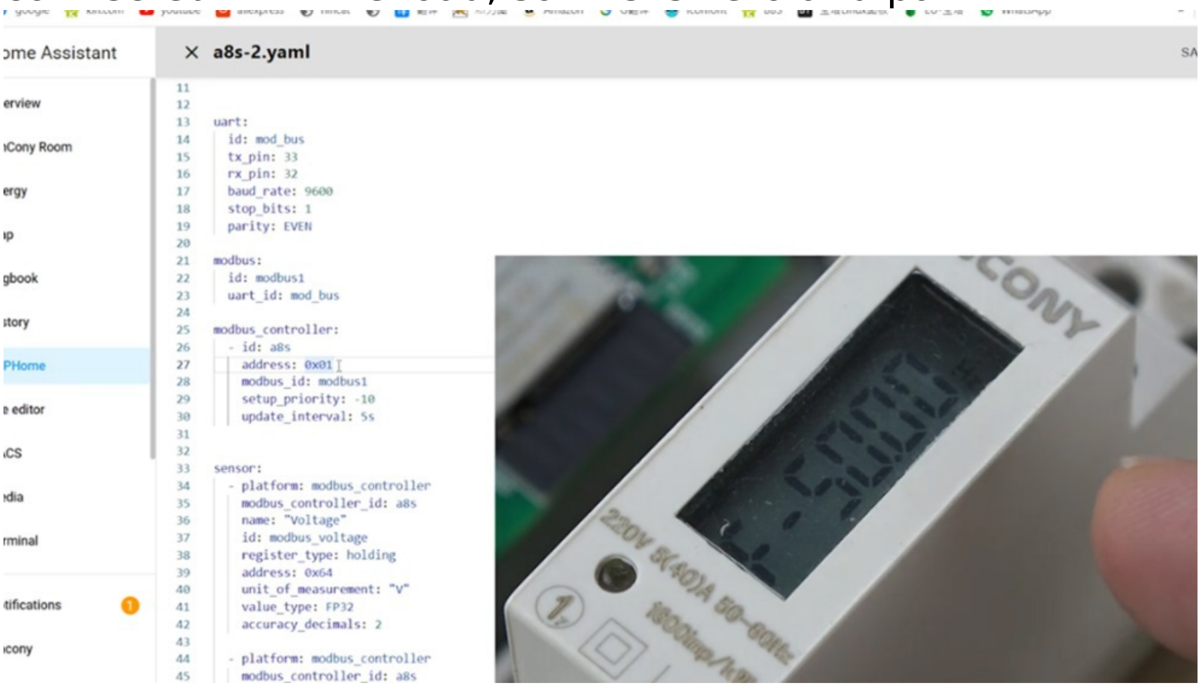
Stop bits: 1

Parity: EVEN

Modbus read function: 03

Register address	data	IEEE-754	funct
0x00 0x01	Power consumption (kWh)	Float	Read

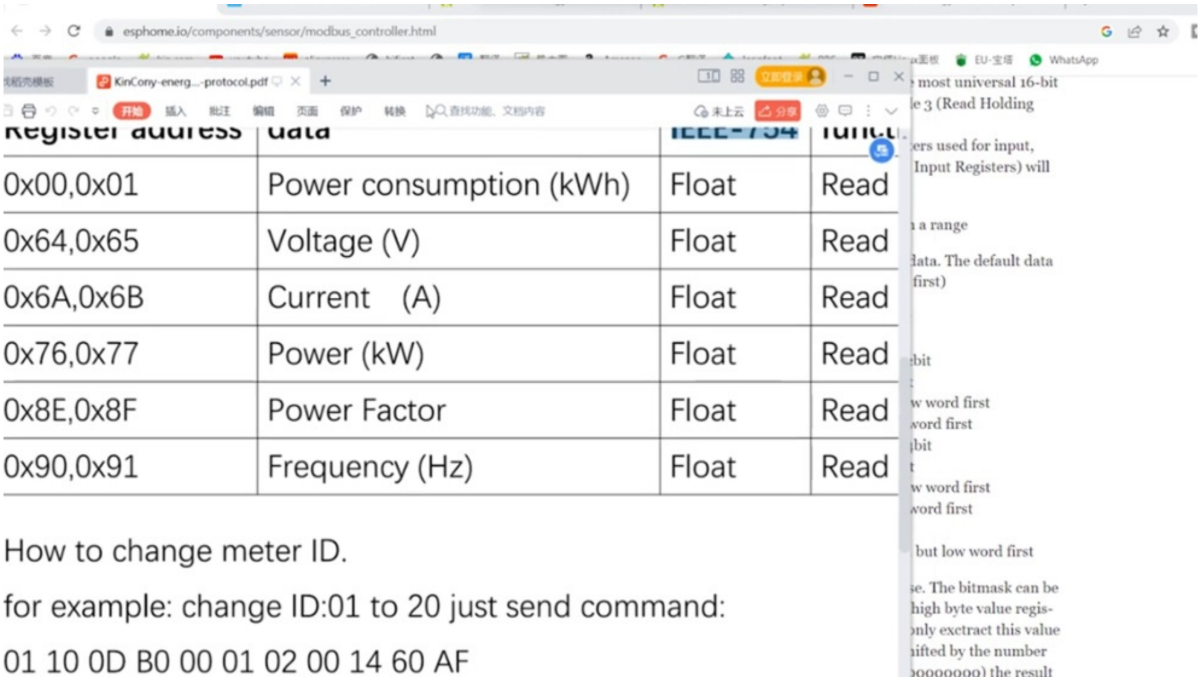
And the next is defines a Modbus this a Modbus ID that is running by yourself. And this is the U AR T ID that we have quit at before. So this ID the modal pass just this one. So if you have clicked a rename this formula plus one or minus two so you can just As simple as before you have defined and you can see here that the Modbus controller, that the ID, you can rename by yourself. And this address, this address means this power meet energy meet address. So this address you can see from your permit. Let's look at the information cam the display Okay, let's look at this LCD display. So that that is frequence this will frequence that is displayed the changed and this is how the totally power consumer this is yellow because the meter is new. And this is address 001 That is the more past address and it is the verge to annual ns 30 watt and this pair because I have not connected with the load, so this is zero and pair.



```
uart:
  id: mod_bus
  tx_pin: 33
  rx_pin: 32
  baud_rate: 9600
  stop_bits: 1
  parity: EVEN

modbus:
  id: modbus1
  uart_id: mod_bus

modbus_controller:
  - id: a8s
    address: 0x01
    modbus_id: modbus1
    setup_priority: -10
    update_interval: 5s


sensor:
  - platform: modbus_controller
    modbus_controller_id: a8s
    name: "Voltage"
    id: modbus_voltage
    register_type: holding
    address: 0x64
    unit_of_measurement: "V"
    value_type: FP32
    accuracy_decimals: 2

  - platform: modbus_controller
    modbus_controller_id: a8s
```

And is power fact also is zero because I have not connected with a load. Okay, let's go. I can do to one is bad. This is one past device ID and this address because we have seen the LCD that is the other one, you can also use the Modbus protocol you can see here, you can change this Modbus energy meter ID. So the default is 01. And we just use the other one directory and this is Modbus ID that defined by yourself. And this is update interview. That means every five seconds will update and eluded this data from the meter. So if you want fast and the leader frequently and you can change to one second or two seconds or three seconds, and the last would just quit the sensor because the watch and pair and the power just to work as a sensor in home assistance. So this in ESP home, we can create our sensor.

So you can see here that the Modbus sensor details in home assistant official website. So there are many settings and details you can see at this webpage, you can see the example how to set the voltage and current and battery capacity. This is for this ng meet. Okay, let's go out. So you can see the first we need to choose the print for that in the mode bus controller and is the mode bus ID so that we need to change to meet one because we'd have chose the name of the meter one meter one okay, I have changed, proceed. And you can see the name The first is watch. And this is an ID just to set by yourself and then we'll just type use the holding just all settings is holding holding because we have a fully held container or relay board that is coin and this is holding and it is addressed according to your motor bus protocol. You can see here we have this table for our energy meter. So you can see if you want to lead the watch. And His address is 54. This is 54 by Hicks from it and it is a measure unit that is a watch. So this watch and there's a polio type that's Fp 32 Why is Fp 32 So you can see here that's how value type. So you can see this is all types of the value but choose which one according to your protocol. So you can you can see here we have used this flood datatype so you can choose this one ie 754 this format so we'll use this one.



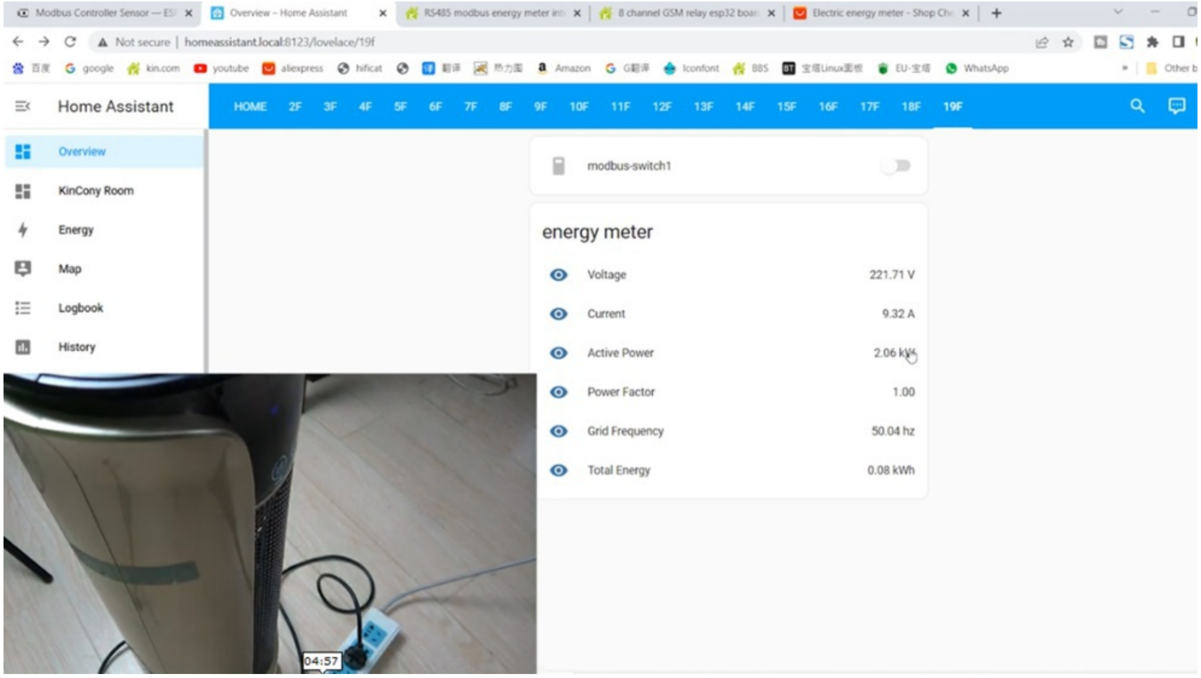
Register address	data	IEEE-754	functi
0x00,0x01	Power consumption (kWh)	Float	Read
0x64,0x65	Voltage (V)	Float	Read
0x6A,0x6B	Current (A)	Float	Read
0x76,0x77	Power (kW)	Float	Read
0x8E,0x8F	Power Factor	Float	Read
0x90,0x91	Frequency (Hz)	Float	Read

How to change meter ID.

for example: change ID:01 to 20 just send command:

01 10 0D B0 00 01 02 00 14 60 AF

But you can see as that is 14.1 to read just because you can see here you can see here ever really resistant. You can see we used to unit to logist. Still We should choose this one FPS 32 And we go back to the home assistant. So, that is Fp 32 And this decimal there are two two. So, that will according to this settings, so, this is for much and we also want to the M pair of the current. So, you can just copy this code and paste it here and just the name this the address you can see you can see the current address is six A. So, I feel this six A and this is the measurement unit and also used by to readjust address. So, this is Fp 32 And also you can step by step this is certain one just the management of the power is the power. So, you can see the table that is 76 This is 76 So, on to copy and paste and just replace all this different address. So, you can use this way for this just this is power effect and frequence and power consumer. So, this is total energy power consumer how much energy you have used. And you can see that is the yellow because this table that is the yellow so you just need to use this table and this document of your energy meet that will be very easy to quit this sensor in Kumasi sent by this ESP home. After all set by yourself you will just click install and we you will use it USB cable so you can see I have used this cable and connect with Raspberry Pi and will download the firmware to this USB so you can click by the USB and click this one and wait for a moment okay, you can see this firmware DOWNLOAD THE COMPLETE and starting connector with a Wi Fi okay you can see the logo file output by the ESP home that have just some different bit of energy meat. So the next we'll call to close the window and click Device and you will see this ad is bug have found and just clip config and click Finish and we can quit app dashboard because I have so much part so we can just use use this one and I can edit and add a card and the entities and we can choose a name and then you meet then you can quit by your name have created a sensor of the multipath such as this the first one is watch. So you can input a watch that will have grit is watched you can see that it displayed 200 and a certain one lot just this is actually detect and the second okay we use this current. So I copy and paste that current that is yellow because no load and the second and the third actually power to power just paste click on this one and the first power fact copy and paste and the next frequence and the next last one the total energy we have used okay you can click Save and down and now you can see in your dashboard you will see this information that have quit. So this will always every five seconds to leader energy meet and show this information on your home assistant. So that will be ready Today let's test it with the load and the connector with a socket. Okay let's look at this plumber hit the last look at the power and watch and you can see here that 220 volts and that the power is too solid in the world what you can see here okay let's test it I will use this socket when I connect with the sockets because this socket is constrained by our energy meet.


Modbus Controller Sensor — ESP
Overview – Home Assistant
RS485 modbus energy meter int
8 channel GSM relay esp32 boar
Electric energy meter - Shop Che
Not secure | homeassistant.local:8123/lovelace/19f
Home Assistant
HOME 2F 3F 4F 5F 6F 7F 8F 9F 10F 11F 12F 13F 14F 15F 16F 17F 18F 19F
Overview
KinCony Room
Energy
Map
Logbook
History
04:57

modbus-switch1

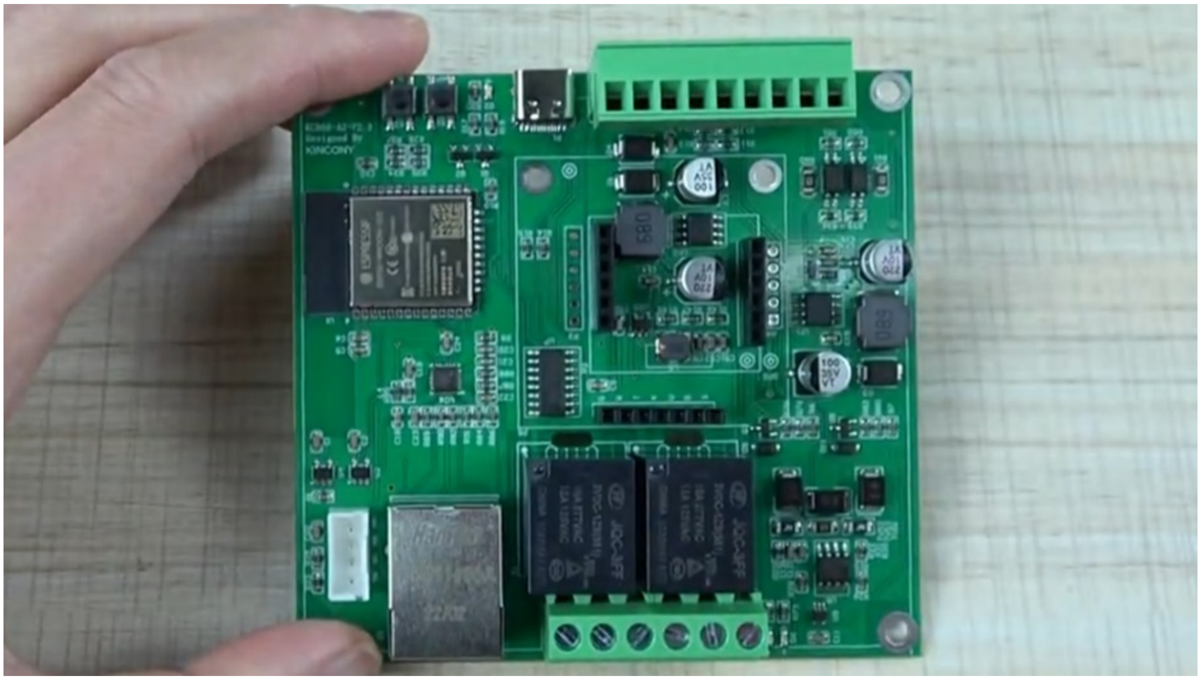
energy meter

Voltage	221.71 V
Current	9.32 A
Active Power	2.06 kW
Power Factor	1.00
Grid Frequency	50.04 hz
Total Energy	0.08 kWh

And then now I can turn on this power. Okay, let's power on and now you can see the home assistant, the current changed under the power it also changed and the let's enable this hot function you can see the sound now it's very hot and we can let change directions and then we can see this almost Tucson in the what and the state also changed about Tampa and this is a watch that have just played. So this is very hot. And this current is very large. And it is a temperature that's how big up you can see if I turn off it because it how activate hang. So you will see the current will be smart. And there's a power also will be smart. And I can turn off this power. And then you can see that we'll begin the cover to zero.

2G SIM800 4G SIM7600 ETHERNET RS485 RELAY _ KC868-A2

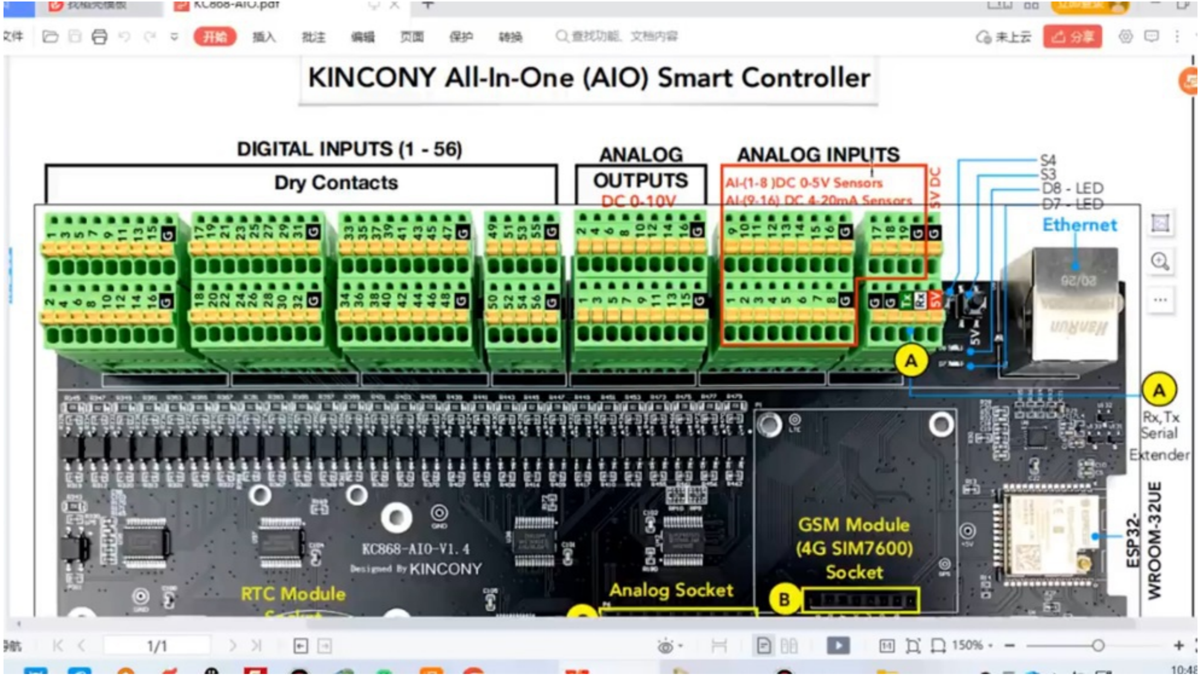
I will show you our new product we have designed this case it six eight a two part is a two part made by ESP 32 module and they are not to release on the PCB part and this time we have had many different communication so, that you can use the AS FOR IT fi and Ethernet and the 4g module or the 2g module back the GSM pay your SIM card. So, you can remote country this Pardubice shot message or by your phone call and this word also was a part of the ESP home so that you can integrate with a home assistant very easy. And also you can write your own Arduino code and for your functions, okay, let's look at the hardware details and this is a two part and the PCB in front of the sides and this is the back Okay, let's look at some details and that is the ESP 32 module.



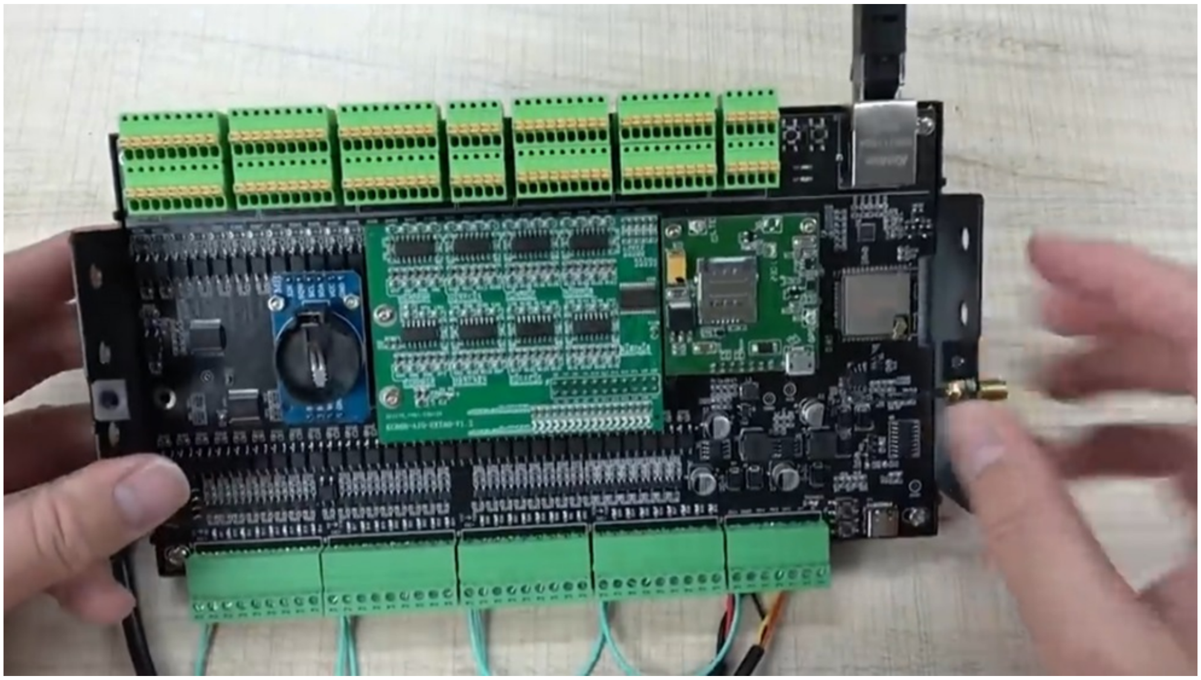
So, that we have this module and this is a Wi Fi reset yes be reset and GPIO zero for defined by yourself and this is USB type C and this terminal. So, we have designed removal. So, you can see here that for DC power supply that support no more much too trendy for watch and is void phi A NP and that is GP house for temperature sensor for the humidity sensor digital input to channel digital input because we needed to use it manually country this really output and so, you can see this part this removable terminal and this is a two channel relay and this relay you can see here that Max is 10 m pair and the elderly will have come and see an N O that inom open normal clothes and come at least the Ethernet port and this is I squared C extend so you can extend your sensor just I squared C device. And here you can see that helps on socket if you want to use a 2g module just like this sim 800 So, you can just install like this plugin this module, so that you can use a 2g franchise and for the SMS to control this relay and if you want use the 4g, so you can see here this module that is 4g module and it is a twitch module. So this is different because some country and Assam city have stopped use the 2g signal. So you can connect with this module at this socket so that you can install the SIM card at here that will be easy to use and also this 4g module will support the MQ TT so you can remote control this part by your SMS or by your mobile phone application. And this is a PCB part. If you want installed pressed cashew, you can see here you can use this show because this show you can see support installed on the DIN rail. And you just you can remove this socket and place that here after the prosecutor show and close it. We can Okay, this is the terminal installed on this box.

ESP32 ALL IN ONE HOME AUTOMATION MODULE FOR HOME ASSISTANT

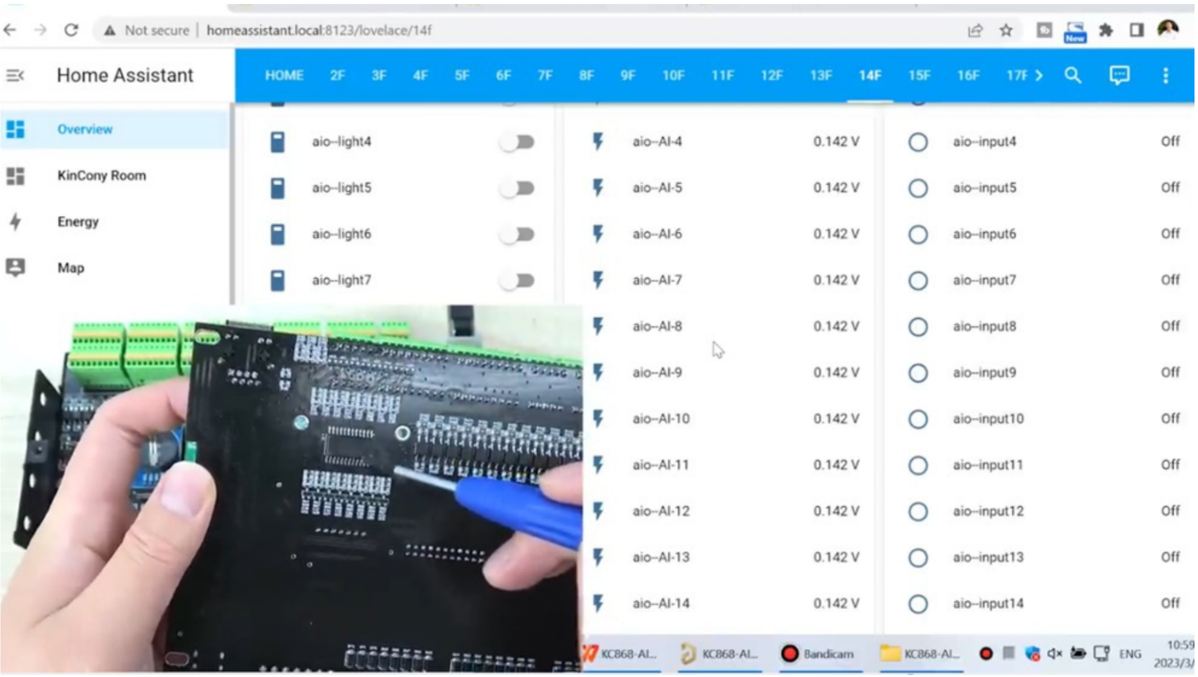
I will show you our new product this will card case 868 ai Oh That means all in one this is all in one resource of the ESP 32 So, this controller is the most powerful ESP 32 part we have integrated the analog input analog output and the digital input and digital output all in one this box and this box we have designed with this metal box just as similar as the edge set it will be pro and kch six eight server so this is very stable and beautiful black box and this box also will support the home assistant by the ESP home and also you can write your own code about Arduino code to drive away all this hardware results and in this video I will show you this PCB design details the hardware details and the height work with the home assistant and with adding it okay let's begin. Okay, let's look at this case it's six eight a I O. So, this is metal box and in the bracket you can see this is a metal and this for many input channel and is the Ethernet port. And this works also support installed on your DIN rail and this is a Wi Fi antenna or 4g module antenna you can install from this hole and also you can install from this site is how many for you to use. And if you have installed into your power distribution box, you can see here that can support installed on the team rail just like this, this will be installed on your power distribution box. Okay, this is a box and let's look at the PCB board we were easy to see the hardware details at this box open this box you will see this PCB board. So you can see that is a IO and in front and this is the back and there's a back sight and we can remove this board. So, you can see here there are many digital input you and I totally have 58 digital input and analog output 16 channel analog outputs and 19 channel analog input and also you can see there are many different terminal on this side and this is for the output that most fit out totally have certainly to channel output and this you can see that for DC power supply. So this product was a product well or 24 What you can see here that it is a power supply and these two cheap shoutouts are for sensor for temperature sensor humidity sensor integrate home assistant by ESP home and this is void fi communication cable a and b and this is a switchboard force power sprang for the sensor such as the DSA 10 p 20 temperature sensor, and these are two buttons that for ESP 32 reset and a four to power zero are defined by yourself and this is a type C USB for download and monitoring led pack for your part and this part were also designed by the ESP 32 module, but that will help IPX socket so you can extend your Wi Fi antenna from this target. And this part is for analog output that is what 60 inch an analog output is that LD indicate. So if you want to use analog output, you just connect this module to this circuit. Actually, this part were encoded with this black part. And this is option for each module. So let's let's look at this clearly. So this is for I squared C RTC module. And here we can see that the socket also in removable, the output is removable and all this input also removable. So you can see I can remove this socket and this is not without score. So you can connect with your one. Just use this this way to fix so you can see we have designed this pod. You can use either two different socket. You can see here that is without school Hello, and this is parallel school. So, if you want I use this type of socket. So, you can remove it you can see this is school version. So, also you can connect with it just as a deferred configuration, we will use this, this socket. Okay, let's look at our form. So, this has a IO. So, you can see here that have many results of the AI obod. And here you can see the hardware details from the datasheet PDF you can see here just this file that is Khinkali a IO that's all in one part.



So, you can see the ESP 32 and different module different module and the hardware details there are so many different dry contact input statistics channel and as you can see, there are two buttons for you to use you can define your function by your code or by your setting and this is the analog output that DC zero to 10 What and either the input ports the port apart, there are five sensors and four to 20 minute and pair sensors. So, I think almost apart everything type of the sensor and this is the output the output you can see that is the MOSFET output solder to channel. So, you can work with a 12 DC or 24 DC and with this extend relay module case it takes it is 16 So, you can country many different DC load such as a stand and a door lock and the boy and the son different contact just the DC load. So you can connect directly and this is the hardware and also we can see this is our PCB design that is oh boy oh you want we have designed this the new yesterday is a 1.4 worsen because we have designed this pod up takes about six months. So how many different pod have features and at last, we have success we're at 1.4 so you can see we can vote be business parties is ready and this is in front of the pod we can see this infarct and the spec found okay, we can change to 2d mode and this is the hardware and the hardware you can see here the pin define the ripping off the ESP 32 How to use and hot config you can from this house you can see the average ship and I squared C for extended the ship house and analog extend all have at here Okay, now, let's look at how to integrate to the home assistant you can see this is our home assistant Yamo file. So, this is the home assistant I have integrated you can see the output analog and analog input and a digital input and output the DS 18 B 20 temperature sensor. Now let's look at this AI robot in query to home assistant I have connect with this terminal and with this power supply at well what will connect to yet and this is a temperature sensor. So it can be 20 I have connected with Chip shell. So I can remove this socket and connect with a new socket I have connected with a wire so, eight channel output there are beings that for power supply because we designed for independently so you can let our eight channel hammer different What have the me so maybe this you have connected well what is your we have connected to different what so the different HTML will output a different watch of the power. So you can see here now I have just a tester with 12 words. So I have connected all this feeling together and this is a power supply and this is a temperature sensor okay and we have used the Ethernet cable to connect with it before our power I will open this metal box and I will let you see how it works.



Okay open this box because I have installed all the modules or you can see here this 4g module that I have installed the SIM card at here so that you can use GPS or use the short message for remote control your output or reader, the sensor state that it outputs and this is LD indicate for 32 channel output, this is 60 inch and output that is for the LD indicate have 60 inch and at the end kit for different value. And this is the RTC module that ts 32 and 31. So this if you if you have installed the battery, so you can save the clock of the system, because the water will send it by the airplane, so we can't install the battery on this socket. Okay. Now you can see this is hardware details, we will begin our arm. And you can see how the red LED is early, early. Okay, let's look at the home assistant, you can see here that is AI Oh, I have quit, actually you just click the ESP home. So you can very easily to integrate to home assistant by ESP home. And there are many many ESP 32 King County Bar. So I can see the RTD API. Oh, so you just added to this API URL and copy this configure Yama file from our company form. So just this file, this file, you can see the content. So you just copy and paste is very easy. And maybe you needed to change your static IP address, but I have used the if you have used DHCP that will be good this is that Skype address of my part. So you just install and connect to USB cable and that will Raspberry Pi or to your KC 868 server that our Raspberry PI Server just downloaded by USB cable is very easy. And after complete, you can see there are our dashboard I have quit 14 You can see that digital output so the tool chain output word this there were LD d sin LD D eight, just there are two there are two LDS we have designed for you to use for some different functions. So, you can see if I have to now also data channel you can see here and here. So, this is output one 232 channel. And you can see here that the seventh and the eighth, so you can see here I can't turn off. So, this is just our LD, so there are two LD for you to use. So, maybe you have some different functions you can define that will be easy to see the early indicate from outside the box. So, you can see here and this is Gs 18 B 20 temperatures. So, you can see here, just the temperature and the you can change this temperature sensor to your Hamilton sensor or other one where sensors just it's up to you. So, this helps set the update time. So, I have set about 16 seconds. So this will be long. And here you can see that is analog input, this is all analog input. We can see here, this PDF file you can see this is analog input from this one to eight and a nine to six because this is for zero to five what and this nine until 16 channel is for four to 20 minute and pair. So it's used for different sensors. So you can see if I have tested this analog one. So this analog one you can see not just the test, you can say this is three volts at here, I just a shot for this and you can see as the watch is changed, this is Channel one is the channel one. So you can connect with a different sensors to monitor the date because you need to configure the ESP home for the different range. So, this you can see a i One two A is 16 but you can see there are 1718 and 19 because this is really tripping out from this ESP 32 directory. And there's this 16 channel analog that helps extend the ship. So, this how all have connected the analog input one become As you can see, this is a chip, it's an I squared C chip will lead to analog input one and external to 16 channel. So, this chip the function makes the ESP 32 have so many different analog input Okay as you can see that is the analog input you can also set the updated 10 You can see here that is analog input different analog input just the Update Interval interview you can set this time this I have set to the five seconds you can see two other seconds.

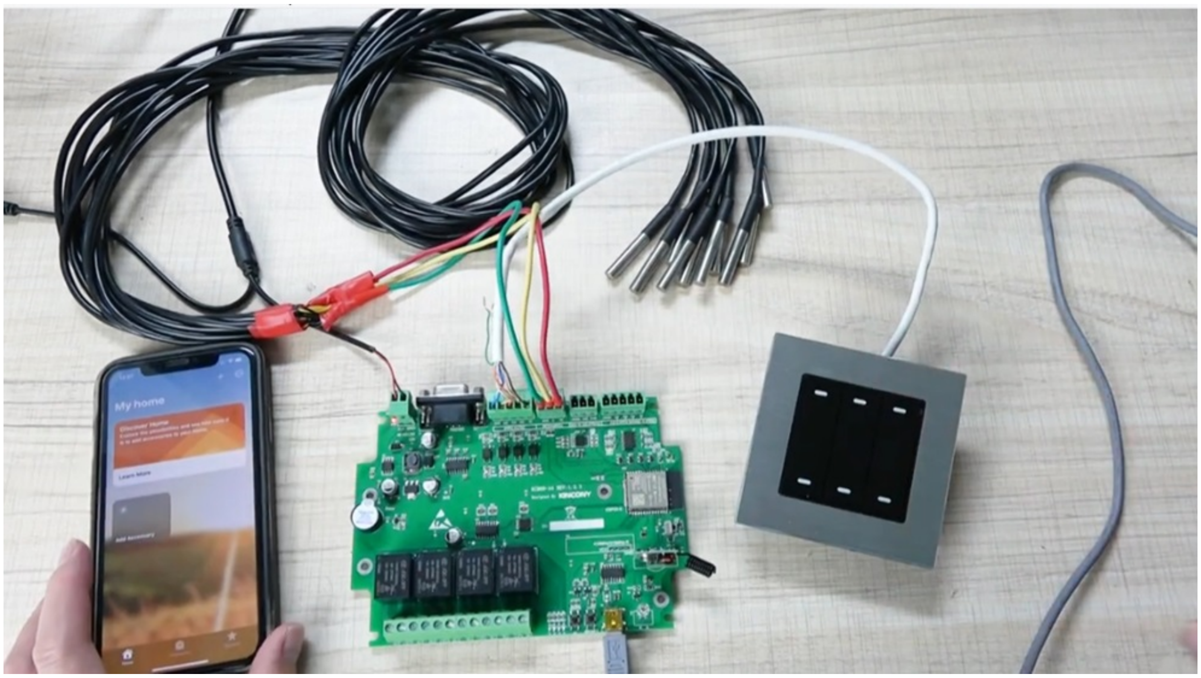


aio-light4		aio-AI-4	0.142 V	aio-input4	Off
aio-light5		aio-AI-5	0.142 V	aio-input5	Off
aio-light6		aio-AI-6	0.142 V	aio-input6	Off
aio-light7		aio-AI-7	0.142 V	aio-input7	Off
		aio-AI-8	0.142 V	aio-input8	Off
		aio-AI-9	0.142 V	aio-input9	Off
		aio-AI-10	0.142 V	aio-input10	Off
		aio-AI-11	0.142 V	aio-input11	Off
		aio-AI-12	0.142 V	aio-input12	Off
		aio-AI-13	0.142 V	aio-input13	Off
		aio-AI-14	0.142 V	aio-input14	Off

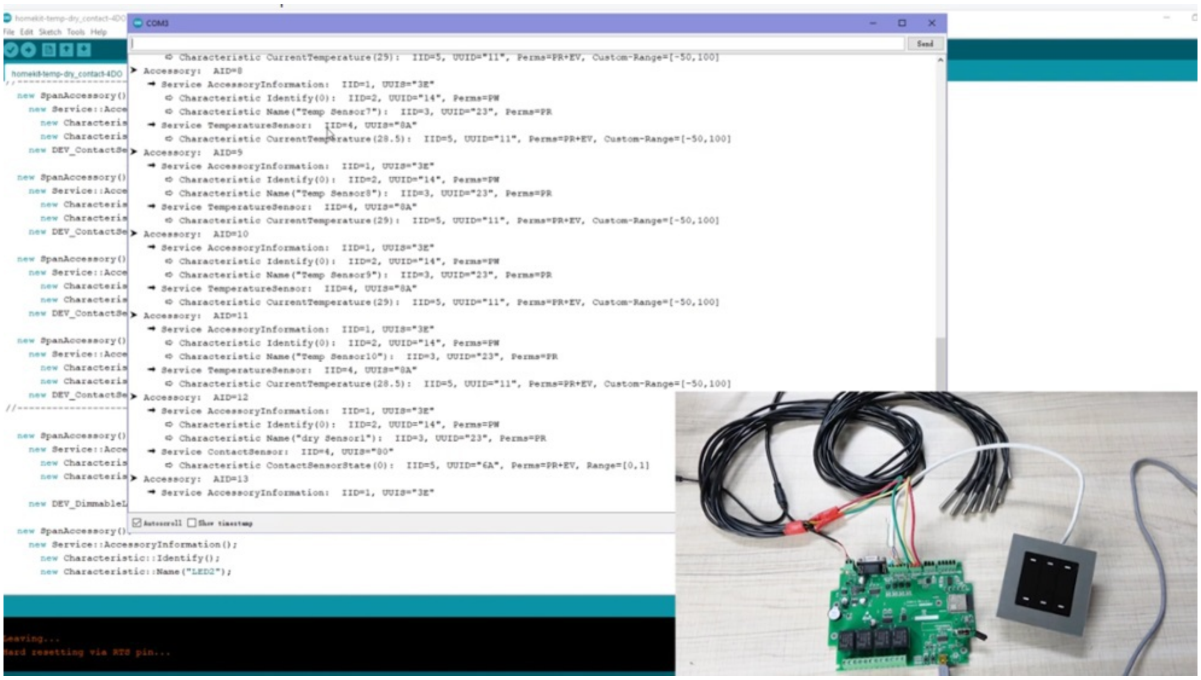
So, this is for analog input and here you can see that the analog outputs that are used for the Timur or you LD strip. So, you can see have 16 channel analog output if you can you can see here as early in the kit, you will see that is picking out is slowly and when I turn off you can see that is our let's look at clearly you can see here that is 16 channel analog output. So, the different voltage the LED will have different light. So, you can see I can have a 16 channel that will be out just as 30 and turn off that will be off. So, if you can see I can turn on this first dimmer or you can see I can change it 50% You can see early let's clearly you can see I can change to 100% that will be very light and reduce reduce and reduce you can see the LED will be change the brightness. So, our channel you can use the dimmer you can see this is number two under the motto and this is 100% And so, you can see here you can see here the PDF file that is the analog output this socket just this socket is analog output. So, if you turn on the channel one and actually not two to 100% So, this two terminal will output a DC 10 vote if you turn off this LD you can see if I can turn off and turn off. So, this terminal will output zero vote. So, this will be different practice will have the different wattage output. So this is just the analog output. And then let's look at this digital input. So, you can see here this digital input one and two digital 56 or but also have 57 and 5958 So, on to this is 56 but there are two buttons there are two buttons that 57 and the 58. So, you can see if I can shoot for this terminal as is the current and this is channel one. So, let's look at this home assistant you can see here I have shot for this this is chapter one, you can see that will be out that the trigger and the shot for this will grant that input the two is shot. So, you can see here we can see the last digital input s three and s four so you can press this button you can see this S one S three when I press this button that will be r and this S four also and AF apply this part. So you can define these two buttons just shut away for your to quit start automation and these two LEDs and these two switches can use by yourself in home assistant. Okay this we have designed this to pattern and 12 day for you to use. And this is the homeless system the dashboard. Okay, let's look at the form. You can see. Not only the home assistant is the park by this AI oh but also can use this ai o pod for Arduino IDE very easily. You can see here we have prepared a sound Arduino demo source code for different hardware resource such as you can use it for the digital input. So this hot related to the input leader the sensor states and how to control this output for r and F. So have this Arduino ID source code, you just download this code you can see this code you can just download and direct it for us that is the zip file have incurred i n o file, so you can easily use any resource in your add in ID.

ESP32 HOME AUTOMATION DIY WITH IOS16 APPLE HOMEKIT

I will show you this a for pod work with a new list HomeKit because I have updated the iPhone iOS with a 16 version and this time I will show you how to do this a for pod work with HomeKit. And this time we have used the fortunately and the fortunate digital input processor because I have used the word switch for test just use any dry contact sensor and connect with this four channel teacher input and this is temperature sensor but this time I have connected with a 10 temperature sensor. So this DS 18 B 20 sensor, but I have used only one to chow pot.



So I have connected with this just to just switch VCC and Ground and the signal. So let's look at how to download is adding a quarter for this HomeKit you can see here we have uploaded the source code to our form. So you can download this 21 source code you can see here you can just download this zip file and the way zip you will see this spouse This is firmware if you want to don't want to use the source code you can just download the firmware to the ESP 32 by this file. This time I have used Arduino you can see I can open this projector and it's the project file and this sound drivers for lamp and for sensors and this for dry contact sensor and this for the s 18 B 20 temperature sensor. And this is for just a relay and you can see here we have defined idle temperature sensor one sensor two sensors for a sensor for so you can see here has four sensors and until 10 sensor and this dry contact sensor so you can see this four channel dry contact sensor until four and this led one or the two and are this way additive for so this just fortunately so we can just download it before you download you just needed to install the part the part you need to install the ESP 32 So you can just input ESP 32 So that will list you must either to install version two I have installed worsen two point 0.4 That is the new list. So just to install it and I have installed and also you need to install the library that card home spare I can input home spent so you can see I have installed also you can install the new rest so you can close it and now you're just click the toy and the chose the part that ESP 32 This one NodeMCU 32 s so as you can see check the come out you can see it see the Comm port on your device manage in your Windows system. So you can see here the compound I have used comes with so I just choose this comport it comes through so you just click upload so that will compare and download is a free whack for ESP 32 So we just wait for a moment okay you can see complete now we can open this remote because I have configured this file before for this part.



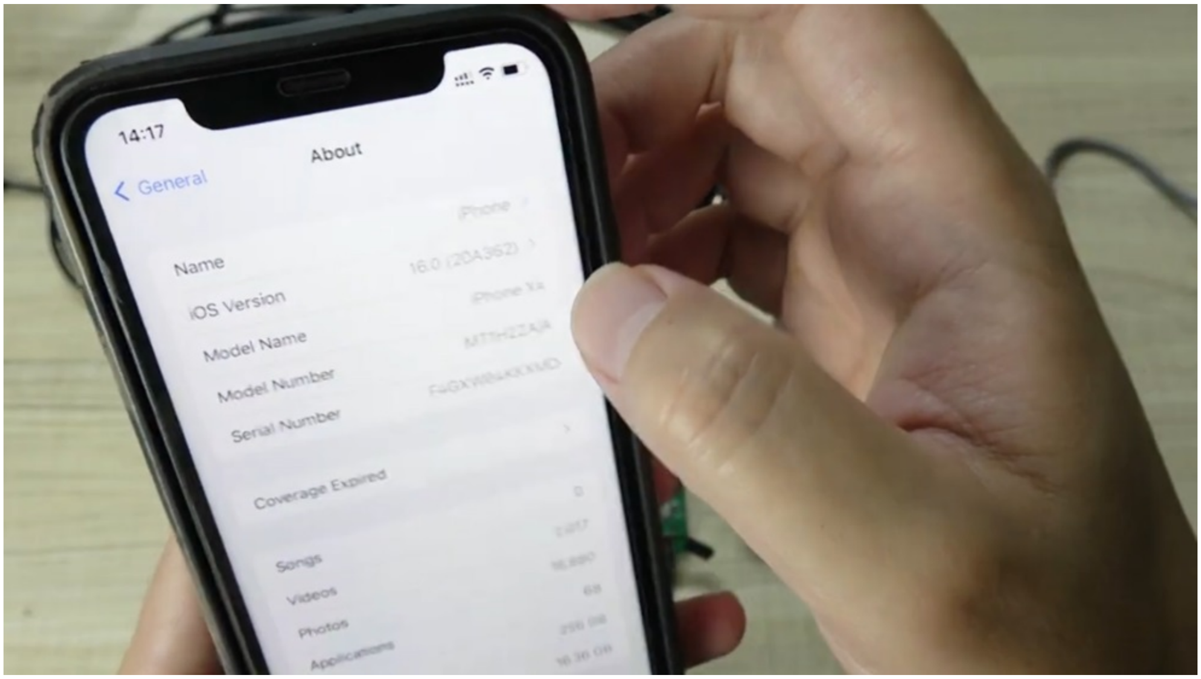
homekit-temp-dry_contact-4DO
File Edit Sketch Tools Help
COM3
Send
☑ Autoscroll ☐ Show timestamp

```
    ⇨ Characteristic CurrentTemperature(29):  IID=5, UUID="11", Perms=PR+EV, Custom-Range=[-50,100]
➤ Accessory:  AID=8
  ➡ Service AccessoryInformation:  IID=1, UUIS="3E"
    ⇨ Characteristic Identify(0):  IID=2, UUID="14", Perms=PW
    ⇨ Characteristic Name("Temp Sensor7"):  IID=3, UUID="23", Perms=PR
  ➡ Service TemperatureSensor:  IID=4, UUIS="8A"
    ⇨ Characteristic CurrentTemperature(28.5):  IID=5, UUID="11", Perms=PR+EV, Custom-Range=[-50,100]
➤ Accessory:  AID=9
  ➡ Service AccessoryInformation:  IID=1, UUIS="3E"
    ⇨ Characteristic Identify(0):  IID=2, UUID="14", Perms=PW
    ⇨ Characteristic Name("Temp Sensor8"):  IID=3, UUID="23", Perms=PR
  ➡ Service TemperatureSensor:  IID=4, UUIS="8A"
    ⇨ Characteristic CurrentTemperature(29):  IID=5, UUID="11", Perms=PR+EV, Custom-Range=[-50,100]
➤ Accessory:  AID=10
  ➡ Service AccessoryInformation:  IID=1, UUIS="3E"
    ⇨ Characteristic Identify(0):  IID=2, UUID="14", Perms=PW
    ⇨ Characteristic Name("Temp Sensor9"):  IID=3, UUID="23", Perms=PR
  ➡ Service TemperatureSensor:  IID=4, UUIS="8A"
    ⇨ Characteristic CurrentTemperature(29):  IID=5, UUID="11", Perms=PR+EV, Custom-Range=[-50,100]
➤ Accessory:  AID=11
  ➡ Service AccessoryInformation:  IID=1, UUIS="3E"
    ⇨ Characteristic Identify(0):  IID=2, UUID="14", Perms=PW
    ⇨ Characteristic Name("Temp Sensor10"):  IID=3, UUID="23", Perms=PR
  ➡ Service TemperatureSensor:  IID=4, UUIS="8A"
    ⇨ Characteristic CurrentTemperature(28.5):  IID=5, UUID="11", Perms=PR+EV, Custom-Range=[-50,100]
➤ Accessory:  AID=12
  ➡ Service AccessoryInformation:  IID=1, UUIS="3E"
    ⇨ Characteristic Identify(0):  IID=2, UUID="14", Perms=PW
    ⇨ Characteristic Name("dry Sensor1"):  IID=3, UUID="23", Perms=PR
  ➡ Service ContactSensor:  IID=4, UUIS="80"
    ⇨ Characteristic ContactSensorState(0):  IID=5, UUID="6A", Perms=PR+EV, Range=[0,1]
➤ Accessory:  AID=13
  ➡ Service AccessoryInformation:  IID=1, UUIS="3E"
```

```
  new SpanAccessory();
    new Service::AccessoryInformation();
      new Characteristic::Identify();
      new Characteristic::Name("LED2");
```

```
Leaving...
Hard resetting via RTS pin...
```

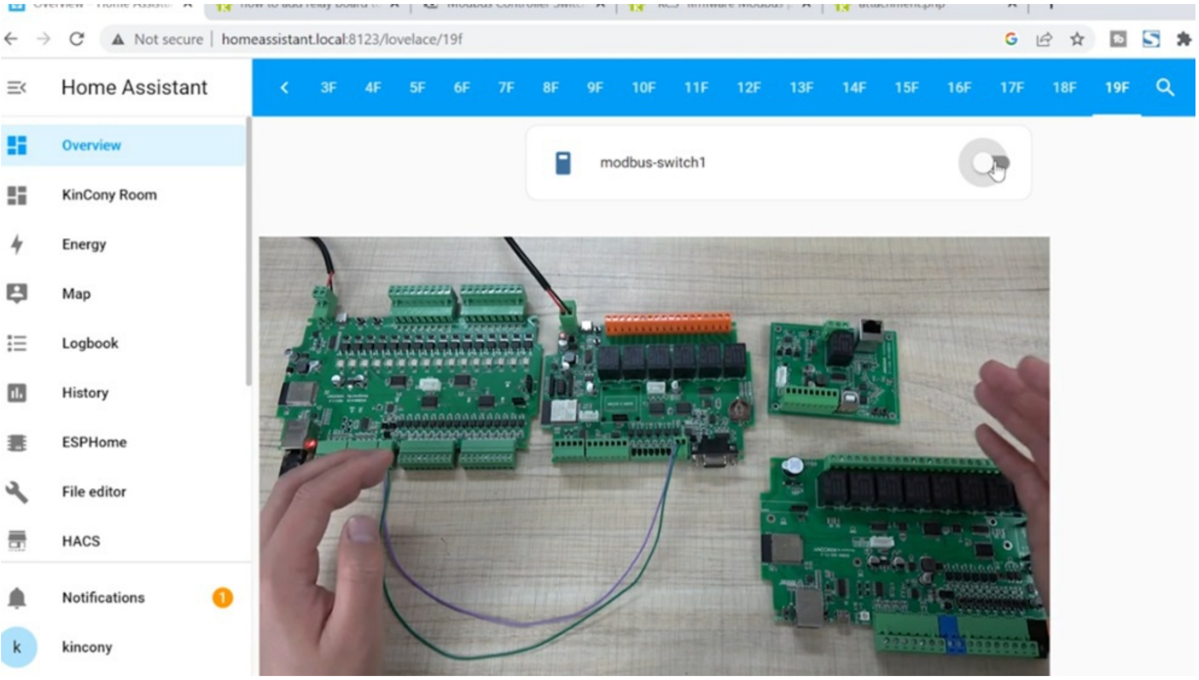
So the first time I needed to input the letter E so that we read and clear audit so I can click send. So you can see here audit it will be starting and this is configured temperature sensor and now you close man is ready because I wait I have imported the E because sometimes you want to change the Wi Fi and change another place you can input the E to erase all information. So now you can input W to configure Wi Fi SSID and the password. So we can input W and press Send and this time postman will scan the network and you can see this illustrated this Wi Fi signal. You can see my mobile phone I have connected to the Wi Fi by the Khinkali SSID. So I also let this a footpod are connected with it. King Kong the SSID so you need to let your pod and your iPhone with one thing load. So I, this is number one, so I can input one and click send. And this time you can see please input the password so you just input the Wi Fi password of your Lord and you can click send and now we're restarting the Infopark you can see we'll restart and here you can see waiting for with connect to King County and this is connected so this is IP address of your a4 bog. Now you just not have prepared you need to use the HomeKit app prepares a four bar. So we can see here we have a code this is the Prepare code. So you can see I can use my mobile phone I have used iOS 16 You can see here I have updated my iPhone for iOS 16 and I will use this HomeKit This is my home kit and click Add accelerator and here as you can see more option and you can see this fan is a home span and click and add any way and this input to your code.



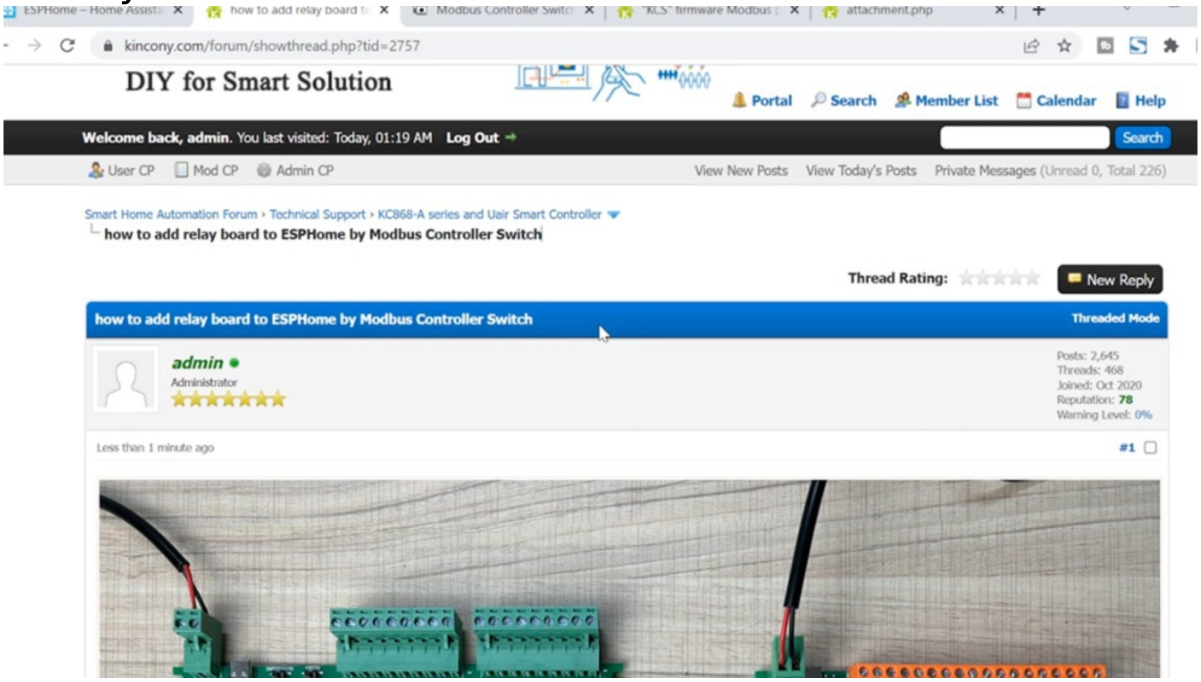
This code is this one you can see here 46637726 This a deferred code off the homestead 46637726 and the press Continue. And here you can see what take a few seconds okay you can see the device is detected I can set this in my bedroom because he is on Chinese words I have used the English words bedroom and press Continue and press Continue and press Continue and this the dry contact sensor this is sensor one can press Continue and also choose to bedroom continue and this is sensor number if you want to rename you can input this to save the time I'm not sure the name yet and his direct contact will list as a contact sensor door sensor character sensor window sensor is up to you I can just click the Contact sensor and continue and this assessor to dry contact sensor to also choose to the bedroom and sensor two I can choose this door door sensor just just let's see that you'll see the different icon and continue and number three sensor bedroom and the continue and carriage continue and for continue and bedroom and continue and I can choose the window says continue okay for dry context since I have added to my home and press Continue and this is the LD set for this for channel relay I can click continue and the bedroom continue and LD one continue and as they say number two and also it shows to the bedroom and continue and I will just weigh and the bedroom and continue and early for and the bedroom and continue. So this afford right have added to my home and consider this a temperature sensor one one of the temperature sensor continue and the bedroom sensor and just next next continue continue for our sensor because I have connected with a 10 sensor. So I just chose to do and the continuity was through and continue choose through and continue continue 789 10 The last one this is the last one. Okay, the 1010 percents I have added to my home down okay you can see the screen is better Good for because this have used our 16 is very beautiful for the HomeKit you can see here this is the iOS 16 kit and you can see here I can click oh and off this and number two oh and off and this way off and the four or you can also click this icon for on and off off off and this is a sensor this is a temperature sensor this is a 10 temperature sensor you can see here I have just did this a turn temperature sensor just this one if I have Hold on I have no doubt in my head and you can see some value have upped or you can see 31 So there too so, the temperature is changed and this is temperature sensor okay and this is a security sensor this is a bedroom and you can see here says a closed sensor does encourage it garage door and this door and it is the window. So, this four channels of digital input sensor track contact sensor, you can see here I can hold down this one you can see this is open this is open to help see it I can release my finger is closed. So I press this button this is open and the release it that closed and this is number two, you can see I can miss a door is open that how open and closed, open and closed and this is number three, you can see here the character is open and that raise my finger that is close and open and closed and it is number four that's a window this a window is open. I can leave that closed, open and closed. So this is the sensor you can see here you can redeem for the strike context sensor. So this is iOS 16 that it can be displayed as a relay and the sensor and Dragonair sensor and a temperature sensor at window. Okay, this is how to use the a4 board.

RS485 RELAY BOARD TO ESPHOME BY MODBUS CONTROLLER SWITCH

I will show you how to add a Modbus relay board to home assistant and by ESP home and you can see this our a 16 part this is the Asics part but this time I have used this is void fi cable connected directly because you can integrate this really bored to home assistant by this Wi Fi or by network this bar no network but if you want stable don't want integrate by Wi Fi. So that you can use this is for it fi cable and connect with the master part and this is live. So this video I will show you our way so you can integrate any size for it fire relay board to the ESP home not only can cornice relay, but just any is void fight mode passport, you can integrate to the ESP home just so you can see here this is the home assistant, I have created this switch this switch is mode by switch so you can say okay and turn off you can see this relay is on and off, but I have not used the Wi Fi I just use this cable. So, if you have the first part me maybe this not have the enough chip child for your use, you can extend by this extender relay module, this is second part and maybe this is the third part and maybe this is first part.



So you can use this cable the first one connect to the second one and the second one connect to the server one and the server connect to the first one. So you can extend the many eyes for it five baud not only really module and also you can connect with the energy meter and Assange as fortified sensor can connect with this cable directly. So this cable will be very long distance when solid meter is no problem because its eyes for it why not you use for the industry widely used. Okay now in this video I will show you how to set up in the home assistant by ESP home. So first you can see here that you need to confer the Tiguan. So, this is a 16 part this a 16 part and this is a six part so this two bar we need the communication with us for it fine. So this have a and b just this two cable, a and b. So this a you can connect with this a a six part A and this B connect with this B so the first hardware you need to just connect this to cable directly. So that will be very easy. Okay, let's look at the second the software and you need to configure your board have the RS 485 Modbus protocol. So this is a sixth part you can find the Modbus protocol document in our form. So that if you have your own relay module, you need to know this protocol and click this code code just you will find this information and we just needed to show you how to control this relay. So you can see set off for every channel of digital output just to this relay output. So this is a command that is sent for Relay one. And so you can see that will be very easy just to command this is sent by the only one and this will turn off really well. So just as this to come out. And let's look at the home assistant. I will show you later how to this details feel to the ESP home. This is our switch now I will show you this ESP home.



kincony.com/forum/showthread.php?tid=2757

DIY for Smart Solution

Welcome back, admin. You last visited: Today, 01:19 AM Log Out

Private Messages (Unread 0, Total 226)

Smart Home Automation Forum › Technical Support › KC868-A series and Uair Smart Controller
└ how to add relay board to ESPHome by Modbus Controller Switch

Thread Rating: ☆☆☆☆☆

how to add relay board to ESPHome by Modbus Controller Switch

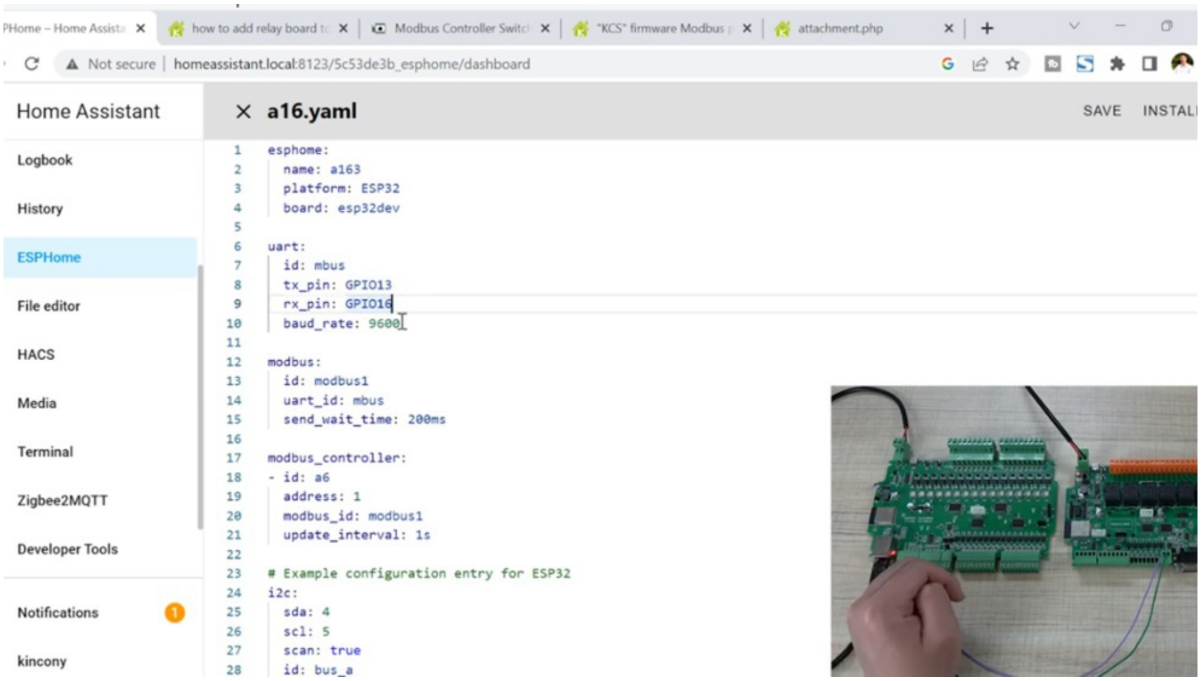
admin ●
Administrator
★★★★★★★

Posts: 2,645
Threads: 468
Joined: Oct 2020
Reputation: 78
Warning Level: 0%

Less than 1 minute ago

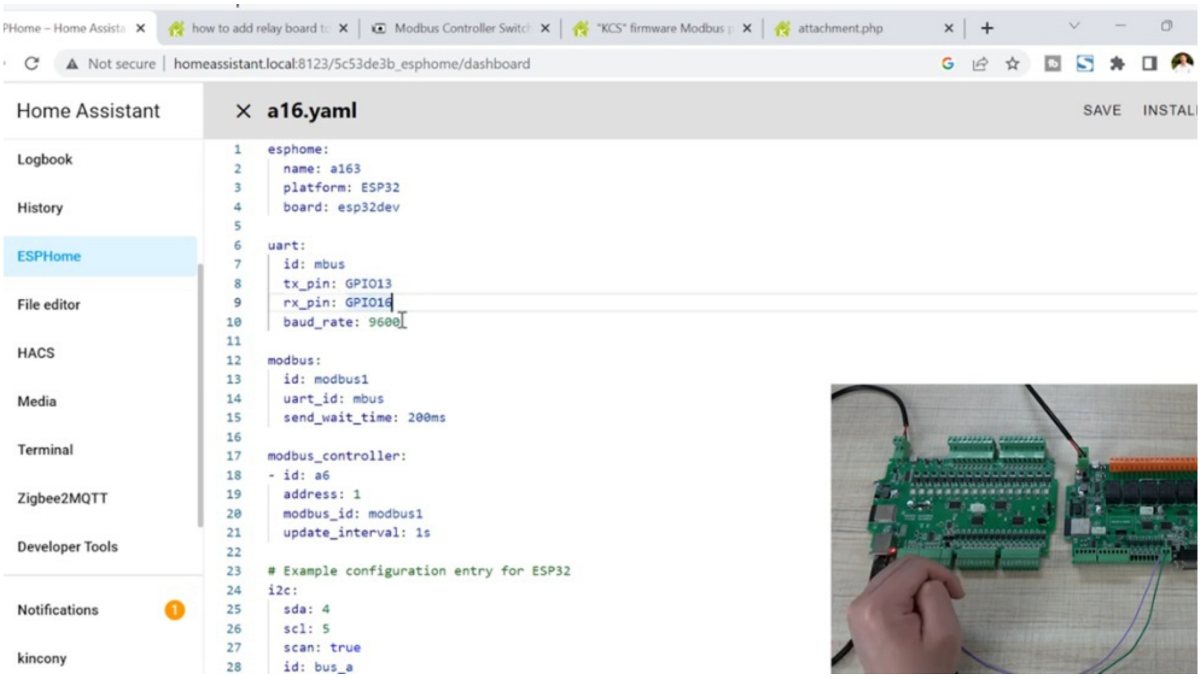
#1 □

And you can see here we can edit this and adjust Edison commands. I have added these commands in our form so you can see here this topic I have upload how to add really powerful ESP home by motor Bus Controller switch so you can see this is a photo and this horsington dashboard and just only this command so that will be very easy to integrate by the Modbus. And how this command works. I will show you and let's look at this one. The first you need to define your you ar t that is server port, the serial port used by the ESP 32. So this part we have used this GPIO setting and shower 16.



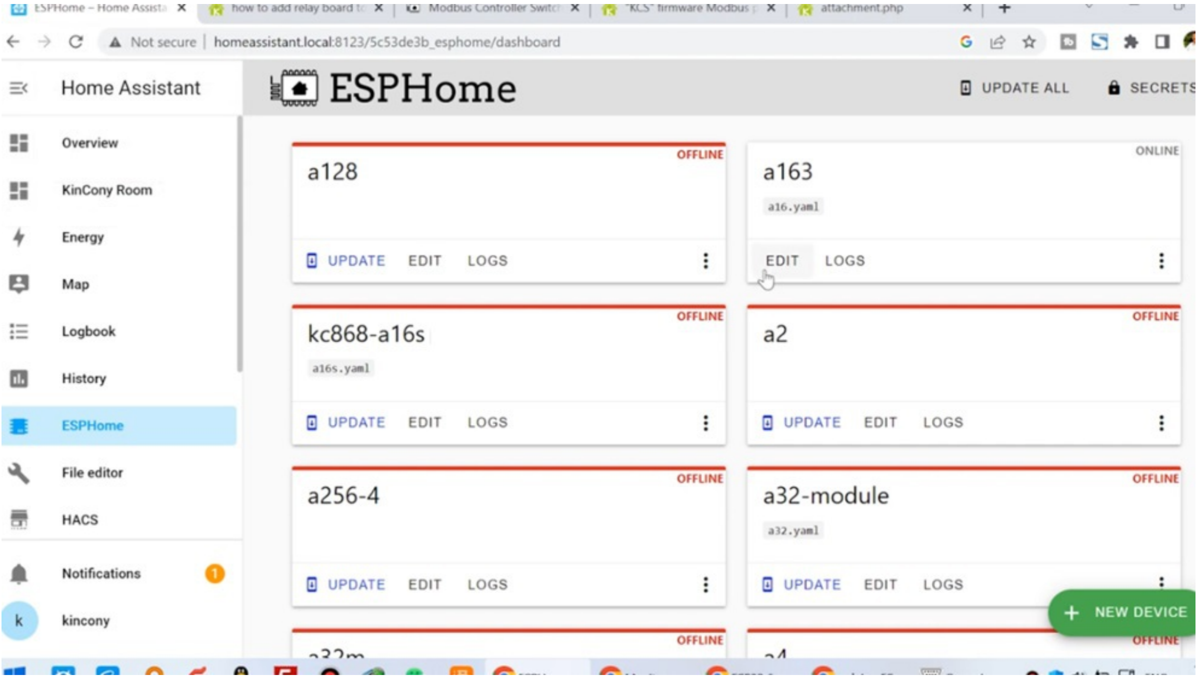
```
esphome:
  name: a163
  platform: ESP32
  board: esp32dev

uart:
  id: mbus
  tx_pin: GPIO13
  rx_pin: GPIO16
  baud_rate: 9600

modbus:
  id: modbus1
  uart_id: mbus
  send_wait_time: 200ms

modbus_controller:
- id: a6
  address: 1
  modbus_id: modbus1
  update_interval: 1s

# Example configuration entry for ESP32
i2c:
  sda: 4
  scl: 5
  scan: true
  id: bus_a
```

And this is the benefit that is this part, the ESP home must support this and must and this is live and this must For So, that is brand rich, because we can set this brand renovate it clear that is a 60 KCS firmware. So, you can enable this is void fi you can see there are many many different protocol you can use, and at this time I have used this is for it fine. So, I can enable this one and in this port code I can choose this RTU as this for our eyes for it by SMA or chopped apart at this time we can choose this one and this is a burn rate. So, I have set this nine six zeros yellow. So, I should set this ESP home also is nice 600 And this is for the UI ui t and in the second we can create the mode pass the mode pass that I can quit the mode bus one this this name just really meant by yourself and this serial port Id just passed just this one, I just feel this that here and pass just this one and this command L command will stand and wait 200 million seconds, this is up to you and not very important. And this quit the mood pest controller because I have this a six. So I can create this for a six if you have another part this is a n so you can create another charts the copy and paste and create another for this one, if you have many Modbus device, so you can create many Modbus container. So, this one, I have only tested with ASICs so I can use this ID is six and this address is a six eyes 485 address. So, you can see here you can see here, this address is one because the local address in the one I have set this one if you have set this part of the second body to or the third part is three. So you can just use a single address this is two and I set this to this is three I set this is three, so just as soon as your controller address and this is Modbus ID as this is set by yourself, I have set the Modbus one and this update interview that is update interview that is very important that I have said every one seconds because that is update state. So you can see here is a home assistant, you can see I can use this web page to turn turn off as this is no problem.



```
esphome:
  name: a163
  platform: ESP32
  board: esp32dev

uart:
  id: mbus
  tx_pin: GPIO13
  rx_pin: GPIO16
  baud_rate: 9600

modbus:
  id: modbus1
  uart_id: mbus
  send_wait_time: 200ms

modbus_controller:
- id: a6
  address: 1
  modbus_id: modbus1
  update_interval: 1s

# Example configuration entry for ESP32
i2c:
  sda: 4
  scl: 5
  scan: true
  id: bus_a
```

But if you have used this, this way, you can see I can use this monitor and the country by the webpage or maybe you can country it by your mobile phone or by your water switch so you can see click on and click off this not by home assistant just by this web service. So you can see I can click on and this will be updated the state is down and I can click this one that is off. So if I can click at this one to turn now, and you can see here the monitor also is turned on so I can close it. So that means every way seconds will update the state found a six part so this is very important for update your state. So we can go out and this interview this cleanse when seconds you can repress by yourself and the next is last step just create a switch as is a Modbus switch and this is a 16 part switch. So I have created this switch for switch one for Modbus the platform is used by the Modbus controller and as the name is renamed by yourself is Modbus switch one and this address is a six address relay address. So the only one that is zero, so no need to that is one so I quit the switch one so I can use address one. If you quit switch to I can use this address is a one and this register type that coin because our user ruling this is the coin of the mod pass command just to coin this a bit musk. Just feel that when that means that you Because this old details have in ESP home mode pest control switch, which I have listed here, you can see many details at here. And now, you just, you can save and install and download to your USB cable because I have not connected with a USB cable, you just download it to your ESP 32 Then we'll begin work. So, if you want to create the switch to so you just copy and paste that here yes, you can just change to switch to and this addresses change to one. So if you have the three, switch three you're just pissed and this is switch Sully and this address is a tool. So until you have six really so you can quit the mobile switcher six and this address will be fine. So, this video I just show you the example the demo how to create one button that will be very easy okay, then after you download, so that you can just use this way and you will find that this mobile switch you can turn on and off and another way So, you can see this is Modbus This is easiest example I have put it as this way, but if you put the code is not as this when that command is this way and off is this way, this is deferred for the ESP home. So that you can use this command very easily. Because our protocol you can see here the is f f 00 Our is m f f 00 and is off and the off is the unknown 000 So 000 So we have designed this part, according to the ESP home. So we'll create this very easily, but don't worry if you report is not as this mount. So, you can send it as a customer command by this way, this is customer Size command by this way.



So, if you want creating your own command, so you can see here I can replace with this code. So, you can see I can just like this delete this one and copy and paste paste it here. So, this is another way so you can see our command that is for owl and off this is a command. So if you want to turn an asset by send it by your own command. So you just copy and paste this command this is there no one and this is yellow file and 00 and this also changed to 00 and this is FF FF and this is the 0000 and it was the last one that is CRC code. This is CRC code. So you not needed to paste that here. Just enter this command. So if you have downloaded this command by the ESP home, this also can work fine. So this just sent all customer Size command by yourself and for any relay pod so you can use this way.