

```
#define BLYNK_TEMPLATE_ID      "TMPL32u7FuTK"  
#define BLYNK_TEMPLATE_NAME    "Quickstart Device"  
#define BLYNK_AUTH_TOKEN  
"VQVdGFEkxzUB1GWN5tym_9pHiiUMgFdI"
```

```
#define BLYNK_PRINT Serial  
#include <Arduino.h>  
#include <analogWrite.h>  
#include <WiFi.h>  
#include <SPI.h>  
#include <WiFiClient.h>  
#include <BlynkSimpleEsp32.h>  
#include <Adafruit_GFX.h>  
#include <Adafruit_SSD1306.h>  
#include <ESP32Servo.h>
```

```
char auth[] = BLYNK_AUTH_TOKEN;  
// Your WiFi credentials.  
// Set password to "" for open networks.  
char ssid[] = "poco m3 pro 5g";  
char pass [] = "12345789";
```

```
#define IR1 5 // IR sensor 1  
#define IR2 18 // IR sensor 2  
#define IR3 19 // IR sensor 3  
#define IR4 23 // IR sensor 4  
#define SCREEN_WIDTH 128 // OLED display width, in pixels  
#define SCREEN_HEIGHT 64 // OLED display height, in pixels  
#define OLED_RESET -1
```

```
#define SCREEN_ADDRESS 0x3C
```

```
//end.....  
...
```

```
Adafruit_SSD1306 display('SCREEN_WIDTH, SCREEN_HEIGHT,  
&Wire, OLED_RESET');  
int pos;  
int med1 = 0;  
int med2 = 0;  
int med3 = 0;  
int med4 = 0;  
boolean counter1 = false;  
boolean counter2 = false;  
boolean counter3 = false;  
boolean counter4 = false;  
int x = 0;  
int y = 16;
```

```
// Servo
```

```
const int servoCount = 4;  
static const int servosPins[servoCount] = {12, 14, 27, 26};  
Servo servos[servoCount] ;  
//-----
```

```
void countmed1 (){  
if (counter1 == true) {  
    counter1 = false;  
    med1 = med1 + 1;  
    Serial.print("Number og medl = ") ;  
    Serial.println (med1/2) ;  
    Blynk.virtualWrite(V0,med1/2) ;  
    if (med1 >= 30) {  
        Blynk.logEvent("containerl_full") ;  
    }  
}
```

```
}  
void countmed2 () {  
if (counter2 == true) {
```

```

    counter2 = false;
    med2 = med2 + 1;
    Serial. print ("Number og med2 = ") ;
    Serial. println (med2/2) ;
    Blynk . virtualWrite (V1, med2/2) ;
    if (med2 >= 30) {
        Blynk.logEvent ("container2_full") ;
    }
}

}
//end.....
...

```

```

void countmed3(){
if (counter3 == true) {
    counter3 = false;
    med3 = med3 + 1;
    Serial. print ("Number og med3 = ");
    Serial. println (med3/2) ;
    Blynk . virtualWrite (V2, med3/2);
    if (med3 >= 30) {
        Blynk.logEvent ("container3_full");
    }
}
}

}
void countmed4() {
    if (counter4 == true) {
        counter4 = false;
        med4 = med4 + 1;
        Serial. print ("Number og med4 = ") ;
        Serial. println (med4/2) ;
        Blynk . virtualWrite (V3, med4/2) ;
        if (med4 >= 30) {
            Blynk.logEvent ("container4_full") ;
        }
    }
}

```

```
//end.....  
}  
}  
  
}
```

```

void docounter1 () {
  counter1 = true;
}
void docounter2 () {
  counter2 = true;
}
void docounter3 () {
  counter3 = true;
}
void docounter4 () {
  counter4 = true;
}
void myTimerEvent()
{
  // You can send any value at any time.
  // Please don't send more that 10 values per second.
  Blynk.virtualWrite (V4, millis() / 1000);
}

```

```

void oledAlert () {

```

```

//end.....

```

```

.....
if (med1 <= 4 && med2 <= 4 && med3 <= 4 && med4<=4 ) {
  display. clearDisplay () ;
  display . setTextSize (2) ;
  display . setTextColor (SSD1306_WHITE) ;
  display . setCursor (1, 4) ;
  display . println (F ("LOW DRUGS") ) ;
  display. setCursor (1, 30) ;
  display . setTextSize (1) ;
  display . println (F ("Please fill") ) ;
  display . setCursor (1, 37) ;
  display. println (F("all container")) ;
  display. display () ;
}
else if (med1 >= 30 && med2 >= 30 && med3 >= 30 && med4 >= 30)
{

```

```
display . clearDisplay ();  
display . setTextSize (2) ;  
display . setTextColor (SSD1306_WHITE) ;  
display . setCursor (1, 4) ;  
display. println (F ("FULL DRUGS") ) ;  
display . setCursor (1, 30) ;  
display . setTextSize (1) ;
```

```
//end.....
```

```
.....  
    display . println (F ("Have a") ) ;  
    display . setCursor (1, 37) ;  
    display . println (F ("nice day") ) ;  
    display . display () ;  
}  
}
```

```
//end.....
```

```
.....  
BlynkTimer timer;
```

```
BLYNK_WRITE (V4) {  
    int value = param. asInt ( ) ;  
    if (value = true) {  
        med1 = 0;  
        med2 = 0;  
        med3 = 0;  
        med4 = 0;  
        Blynk . virtualWrite (V0, med1/2) ;  
        Blynk . virtualWrite (V1, med2/2) ;  
        Blynk . virtualWrite (V2, med3/2) ;  
        Blynk . virtualWrite (V3, med4/2) ;  
        display . clearDisplay () ;  
        display . setTextSize (2) ;  
        display . setTextColor (SSD1306_WHITE) ;  
        display . setCursor (20, 2) ;  
        display . println (F ("RESETTING") ) ;  
        display . display () ;  
        delay (3000) ;  
    }  
}
```

```
//end.....
```

```
.....
```

```
BLYNK_WRITE (V5) {  
  int value = param. asInt () ;  
  int pos = 0 ;  
  Serial. println ("Now, Container 1 is working") ;  
  Serial. println (value) ;  
  if (value >= 1) {
```

```

med1 = med1 - 2;
Serial. print ("Number og medi = ") ;
Serial. println (med1/2) ;
Blynk . virtualWrite (V0, med1/2) ;
for (pos = 0; pos <= 61; pos += 1) {
  servos [0] . write (pos) ;
  delay (15) ;
}
for (pos = 61; pos >= 0; pos -= 1) {
  servos [0].write (pos) ;
}
if (med1 <= 0 && med2 <= 0 && med3 <= 0 && med4<=0) {
  Blynk.logEvent("out_off medicine", "Please fill up all containers !
!!!");
  display.clearDisplay();
  display.setTextSize(2);
  display.setTextColor(SSD1306_WHITE);
  display.setCursor(1,4);
  display.println (F("OUT OFF"));
  display.setCursor (1,20);
  display.println (F ("Medicine"));

```

```

//end.....
.....

```

```

}
else if (med1 <= 4) {
  Blynk.logEvent("container_1_low","Please fill medicine in the
container 1" ) ;
}
}
}
BLYNK_WRITE (V6) {
  int value = param. asInt () ;

```

```
int pos = 0 ;
Serial.println ("Now, Container 2 is working") ;
if (value >= 1) {
  med2 = med2 - 2;
  Serial.print ("Number og med2 = ") ;
  Serial . println (med2/2) ;
//end.....
.....
```

```

    delay (15) ;
}
for (pos = 61; pos >= 0; pos -= 1) {
    servos [0].write(pos) ;
}
if (med1 <= 0 && med2 <= 0 && med3 <= 0 && med4<=0) {
    Blynk.logEvent ("out_off medicine", "Please fill up all containers !!
!!");
    display.clearDisplay ();
    display.setTextSize (2) ;
    display.setTextColor(SSD1306_WHITE);
    display.setCursor (1,4);
    display.println (F("OUT OFF"));
    display.setCursor (1,20) ;
    display.println (F ("Medicine"));
}
else if (med1 <= 4)
{
    Blynk.logEvent("container_1_low","Please fill medicine in the
container 1" );
}
}

```

```

BLYNK_WRITE (V7) {
    int value = param.asInt();
    int pos = 0 ;
    Serial.println("Now, Container 2 is working") ;
    if (value >= 1)
    {
        med2 = med2 - 2;
        Serial.print ("Number og med2 = ") ;
        Serial.println (med2/2) ;
        Blynk.virtualWrite(V1,med2/2);
    }
}

```

```

//end.....
.....

```

```
}  
void setup(){  
  Serial.begin(115200);  
  Blynk.begin(auth,ssid,pass) ;  
  pinMode(IR1, INPUT);  
  pinMode(IR2, INPUT);  
  pinMode(IR3, INPUT);
```

```

pinMode(IR4, INPUT);
attachInterrupt(digitalPinToInterrupt(IR1) , docounter1, RISING) ;
attachInterrupt(digitalPinToInterrupt(IR2) , docounter2, RISING) ;
attachInterrupt(digitalPinToInterrupt(IR3), docounter3, RISING) ;
attachInterrupt(digitalPinToInterrupt(IR4) , docounter4, RISING) ;
timer.setInterval(1000L, myTimerEvent);
if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3c)) {
  Serial.println(F("SSD1306 allocation failed") ) ;
  for (;;) ; // Don't proceed, loop forever
}
display . display () ;
delay (2000) ;
// Clear the buffer
display.clearDisplay();
display.setTextSize(2); // Normal 1:1 pixel scale
display.setTextColor(SSD1306_WHITE); // Draw white text
display.setCursor (1,4); // Start at top-left corner
display.println (F("STARTING.."));
display.setTextSize(1.5);
display.setCursor(2,25); // Start at top-left corner

//end.....
.....

display.println(F("PLEASE SETUP"));
display.setCursor(2,40);
display.println(F("ON BLYNK"));
display.drawLine(0, 0, 0, 60, SSD1306_WHITE);
display.drawLine(0, 60, 127, 60, SSD1306_WHITE);
display.drawLine(127, 60, 127, 0, SSD1306_WHITE);
display.drawLine(127, 0, 0, 0, SSD1306_WHITE);
display.display();
delay(5000);
for(int i = 0; i < servoCount; ++i) {
  servos [i] .attach (servosPins[i]);
}
}
void loop() {
Blynk.run () ;
timer.run () ;
countmed1 () ;

```

```
countmed2 ();  
countmed3 ();  
countmed4 ();  
oledAlear () ;  
}
```

//end