變數不能大寫、有單引號(')、換行(\n)要設為脫逸字元、~~字串變數不用留雙引號~~

★字首單引號前加空格,避免被excel去掉

|  |  |
| --- | --- |
| \_01imi\_unogoods\_init  iMi物流模組UNO初始化  CNC腳位  UART通訊  撿貨伺服馬達  \_02imi\_unogoods\_cnc  致能 x方向 x步進 y方向 y步進  0  0  0  0  0  '// 定義電機控制用常數'  'const int enablePin = '+value\_en+'; // 啟用控制引腳'  'const int xdirPin = '+value\_xdir+'; // x方向控制引腳'  'const int xstepPin = '+value\_xstep+'; // x步進控制引腳'  'const int ydirPin = '+value\_ydir+'; // y方向控制引腳'  'const int ystepPin = '+value\_ystep+'; // y步進控制引腳'  \_03imi\_unogoods\_uart  RX TX  0  0  '// UART通訊'  'SoftwareSerial ESP32Serial('+value\_rx+',' +value\_tx+');'  \_04imi\_unogoods\_servopin  腳位  0  '// 伺服馬達腳位(接在Z+是pin11)'  'const int servoPin = '+value\_servo+';' | #include <AccelStepper.h>  #include <Servo.h>  #include <SoftwareSerial.h>  //====插入變數Start====  Statements區塊  //====插入變數End====  // 伺服馬達  Servo servoPick;  const int xMoveSteps = 500; // 測試電機1運行使用的運行步數  const int yMoveSteps = 500; // 測試電機2運行使用的運行步數  AccelStepper stepper1(1, xstepPin, xdirPin); // 建立步進電機物件1  AccelStepper stepper2(1, ystepPin, ydirPin); // 建立步進電機物件2  const char\* GOODS\_LOAD = "goodsLoad";  void setup() {  Serial.begin(9600);  // UART  ESP32Serial.begin(9600);  // 伺服馬達  servoPick.attach(servoPin);  servoPick.write(0);  pinMode(xstepPin, OUTPUT); // Arduino控制A4988x步進引腳為輸出模式  pinMode(xdirPin, OUTPUT); // Arduino控制A4988x方向引腳為輸出模式  pinMode(ystepPin, OUTPUT); // Arduino控制A4988y步進引腳為輸出模式  pinMode(ydirPin, OUTPUT); // Arduino控制A4988y方向引腳為輸出模式  pinMode(enablePin, OUTPUT); // Arduino控制A4988啟用引腳為輸出模式  digitalWrite(enablePin, LOW); // 將啟用控制引腳設置為低電平以使電機驅動板進入工作狀態  stepper1.setMaxSpeed(300.0); // 設置電機最大速度300  stepper1.setAcceleration(20.0); // 設置電機加速度20.0  stepper2.setMaxSpeed(300.0); // 設置電機最大速度300  stepper2.setAcceleration(20.0); // 設置電機加速度20.0    //歸零  resetStepper();  }  void findGoods(int x,int y){  //分成3\*3格倉庫的xy步數  int xSteps=xMoveSteps/2;  int ySteps=yMoveSteps/2;  //電機移動至該貨物格子  stepper1.moveTo(xSteps\*x);  stepper2.moveTo(ySteps\*y);  }  void resetStepper(){  stepper1.moveTo(0);  stepper2.moveTo(0);  }  void testStepper(){  // 控制步進電機1往復運動  if ( stepper1.currentPosition() == 0 ) {  stepper1.moveTo(xMoveSteps);  servoPick.write(0);// 伺服馬達轉0度  } else if ( stepper1.currentPosition() == xMoveSteps ) {  stepper1.moveTo(0);  servoPick.write(90);// 伺服馬達轉90度  }  // 控制步進電機2往復運動  if ( stepper2.currentPosition() == 0 ) {  stepper2.moveTo(yMoveSteps);  } else if ( stepper2.currentPosition() == yMoveSteps ) {  stepper2.moveTo(0);  }  stepper1.run(); // 1號電機運行  stepper2.run(); // 2號電機運行  } |
| \_05imi\_unogoods\_esp32  當接收ESP32訊息時  \_06imi\_unogoods\_xy  取得貨物在倉庫xy座標  'int x = (int)(str.charAt(0) - \'0\');'  'int y = (int)(str.charAt(1) - \'0\');'  \_07imi\_unogoods\_findgoods  CNC開始依座標找尋貨物  '//CNC開始依座標找尋貨物'  'findGoods(x,y);'  \_08imi\_unogoods\_servo  設定為 角度  撿貨伺服馬達  推貨  '// 撿貨伺服馬達'  dropdown\_servo+'.write('+dropdown\_angle+');'  'delay(2000);'  \_09imi\_unogoods\_reset  電機位置歸零  '//歸零'  'resetStepper();'  \_10imi\_unogoods\_sendesp32  [傳送]訊息至ESP32  GOODS\_LOAD  '// 完成撿貨通知ESP32'  'ESP32Serial.print(GOODS\_LOAD);' | // 接收訊息：ESP32→Arduino  while (ESP32Serial.available()) {  //取得貨物在倉庫xy座標  String str = ESP32Serial.readString();// 格式xy(例如13)  Statements區塊  }//end while |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |