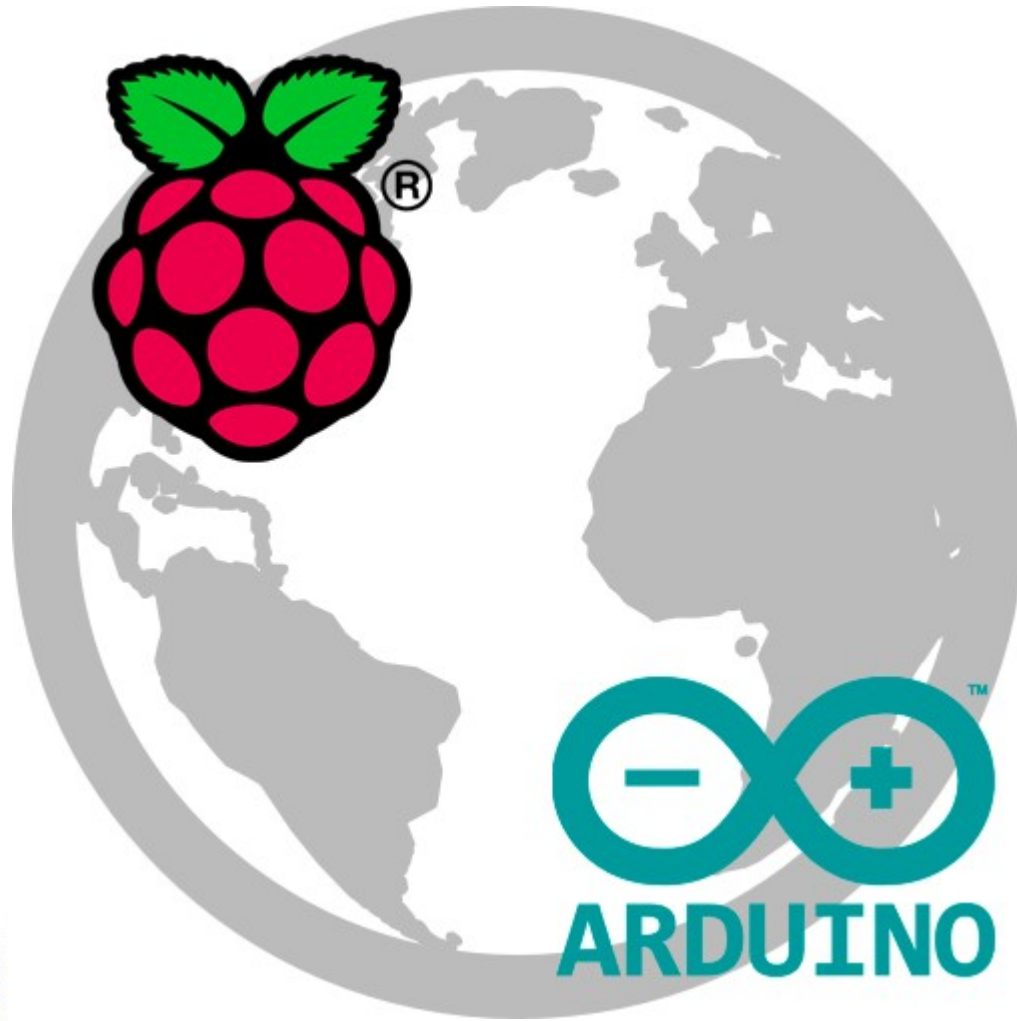


Arduino i Raspberry Pi



Telecos.cat
enginyers de telecomunicació,
electrònica i multimèdia-audiovisual

**22, 24, 29 de novembre
i 1 de desembre 2016**



Arduino i Raspberry Pi

- * Maneres de programar un Arduino (per blocs i des de l'IDE d'Arduino)
- * Connexió de perifèrics a l'Arduino (entrades i sortides digitals, **SPI**, **I2C**, UART)
- * **Comunicació entre l'Arduino i l'ordinador**
- * Bluetooth i **RS485**
- * **ESP8266**
- * **Maneres de programar una Raspberry Pi (Python, BASH, C++, Qt)**
- * **Connexió de perifèrics a la Raspberry Pi**
- * Automatització de processos amb la Raspberry Pi (sense entorn gràfic i amb entorn gràfic. Mode quiosc)
- * Comunicació entre la Raspberry Pi i l'Arduino
- * **Comunicació entre la Raspberry Pi i l'ordinador**
- * Comunicació entre la Raspberry Pi i el núvol (Introducció a Internet de les coses i seguretat en les comunicacions)

Arduino i Raspberry Pi

Dubtes del dia anterior

Pràctica SPI

Pràctica amb SPI (RFID)

Pràctica I2C (PCF8574)

Pràctica amb RS485 (SoftwareSerial.ino)

Pràctica Ethernet

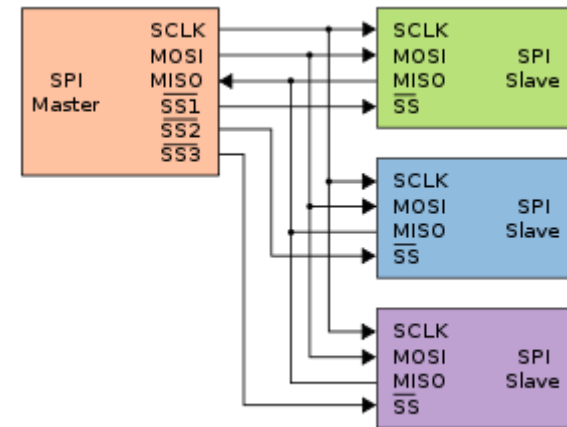
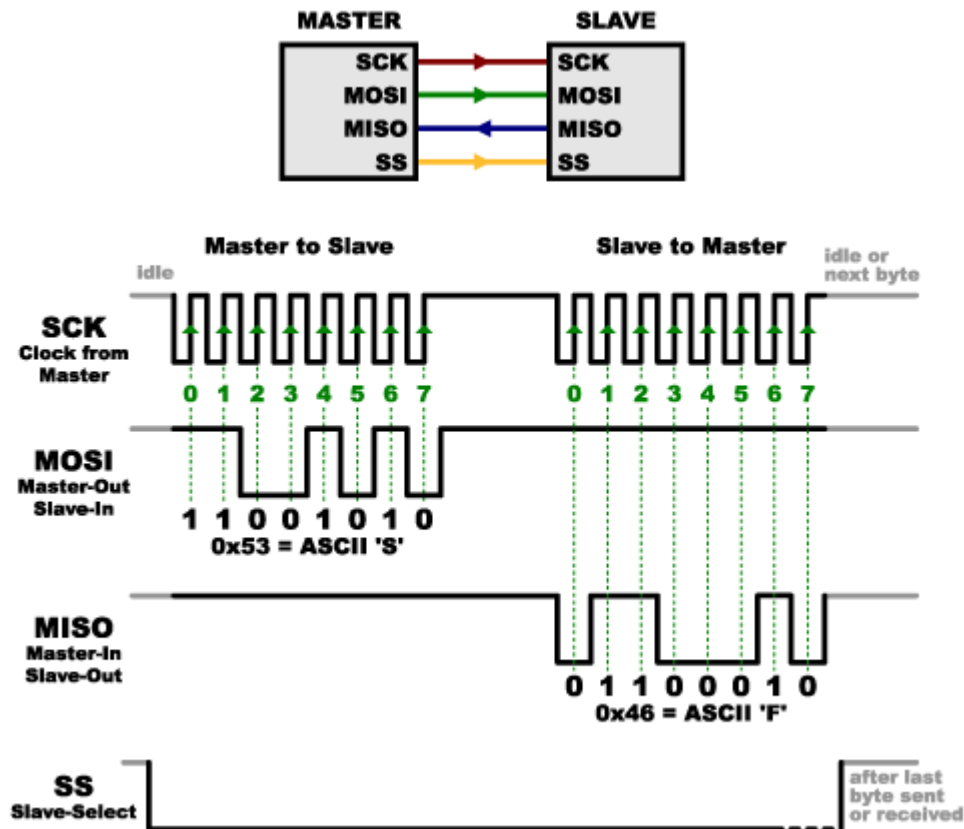
Pràctiques ESP8266

Introducció a la Raspberry Pi



Arduino i Raspberry Pi

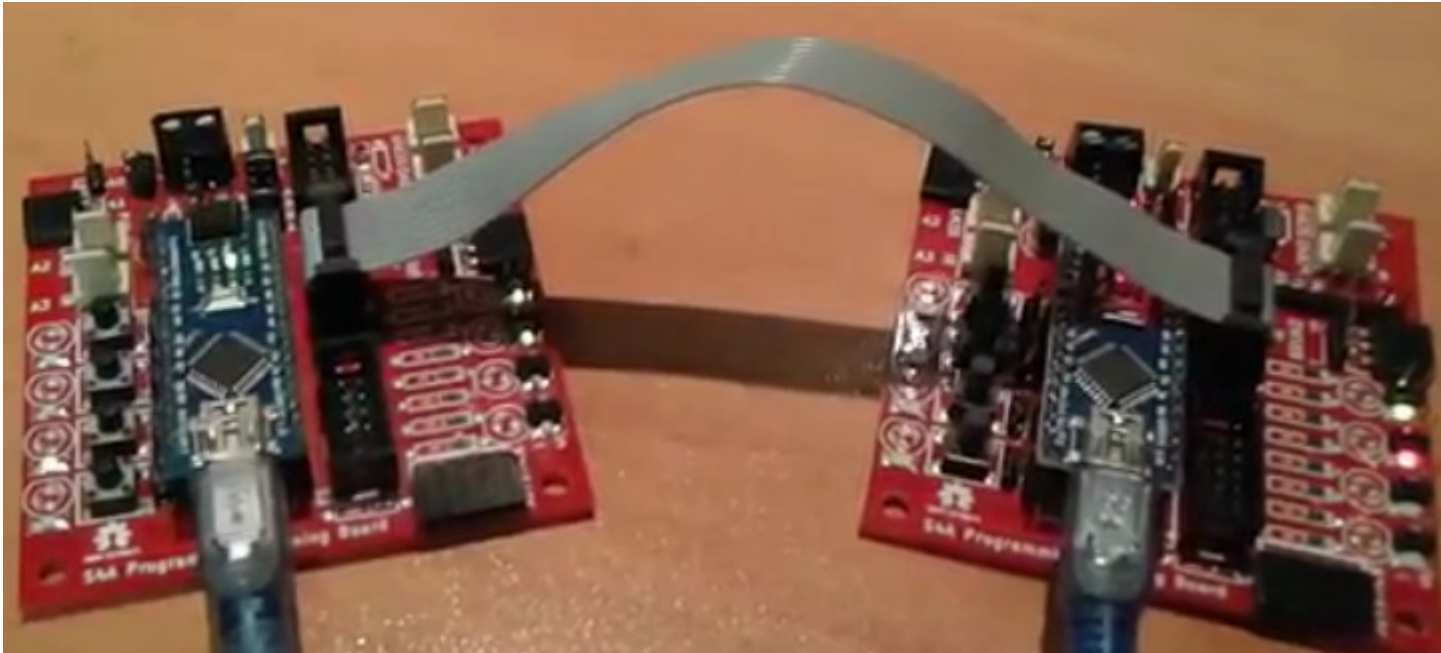
SPI



Explicació del bus SPI

Arduino i Raspberry Pi

SPI



[Arduino's SPI master code](#)

[Arduino's SPI slave code](#)

[Vídeo de funcionament](#)



Arduino i Raspberry Pi

SPI

```

Serial.println("Master Initialized");
}
// The loop() function runs continuously after setup
void loop() {
  // Master button pressed?
  if (!digitalRead(btn)) {
    // Yes
    Serial.println("Master Button Pressed.");
    // Select and wait for slave.
    digitalWrite(SS, LOW);
    Serial.println("***Slave Selected.");
    delay(20);
    // Send cmdBtn
    SPI.transfer(cmdBtn);
    Serial.println("cmdBtn Sent.");
    // Wait for slave.
    delay(20);
    // Get slave response.
    byte rx = SPI.transfer(255);
    // Acknowledged?
    if (rx == cmdBtn) {
      // Yes.
      Serial.println("Slave acknowledged cmdBtn.");
    }
    // Command not recognized?
  }
}

```

Arduino's SPI master code

Arduino's SPI slave code

```

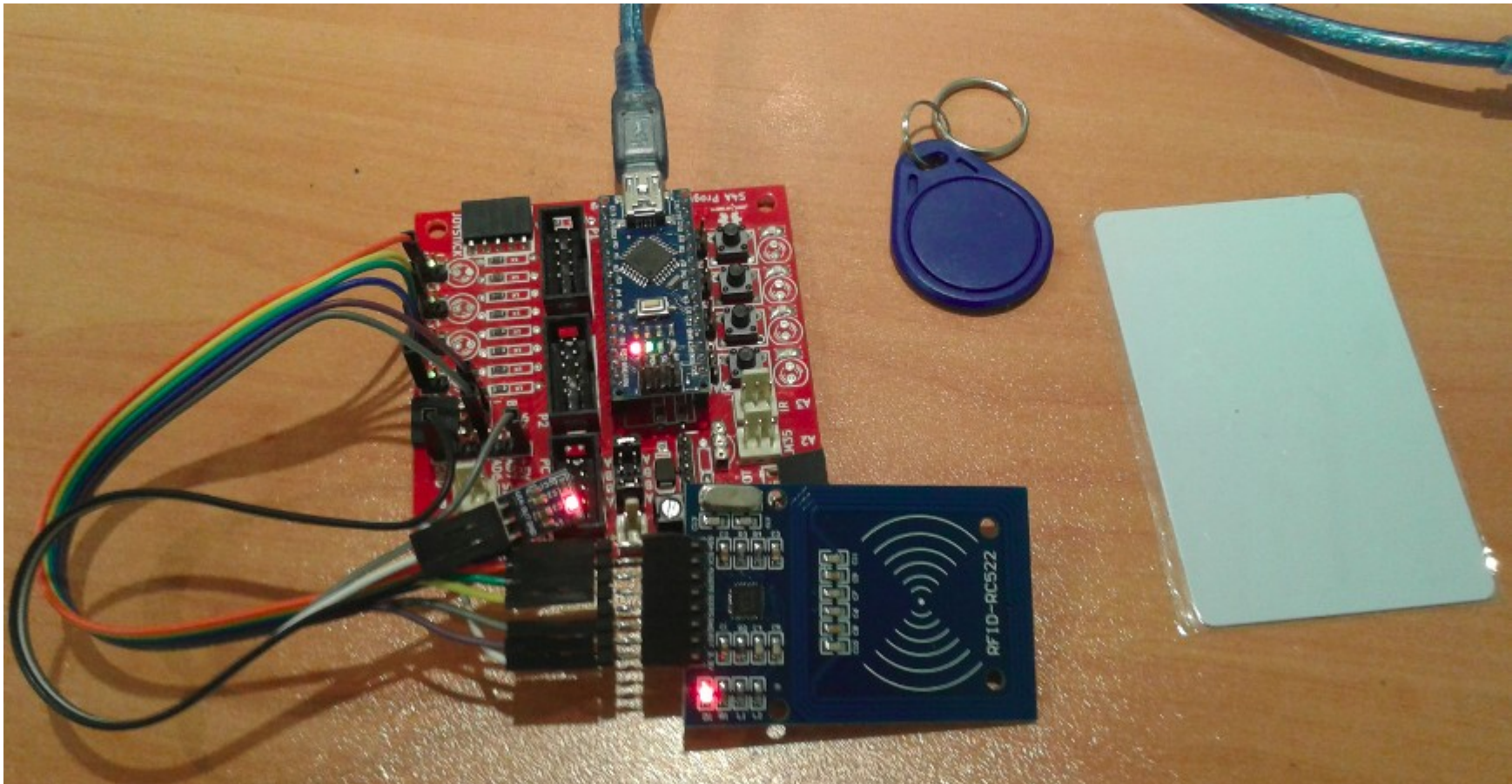
void loop() {
  // Slave Enabled?
  if (!digitalRead(SS)) {
    // Yes, first time?
    if (SSlast != LOW) {
      // Yes, take MISO pin.
      pinMode(MISO, OUTPUT);
      Serial.println("***Slave Enabled.");
      // Write -1 slave response code and receive master command code
      byte rx = SPI.transfer(255);
      Serial.println("Initial -1 slave response code sent");
      Serial.println("rx:" + String(rx) + ".");
      // cmdBtn?
      if (rx == cmdBtn) {
        // Acknowledge cmdBtn.
        byte rx = SPI.transfer(cmdBtn);
        Serial.println("cmdBtn Acknowledged.");
        Serial.println("rx:" + String(rx) + ".");
        // Toggle LED State
        ledState = !ledState;
        digitalWrite(led, ledState);
      }
      // cmdLEDState?
    } else if (rx == cmdLEDState) {
      // Acknowledge cmdLEDState.
      byte rx = SPI.transfer(cmdLEDState);
      Serial.println("cmdLEDState Acknowledged.");
      Serial.println("rx:" + String(rx) + ".");
      rx = SPI.transfer(ledState);
      Serial.println("ledState:" + String(ledState) + " Sent.");
      Serial.println("rx:" + String(rx) + ".");
    }
  }
}

```




Arduino i Raspberry Pi

SPI



Com connectar el mòdul RFID-RC522 a Arduino



Arduino i Raspberry Pi

SPI

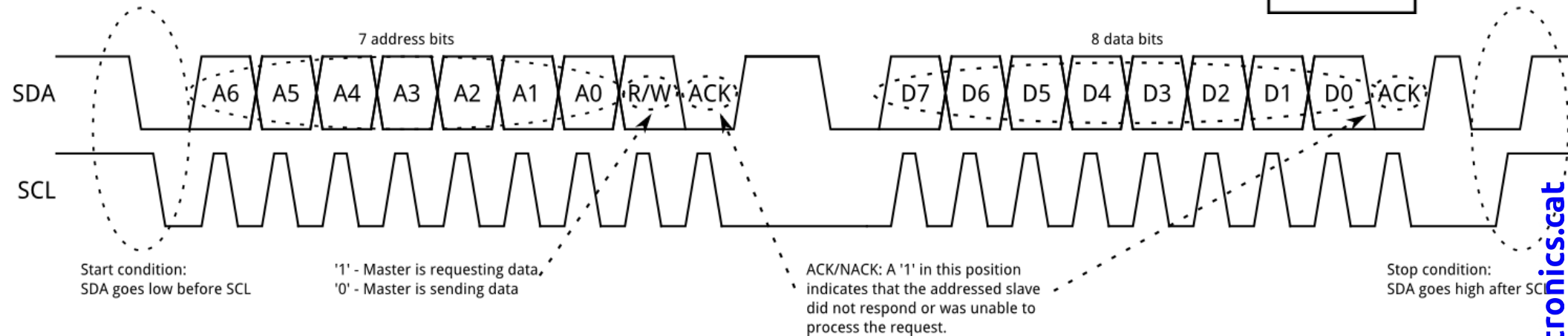
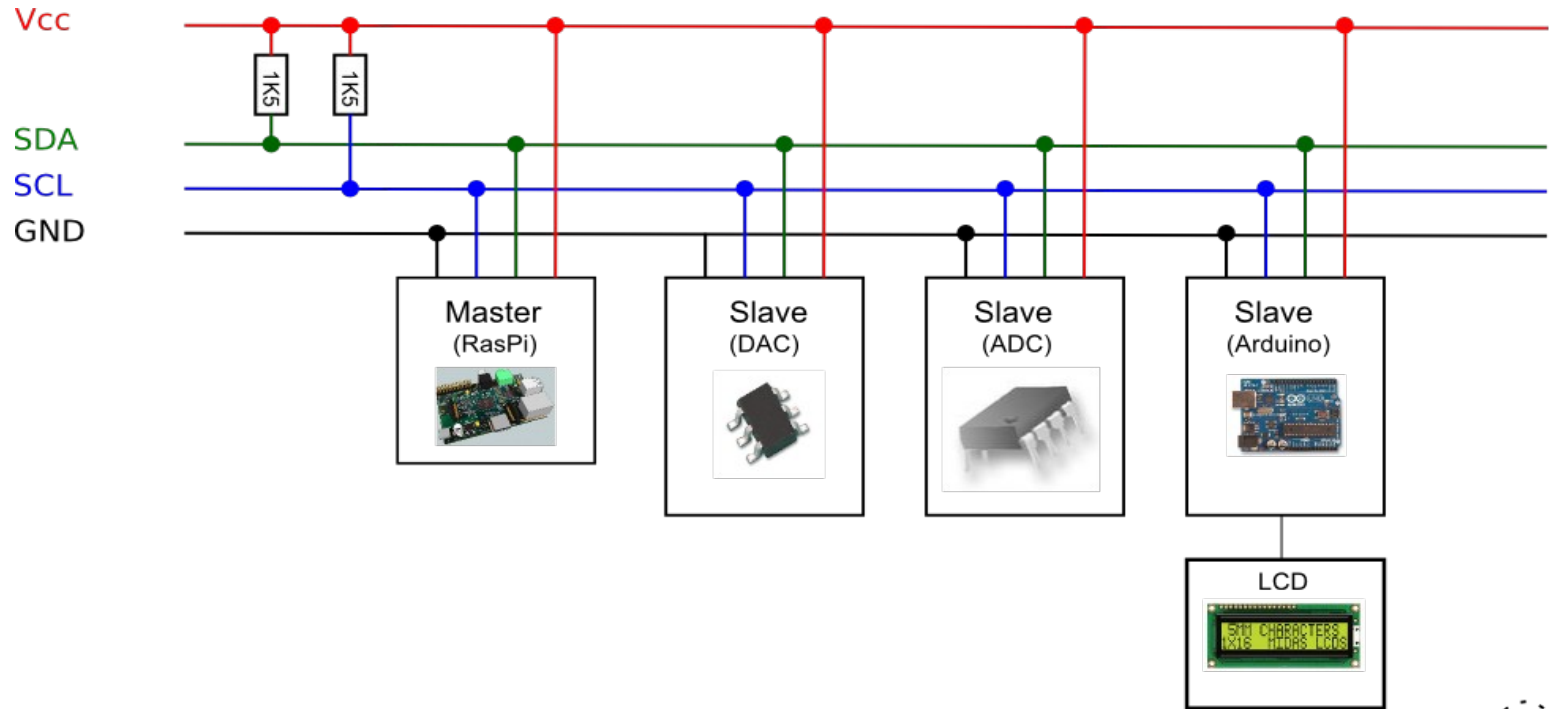
```
/dev/ttyUSB0
| Envia
This code scan the MIFARE Classsic NUID.
Using the following key: FF FF FF FF FF FFPICC type: MIFARE 1KB
A new card has been detected.
The NUID tag is:
In hex: 35 07 10 22
In dec: 53 07 16 34
PICC type: MIFARE 1KB
A new card has been detected.
The NUID tag is:
In hex: F1 45 25 61
In dec: 241 69 37 97

☒ Desplaçament automàtic  Ambdós NL & CR  9600 baud
```



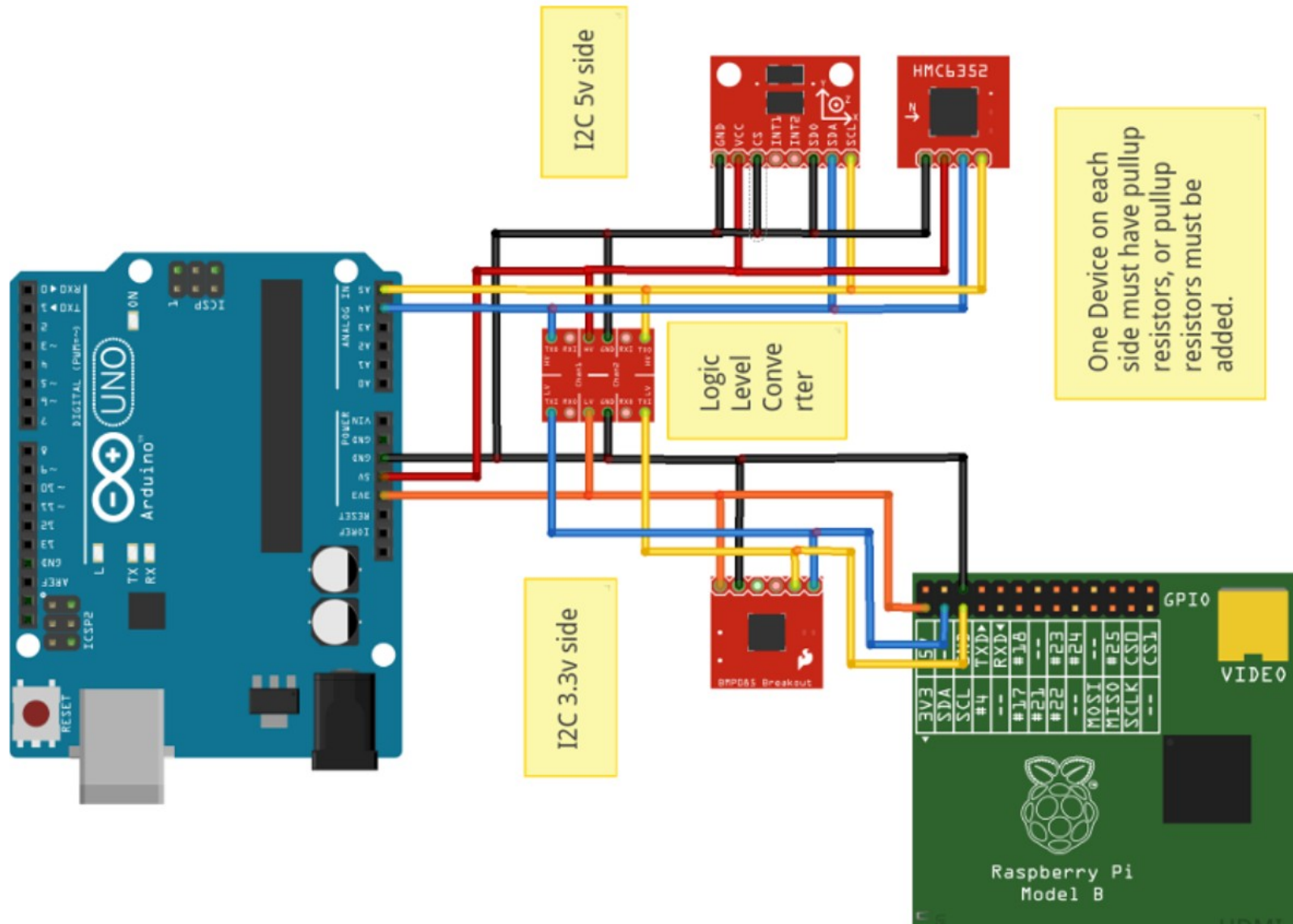

Arduino i Raspberry Pi

I2C



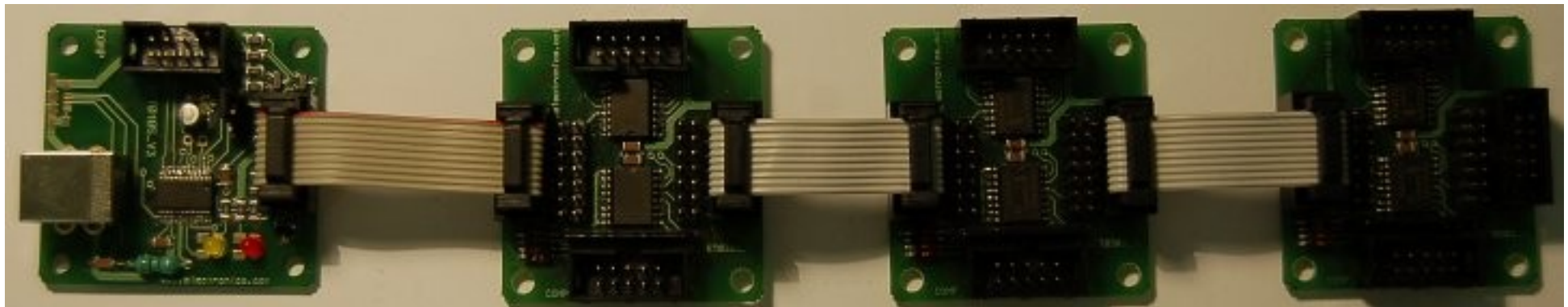
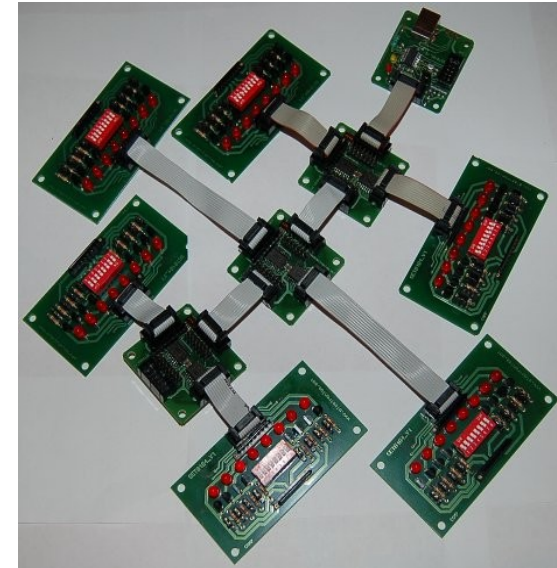
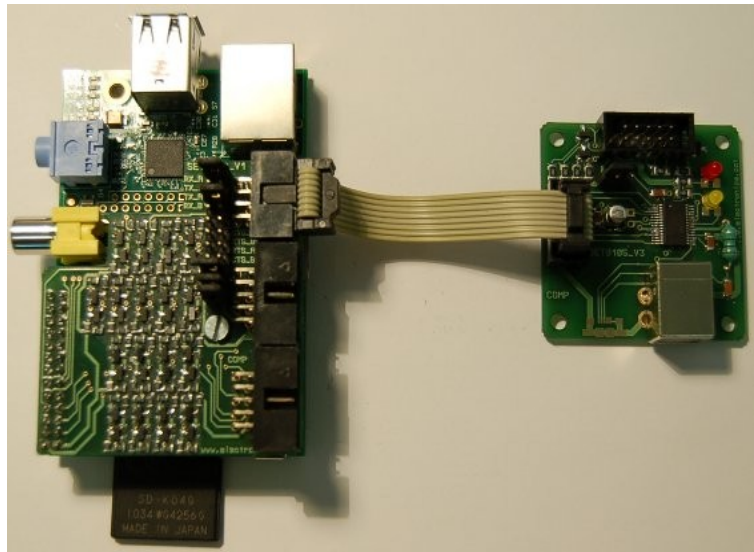
Arduino i Raspberry Pi

I2C



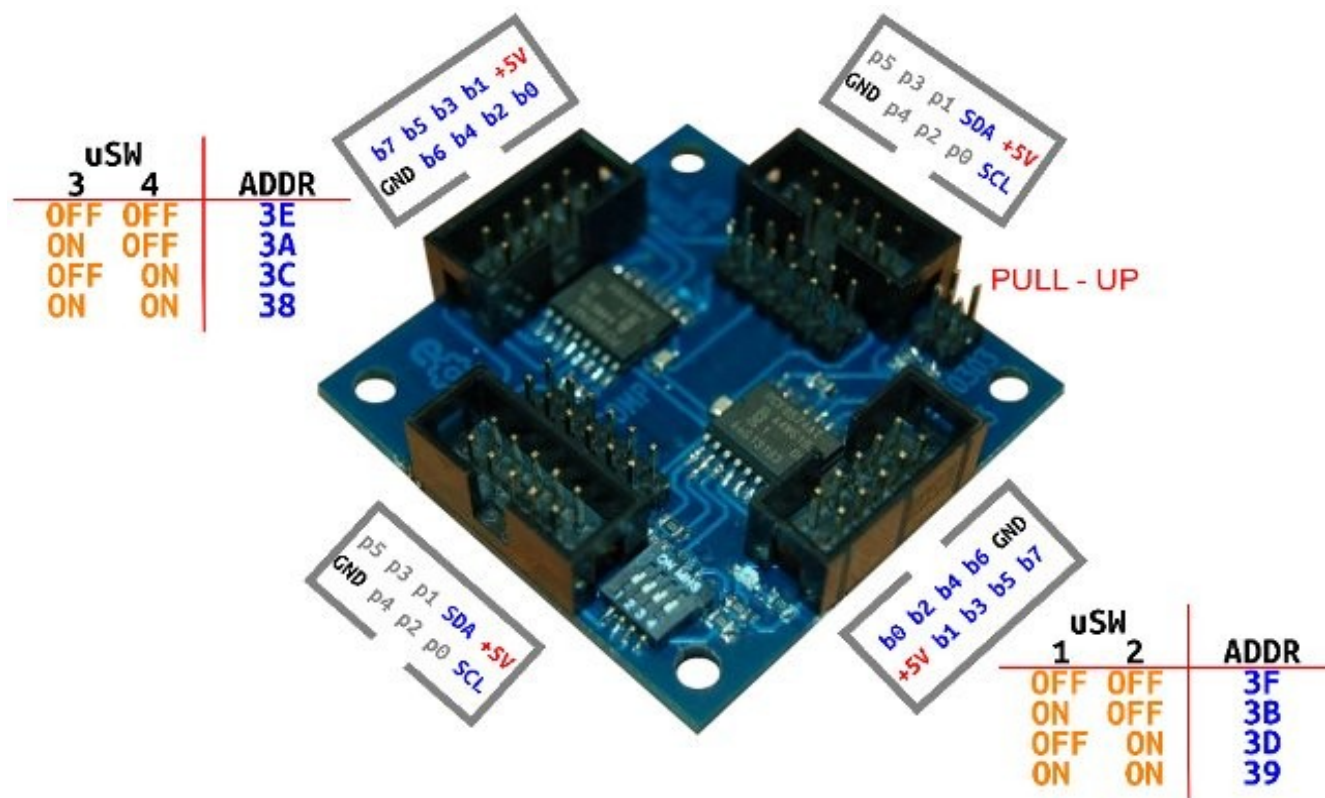
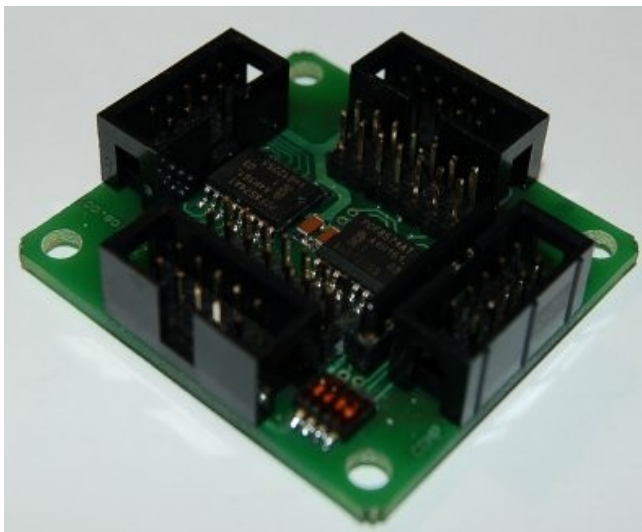
Arduino i Raspberry Pi

I2C



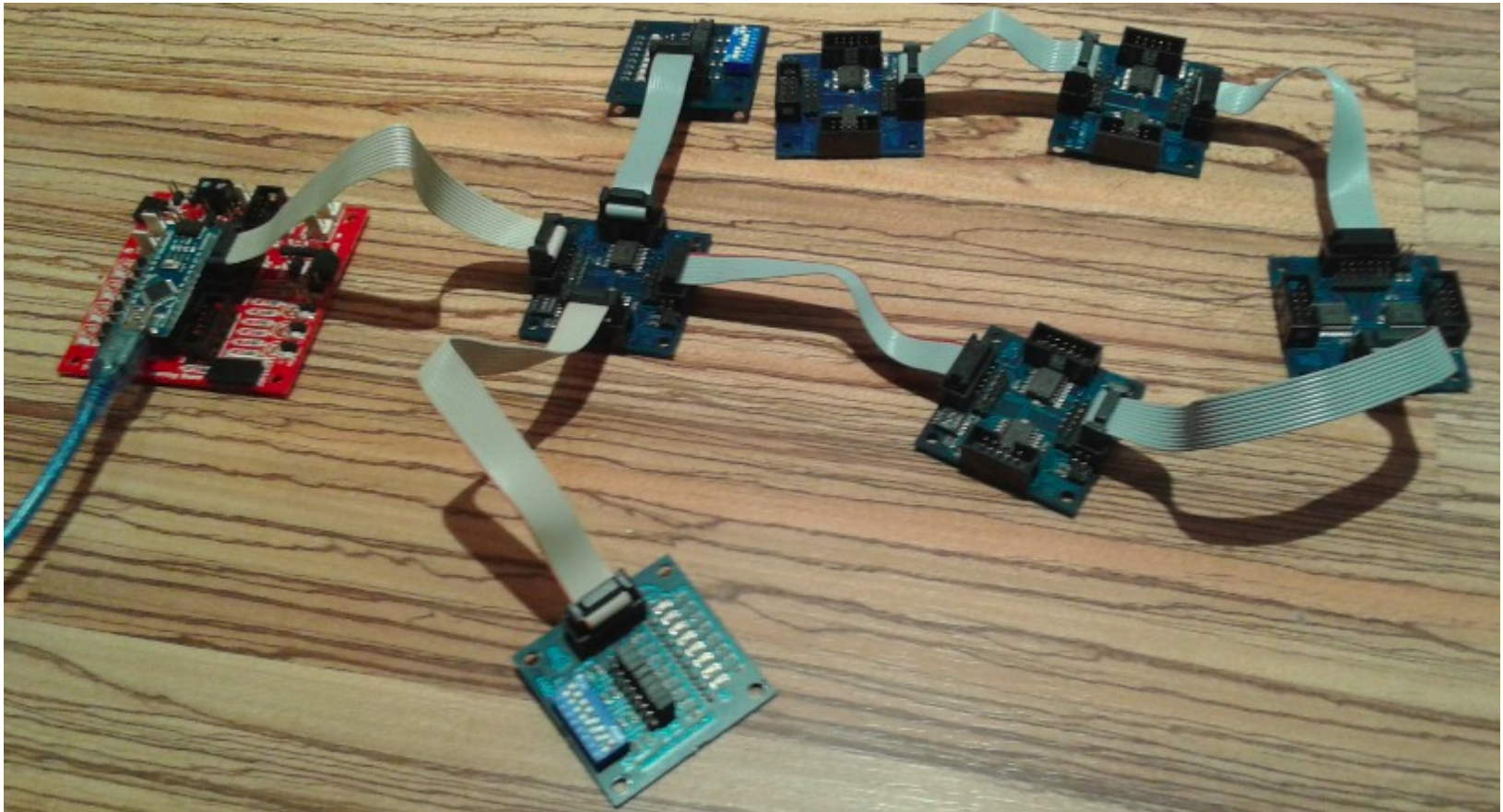
Arduino i Raspberry Pi

I2C



Arduino i Raspberry Pi

I2C



Codis I2C



Arduino i Raspberry Pi

I2C

I2cScanner.ino

```
void loop()
{
  byte error, address;
  int nDevices;

  Serial.println("Scanning...");

  nDevices = 0;
  for(address = 1; address < 127; address++)
  {
    // The i2c_scanner uses the return value of
    // the Write.endTransmission to see if
    // a device did acknowledge to the address.
    Wire.beginTransmission(address);
    error = Wire.endTransmission();

    if (error == 0)
    {
      Serial.print("I2C device found at address 0x");
      if (address<16)
        Serial.print("0");
      Serial.print(address,HEX);
      Serial.println("  !");

      nDevices++;
    }
  }
}
```




Arduino i Raspberry Pi

I2C

I2cScanner.ino

```
/dev/ttyUSB0
done
Scanning...
I2C device found at address 0x20 !
I2C device found at address 0x21 !
I2C device found at address 0x22 !
I2C device found at address 0x23 !
I2C device found at address 0x24 !
I2C device found at address 0x25 !
I2C device found at address 0x26 !
I2C device found at address 0x27 !
I2C device found at address 0x38 !
I2C device found at address 0x39 !
done
```

☒ Desplaçament automàtic Ambdós NL & CR 9600 baud



Arduino i Raspberry Pi

I2C

[// http://playground.arduino.cc/Code/I2CPortExpander8574](http://playground.arduino.cc/Code/I2CPortExpander8574)

```
#include <Wire.h>
// 8574 Address range is 0x20-0x27
// 8574A Address range is 0x38-0x3F
// 9555 Address range is 0x20-0x27 (same as 8574, bummer)

#define INAddr 0x20
#define OUTAddr 0x21

void setup()
{
  pinMode(12, INPUT); // to read /INT
  pinMode(13, OUTPUT); // to show we are working
  Wire.begin();
  expanderSetInput(INAddr, 0xFF);
}
```

```
void loop(){
  static byte data = 0x01;

  expanderWrite(OUTAddr, (byte)data);
  data <=< 1;
  if(!data)
    data = 0x01;
  delay(200);
}
```

```
void loop(){
  static byte data = 0x01;

  expanderWrite(OUTAddr, data);
  expanderWrite(INAddr, ~data);
  data <=< 1;
  if(!data)
    data = 0x01;
  delay(200);
}
```

```
byte expanderRead(int i2caddr) {
  int _data = -1;
  Wire.requestFrom(i2caddr, 1);
  if(Wire.available()) {
    _data = Wire.read();
  }
  return _data;
}

void expanderWrite(int i2caddr, byte data)
{
  Wire.beginTransmission(i2caddr);
  Wire.write(data);
  Wire.endTransmission();
}
```

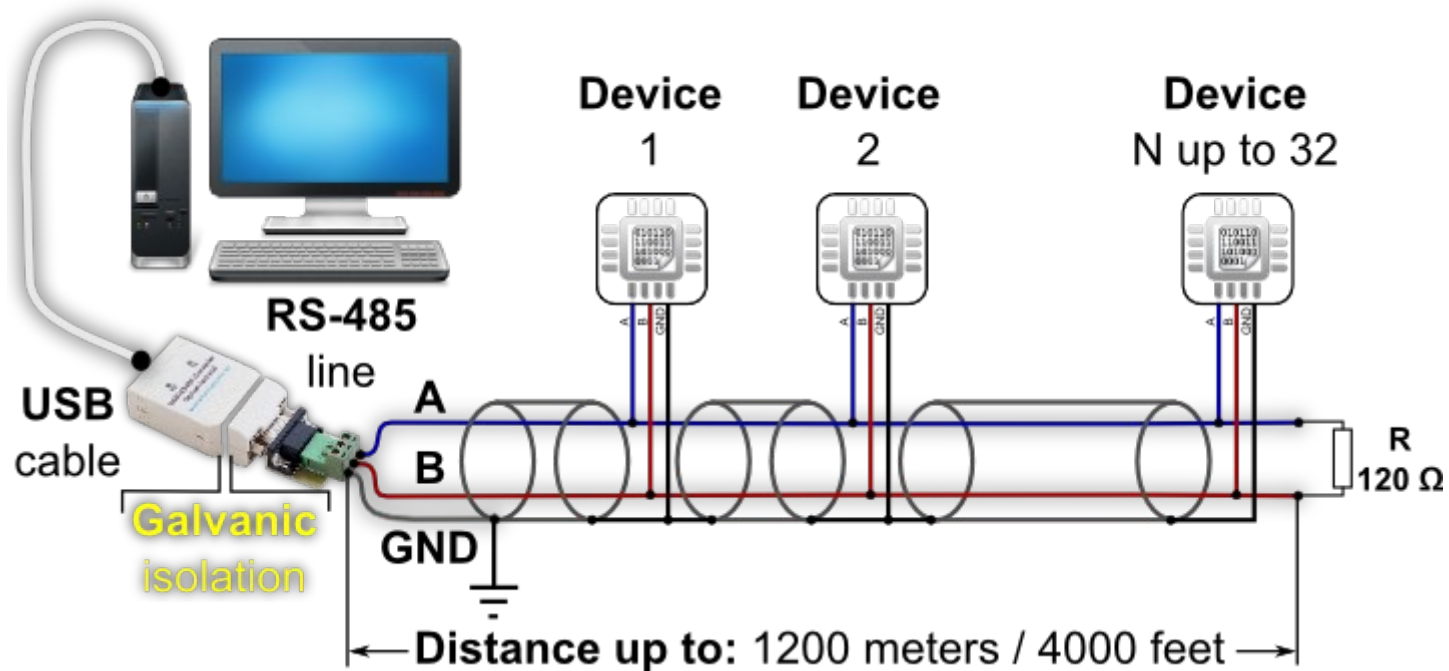
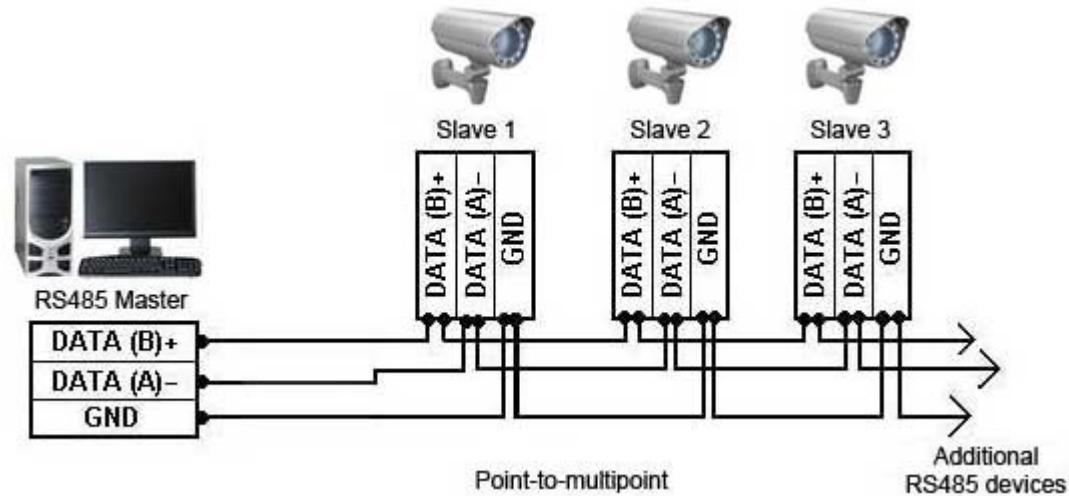
```
void loop(){
  byte data = expanderRead(INAddr);

  expanderWrite(OUTAddr, data);
}
```

Codis I2C

Arduino i Raspberry Pi

RS485





Arduino i Raspberry Pi

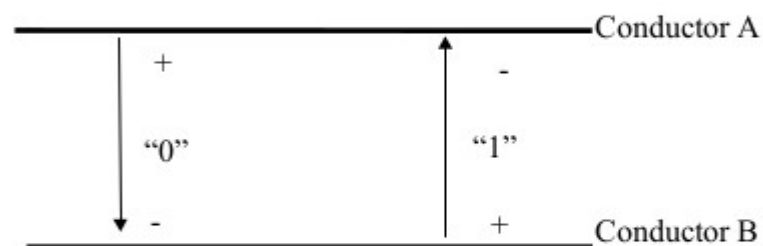
RS422

Comunicación Serie RS-422

Descripción

La RS-422 trabaja en forma diferencial con las líneas que transmite y recibe, el circuito tiene solo dos hilos sin que exista una línea de masa común. Los unos y ceros lógicos se establecen en función de la diferencia de tensión ambos conductores del circuito.

Especificaciones		RS422
Modo de Operación		DIFERENCIAL
Número de dispositivos		1 EMISOR 10 RECEPTORES
Máxima longitud del cable		1200 metros
Máxima velocidad de transmisión		10 Mb/s
Rango de trabajo		+/-10V
Rango de señal	Alto	+/-6V
	Bajo	+/-2V
Sensibilidad de entrada receptor		+/-200mV
Resistencia de entrada receptor		>=4K



"0" $1,5 \text{ V} < \Delta V_{AB} < 5 \text{ V}$

"1" $-1,5 \text{ V} > \Delta V_{AB} > -5 \text{ V}$



Arduino i Raspberry Pi

RS485

Comunicación Serie RS-485 (I)

Descripción

La RS-485 es una leve modificación de la RS-422, redefiniendo características eléctricas para asegurar un nivel de tensión adecuado a la máxima carga, incrementándose el número de dispositivos.

En una red de dispositivos sobre una simple línea, es necesario direccionar uno en particular. Esto se puede realizar simplemente utilizando caracteres ASCII, constituyendo comandos de identificación del dispositivo y que este a su vez responde con los datos. Esto es un esquema básico de protocolo de comunicación denominado comúnmente maestro/esclavo (Master/Slave).

Especificaciones		RS485
Modo de Operación		DIFERENCIAL
Número de dispositivos		32 EMISORES 32 RECEPTORES
Máxima longitud del cable		1200 metros
Máxima velocidad de transmisión		10 Mb/s
Rango de trabajo		-7V a +12V
Rango de señal	Alto	+/-6V
	Bajo	+/-1,5V
Sensibilidad de entrada receptor		+/-200mV
Resistencia de entrada receptor		>=12K



Arduino i Raspberry Pi

RS485

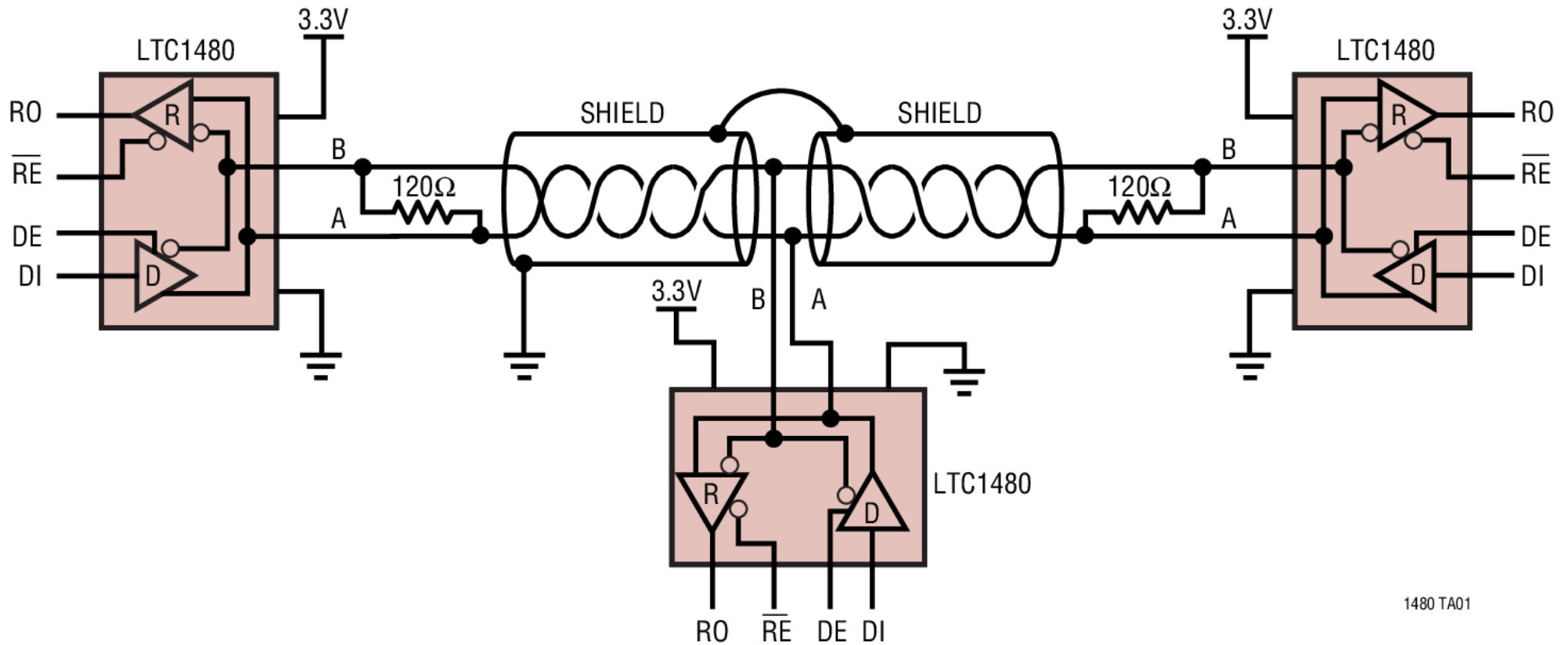
Comunicación Serie en Tensión – Comparativa

Especificaciones		RS232	RS423	RS422	RS485
Modo de Operación		NO DIFERENCIAL	NO DIFERENCIAL	DIFERENCIAL	DIFERENCIAL
Número de dispositivos		1 EMISOR 1 RECEPTOR	1 EMISOR 10 RECEPTORES	1 EMISOR 10 RECEPTORES	32 EMISORES 32 RECEPTORES
Máxima longitud del cable		15 metros	1200 metros	1200 metros	1200 metros
Máxima velocidad de transmisión		19,2 Kb/s	100 Kb/s	10 Mb/s	10 Mb/s
Rango de trabajo		+/-25V	+/-6V	+/-10V	-7V a +12V
Rango de señal	Alto	+/-15V	+/-6V	+/-6V	+/-6V
	Bajo	+/-3V	+/-3,6V	+/-2V	+/-1,5V
Sensibilidad de entrada receptor		+/-3V	+/-200mV	+/-200mV	+/-200mV
Resistencia de entrada receptor		7K	4K	$\geq 4K$	$\geq 12K$

Arduino i Raspberry Pi

RS485

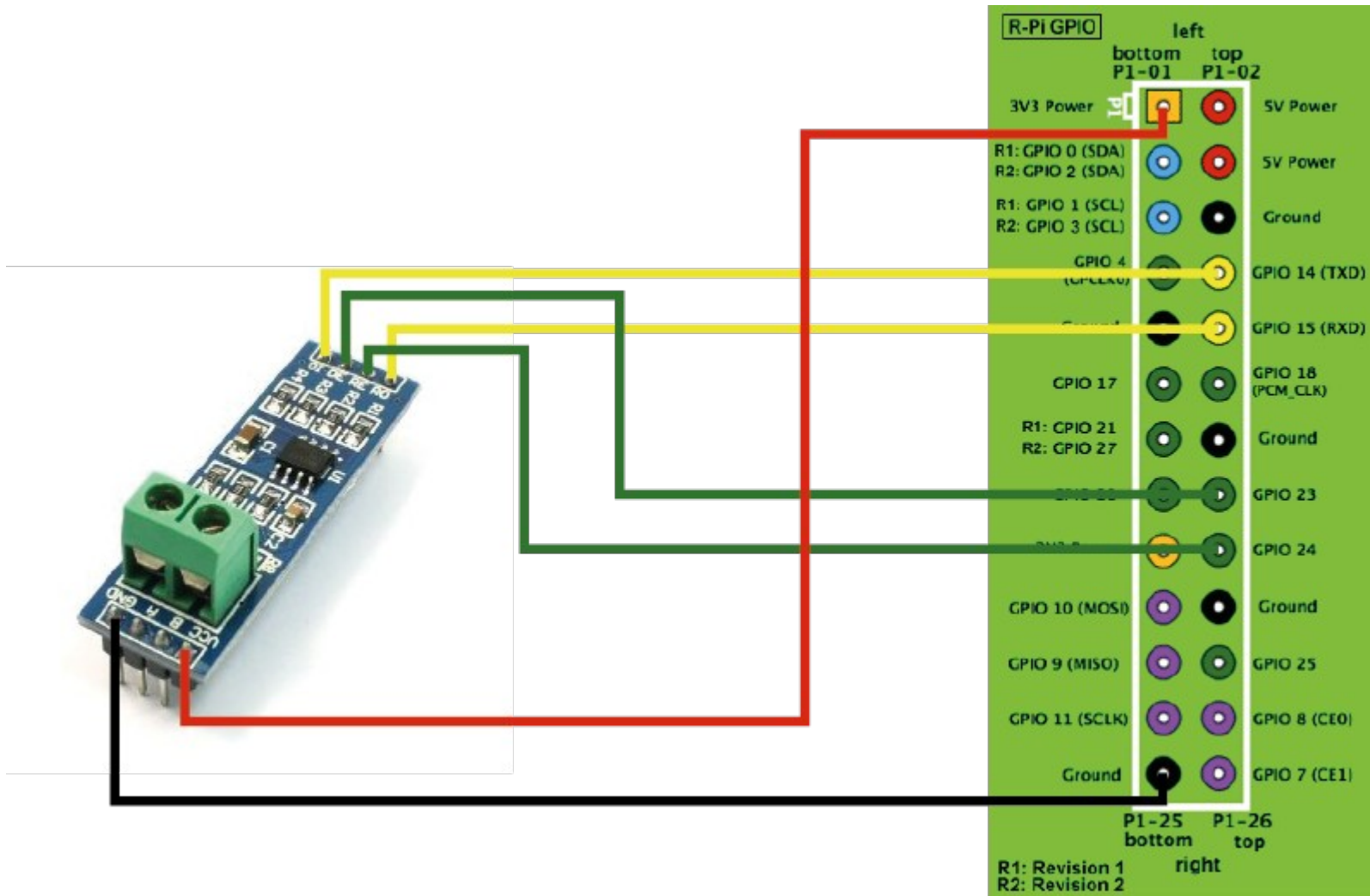
3.3V RS485 Network



1480 TA01

Arduino i Raspberry Pi

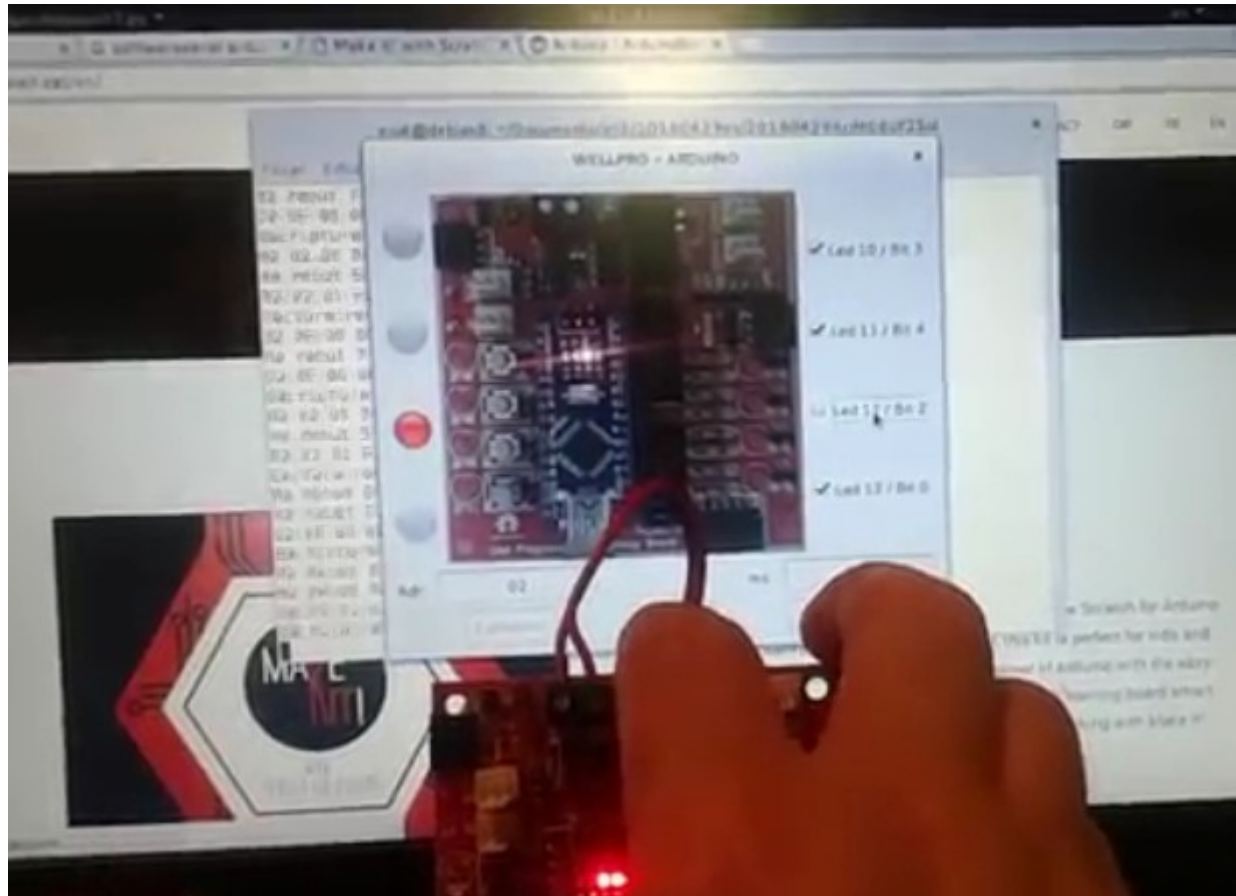
RS485





Arduino i Raspberry Pi

RS485 -ModBusRTU

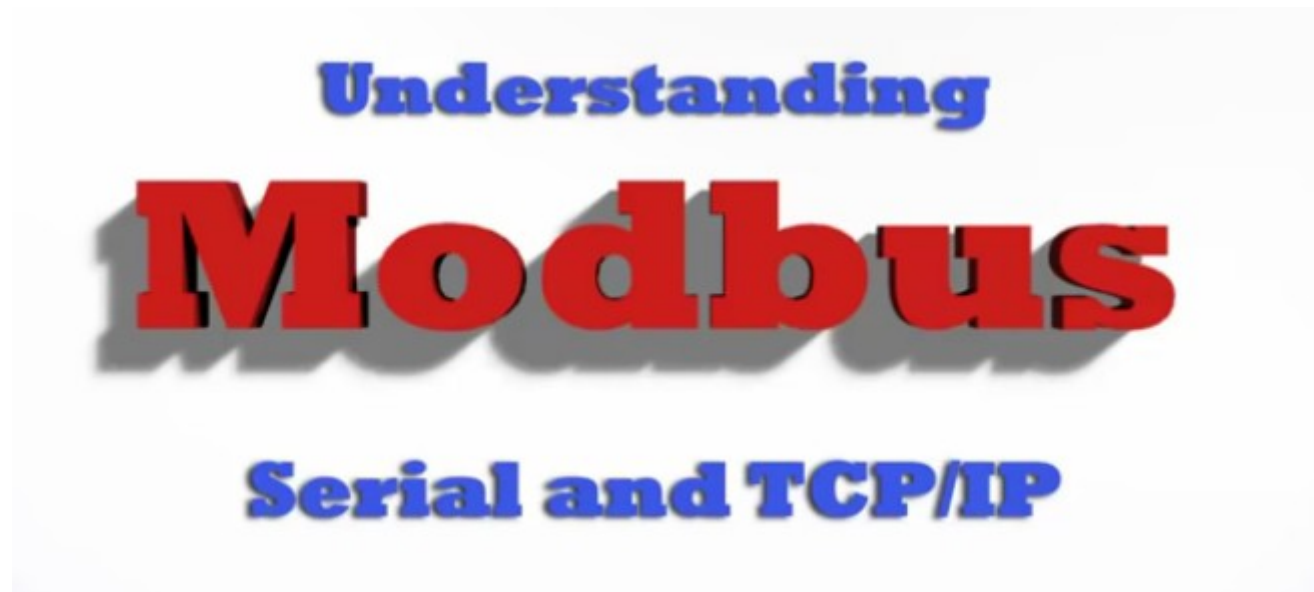


ModBus RTU sobre RS485 emprant Arduino i pyQt
 Modbus sobre USB emprant Arduino i pyQt
 Modbus RTU sobre un mòdul industrial



Arduino i Raspberry Pi

RS485 -ModBusRTU

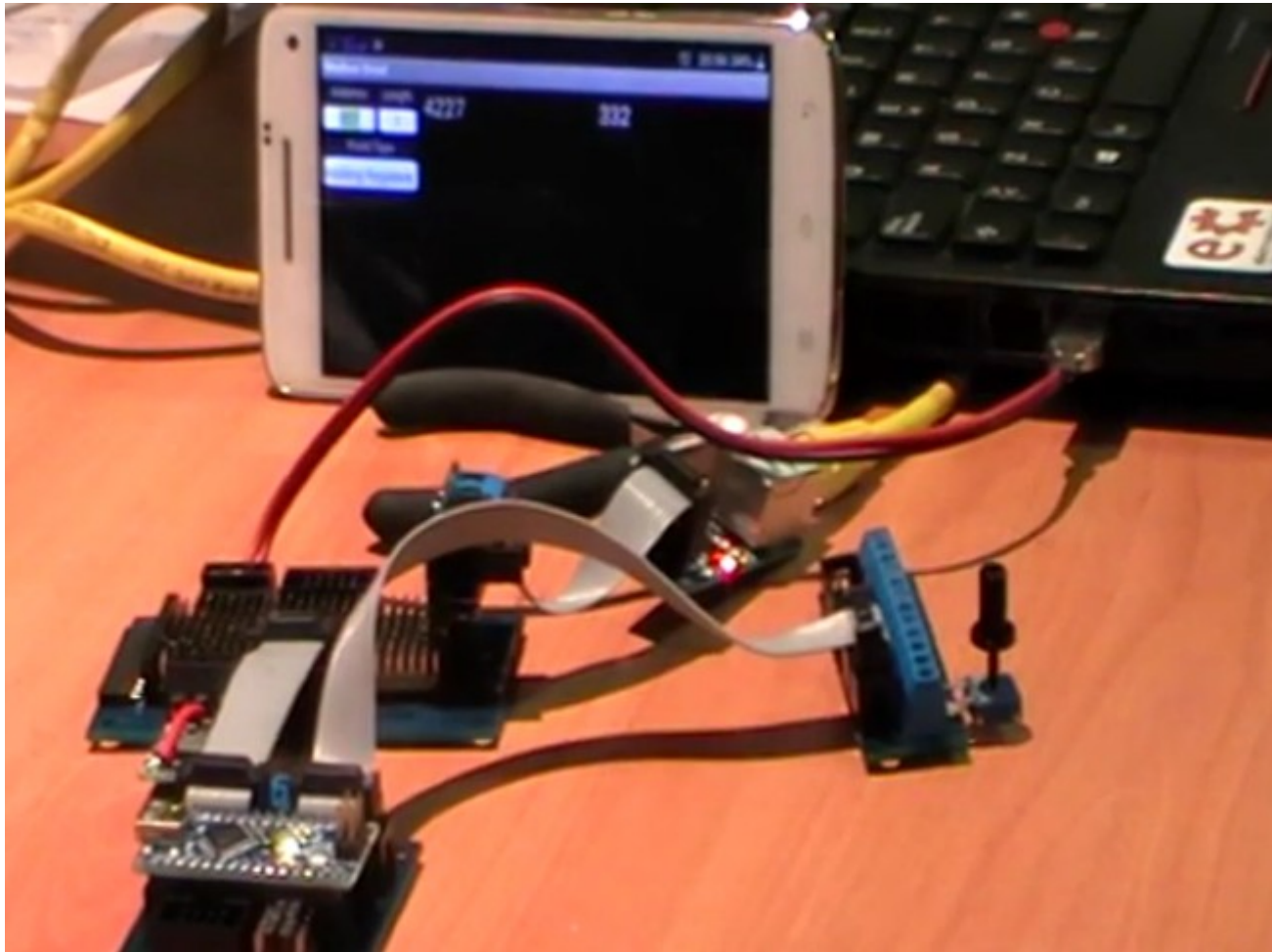


Vídeo explicatiu del ModBusRTU i el ModBusTCP



Arduino i Raspberry Pi

ModBusTCP

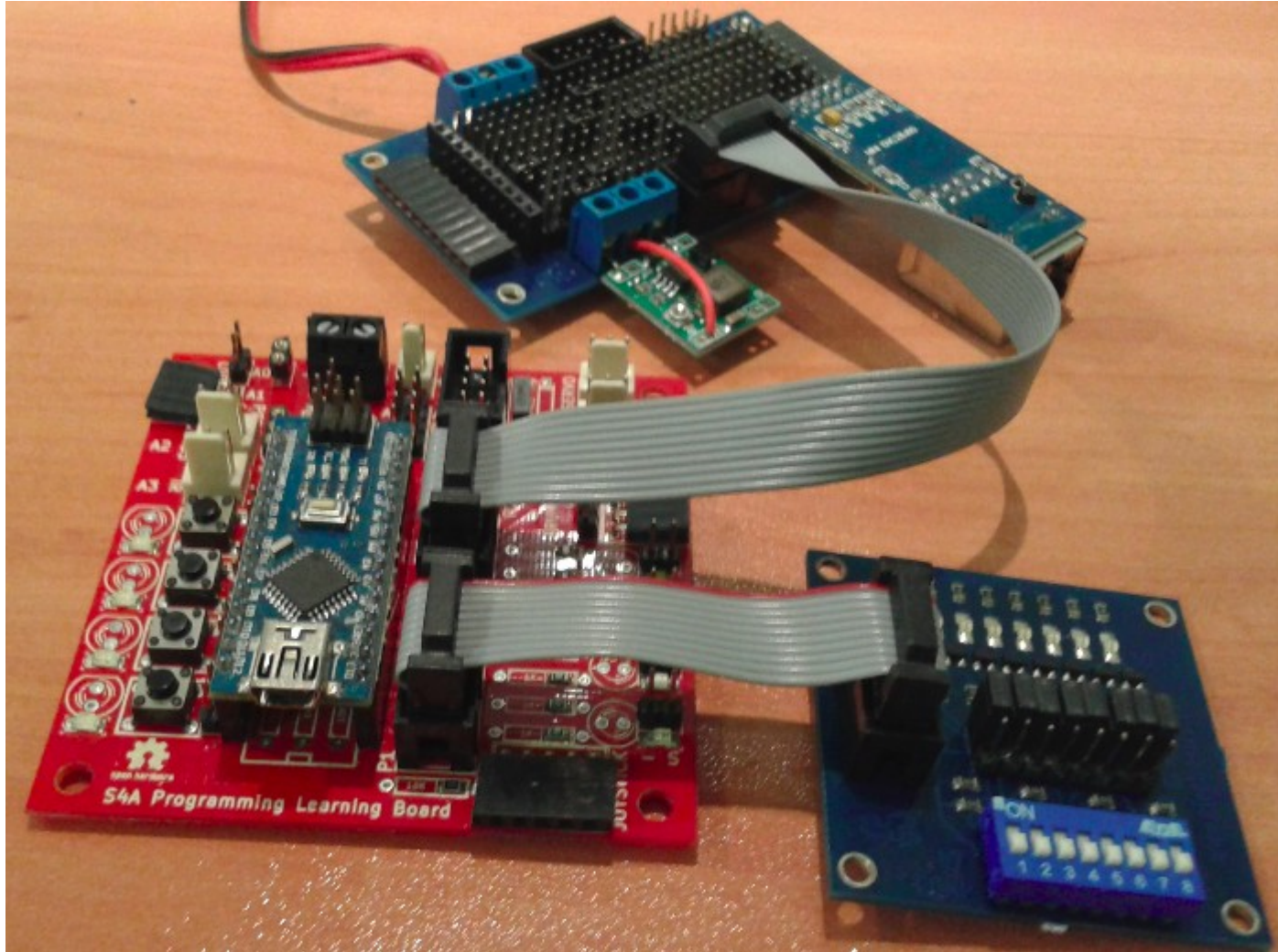


<http://www.binefa.cat/php/doc/modbus/>



Arduino i Raspberry Pi

Ethernet

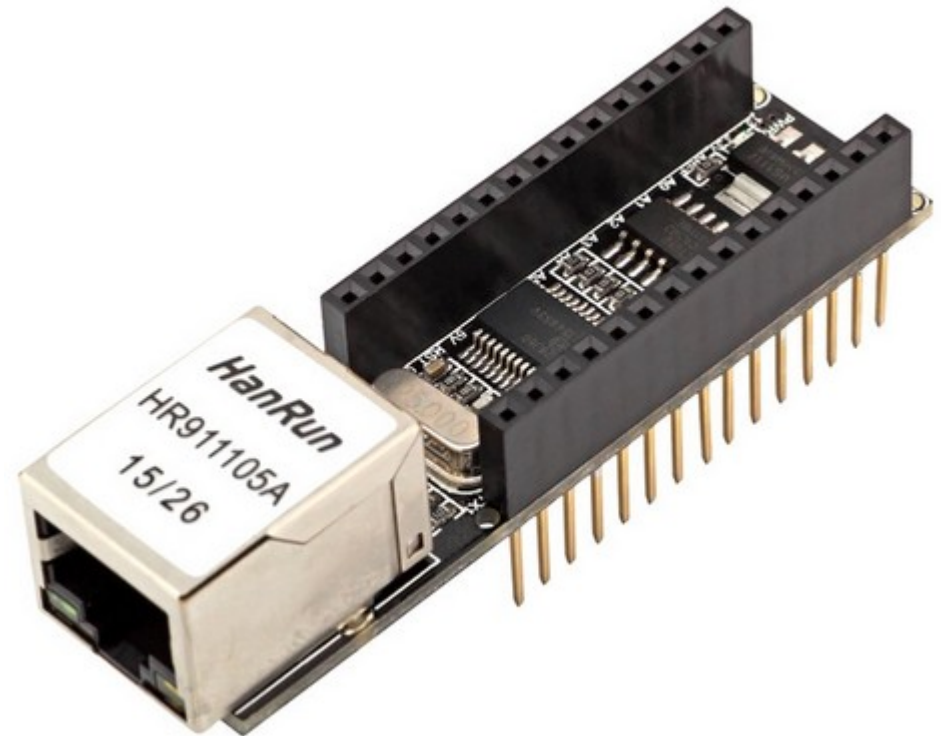


Codi per a Arduino UdpServer01b



Arduino i Raspberry Pi

Ethernet – Mòdul ENC28J60

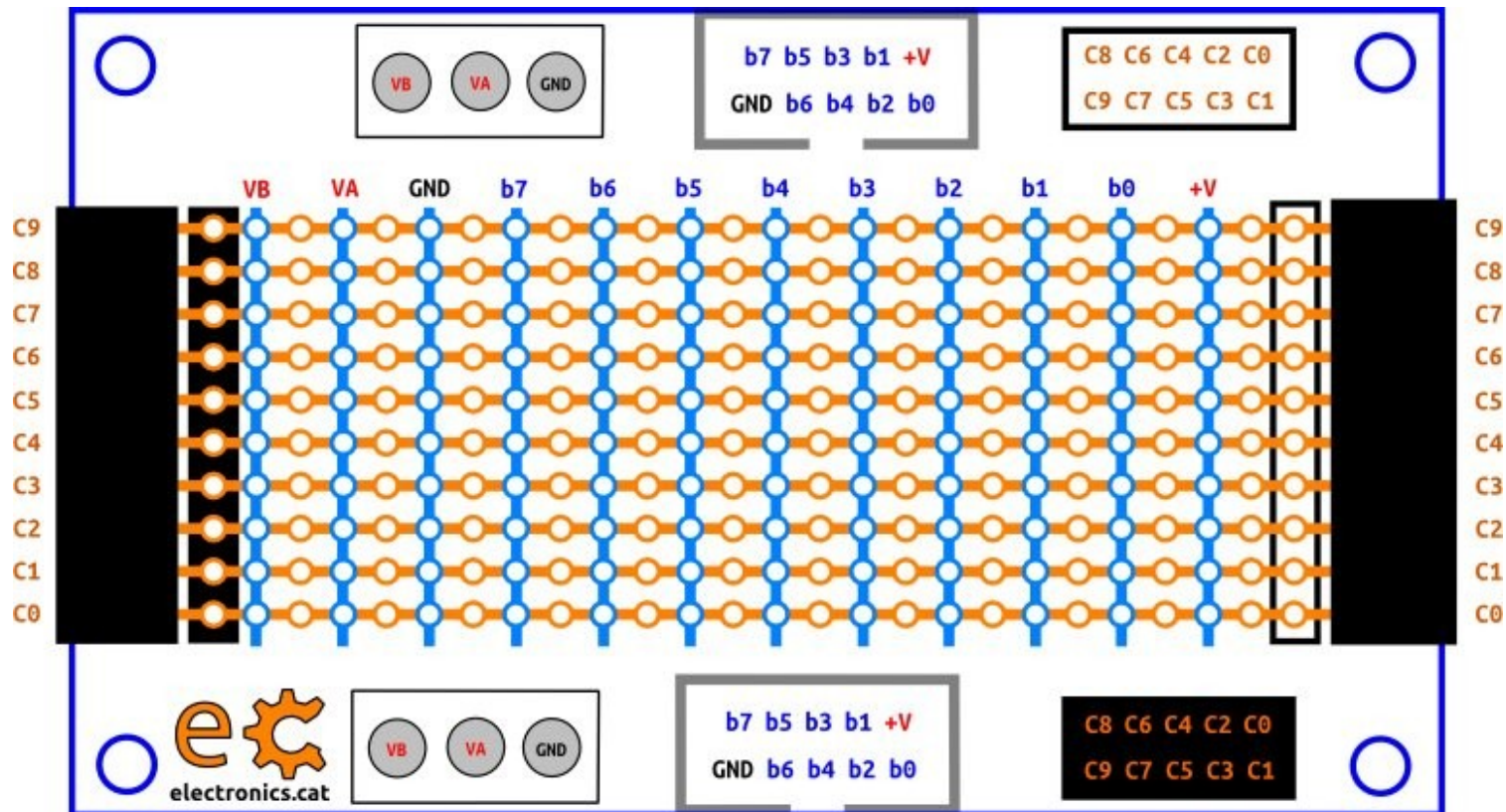


Biblioteca UIPEthernet per Arduino i ENC28J60



Arduino i Raspberry Pi

Matriu

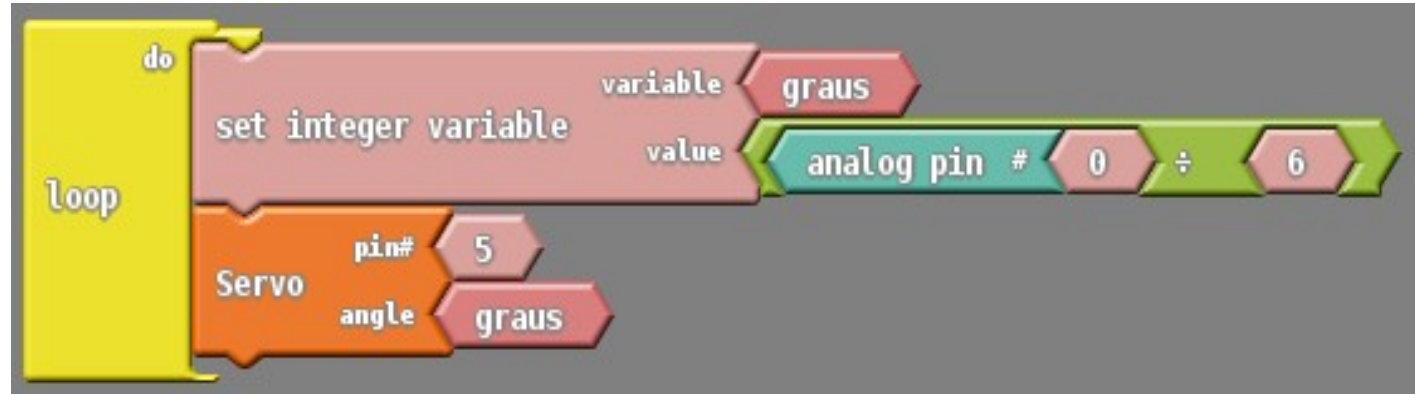
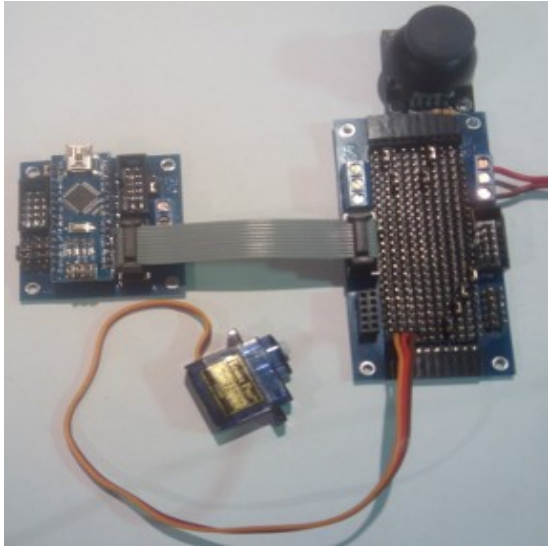


<http://electronics.cat/php/common/index.php?lang=ca&page=517>

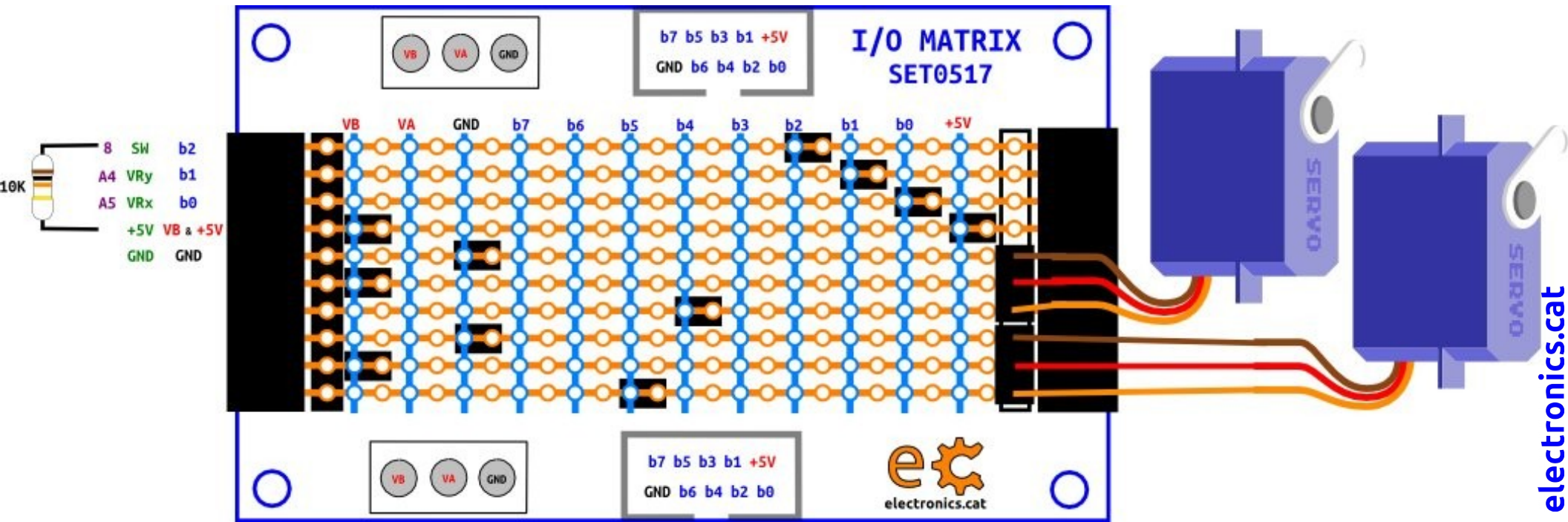


Arduino i Raspberry Pi

Exemple d'ús de la matriu



<http://www.binefa.cat/php/arduino/ardublock/pr04joystickServo.abp>





Arduino i Raspberry Pi

Matriu

Connexió a la matriu inversa (pàgina 100 del pdf)

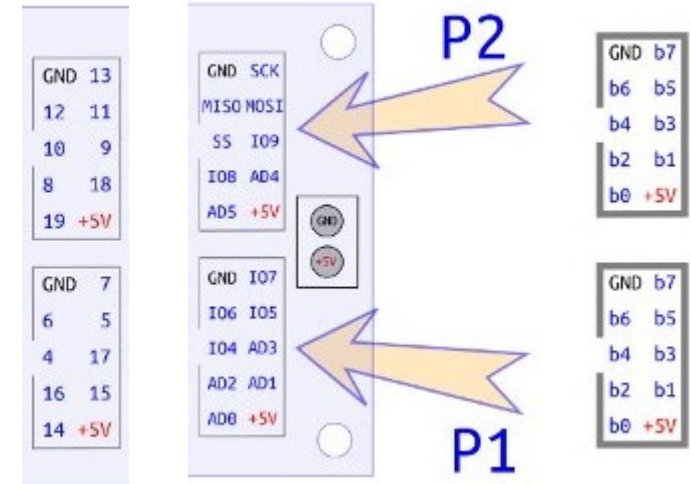
C8	C6	c4	C2	C0
C9	C7	C5	C3	C1

Connexió de l'ENC28J60

CLK	WOL	SI	CS	VCC
INT	S0	SCK	RST	GND

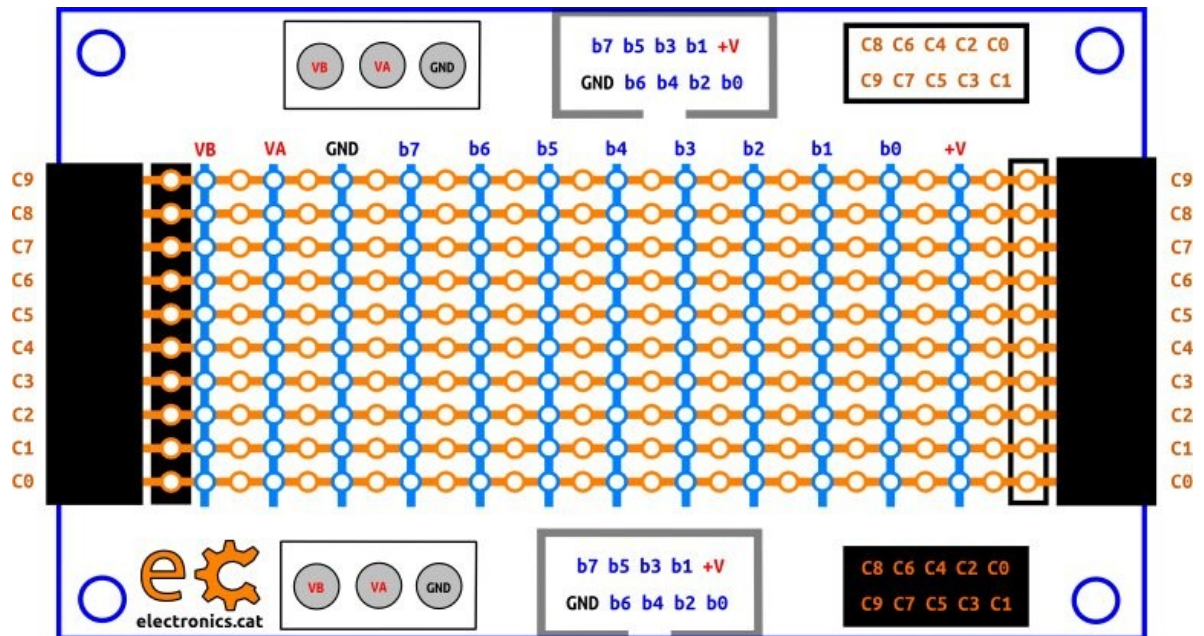
Connexió bus ecat

N	NC	b5	b4	3V3
NC	b6	b7	+5V	GND



Per pota de connector ecat

1	CK	NC
2	INT	NC
3	WOL	NC
4	S0	b6
5	SI	b5
6	SCK	b7
7	CS	b4
8	RST	+5V
9	VCC	3V3
10	GND	





Arduino i Raspberry Pi

Ethernet – Recepció de datagrames UDP

```
#include <UIPEthernet.h>
#include <ecat.h>

EthernetUDP udp;
Ecat ecat;

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

  uint8_t mac[6] = {0x00,0x01,0x02,0x03,0x04,0x05};

  Ethernet.begin(mac,IPAddress(192,168,1,108));
  //Ethernet.begin(mac,IPAddress(172,20,1,168));

  int success = udp.begin(5000);

  Serial.print("initialize: ");
  Serial.println(success ? "success" : "failed");
  ecat.setupHighNibbleP1(OUTPUT);
}
```

```
void loop() {
  //check for new udp-packet:
  int size = udp.parsePacket();
  if (size > 0) {
    do
    {
      char* msg = (char*)malloc(size+1);
      int len = udp.read(msg,size+1);
      msg[len]=0;
      Serial.print("received: ");
      Serial.print(msg);
      if(msg[0]=='0'){
        digitalWrite(ecat.nPinP1B7, LOW);
      }else{
        if(msg[0]=='1'){
          digitalWrite(ecat.nPinP1B7, HIGH);
        }
      }
      free(msg);
    }
    while ((size = udp.available())>0);
    //finish reading this packet:
    udp.flush();
    Serial.println("");
  }
}
```



Arduino i Raspberry Pi

Ethernet – Enviament de datagrames UDP

```

int success;
do
{
    Serial.print("remote ip: ");
    Serial.println(udp.remoteIP());
    Serial.print("remote port: ");
    Serial.println(udp.remotePort());
    //send new packet back to ip/port of client. This also
    //configures the current connection to ignore packets from
    //other clients!
    success = udp.beginPacket(udp.remoteIP(),udp.remotePort());
    Serial.print("beginPacket: ");
    Serial.println(success ? "success" : "failed");
    //beginPacket fails if remote ethaddr is unknown. In this case an
    //arp-request is send out first and beginPacket succeeds as soon
    //the arp-response is received.
}
while (!success);

success = udp.println("hello world from arduino");

Serial.print("bytes written: ");
Serial.println(success);

success = udp.endPacket();

Serial.print("endPacket: ");
Serial.println(success ? "success" : "failed");

udp.stop();
//restart with new connection to receive packets from other clients
Serial.print("restart connection: ");
Serial.println (udp.begin(5000) ? "success" : "failed");
}
}

```

Codi per a Arduino UdpServer01b



Arduino i Raspberry Pi

Ethernet – Enviament de datagrames UDP

```
$ ./udpOn.py 192.168.1.108
```

```
#!/usr/bin/python

import socket
import sys

szServer = str(sys.argv[1])
nUdpPort = 5000

client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
client_socket.sendto("1", (szServer, nUdpPort))
client_socket.close()
```

```
$ ./udpOff.py 192.168.1.108
```

```
#!/usr/bin/python

import socket
import sys

szServer = str(sys.argv[1])
nUdpPort = 5000

client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
client_socket.sendto("0", (szServer, nUdpPort))
client_socket.close()
```

Codis font dels codis d'enviament de datagrames UDP en Python



Arduino i Raspberry Pi

ESP8266

Programa per a carregar microprogramari a l'ESP8266
Binari amb ordres AT per a l'ESP8266

```
jordi@debianJB:~/Documents/electronics.cat/presentacions/20161129_telecosCat/codis/ESP8266$ python esptool.py  
--port /dev/ttyUSB0 write_flash 0x000000 v0.9.5.2\_ AT_Firmware.bin  
Connecting...  
Erasing flash...  
Wrote 520192 bytes at 0x00000000 in 50.0 seconds (83.2 kbit/s)...  
  
Leaving...  
jordi@debianJB:~/Documents/electronics.cat/presentacions/20161129_telecosCat/codis/ESP8266$
```





Arduino i Raspberry Pi

ESP8266

The screenshot displays the ESPlorer IDE interface. On the left, the 'Basic AT commands' section includes buttons for AT, RST, GMR, GSLP, ATE0, ATE1, and UPD. Below this, the 'Common WiFi commands' section features buttons for CWMODE=? (Get available...), CWMODE=1 Station, CWMODE=? (Get current m...), CWMODE=2 softAP, CWLAP - Get AP list, CWMODE=3 softAP + Stati..., 0 - Enable ..., 1 - Set ..., and AT+CWDHCP DHCP control. The 'WiFi Station' section includes buttons for CWJAP? - Connection info, CWQAP - Disconnect fr..., CIPSTAMAC? Get MAC, CIPSTAMAC= Set MAC S..., and CIPSTA? Get Station IP. On the right, the terminal window shows the output of the AT command: 'OK', 'Ai-Thinker Technology Co. Ltd.', 'ready', 'AT+CWMODE=1', 'OK', 'AT+CWLAP', and a list of available WiFi networks with their SSIDs, signal strengths, and MAC addresses.

Programa ESPlorer

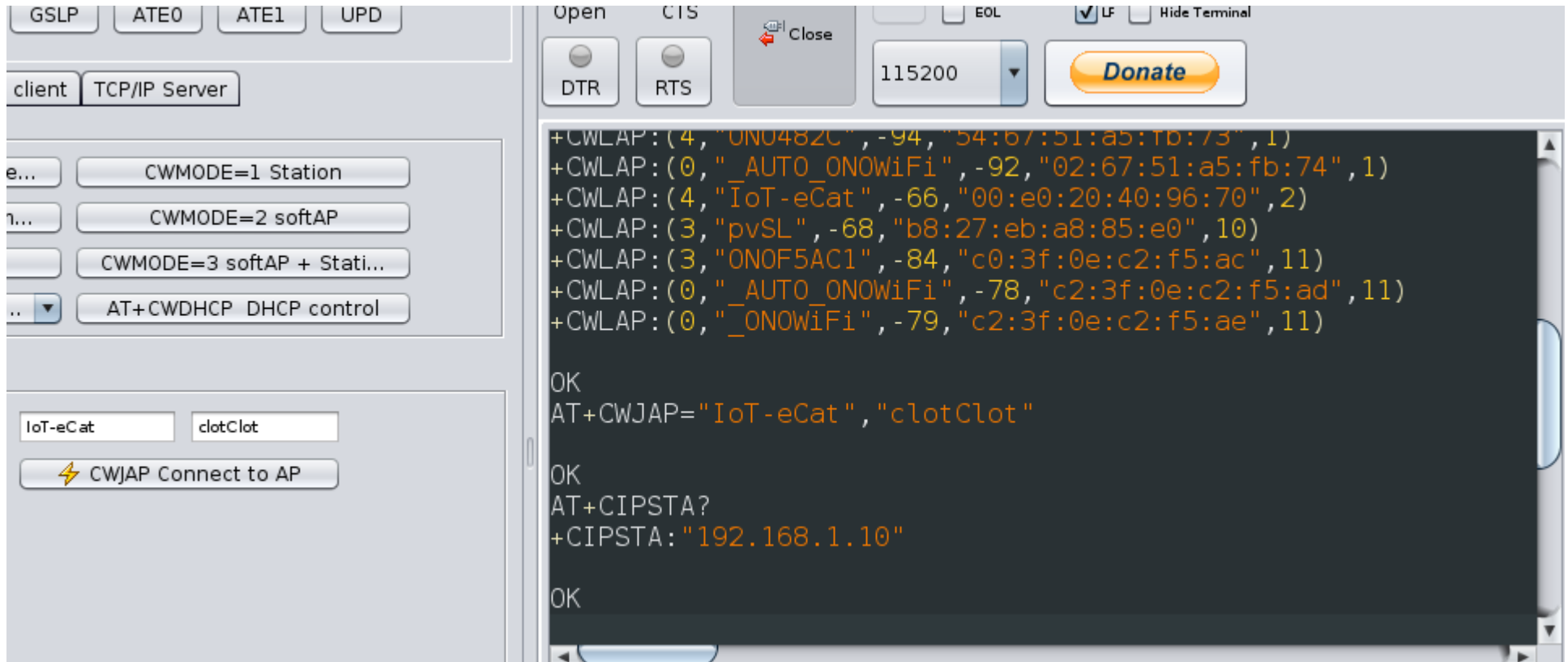
\$ java -jar ESPlorer.jar





Arduino i Raspberry Pi

ESP8266



Llistat d'ordres AT





Arduino i Raspberry Pi

ESP8266

Microprogramari `nodemcu_integer_0.9.6-dev_20150704.bin`

```
jordi@debianJB:~/Documents/electronics.cat/presentacions/20161129_telecosCat/ESP8266$  
./esptool.py --port /dev/ttyUSB0 write_flash 0x000000 nodemcu_integer_0.9.6-dev_20150704.bin  
Connecting...  
Erasing flash...  
Wrote 450560 bytes at 0x00000000 in 43.9 seconds (82.1 kbit/s)...  
Leaving...  
jordi@debianJB:~/Documents/electronics.cat/presentacions/20161129_telecosCat/ESP8266$
```





Arduino i Raspberry Pi

ESP8266

```

/dev/ttyUSB0
[Open] [CTS] [Close] [Refresh] [AutoScroll] [CR] [Hide Editor]
[DTR] [RTS] [9600] [Donate]
NodeMCU 0.9.6 build 20150704 powered by Lua
lua: cannot open init.lua
>
=wifi.getmode()
=wifi.getmode()
1
>
wifi.setmode(wifi.STATION)
wifi.setmode(wifi.STATION)
>
wifi.sta.config("IoT-eCat","clotClot")
wifi.sta.config("IoT-eCat","clotClot")
>
=wifi.sta.getip()
=wifi.sta.getip()
192.168.1.11    255.255.255.0    192.168.1.1
>

```





Arduino i Raspberry Pi

ESP8266

```
> w([==[end]==]),
> w([==[end]==]);
> file.close();
> dofile("clientWeb.lua");
nil
> file.remove("closeTcp.lua");
> file.open("closeTcp.lua","w+");
> w = file.writeline
> w([==[srv.close(srv)==]);
> w([==[srv=nil]==]);
> file.close();
> dofile("closeTcp.lua");
>
dofile("clientWeb.lua")
dofile("clientWeb.lua")
nil
```



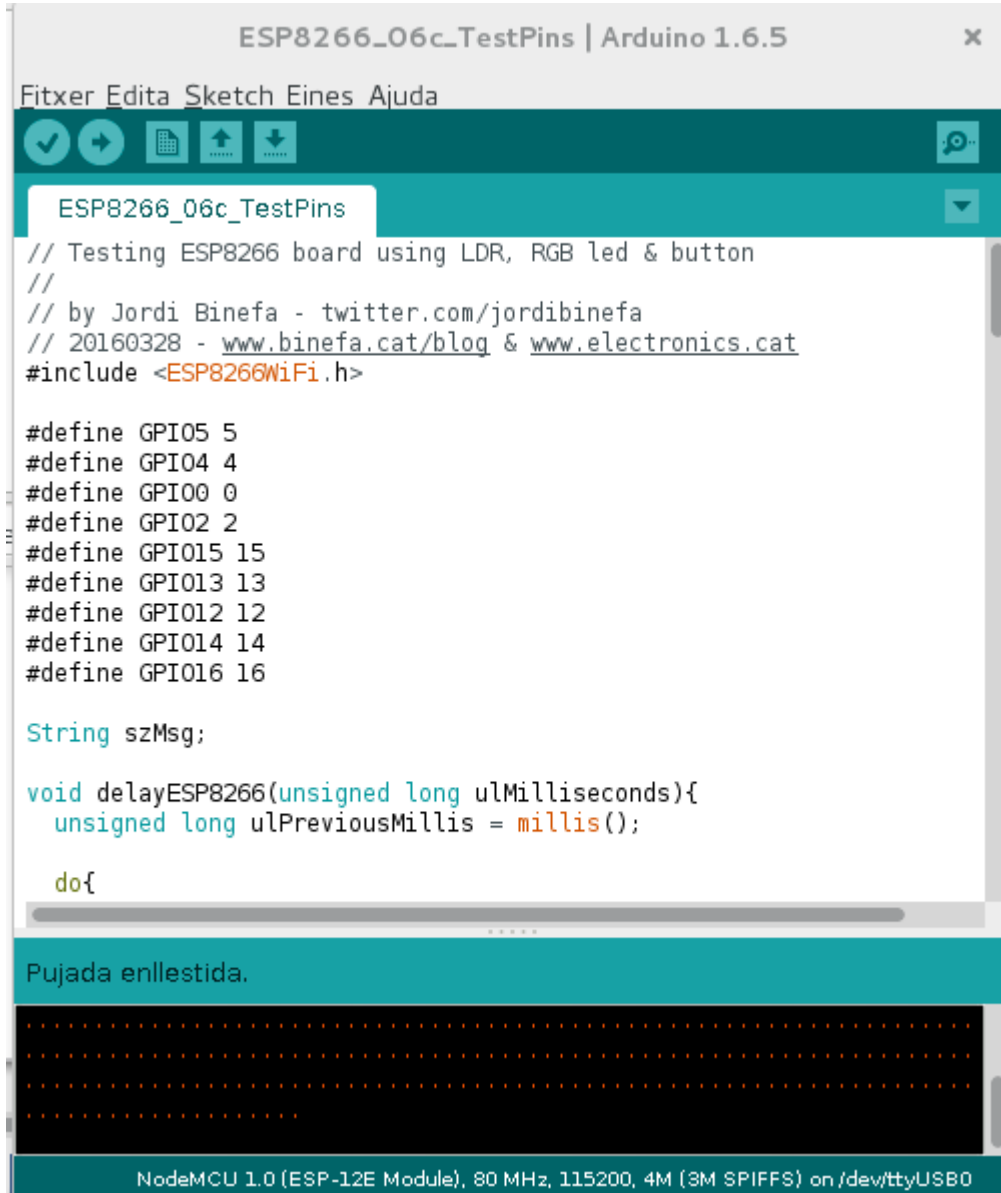
Client web en Lua

Per tancar la connexió TCP en Lua



Arduino i Raspberry Pi

ESP8266



```

ESP8266_06c_TestPins | Arduino 1.6.5
Fitxer Edita Sketch Eines Ajuda

ESP8266_06c_TestPins
// Testing ESP8266 board using LDR, RGB led & button
//
// by Jordi Binefa - twitter.com/jordibinefa
// 20160328 - www.binefa.cat/blog & www.electronics.cat
#include <ESP8266WiFi.h>

#define GPIO5 5
#define GPIO4 4
#define GPIO0 0
#define GPIO2 2
#define GPIO15 15
#define GPIO13 13
#define GPIO12 12
#define GPIO14 14
#define GPIO16 16

String szMsg;

void delayESP8266(unsigned long ulMilliseconds){
  unsigned long ulPreviousMillis = millis();

  do{

```

Pujada enllestida.

NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on /dev/ttyUSB0

Microprogramari de verificació





Arduino i Raspberry Pi

ESP8266

/dev/ttyUSB0

Envia

UDPClientMAC02 | Arduino 1.6.5

Fitxer Edita Sketch Eines Ajuda

UDPClientMAC02

```

Serial.println();
Serial.println();
Serial.print("Connecting to ");
Serial.println(ssid);

WiFi.begin(ssid, password);

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

Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

// Udp.beginPacket("192.168.1.105", 888);
// Udp.beginPacket("192.168.1.13", 6000);
// Udp.beginPacket("192.168.1.52", 6000);
Udp.beginPacket("192.168.1.12", 6000);
        
```

UDPdIlg

Text :

IP servidor :

Port Tx : <-> Port Rx :

Tramet

```

00004) 5c:cf:7f:c0:84:07
00005) 5c:cf:7f:c0:84:07
00006) 5c:cf:7f:c0:84:07
00007) 5c:cf:7f:c0:84:07
00008) 5c:cf:7f:c0:84:07
00009) 5c:cf:7f:c0:84:07
00010) 5c:cf:7f:c0:84:07
00011) 5c:cf:7f:c0:84:07
00012) 5c:cf:7f:c0:84:07
00013) 5c:cf:7f:c0:84:07
00014) 5c:cf:7f:c0:84:07
00015) 5c:cf:7f:c0:84:07
00016) 5c:cf:7f:c0:84:07
        
```





Arduino i Raspberry Pi

ESP8266

A partir de la pàgina 18 del document [ethernetTcpUdp_00.pdf](#)

Torn de preguntes ...



... i sessió pràctica.



Arduino i Raspberry Pi

Presentació descarregable a : <http://binefa.cat/blog>

Correu electrònic de contacte : jordibinefa@electronics.cat

twitter



<https://twitter.com/JordiBinefa>

<https://twitter.com/electronicscat>



<http://es.linkedin.com/pub/jordi-binefa/13/717/90b>

Plaques aviat disponibles a :

<http://www.electronics.cat>

<http://www.makeit.cat>

Moltes gràcies per la vostra atenció