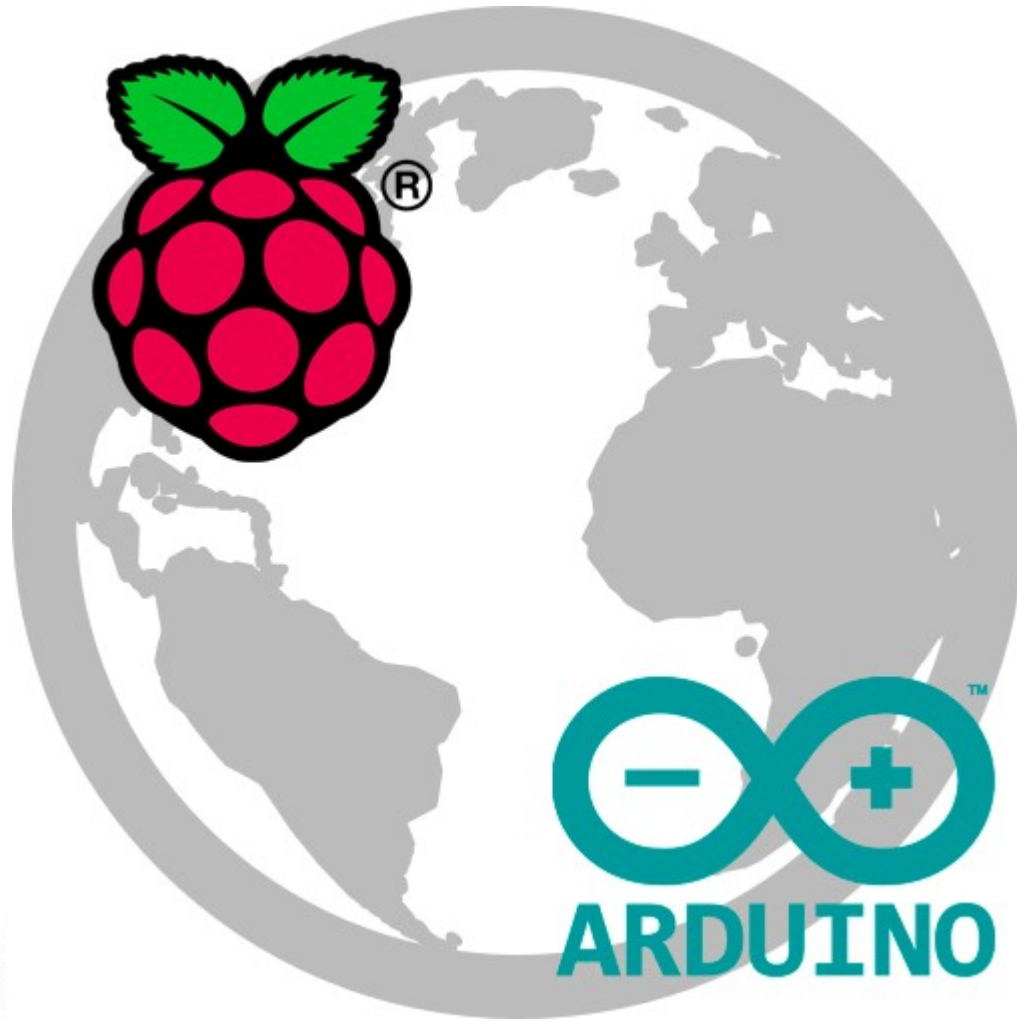


Arduino i Raspberry Pi



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**
- * 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 Joystick + Servos

Pràctica ultrasons

Biblioteca ecat.h

Pràctica pantalles LCD 16x2

Pràctica SoftwareSerial

Pràctica amb Bluetooth . Variació emprant SoftwareSerial.

Pràctica amb RS485

Pràctica I2C (PCF8574)

Pràctica SPI

Pràctica Ethernet

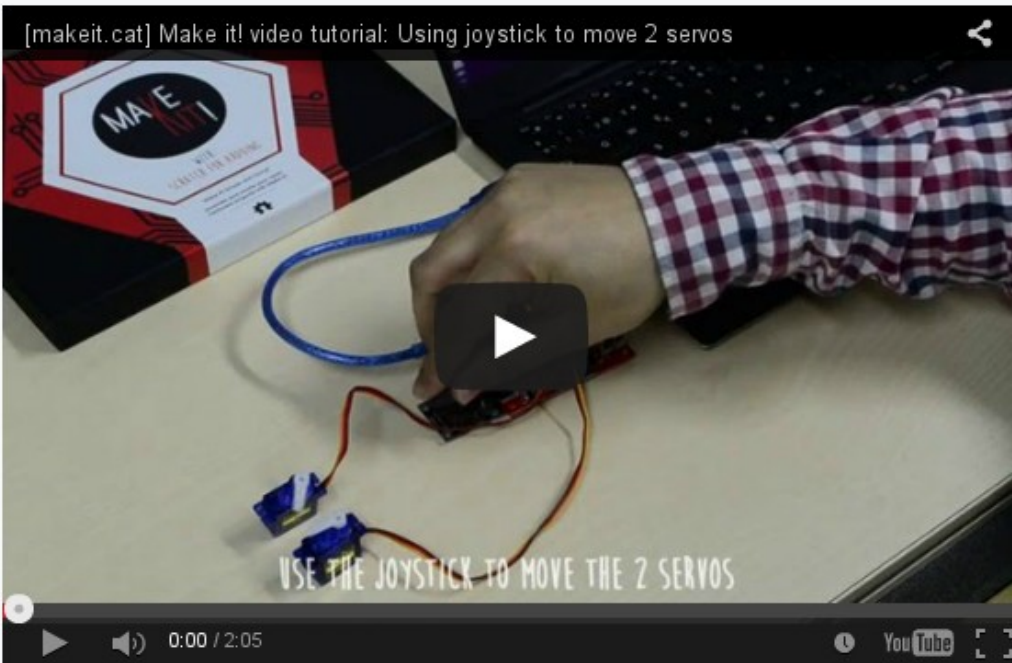
Pràctica ESP8266

Exercici emprant ultrasons , servo i sensor IR



Arduino i Raspberry Pi

Joystick + Servos (S4A)



Tutorial *Make it!*: Utilitzant un joystick per moure 2 servos

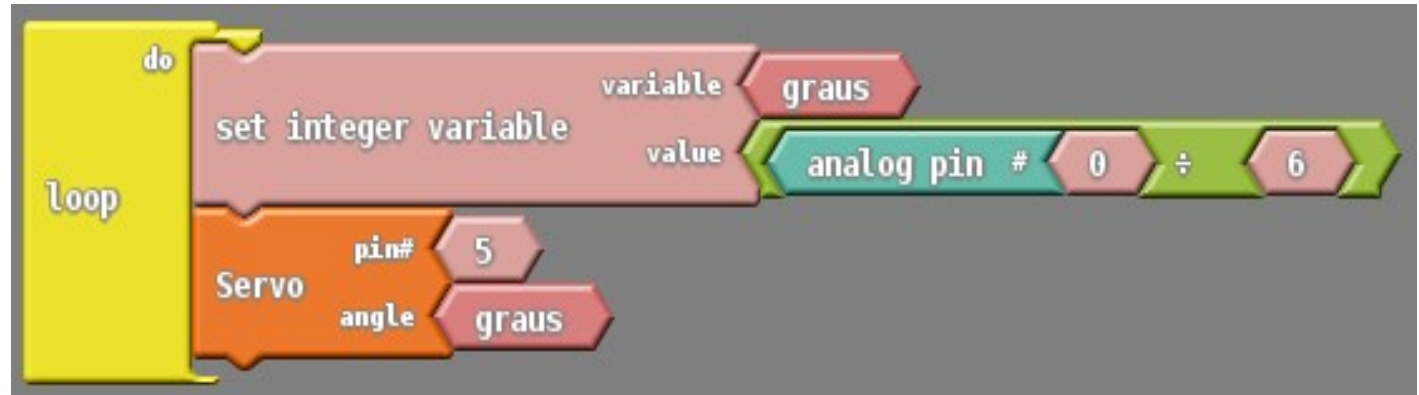
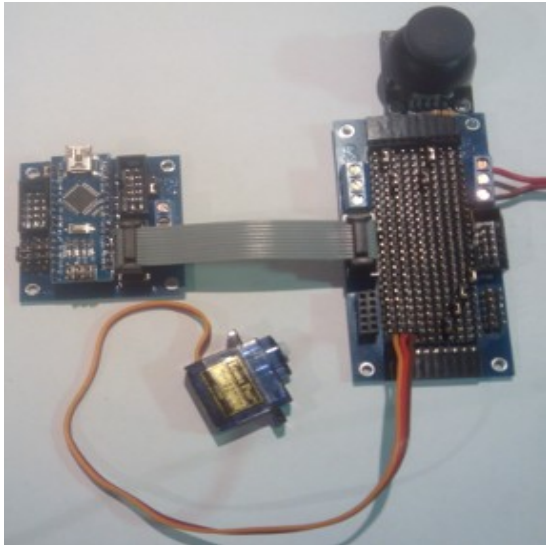
Com moure 2 servos utilitzant un joystick, el *Make it!* i el programa Scratch for Arduino.

<http://makeit.cat/cat/>



Arduino i Raspberry Pi

Joystick + Servos (Ardublock)

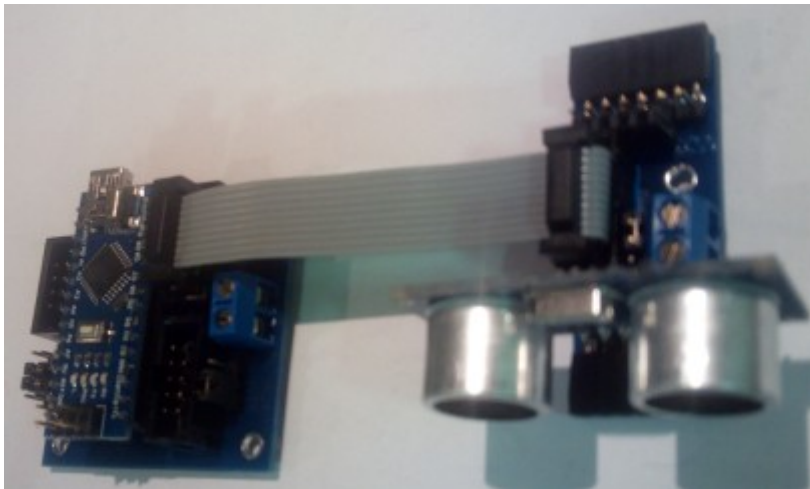
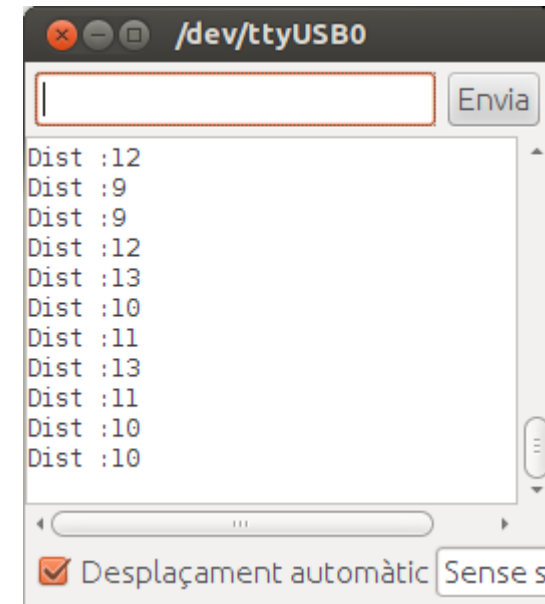


<http://www.binefa.cat/php/arduino/ardublock/pr04joystickServo.abp>



Arduino i Raspberry Pi

pr05ultrasons.abp Mesura ultrasònica



<http://www.binefa.cat/php/arduino/ardublock/pr05ultrasons.abp>

<http://www.binefa.cat/php/doc/pr002/>



Arduino i Raspberry Pi

Exercici Ultrasons + Servo

Emprant l'ArduBlock feu que el servo es mogui en funció de la distància llegida pels ultrasons.

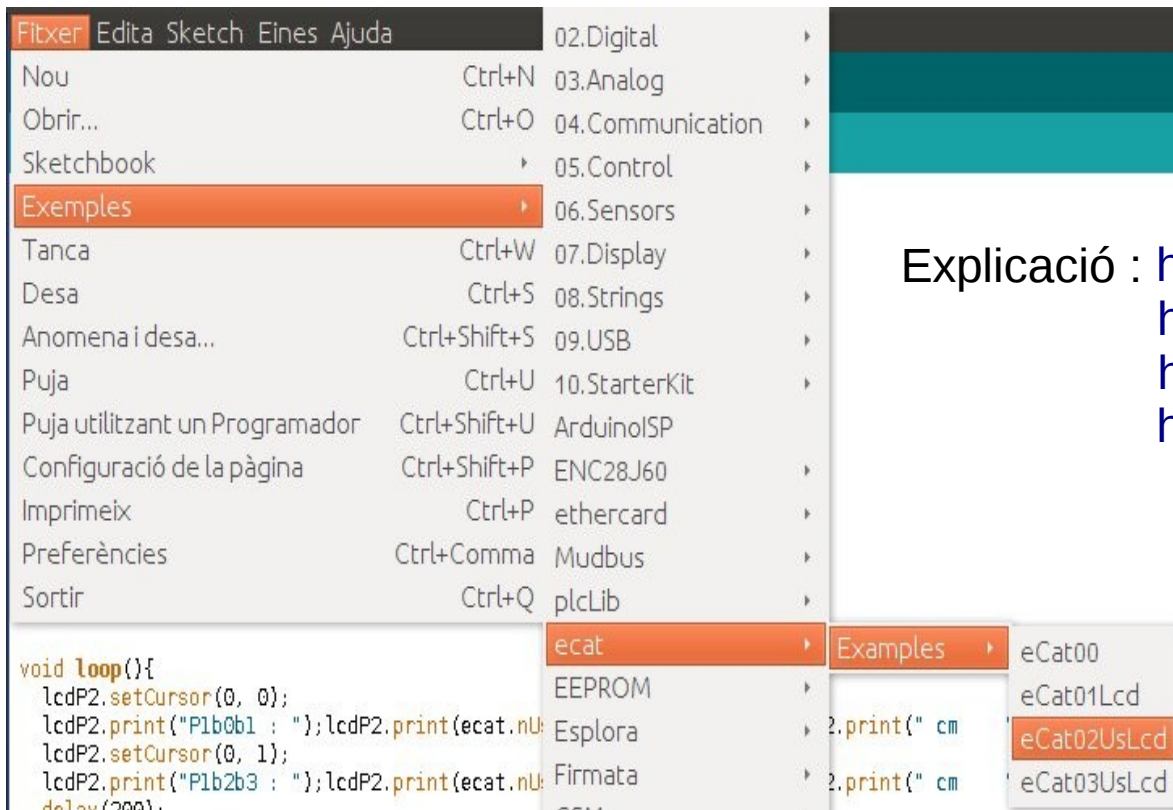
Haureu de tenir en compte que el servo accepta un valor mínim i un màxim.



Arduino i Raspberry Pi

Biblioteca **ecat.h** per a l'IDE d'Arduino

<http://www.electronics.cat/doc/arduinoLib/ecat.zip>



Explicació : <http://www.binefa.cat/php/doc/lib-eCat01/>
<http://www.binefa.cat/php/doc/lib-eCat02/>
<http://www.binefa.cat/php/doc/lib-eCat03/>
<http://binefa.cat/blog/?p=140>



Arduino i Raspberry Pi

Arduino Nano + LCD + Joystick + Servo

```
#include <LiquidCrystal.h>
#include <Servo.h>

LiquidCrystal lcd(16, 17, 4, 5, 6, 7); // P1
//LiquidCrystal lcd(8,9,10,11,12,13); // P2
Servo servo1; Servo servo2;

const int buttonPin = 8;
//const int buttonPin = 16;
int analogPinX = 5, analogPinY = 4;
//int analogPinX = 0, analogPinY = 1;
int buttonState = 0;

void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  lcd.print("electronics.cat");
  pinMode(buttonPin, INPUT);
  servo1.attach(10);
  servo2.attach(11);
  servo1.write(90);
  servo2.write(90);
}
```

```
unsigned long int nAnalogValue2Degrees(unsigned long int nAnalogValue){
  return (nAnalogValue * 180) / 1023;
}

void loop() {
  buttonState = digitalRead(buttonPin);
  unsigned long int degX = nAnalogValue2Degrees(analogRead(analogPinX));
  unsigned long int degY = nAnalogValue2Degrees(analogRead(analogPinY));
  // set the cursor to column 0, line 1 (note: line 1 is the second row, since counting begins from 0)
  lcd.setCursor(0, 1);
  if (buttonState == HIGH) {
    lcd.print("F ");
  } else {
    lcd.print("N ");
  }
  lcd.print("X:"); lcd.print(degX); lcd.print("d Y:"); lcd.print(degY); lcd.print("d ");
  servo1.write(degX);
  servo2.write(degY);
}
```



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

<http://www.binefa.cat/php/arduino/ardublock/lcdJsServo/lcdJsServo.ino>



Arduino i Raspberry Pi

Biblioteca **ecat.h** per a l'IDE d'Arduino

<http://www.electronics.cat/doc/arduinoLib/ecat.zip>

Explicació : <http://www.binefa.cat/php/doc/lib-eCat01/>
<http://www.binefa.cat/php/doc/lib-eCat02/>
<http://www.binefa.cat/php/doc/lib-eCat03/>
<http://binefa.cat/blog/?p=140>

```

1  #include <ecat.h>
2
3  #define ROBOT_ATURAT      0
4  #define ROBOT_ENDAVANT    1
5  #define ROBOT_ENDARRERA   2
6  #define ROBOT_DRETA       3
7  #define ROBOT_ESQUERRA    4
8
9  String szMissatge;
10 Ecat ecat;
11 boolean bConnectat;
12 int nG,nA,nB,nEstatActual;
13
14 void setup(){
15     ecat.setupNibbleMode(NIBBLE_H_P1,OUTPUT);
16     ecat.vUltrasonicSensorP1b0b1_init();
17     pinMode(ecat.nPinP1B2,INPUT);
18     pinMode(ecat.nPinP1B3,INPUT);
19     pinMode(ecat.nPinP2B7,OUTPUT);
20     pinMode(ecat.nPinP2B6,INPUT);
21     pinMode(ecat.nPinP2B5,INPUT);
22     pinMode(ecat.nPinP2B4,INPUT);
23     ecat.setupNibbleMode(NIBBLE_L_P2,INPUT);
24     Serial.begin(9600);
25     nG = 150;
26     nA = nB = 30;
27     bConnectat = false;
28     nEstatActual = ROBOT_ATURAT;
29 }
30

```

<http://electronics.cat/downloads/code/robot08.ino>



Arduino i Raspberry Pi

Arduino Nano + LCD + Joystick + Servo

```
#include <LiquidCrystal.h>
#include <Servo.h>

LiquidCrystal lcd(16, 17, 4,5, 6, 7); // P1
//LiquidCrystal lcd(8,9,10,11,12,13); // P2
Servo servol; Servo servo2;

const int buttonPin = 8;
//const int buttonPin = 16;
int analogPinX = 5,analogPinY = 4;
//int analogPinX = 0,analogPinY = 1;
int buttonState = 0;

void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  lcd.print("electronics.cat");
  pinMode(buttonPin, INPUT);
  servol.attach(10);
  servo2.attach(11);
  servol.write(90);
  servo2.write(90);
}
```

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

<http://www.binefa.cat/php/arduino/ardublock/lcdJsServo/lcdJsServo.ino>



Arduino i Raspberry Pi

Arduino Nano + LCD + Joystick + Servo

```

unsigned long int nAnalogValue2Degrees(unsigned long int nAnalogValue){
    return (nAnalogValue * 180) / 1023;
}

void loop() {
    buttonState = digitalRead(buttonPin);
    unsigned long int degX = nAnalogValue2Degrees(analogRead(analogPinX));
    unsigned long int degY = nAnalogValue2Degrees(analogRead(analogPinY));
    // set the cursor to column 0, line 1 (note: line 1 is the second row, since counting begins with 0):
    lcd.setCursor(0, 1);
    if (buttonState == HIGH) {
        lcd.print("F ");
    }else{
        lcd.print("N ");
    }
    lcd.print("X:");lcd.print(degX); lcd.print("d  Y:");lcd.print(degY);lcd.print("d      ");
    servo1.write(degX);
    servo2.write(degY);
}

```



Arduino i Raspberry Pi

SoftwareSerial

Carregar el programa serialPass.ino



Arduino i Raspberry Pi

Preparant la comunicació BlueTooth

```
#include <ecat.h>

Ecat ecat;
boolean bP2B0;

void setup(){
  pinMode(ecat.nPinP2B7, OUTPUT);
  pinMode(ecat.nPinP2B0, INPUT);
  Serial.begin(9600);
  bP2B0 = digitalRead(ecat.nPinP2B0);
}

void loop(){
  String szMsg;

  while(Serial.available()){
    delay(3);
    char c = Serial.read();
    szMsg += c;
  }
  if(szMsg == "n"){
    digitalWrite(ecat.nPinP2B7, HIGH);
  }
  if(szMsg == "f"){
    digitalWrite(ecat.nPinP2B7, LOW);
  }
  if(bP2B0 != digitalRead(ecat.nPinP2B0)){
    bP2B0 = digitalRead(ecat.nPinP2B0);
    if(bP2B0){
      Serial.println("P2B0 HIGH");
    }else{
      Serial.println("P2B0 LOW");
    }
  }
  szMsg = "";
}
```



<http://electronics.cat/doc/hc06/bluetooth04.ino>



Arduino i Raspberry Pi

Preparant la comunicació Bluetooth

```
#include <ecat.h>

Ecat ecat;
boolean bP2B0;

void setup(){
  pinMode(ecat.nPinP2B7, OUTPUT);
  pinMode(ecat.nPinP2B0, INPUT);
  Serial.begin(9600);
  bP2B0 = digitalRead(ecat.nPinP2B0);
}
```


Arduino i Raspberry Pi

Preparant la comunicació Bluetooth

```
void loop(){
  String szMsg;

  while(Serial.available()){
    delay(3);
    char c = Serial.read();
    szMsg += c;
  }
  if(szMsg == "n"){
    digitalWrite(ecat.nPinP2B7,HIGH);
  }
  if(szMsg == "f"){
    digitalWrite(ecat.nPinP2B7,LOW);
  }
  if(bP2B0 != digitalRead(ecat.nPinP2B0)){
    bP2B0 = digitalRead(ecat.nPinP2B0);
    if(bP2B0){
      Serial.println("P2B0 HIGH");
    }else{
      Serial.println("P2B0 LOW");
    }
  }
  szMsg = "";
}
```

<http://electronics.cat/doc/hc06/bluetooth04.ino>

Arduino i Raspberry Pi

Preparant la comunicació BlueTooth

Verifiquen el funcionament del programa bluetooth04.ino interactuant amb qualsevol dels programes de comunicacions que coneixeu



<http://electronics.cat/doc/hc06/bluetooth04.ino>



Arduino i Raspberry Pi

App Inventor

MIT App Inventor | Explore MIT App Inventor - Mozilla Firefox

MIT App Inventor | E... x

appinventor.mit.edu/explore/ Cerca

MIT App Inventor About News & Stories Resources Create!

Check out the new App Inventor Gallery! Google Custom Search

MIT App Inventor noted in US News & World Report.

MIT students use App Inventor to create UMATI, a crowdsourced app to track bus routes in Nairobi

Recent News

- MIT App Inventor at ISTE 2015
- College Board Announces Endorsement of Advanced Placement Mobile CSP Course
- MIT App Inventor Classic to MIT App Inventor 2 Converter Released
- MIT App Inventor usage triples over the 2014-2015 academic year

More...

edX

Register for a free online edX App Inventor course!

Tweets

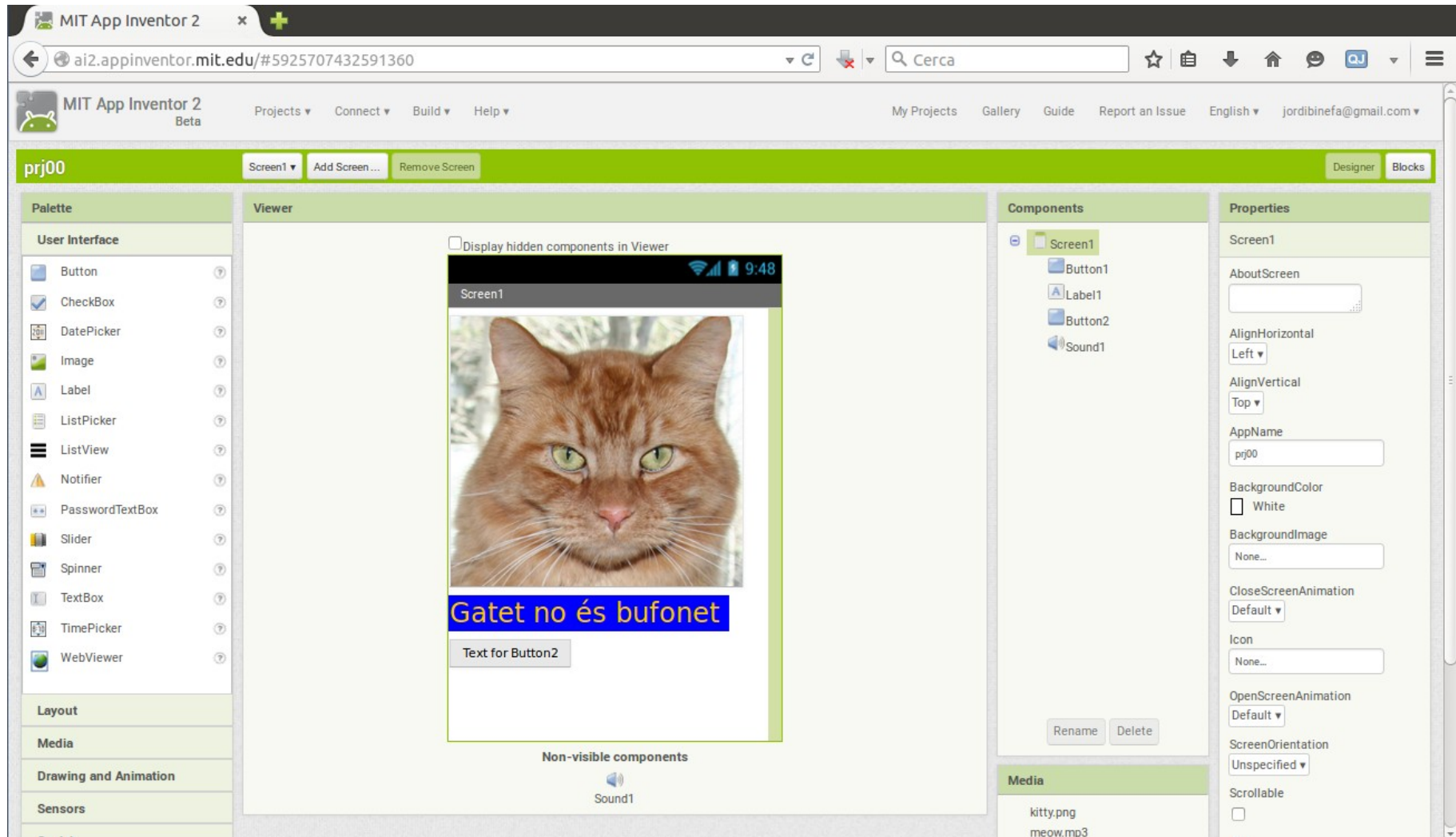
MIT App Inventor @MITAppInventor 6h

<http://appinventor.mit.edu/>



Arduino i Raspberry Pi

App Inventor

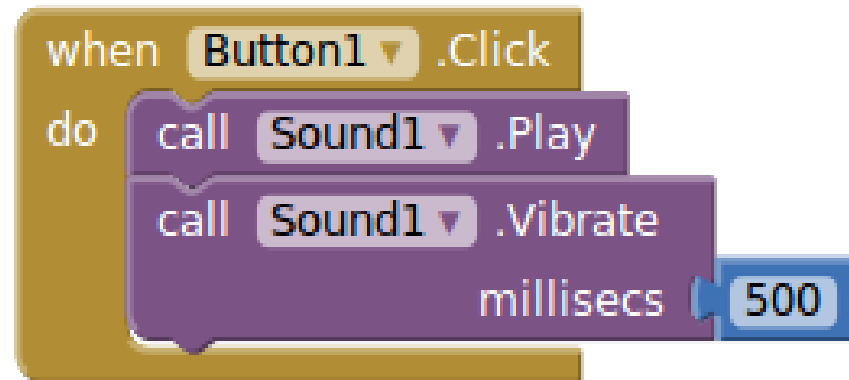


<http://binefa.cat/php/appinventor/prj00.aia>



Arduino i Raspberry Pi

App Inventor



<http://binefa.cat/php/appinventor/prj00.aia>



Arduino i Raspberry Pi

App Inventor

Un cop us heu validat a l'App Inventor, importeu l'arxiu prj00.aia -Projects / Import projects (.aia) from my computer- i genereu l'arxiu prj00.apk -Build / App (save .apk to my computer)-.

Un cop generat l'arxiu .apk el passeu al vostre mòbil Android fent servir el cable USB, trametent-lo mitjançant correu electrònic o anant a l'adreça de sota mitjançant el vostre navegador.

A l'hora de fer la instal·lació el telèfon us avisarà de que la font del programa no és l'estàndard. Temporalment, doneu al vostre mòbil permisos per a fer una instal·lació des de font desconeguda.

Verifiquen el seu funcionament.

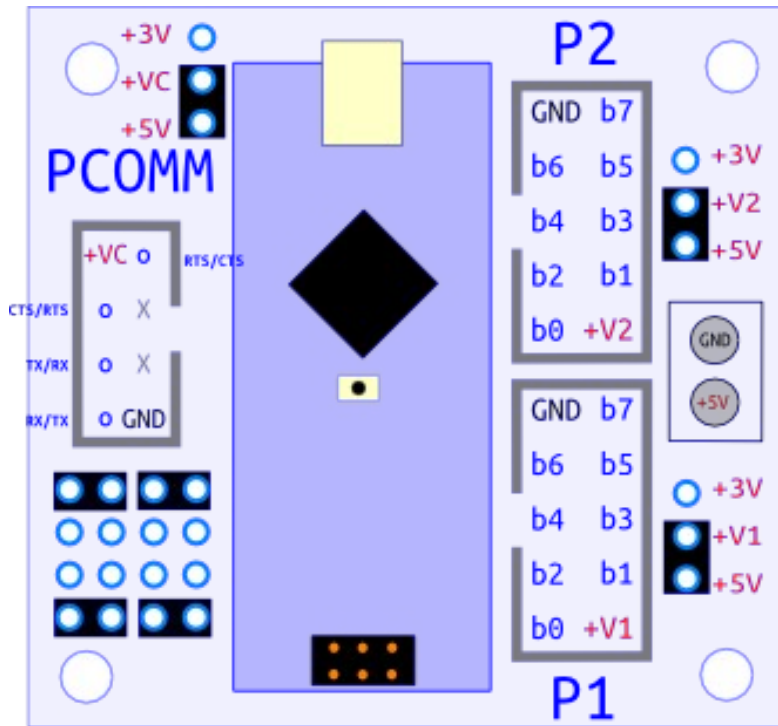
<http://binefa.cat/php/appinventor/prj00.aia>

<http://binefa.cat/php/appinventor/prj00.apk>



Arduino i Raspberry Pi

Comunicació mitjançant Bluetooth



- Carregueu bluetooth04.ino a l'Arduino Nano
- Desconnecteu cable mini-USB de l'Arduino Nano
- Alimenteu separatament la placa nano-eCat (cal tornavís)
- Assegureu-vos de la posició dels ponts (jumpers) de comunicació
- Preneu nota del número identificador de Bluetooth
- Connecteu placa set0525 a la nano-eCat
- Vinculeu el dispositiu BlueTooth al vostre mòbil Android (contrasenya : 1234)
- Proveu el fi  direccional

<http://electronics.cat/doc/hc06/bluetooth04.ino>

http://electronics.cat/doc/hc06/HC06_04b.aia

http://electronics.cat/doc/hc06/HC06_04b.apk



Arduino i Raspberry Pi

Com trobar pel terminal el número ID del
Bluetooth

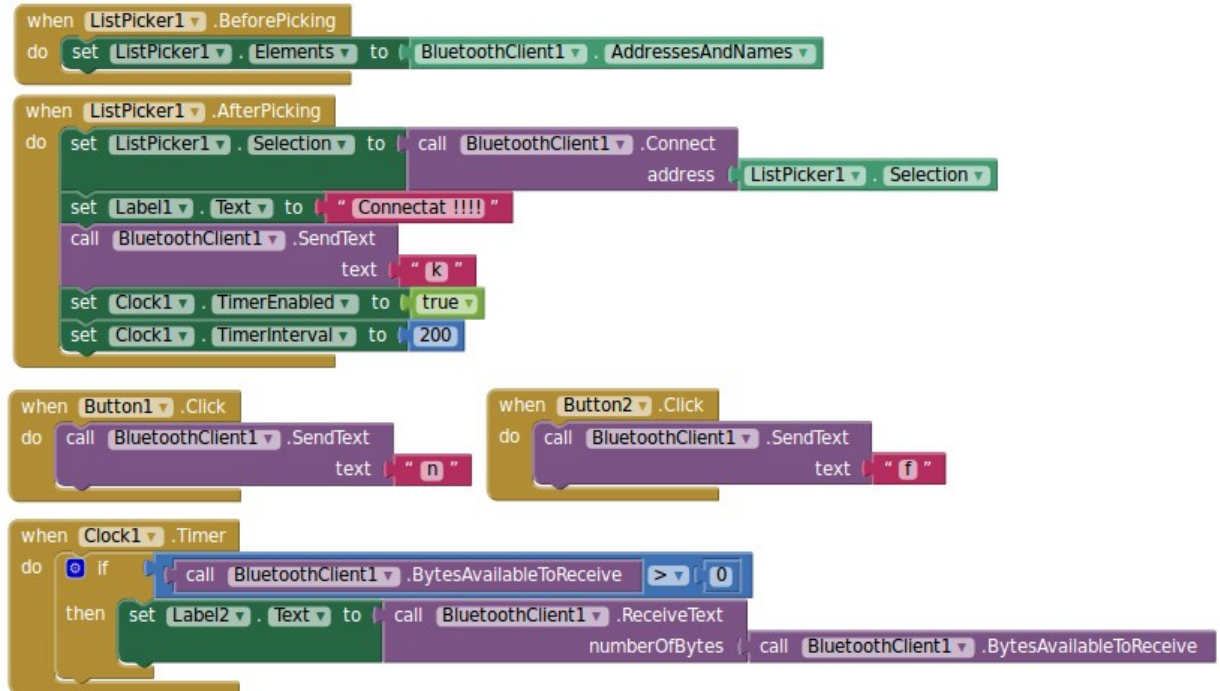
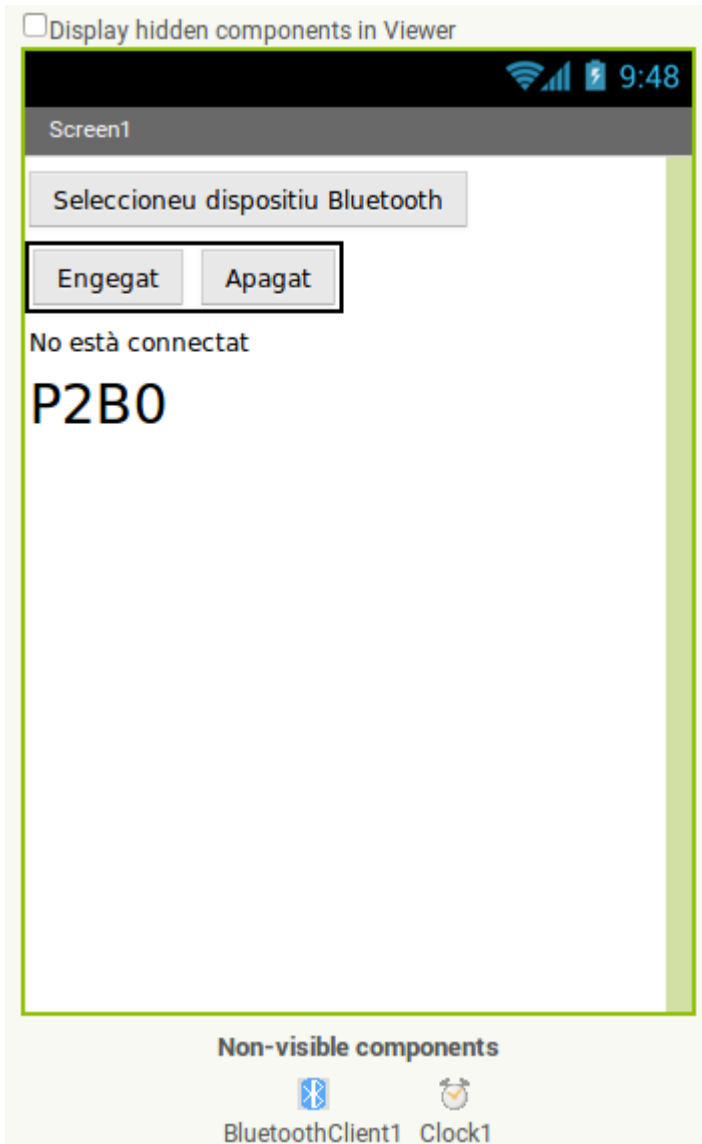
```
$ hcitool scan  
Scanning ...  
98:D3:31:30:2C:0D HC-06
```





Arduino i Raspberry Pi

App Inventor - Comunicació mitjançant Bluetooth





Arduino i Raspberry Pi

App Inventor - Comunicació mitjançant Bluetooth

```

when ListPicker1.BeforePicking
do
  set ListPicker1.Elements to BluetoothClient1.AddressesAndNames

when ListPicker1.AfterPicking
do
  set ListPicker1.Selection to call BluetoothClient1.Connect
                                address ListPicker1.Selection
  set Label1.Text to "Connectat !!!! "
  call BluetoothClient1.SendText
                                text "k"
  set Clock1.TimerEnabled to true
  set Clock1.TimerInterval to 200

when Button1.Click
do
  call BluetoothClient1.SendText
                                text "n"

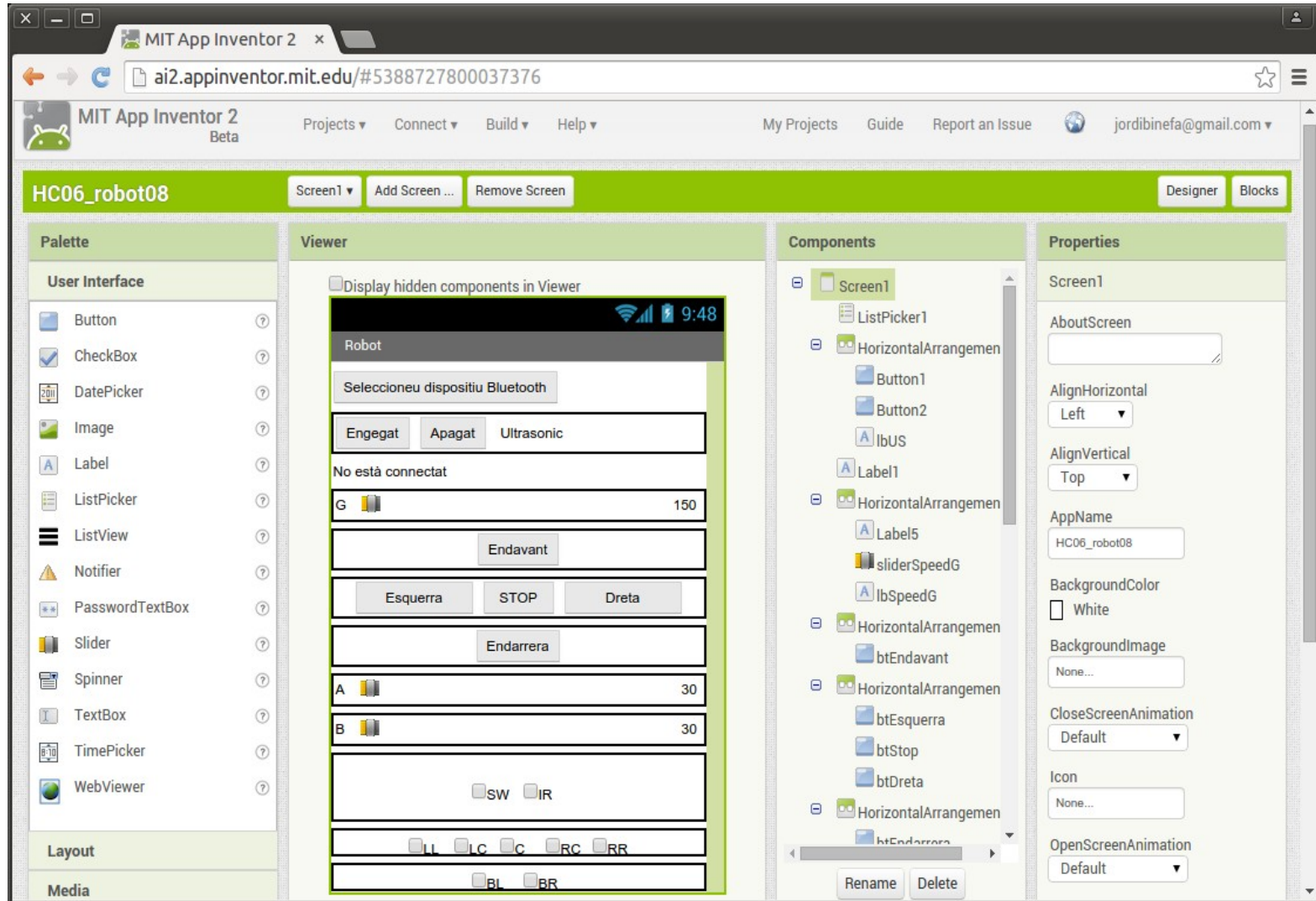
when Button2.Click
do
  call BluetoothClient1.SendText
                                text "f"

when Clock1.Timer
do
  if call BluetoothClient1.BytesAvailableToReceive > 0
  then
    set Label2.Text to call BluetoothClient1.ReceiveText
                                numberOfBytes call BluetoothClient1.BytesAvailableToReceive
  
```



Arduino i Raspberry Pi

Programar al mòbil Android – App Inventor

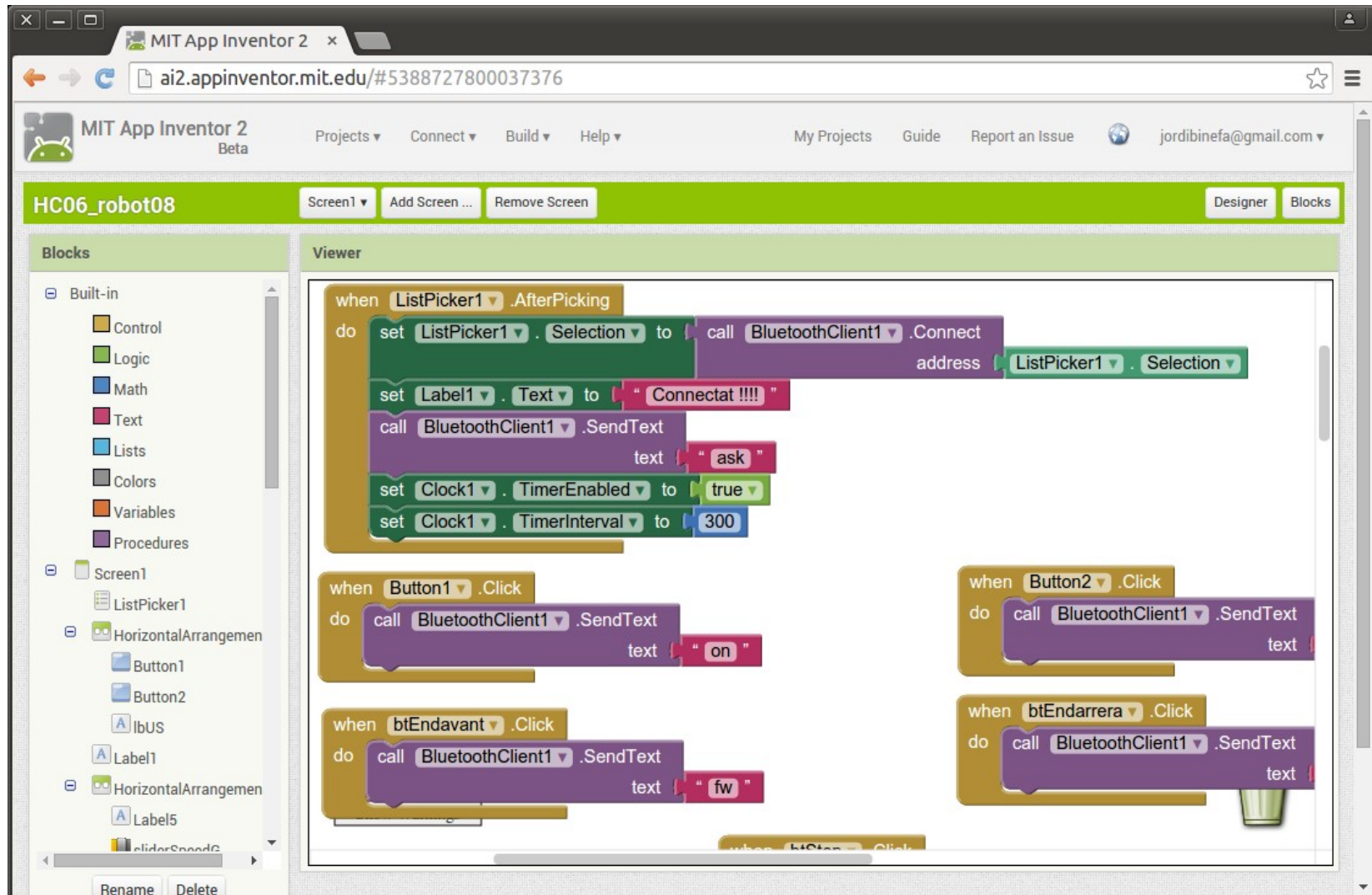


<http://appinventor.mit.edu/>



Arduino i Raspberry Pi

Programar al mòbil Android – App Inventor



http://electronics.cat/downloads/code/HC06_robot08.aia

<http://appinventor.mit.edu/>



Arduino i Raspberry Pi

Arduino i App Inventor - Comunicació

```
147 void vManageMsg(){
148     vSliders();
149     if(szMissatge == "on"){
150         //digitalWrite(ecat.nPinP2B7,HIGH);
151         analogWrite(ecat.nPinP2B7,nG);
152     }
153     if(szMissatge == "off"){
154         digitalWrite(ecat.nPinP2B7,LOW);
155     }
156     if(szMissatge == "ask"){
157         bConnectat = true;
158         vInformaSensors();
159     }
160     if(szMissatge == "fw"){
161         vRobotEndavant();
162     }
163     if(szMissatge == "bw"){
164         vRobotEndarrera();
165     }
166     if(szMissatge == "st"){
167         vRobotAturat();
168     }
169     if(szMissatge == "le"){
170         vRobotEsquerra();
171     }
172     if(szMissatge == "ri"){
173         vRobotDreta();
174     }
175 }
```

<http://electronics.cat/downloads/code/robot08.ino>

http://electronics.cat/downloads/code/HC06_robot08.aia

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