

3.2.8. Giróscopo: Mover un robot

Vamos a realizar una actividad que utilice el Giróscopo de un móvil o tablet para controlar los movimientos de un robot.



Figura 3.2.8-1 Ejemplo de giróscopo

3.2.8.1. Programa para Android (código de Bloques en AppInventor)

- **Diseño**

- Utilizamos como base el programa de encender/apagar el led utilizado en la sección 3.2.4.1.

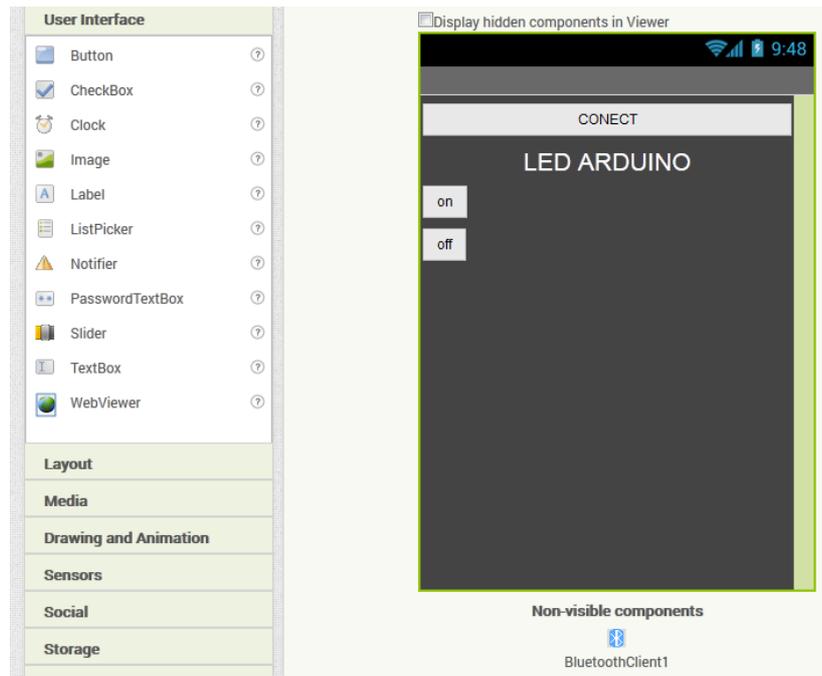


Figura 3.2.8-2 Diseño del que partimos

- Añadimos dos etiquetas (roll y pitch) y dos cuadros de texto para mostrar los valores del giróscopo.
- Añadimos el sensor OrientationSensor.

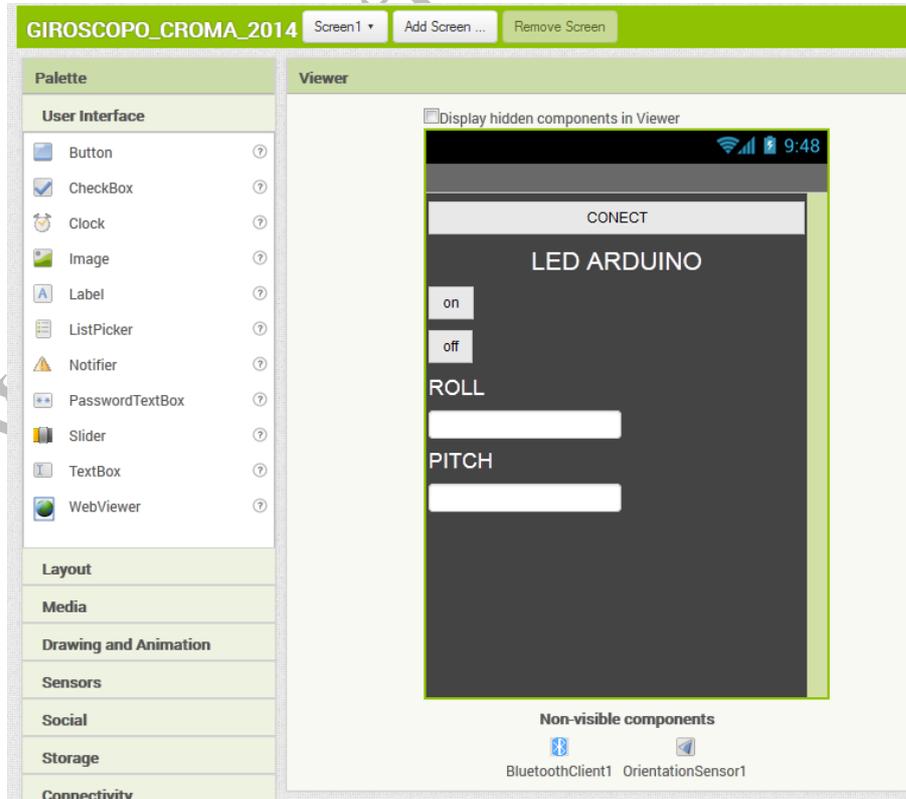


Figura 3.2.8-3 Diseño final de la aplicación de la actividad en AppInventor

- Programación

- Enviamos por el bluetooth los datos roll y pitch cuando estos cambien. Para ello, añadimos un evento del sensor de orientación.

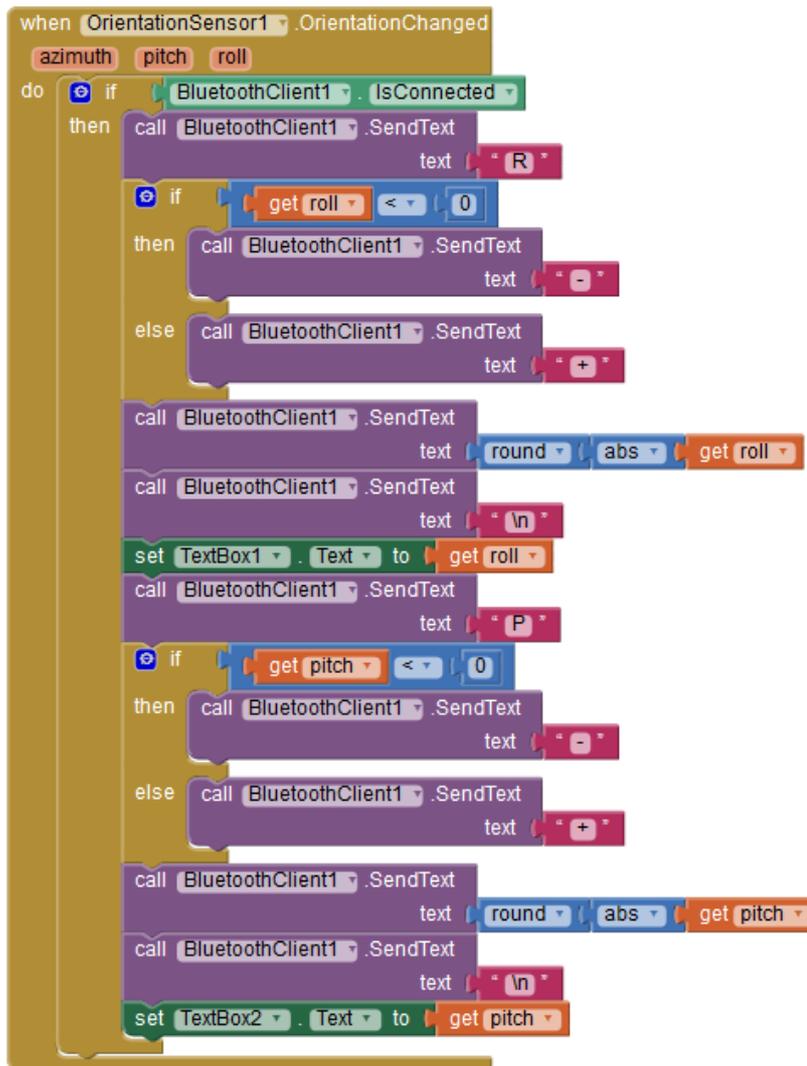


Figura 3.2.8-4 Programa en bloques en AppInventor de la actividad

3.2.8.2. Programa en Arduino (con el IDE de Arduino)

Utilizaremos un código basado en actividades anteriores, aunque incluyendo algunas modificaciones, como por ejemplo, la decodificación de la trama recibida:

R (+/-) dato, P (+/-) dato

- Si tenemos un valor de roll absoluto mayor que 10 grados giramos en el sentido que marca el signo de dicho valor.

- Si tenemos un valor de pitch absoluto mayor que 10 grados avanzamos o retrocedemos, según el signo.
- Si los valores son menores que 10° paramos.

Se propone modificar el código para que avance con mayor o menor velocidad según la inclinación que metamos.

```
#include <SoftwareSerial.h>
#include <Servo.h>
char comando;
int led = 13;
int int_roll;
int int_pitch;

//Parte Motores
Servo servo_6;
Servo servo_9;

void setup()
{
  Serial.begin(19200);
  pinMode(led, OUTPUT);
  servo_6.attach(6);
  servo_9.attach(9);
}

void loop() // run over and over
{
  comando=Serial.read();
  //Según el dato recibido
  switch (comando){
    case 'P':
      procesar_entrada_giroscopo_pitch();
      break;
    case 'R':
      procesar_entrada_giroscopo_roll();
      break;
    case 'H':
      digitalWrite(led, HIGH);
      break;
    case 'L':
      digitalWrite(led, LOW);
      break;
    //case '\-':
    // digitalWrite(led, HIGH);
    //break;
    default:
      break;
  }
  //para comprobacion roll
  if (int_roll>0)
    digitalWrite(led, HIGH);
  else
    digitalWrite(led, LOW);

  if (abs(int_roll)>10){
    if (int_roll>0)
      gira_izq(150);
    else
```

```

        gira_dch(150);
    }else {
        if (abs(int_pitch)>10){
            if (int_pitch>0)
                avanza(150);
            else
                retrocede(150);
        }else
            detente();
    }
}
void procesar_entrada_giroscopo_pitch()
{
    int signo_pitch;

    while(comando!='\n'){
        if(Serial.available())
        {
            comando=Serial.read();
            switch(comando){
                case '\n':
                    break;
                case '+':
                    signo_pitch=1;
                    break;
                case '-':
                    signo_pitch=-1;
                    break;
                default:
                    int_pitch=Serial.parseInt();
                    break;
            }
        }
    }
    int_pitch=int_pitch*signo_pitch;
}
void procesar_entrada_giroscopo_roll()
{
    int signo_roll;
    int cont=0;

    while(comando!='\n'){
        if(Serial.available())
        {
            comando=Serial.read();
            switch(comando){
                case '\n':
                    break;
                case '+':
                    signo_roll=1;
                    break;
                case '-':
                    signo_roll=-1;
                    break;
                default:
                    int_roll=Serial.parseInt();
                    break;
            }
        }
    }
    int_roll=int_roll*signo_roll;
}
}

```