



FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA DE SISTEMAS

ASIGNATURA:

INTERNET DE LAS COSAS IoT

GUÍA DE LABORATORIO: TEORÍA

PRÁCTICA DE LABORATORIO N.º 2

PROTOCOLOS DE COMUNICACIÓN

*Este material de apoyo académico
se hace para uso exclusivo de los alumnos
de la Universidad de Lima y en concordancia
con lo dispuesto por la legislación sobre
los derechos de autor: Decreto Legislativo 822*

PRÁCTICA DE LABORATORIO N.º 2 PROTOCOLOS DE COMUNICACIÓN

1. OBJETIVOS

- Uso de un módulo bluetooth HC-05, RTC DS1307 y módulo adaptador LCD a I2C.
- Comprensión de las librerías I2C y RTCLib
- Realizar un correcto cableado entre el módulo y el ESP32.
- Proyección de la fecha y hora en una pantalla LCD 16x2
- Encendido de diversos LED y muestra en el monitor serial.

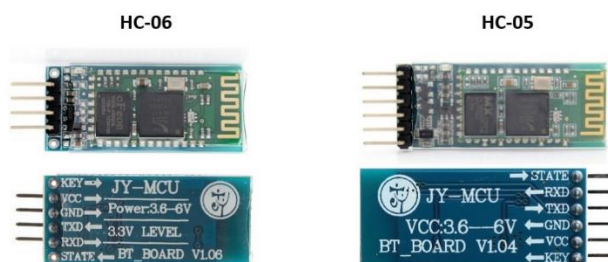
2. FUNDAMENTO TEÓRICO

2.1 Módulo Bluetooth

El Bluetooth es un estándar de comunicación inalámbrica que permite la transmisión de datos a través de radiofrecuencia en la banda de 2,4 GHz. Existen muchos módulos Bluetooth para usarlos en nuestros proyectos de electrónica, pero los más utilizados son los módulos de JY-MCU, ya que son muy económicos y fáciles de encontrar en el mercado. Son módulos pequeños y con un consumo muy bajo que nos permitirán agregar funcionalidades Bluetooth a nuestro Arduino. Estos módulos contienen el chip con una placa de desarrollo con los pins necesarios para la comunicación serie.

Existen dos modelos de módulos Bluetooth: el HC-05 que puede ser maestro/esclavo (master/slave), y el HC-06 que solo puede actuar como esclavo (slave). La diferencia entre maestro y esclavo es que en modo esclavo es el dispositivo quien se conecta al módulo, mientras que en modo maestro es el módulo quien se conecta con un dispositivo. Físicamente, los dos módulos son muy parecidos, solo varían algunas conexiones. Los pins que encontraremos son los siguientes:

- **VCC:** Alimentación del módulo entre 3,6V y 6V.
- **GND:** La masa del módulo.
- **TXD:** Transmisión de datos.
- **RXD:** Recepción de datos a un voltaje de 3,3V.
- **KEY:** Poner a nivel alto para entrar en modo configuración del módulo (solo el modelo HC-05)
- **STATE:** Para conectar un led de salida para visualizar cuando se comuniquen datos.



• Imagen 1: Módulos Bluetooth

2.2. Comandos AT

Los comandos AT son un tipo de comandos que sirven para configurar el módulo Bluetooth a través de un microcontrolador, un ordenador o con cualquier dispositivo que posea una comunicación serie (Tx/Rx). Son unas instrucciones que nos permiten cambiar los baudios del módulo, el PIN, el nombre, etc. Para usar los comandos AT el módulo Bluetooth no debe estar vinculado a ningún dispositivo (led rojo del módulo parpadeando). Según las especificaciones del módulo, el tiempo que se tiene que respetar entre el envío de un comando AT y otro tiene que ser de 1 segundo. Si se envía un comando AT y en menos de un segundo se envía otro, el módulo no devuelve respuesta. A continuación, se muestran las tablas de los comandos AT para los módulos HC-06 (slave) y HC-05 (master/slave)

Comando AT	Descripción	Respuesta
AT	Test de comunicación.	Responde con un OK
AT+VERSION	Retorna la versión del Modulo	OKlinvorV1.8
AT+BAUDx	Configura la velocidad de transmisión del modulo según el valor de "x" 1 = 1200 bps 2 = 2400 bps 3 = 4800 bps 4 = 9600 bps (por defecto) 5 = 19200 bps 6 = 38400 bps 7 = 57600 bps 8 = 115200 bps 9 = 230400 bps A = 460800 bps B = 921600 bps C = 1382400 bps	AT+BAUD4 Configura la velocidad a 9600 baud rate Responde con OK9600
AT+NAMEx	Configura el nombre con el que se visualizara el modulo, soporta hasta 20 caracteres	AT+NAMEDIYMakers Configura el nombre del modulo a DIYMakers Responde con OKsetname
AT+PINxxxx	Configura el Pin de acceso al modulo (password).1234 por defecto.	AT+PIN1122 Configura el pin a 1122 Responde con OKsetPIN

Tabla 1: Comandos AT del módulo HC-06

Comando AT	Descripción	Respuesta
AT+ORGL	Restaura los valores por defecto del módulo Bluetooth HC-05	Responde con un OK . (+ROLE:0; +CMOD:0; +UART:38400,0,0; +PSWD:1234 y Nombre: "H-C-2010-06-01" o "HC-05").
AT	Test de comunicación.	Responde con un OK .
AT+VERSION	Retorna la versión del módulo	Responde con la versión (+VERSION:2.0-20100601) .
AT+UART=X,Y,Z	Configura la velocidad de transmisión del módulo según el valor de "X": 1200 bps, 2400 bps, 4800 bps, 9600 bps, 19200 bps, 38400 bps (por defecto), 57600 bps, 115200 bps, 230400 bps, 460800 bps, 921600 bps, 1382400 bps. Con el valor de "Y" se configura el bit de parada: 0 -> 1 bit. 1 -> 2 bits. Con el valor de "Z" se configura el bit de paridad: 0 -> Sin paridad. 1 -> Paridad impar. 2 -> Paridad Par.	AT+UART=38400,0,0 configura el módulo Bluetooth HC-05 con una velocidad de comunicación de 38400 bps, con un bit de parada y sin paridad. Responde con un OK . AT+UART. Preguntamos por el estado de la comunicación UART. Responde con +UART:38400,0,0 .
AT+NAME=XXXX	Configura el nombre con el que se visualizara el módulo, soporta hasta 20 caracteres.	AT+NAME=Arduino_HC-0 configura el nombre del módulo a Arduino_HC-05. Responde con un OK .
AT+PSWD=XXXX	Configura el Pin o la contraseña de acceso al módulo Bluetooth HC-05.	AT+PSWD=1234 configura el pin a 1234. Responde con un OK . AT+PSWD. Preguntamos por el pin. Nos responde con +PSWD:1234 .
AT+CMODE = X	Cambiamos el modo de conexión del módulo según el valor de "X": 0 -> Se conecta al dispositivo Bluetooth especificado. 1 -> Se conecta a cualquier dispositivo Bluetooth disponible.	AT+CMODE = 1 configura el módulo Bluetooth HC-05 para que pueda conectarse a cualquier dispositivo Bluetooth disponible (sin dirección requerida). Responde con un OK . AT+CMODE: Preguntamos por el modo de conexión del módulo. Responde con +CMOD:1 .
AT+ROLE = X	Cambiamos el modo de trabajo del módulo, según el valor de "X": 0 -> Modo Esclavo. 1 -> Modo Maestro.	AT+ROLE = 0 configura el módulo Bluetooth para que trabaje como Esclavo de otro dispositivo Bluetooth. Responde con un OK . AT+ROLE: Preguntamos por el modo de trabajo del módulo. Responde con +ROLE:0 .

Tabla 2: Comandos AT módulo HC-05

2.3 APP Inventor

Es una aplicación originalmente desarrollada por Google y mantenida ahora por el Instituto de Tecnología de Massachusetts. Permite que cualquier persona, incluyendo las no familiarizadas con la programación y SDK de Android, pueda crear aplicaciones de Software para Android.

Utiliza una interfaz gráfica, muy similar al Scratch y el StarLogo, que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android. Google puso fin al desarrollo el 31 de diciembre de 2011 cediéndole el código al MIT, quién lo ha puesto a disposición de todos.

Se ejecuta vía web, sólo es necesario ejecutar una pequeña aplicación para mostrar el Editor de Bloques (Block Editor) y el emulador, incluso las aplicaciones se guardarán en la web. Por lo cual se puede acceder a nuestros proyectos desde cualquier equipo conectado a Internet. Permite conectarse directamente al equipo Android, aunque en

algunos casos no será posible, por ello incluye un emulador para probar las aplicaciones que se desarrollen.

Permite guardar el proyecto en PC, para enviarlo a otros usuarios o para hacer copia de seguridad y permite también descargar la aplicación .apk compilada para ser instalada en el dispositivo Android. Por lo cual no se necesita ningún tipo de compilador ni nada por el estilo, todo se realiza en el entorno de App Inventor.

2.4 MÓDULO ADAPTADOR LCD A I2C - PCF8574

Este módulo facilita considerablemente el uso de una pantalla LCD 16x2 o también de las pantallas 20x4. Dicho módulo solo utiliza los pines SDA y SCL para la comunicación entre microcontrolador y pantalla, aparte de la alimentación (5Vdc) y tierra (GND). Está basado en el controlador I2C PCF8574 que es un expensor de entradas y salidas digitales controlado por I2C. Por el diseño del PCB este módulo se usa especialmente para controlar un LCD Alfanumérico. Si en nuestro proyecto contamos con varios dispositivos I2C, este módulo se conecta en el mismo bus y no utiliza más pines. Para cambiar la dirección I2C basta soldar unos puentes que trae el módulo, por ejemplo, para utilizar hasta 8 Conversores I2C + LCD y así poder manejar cada LCD de forma independiente.

Este módulo es de fácil conexión por tener una manera única de soldar, tal como se muestra en la figura.

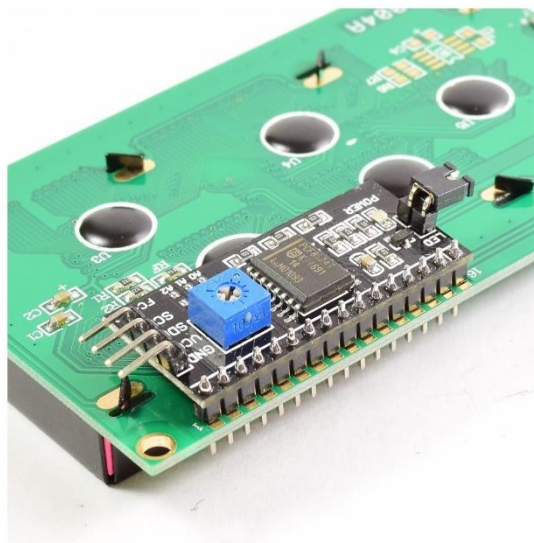


Imagen 2: Módulo I2C

Para un correcto uso y comunicación con el microcontrolador, se debe de usar la librería LiquidCrystal_I2C, los comandos utilizados en esta librería son muy similares a la librería LiquidCrystal normal. Algunos comandos si cambian como por ejemplo: .init() en lugar del comando .begin(). Se recomienda descargar la librería en el siguiente link: [Librería](#)

2.5 MÓDULO RTC DS1307

Los RTC (Real Time Clock) o reloj en Tiempo Real son la solución ideal cuando necesitamos integrar mediciones de tiempo a nuestros proyectos. Los RTC son de muy bajo consumo por lo que pueden ser alimentados por baterías y de esa forma no perder la sincronización. Si bien los microcontroladores poseen contadores internos, estos no son tan exactos como un RTC dedicado.

El módulo está basado en el RTC DS1307 de MAXIM y la EEPROM AT24C32 de ATMEL. Ambos circuitos integrados comparten el mismo bus comunicación con el Protocolo I2C. La memoria EEPROM AT24C32 te permite almacenar 32Kbits (4K Bytes) de datos de manera permanente.

Agregando el módulo El RTC a nuestro Arduino es como darle un reloj y un calendario, con el DS1307 nuestro Arduino sabrá la hora exacta como segundos, minutos y horas. También podremos saber el día del mes, día de la semana, mes, y año. EL ds1307 tiene implementado su calendario hasta el 2100 y tiene en cuenta también los años bisiestos. La comunicación entre el Arduino y el Módulo RTC es por I2C (pines SDA y SCL), También tiene un pin (SQ) que envía un pulso cada segundo (1 Hz), que nos puede servir para sincronizar otros componentes o para realizar un contador de segundos. A su vez, el módulo tiene un espacio para soldar un DS18B20 (sensor de temperatura) y acceder a este a través del pin DS.

Al reverso tiene un socket para insertar una batería de 3V, que le permite al RTC mantener en funcionamiento la hora y fecha cuando se apague el Arduino o se le quite la alimentación principal. Esta batería dependiendo de la marca y del tiempo apagado, puede durar desde los 3 hasta los 10 años aproximadamente.

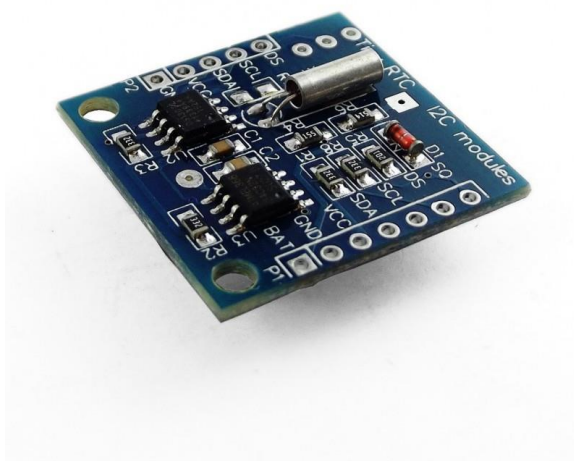


Imagen 3: Módulo RTC

3. CONFIGURACIÓN DE LOS MÓDULOS I2C

3.1 Módulo adaptador LCD a I2C - PCF8574

El primer paso que se debe de realizar es instalar la librería correspondiente al adaptador mencionado líneas arriba ([Librería](#)), si no se instala dicha librería no podremos tener un correcto uso del módulo y la pantalla, se recomienda utilizar un código escáner I2C para poder obtener la dirección I2C correspondiente al módulo. Por defecto la dirección puede ser 0x27 o 0x3F, aunque dichas direcciones se pueden modificar realizando una simple soldadura en los puntos A0, A1 y A2 del módulo. OJO: al realizar esta soldadura se debe de tener pre-estañado la placa y usar poco estaño para no averiar la placa. Conocida la dirección de la pantalla se usa el siguiente comando:

LiquidCrystal_I2C “nombre_de_la_pantalla”(dirección_I2C,16,2);

Las palabras en cursivas se tienen que modificar según el requerimiento del programador, es decir que puede colocar el nombre que más le guste a la pantalla, además de usar la dirección correspondiente al escáner I2C. Las siguientes líneas a usar deben de ser:

nombre_de_la_pantalla.init();
nombre_de_la_pantalla.backlight();

El primer comando es para inicializar la pantalla y la segunda para encender la luz led de la pantalla. En algunas ocasiones es necesario colocar el comando .init() al menos dos veces para que pueda conectar adecuadamente la pantalla, esto depende mucho de la versión de la librería o tipo de librería instalada.

A partir de ahora los comandos serán los mismos dentro del loop, en la cual vamos a definir la posición del cursor dentro de la pantalla con el comando nombre_de_la_pantalla.setCursor(); y el comando nombre_de_la_pantalla.print(); para escribir en la pantalla. Dentro del paréntesis debemos escribir la variable a imprimir o una palabra entre comillas. OJO: la pantalla no puede escribir letras propias de nuestro idioma como la letra ñ o cualquier vocal con tilde, para ello debemos de crear el carácter y escribirlo con el comando .write()

3.2 Módulo RTC DS1307

Es de suma importancia que la librería instalada sea la creada por Adafruit para el correcto desarrollo del laboratorio, otras librerías no son compatibles con el ESP32 al contar con una arquitectura distinta al Arduino, muchas librerías están pensadas para arduino y evoluciones de dicha placa. Se muestra una imagen de la librería que se debe de instalar.

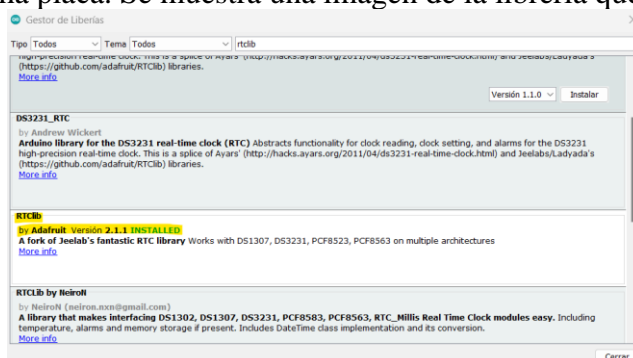


Imagen 4: Librería requerida

El cableado a realizar con el ESP32 es sencillo, solo se debe de conectar la alimentación, tierra y los pines SDA y SCL. Se recomienda usar los siguientes pines del módulo:

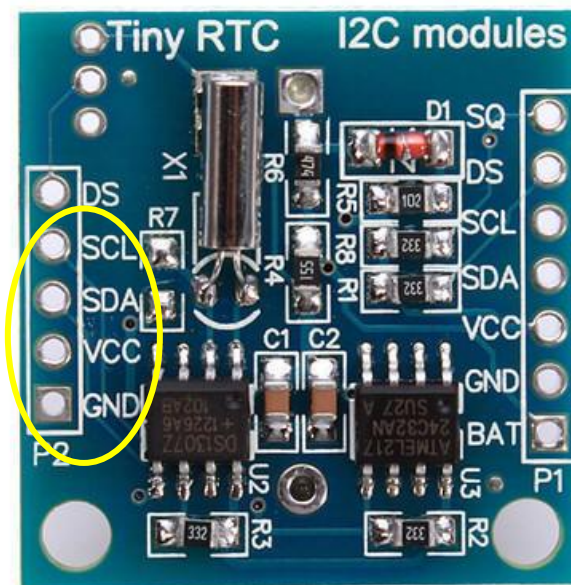


Imagen 5: Pines del módulo RTC

Algo importante a tener en cuenta es que el módulo imprime la fecha por defecto en el formato de fecha estadounidense, es decir: MM/DD/AAAA. Cuando se requiera un dato en concreto se debe de especificar.

El primer paso que se debe de realizar es definir el nombre que vamos a poner al módulo, en este caso no será necesario cambiar la dirección del módulo I2C ya que no es para nada necesario usar más de un módulo RTC, por defecto la librería se comunica con el módulo. El comando a usar será:

RTC_DS1307 nombre_del_reloj;

Como en los casos anteriores, cada uno puede definir el nombre del reloj según su gusto, pero si se debe especificar que se usará el modelo DS1307, la librería toma otros módulos RTC del mercado.

Si se desea tomar los nombres del día, se necesita crear una matriz de 7x12 correspondiente a los días de la semana y los doce meses. OJO: el RTC toma como el primer día de la semana el domingo, por lo cual se recomienda crear la siguiente matriz:

```
char diasSemana[7][12]={"Domingo","Lunes","Martes","Miercoles","Jueves","Viernes","Sabado"};
```

Los comandos mencionados con anterioridad se deben de definir antes del setup, porque solo se ejecutarán una única vez.

Dentro del setup se debe de inicializar el módulo, de la misma forma que se realizó con la pantalla LCD, se debe de usar un comando especial como el siguiente:

`nombre_del_reloj.begin();`

Si no se inicializa el módulo, este tendrá muchas fallas al no reconocer adecuadamente el RTC. Finalmente, dentro del loop vamos a realizar de manera constante una adquisición de la fecha y hora en el instante, de esta manera tenemos una actualización en tiempo real de la información del RTC. La línea de código a usar dentro del loop es:

`DateTime now= nombre_del_reloj.now();`

Dentro de la variable now se encuentra la información del RTC. Se muestra una tabla con los comandos utilizados para obtener una información en concreto:

Comando	Información
<code>now.day()</code>	Día
<code>now.month()</code>	Mes
<code>now.year()</code>	Año
<code>now.hour()</code>	Hora
<code>now.minute()</code>	Minuto
<code>now.second()</code>	Segundo

En caso de necesitar el nombre del día, podemos aprovechar el array creado y usar el siguiente comando: `diasSemana[now.dayOfTheWeek()]`

Se podría crear un array 12x12 para los meses del año como por ejemplo:

`char nombreMes[12][12]= {"Enero", "Febrero", "Marzo", "Abril", "Mayo", "Junio", "Julio", "Agosto", "Septiembre", "Octubre", "Noviembre", "Diciembre"};`

El detalle será que al imprimir el mes del año se debe de restar 1 al mes, algo similar a la siguiente línea de código: `nombreMes[now. month ()-1]`

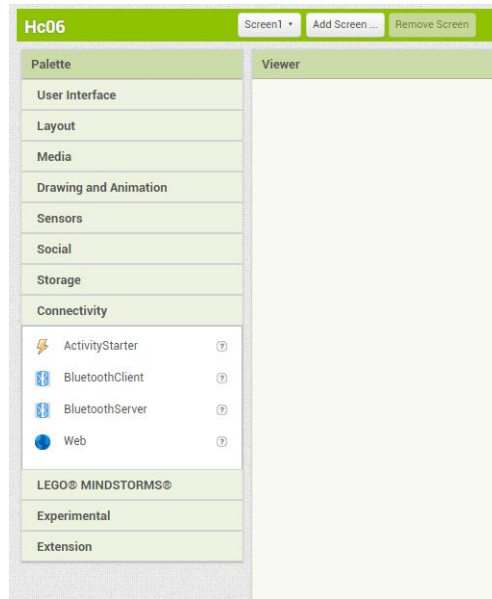
3.3 CREACION DE UN APP CON EL APP INVENTOR

Ingresamos en la web de app inventor (<http://appinventor.mit.edu/explore/>), seleccionamos Create apps! Y vinculamos con un correo Gmail.

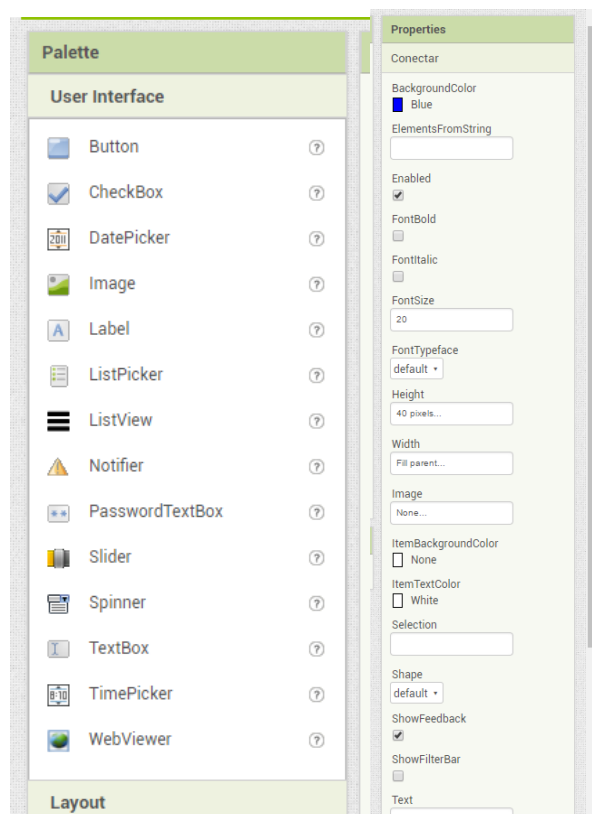
Seleccionamos crear nuevo proyecto y escribir un nombre a nuestro nuevo proyecto.

En este apartado podemos empezar a crear los bloques de programación y la apariencia de nuestra app.

El primer paso es crear la conexión con el bluetooth, para ello seleccionamos el bluetooth client en el apartado de conectividad y lo llevamos al viewer:

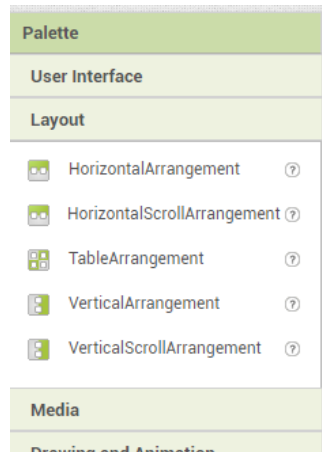


El siguiente paso será crear un botón para conectar el módulo, se selecciona en el menú de interfaz de usuario la opción “Listpicker” y se modifica los siguientes parámetros de sus propiedades:



Color de fondo: azul.
Tamaño de fuente: 30
Alto: 40 pixeles
Ancho: Fill parent
Texto: Conectar Bluetooth
TextAlignment: centrar

Procedemos a crear un espacio en blanco para tener un orden, para ello buscamos la opción HorizontalArrangement en la opción Layout con las siguiente propiedades:

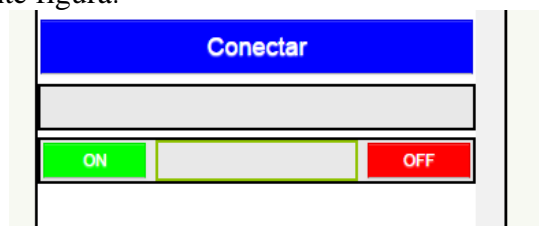


Alto: 30 pixeles

Ancho: Fill Parent

El siguiente paso será crear 2 botones, para ello creamos otro “HorizontalArrangement” y se coloca otro “HorizontalArrangement” en su interior.

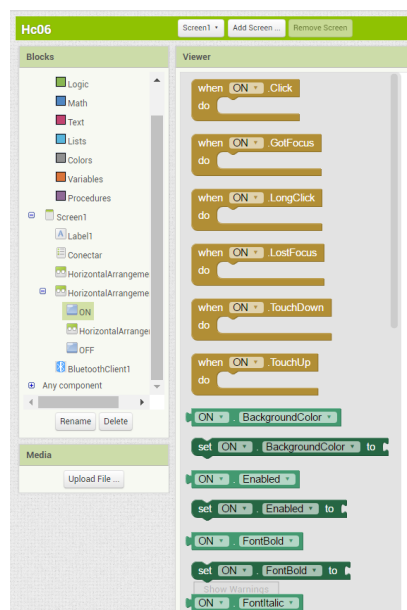
Se procede a colocar 2 botones y se configura sus propiedades para obtener un resultado similar al de la siguiente figura:



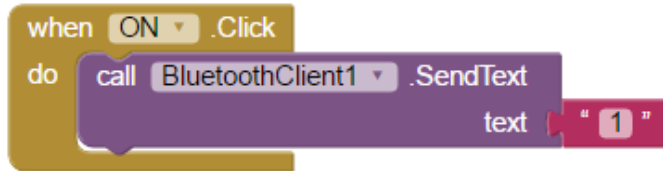
Una vez realizado nuestra interface, se debe de realizar la programación para tener una comunicación con el Arduino. Para ello entramos a blocks.

A la izquierda podemos observar todos los bloques previamente creados para empezar a programarlos. Empezamos con los botones para el encendido y apagado.

Al dar click en el botón ON seleccionamos When “ON”. Click do..



El objetivo del botón es enviar un texto al Arduino por medio del bluetooth y este sepa qué hacer con dicha información, para ello se busca en bluetoothclient la opción de enviar y texto. Al inicio elegimos el bloque texto y seleccionamos el bloque en blanco para colocar 1 para el encendido. El resultado final debe ser similar al siguiente:

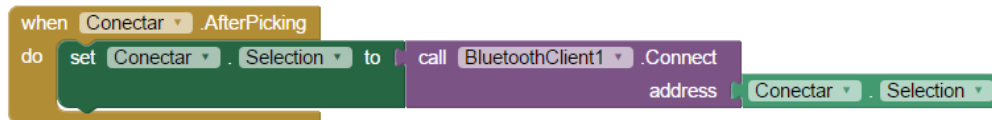


Para el botón de apagado, hacemos doble click y seleccionamos la opción duplicar. Solo debemos de configurar el botón OFF.

El siguiente paso es configurar la conectividad del bluetooth, para ello seleccionamos las opciones before picking y after picking del listpicker para poder realizar lo siguiente: El before picking debe de crear unos elementos al listpicker , que serán las direcciones y nombres de los dispositivos bluetooth existentes.



El siguiente caso; afterpicking debe de hacer una selección de la conexión de la dirección del módulo bluetooth del listpicker anterior:

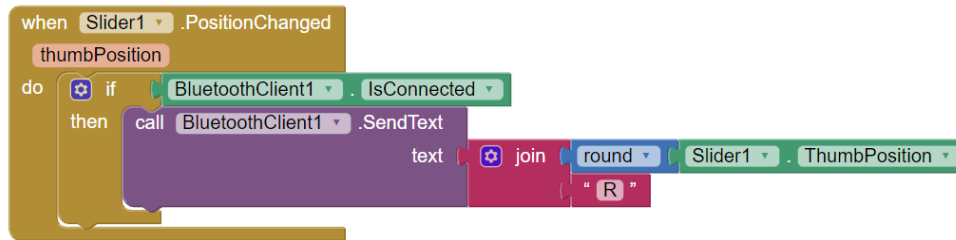


El siguiente pase será instalar el apk a nuestro dispositivo Android. Para ello damos click en build y seleccionamos la opción .apk. Si damos a la opción por código QR, debemos de estar conectados a la misma red que nuestra computadora, sino con el cable USB del dispositivo Android pasar el archivo a la memoria interna e instalarlo por ahí. Comprobar el correcto funcionamiento del app.

Si deseamos crear un aplicativo que contenga el control PWM del ESP32, debemos de crear una app que contenga un slider de la siguiente manera:



A continuación, debemos de programar los bloques de activación del bluetooth y además de crear los siguientes bloques:



Este caso lo que hará es enviar un texto que será el valor del slide como un texto al Arduino, seguidamente del texto "R"