

PROYECTO FINAL

PROYECTO: Control de relé mediante Bluetooth

CURSO: 7^º1^º

PROF: Dellapaolera - Cottone

ALUMNO: Agustín Ostua

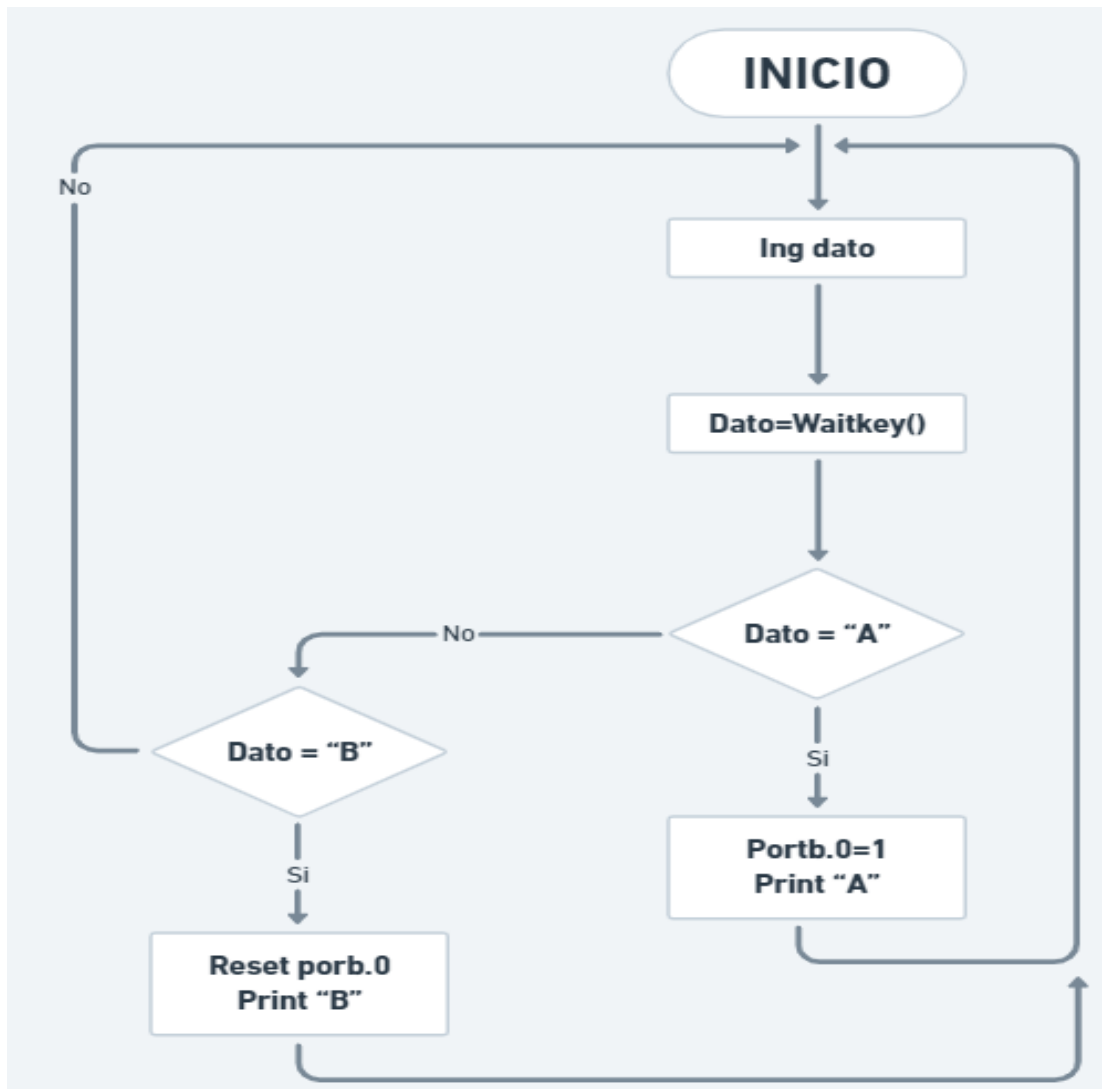
Explicación de proyecto

-COMPONENTES

- Resistencias
- Atmega 328p
- Led
- Módulo relé (5v)
- Modulo Bluetooth HC-05
- Cables Macho-Macho / Macho-Hembra
- Cable USB (Entrada Arduino 5v)
- Batería 9v
- Porta batería

DIAGRAMA DE FLUJO y CODIGO BASCOM

-Diagrama de flujo



Código bascom

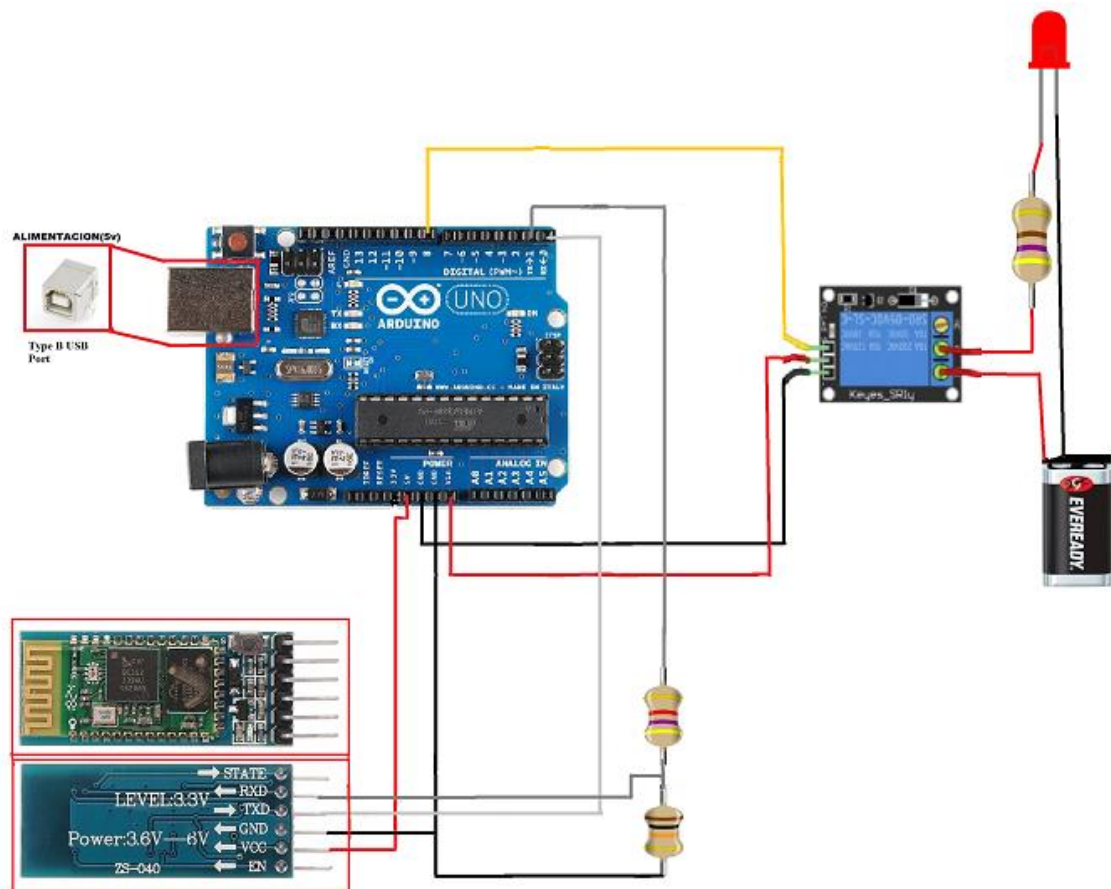
```
*****CONFIGURO EL MICRO*****
Sregfile = "m328pdef.dat"
$crystal = 16000000
$hwstack = 40
$swstack = 16
$framesize = 32

***** CONFIGURACION DE PINES Y VARIABLES *****
Config PORTB = Output           'configuro puerto B como salida
Dim Dato As Byte                'declaro variable "dato" tipo byte

*****BUCLE PRINCIPAL*****
Do
    Dato = Waitkey()             'con la variable dato leo los datos que puedo recibir mediante la conexcion
    If Dato = "A" Then           'si dato es A, entonces enciendo el rele y mando un aviso con A
        PORTB 0 = 1
        Print "A"               'imprimo el caracter "A", es decir, mando al bluetooth el aviso de estado de rele
    End If
    If Dato = "B" Then           'si dato es B, entonces apago el rele y mando un aviso con B
        Reset PORTB 0
        Print "B"               'imprimo el caracter "B", es decir, mando al bluetooth el aviso de estado de rele
    End If
Loop
*****
```

Se colocaron comentarios en cada línea de código para seguir el proceso del mismo y entender su funcionamiento

CIRCUITO (Esquema)



-Rled = 470 ohm

-Divisor de tensión (modulo Bluetooth)

R2 = 10k - R1 = 4,7K

Se utiliza un divisor de tensión (en la terminal RX del módulo bluetooth), para tener la adaptación de niveles, ya que en la entrada de este módulo no hay toleración de 5v. Por lo tanto se debe tener en cuenta tanto el voltaje de entrada y el voltaje de salida deseado.

Entonces => Adoptando R2 = 10k => $R1 = (Vi/Vo - 1) * R2$

Reemplazando los valores obtenemos una R1 = 4,7K

Verificando esto se utiliza la fórmula para ver el voltaje que tendremos en RX.

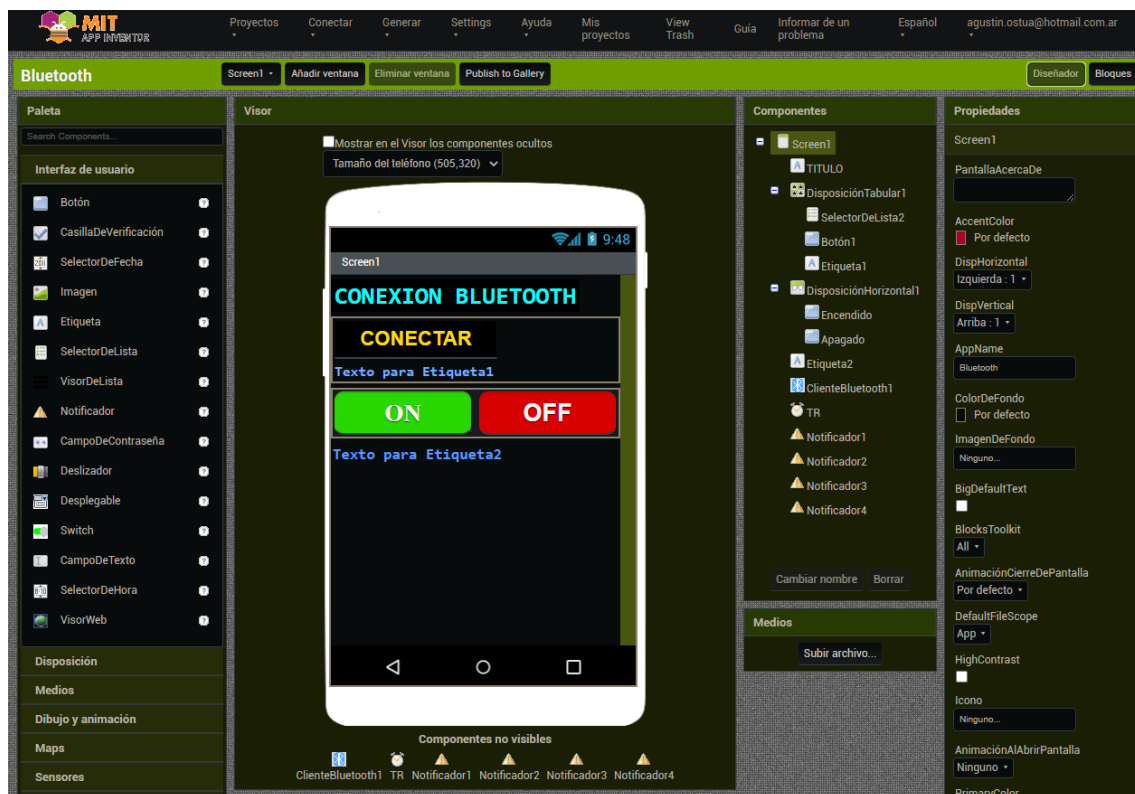
$$Vi * \frac{R2}{R1 + R2} \Rightarrow Vo = 3,4v$$

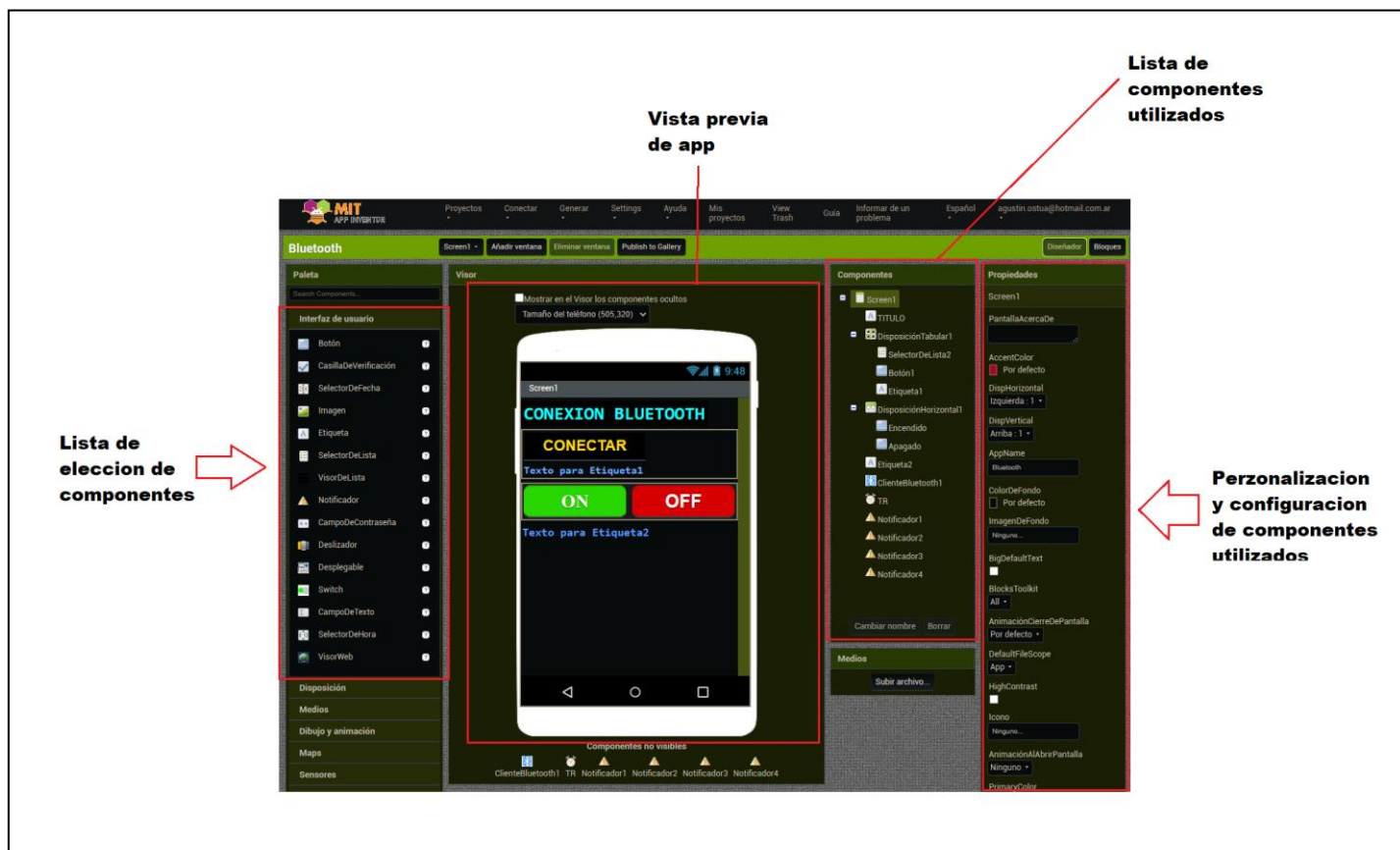
Con esta operación se obtiene el resultado deseado para utilizar y conectar el modulo bluetooth.

CONFIGURACION DE BLUETOOTH (Programando en app inventor)

Proceso de aplicación

-Interfaz



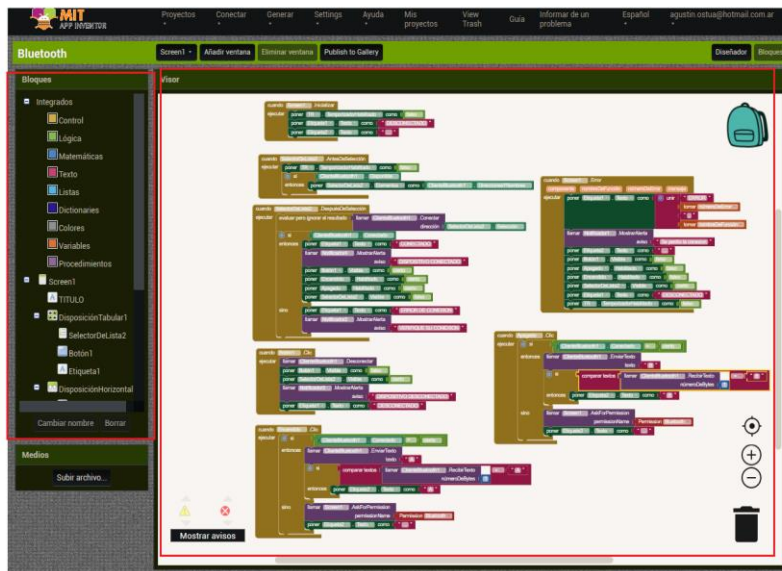


Como se ve en las imagenes, existen tanto componentes visibles como no visibles, por ejemplo, los botones y selectores son visibles, y las conexiones o notficadores son no visibles ya que solo se configuran a gusto del usuario, también existe la opción de que un componente visible sea no visible configurándolo en la sección de “Personalización y configuración de componentes utilizados”.

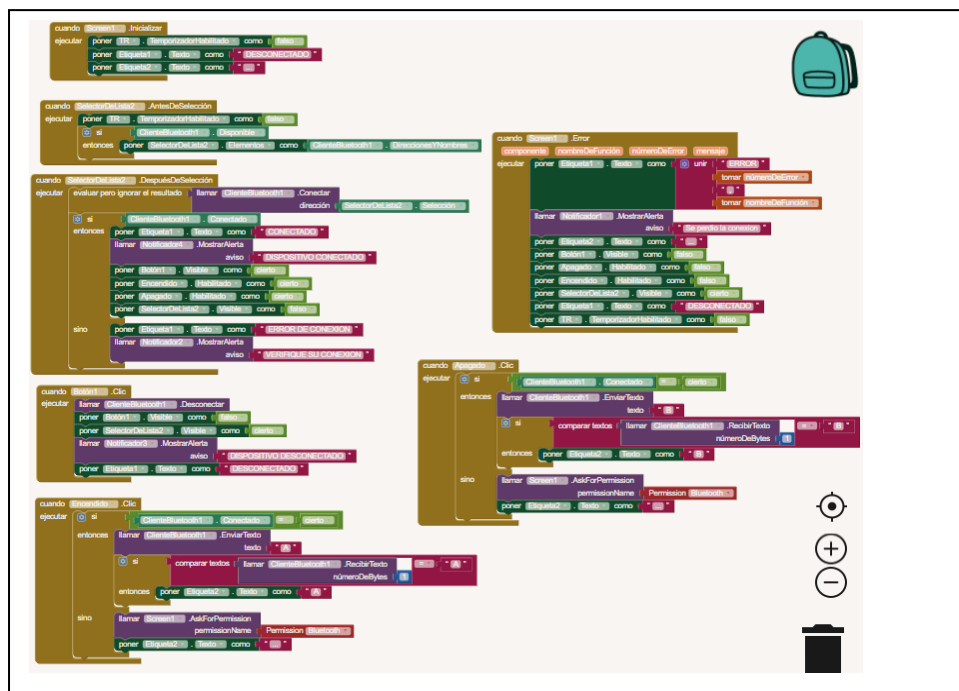
Configuración por medio de Diagrama en bloque

-Interfaz y proceso de programa

Lista de
selección de
variables y
manejo de
componentes
utilizados



Proceso de
programación



Explicación de programa

Una vez puesta en marcha la app se verán, un título, una opción de conexión bluetooth (selección de lista), los botones “ON” “OFF” y sus respectivos textos de aviso; Como al iniciar no hay conexión alguna se mostrara un texto de “DESCONECTADO” y otro en espera de estado “...”.

Al querer hacer conexión se debe presionar el botón “CONECTAR”, el cual es un selector de lista de dispositivos y por lo tanto mostrara dispositivos disponibles para conectar, en caso de lograr una conexión, este ya dejaría de mostrarse y pasara a mostrar un botón de desconexión (“DESCONECTAR”) si busca salir de esa conexión, también se mostrara un aviso de “DISPOSITIVO CONECTADO” mediante un notificador el cual fue configurado en esta aplicación. Una vez conectados podemos manejar el estado de aquel componente el cual se quiera cambiar de estado, en este caso un relé que accionara un LED. Al presionar “ON” se enviara un dato / carácter al arduino (“A”), este lo leerá y evaluara ese dato para luego encender el relé, una vez hecho esto el mismo arduino enviara a la aplicación un dato / carácter para avisar que cambio el estado del componente, teniendo así en la app una actualización o aviso de en qué estado está el relé / led; La aplicación evaluara el dato recibido y actuara en consecuencia mostrando el estado.

En caso de presionar el botón “OFF”, ocurrirá el mismo proceso con la diferencia de que cambiara el dato / carácter que recibirá tanto el arduino como la aplicación y luego se verá en pantalla el nuevo estado del componente.

La condición necesaria para que podamos verificar que estos estados son reales, es que, al presionar cada botón

se debe constatar de que la conexión aún está corriendo y si esto se cumple se llevara el proceso antes mencionado.

Una vez visto cómo se puede cambiar el estado del componente y finalizada la acción, fácilmente presionando el botón “DESCONECTAR”, dejaremos de tener conexión y aparecerá un notificador que mostrara “DISPOSITIVO DESCONECTADO” y al mismo tiempo, los respectivos textos de cada botón pasaran a su estado inicial esperando así una nueva conexión para llevar a cabo su función.

En caso de no haber finalizado la acción y querer continuar con el manejo de cambio de estado, habrá 2 opciones, una opción es seguir normalmente utilizando esta conexión y la segunda opción es la siguiente:

Si el usuario utiliza la aplicación y por alguna razón se percibe una desconexión ya sea desde el arduino, modulo bluetooth o la aplicación, entonces la misma aplicación detectara esa desconexión los notificadores actuaran mostrando los textos “Se perdió la conexión” y “Verifique su conexión”, y todo pasara a su estado inicial a excepción del texto de aviso correspondiente a los botones ON OFF, el cual mostrara el ultimo estado en que se supone quedo el relé, colocando al usuario en la obligación de verificar manualmente que ese estado es correcto.

Los textos “Se perdió la conexión” y “Verifique su conexión” también abarcaran para cuando el usuario desee obtener una nueva conexión y esta falle, teniendo así que verificar que todo esté en orden para volver a conectar.