

Sesión 49 – Bluetooth + KYMO

(Guía formador primaria)

Meta

Interactuar por medio del modulo bluetooth y la kymo con él Pygmo.

Resumen

En esta actividad se programará al Pygmo con el fin de controlarlo por medio de la kymo, utilizando dos módulos bluetooth HC05..

Al completar esta actividad, los estudiantes:

- Identificar el funcionamiento básico del dispositivo bluetooth en el Pygmo.
- Conocer los fundamentos básicos de programación.
- Incentivar la exploración de la programación de los motores del robot por medio del bluetooth.

Vocabulario contextualizado de la sesión

¿Qué significa lenguaje de programación C++?

Es un lenguaje de programación es un lenguaje formal diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana.

Funciones void setup() y void loop()

En CODE, siempre vamos a encontrar estas dos funciones principales:

La función **void setup()** se usa para establecer las configuraciones iniciales de nuestro robot, por ejemplo, si deseamos que los puertos que vamos a utilizar, sean de entrada o salida, si queremos establecer alguna comunicación con el computador y mucho más. Esta función se ejecutará una única vez cuando se conecte la energía a la tarjeta de control de Innobot o cuando se pulse el botón de reinicio.

El lenguaje de programación C++ es sumamente preciso.

Las siguientes formas son incorrectas, miremos por qué.

Forma Incorrecta	Descripción
setup()	Si omites void el compilador te arroja un error.
voidsetup()	Las palabras void y setup deben estar separadas por un espacio
void setup	Si omites los paréntesis () el compilador te arroja un error.
void Setup()	C++ identifica minúsculas y mayúsculas, setup() no es igual a Setup() .

*La función **void loop()** se usa para realizar una tarea una y otra vez, como un bucle o loop en inglés, por ejemplo: revisar un sensor una y otra vez, hasta que se desconecte la energía a la tarjeta de control Innobot.*

"Esta Metodología, y todas sus guías, manuales y componentes, fue desarrollada por PYGMALION (R), quien tiene los derechos de uso, distribución, comercialización y autorización a terceros. Queda prohibida la reproducción parcial o total del presente documento, por medio de cualquier proceso reprográfico, sea fónico, microfilme, mimeográfico, offset, electrónico o por fotocopia. Esta edición y sus características gráficas son propiedad de FUNDACIÓN PARQUE DEL SOFTWARE MEDELLÍN, ParqueSoft Medellín. Todos los derechos Reservados"

Es donde se controla de forma activa la tarjeta de control de InnoBot pues dentro de las llaves irá encerrado nuestro programa.

Las llaves { }

Has notado que void **setup()** y void **loop()** comienzan con una llave { y luego terminan con otra llave }

Esto tiene un por qué: toda función tiene un inicio y tiene un final, se usa llave { para abrir, y llave para cerrar }

El programa ejecutará todo lo que encuentre entre las llaves. Si olvidas una de estas el Compilador te dirá que tienes un error ¡No lo olvides!

El lenguaje de programación C++ es sumamente preciso, las siguientes formas son incorrectas, miremos por qué.

Forma Incorrecta	Descripción
<pre>void loop() { digitalWrite(13, HIGH); delay(1000); digitalWrite(13, LOW); delay(1000); }</pre>	Si omites la llave de cierre } nunca se sabrá dónde termina el void loop() .
<pre>void setup() pinMode(13, OUTPUT); }</pre>	Si omites la llave que abre { nunca se sabrá dónde comienza el void setup ()

El punto y coma ;

Has notado también que las funciones pinMode(), digitalWrite() y delay() terminan en ; significa que la instrucción o paso termina ahí. Si lo olvidas el compilador te advertirá que te hace falta.

Los paréntesis ()

¿Has notado que las funciones pinMode, digitalWrite y delay, tienen unos paréntesis ()? Significa que los parámetros que tienen una función. Por ejemplo: pinMode(13, OUTPUT) tiene dos parámetros

- (1) El pin que deseas establecer el modo
- (2) El modo que deseas establecer, en este caso SALIDA(OUTPUT).

Algunas excepciones

C++ permite algunas maneras distintas de escribir el código siempre y cuando se cumplan las reglas básicas de abrir y cerrar paréntesis, abrir y cerrar llaves, de cerrar cada instrucción con punto y coma (;) y de escribir correctamente las minúsculas y las mayúsculas.

Los siguientes ejemplos no tienen errores de sintaxis o escritura de código.

<pre>void loop() { digitalWrite(13, HIGH); }</pre>	<pre>void loop() {</pre>
--	--------------------------

<pre>delay(1000); digitalWrite(13, LOW); delay(1000); }</pre>	<pre>digitalWrite(13, HIGH); delay(1000); digitalWrite(13, LOW); delay(1000); }</pre>
<pre>void loop() { digitalWrite(13, HIGH); delay(1000); digitalWrite(13, LOW); delay(1000); }</pre>	<pre>void loop() { digitalWrite(13, HIGH); delay(1000); digitalWrite(13, LOW); delay(1000); }</pre>

Módulo bluetooth: Es una especificación tecnológica para redes inalámbricas que permite la transmisión de voz y datos entre distintos dispositivos mediante una radiofrecuencia segura (2,4 GHz). Esta tecnología, por lo tanto, permite las comunicaciones sin cables ni conectores y la posibilidad de crear redes inalámbricas domésticas para sincronizar y compartir la información que se encuentra almacenada en diversos equipos.

Datos: Se conoce que la palabra Datos proviene del latín “Dtum” cuyo significado es “lo que se da”. Los datos son la representación simbólica, bien sea mediante números o letras de una recopilación de información la cual puede ser cualitativa o cuantitativa, que facilitan la deducción de una investigación o un hecho.

Lógica: Es la ciencia que se basa en las leyes, modalidades y formas del conocimiento científico se conoce bajo el nombre de lógica. Se trata de una ciencia de carácter formal que carece de contenido ya que hace foco en el estudio de las alternativas válidas de inferencia. Es decir, propone estudiar los métodos y los principios adecuados para identificar al razonamiento correcto frente a que no lo es.

Programar: Es crear una serie de códigos en la computadora, para que un robot, una maquina o hasta la misma computadora, realice unas tareas específicas.

Robótica: Es la ciencia y la técnica que está involucrada en el diseño, la fabricación y la utilización de robots. Un robot es, por otra parte, una máquina que puede programarse para que interactúe con objetos y lograr que imite, en cierta forma, el comportamiento humano o animal. La informática, la electrónica, la mecánica y la ingeniería son sólo algunas de las disciplinas que se combinan en la robótica. El objetivo principal de la robótica es la construcción de dispositivos que funcionen de manera automática y que realicen trabajos dificultosos o imposibles para los seres humanos.

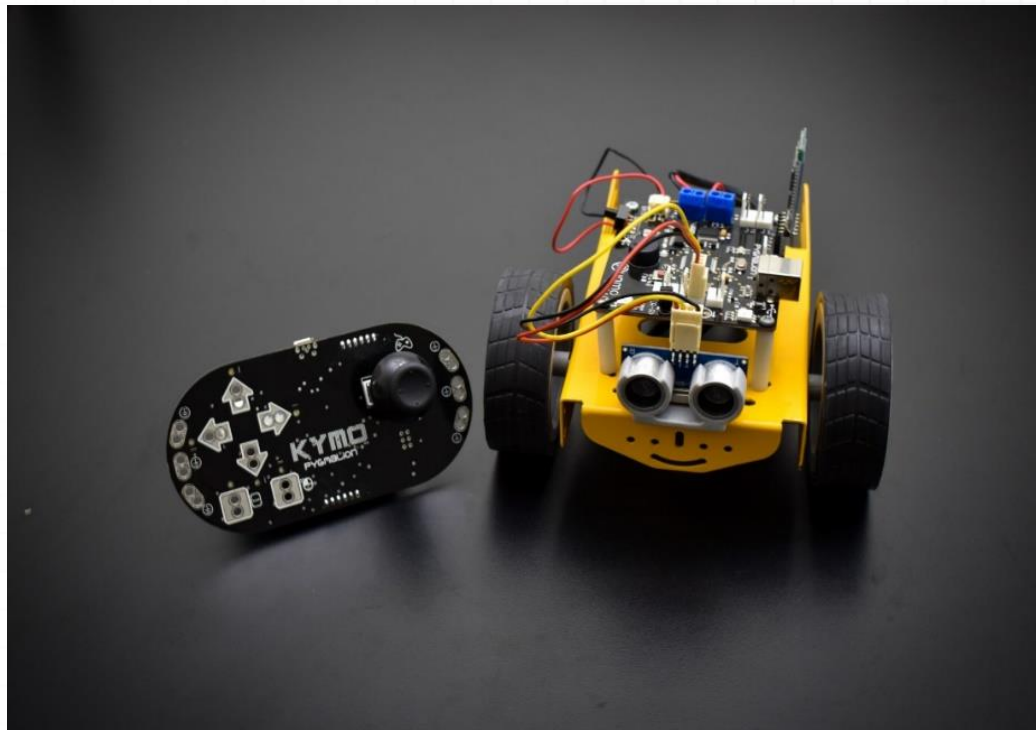
Flujo de la sesión

No.	Actividad	Descripción	Recursos	Tiempo
1	Presentación	Toma de asistencia	Lista de asistencia	10 min
2	Explicación	En este espacio explique lo que se realizará en la sesión.		10 min

3	Configuración de los bluetooth como maestro y esclavo	Realice el código en CODE, y realice una breve descripción de las funciones.	Pygmalion IDE CODE	20 min
4	Conectividad entre bluetooth (maestro y esclavo)	Ensayar si ya se conectan entre sí los dos dispositivos	Módulo Bluetooth	10 min
5	Programación de control de kymo y Pygmo	Realice la programación	Pygmalion IDE - CODE	30 min
6	Juega con Pygmo	El facilitador se ingenia una carrera para poner a prueba la destreza de los estudiantes con la kymo.	Pygmo kymo	30 min
6	Cierre	Resuelva dudas e inquietudes, realice la recolección del material		10 min

Descripción de las actividades

1. Inicie la sesión con la toma de asistencia de los estudiantes.
2. Realice una breve explicación de lo que se realizara en la sesión.

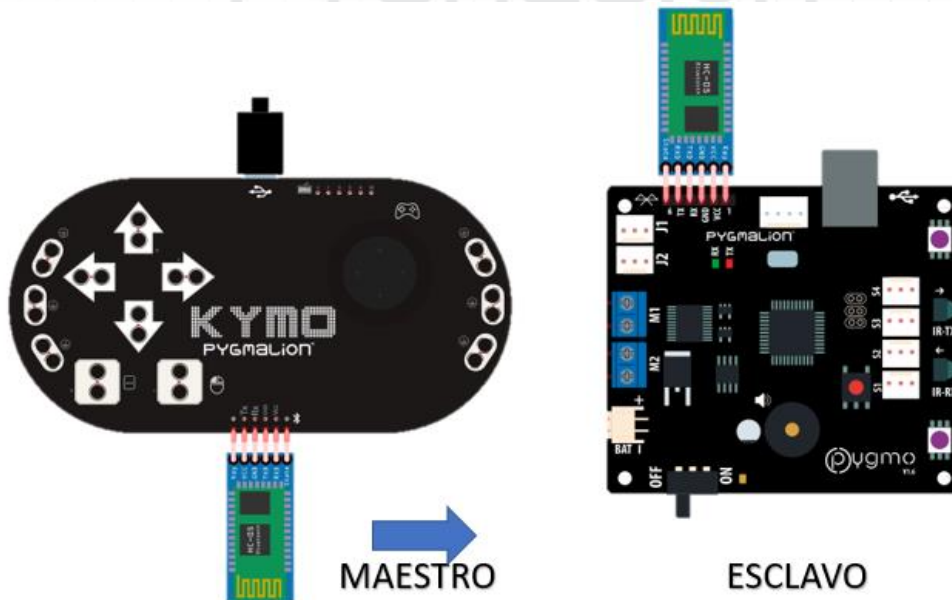


Acercamiento y construcción de conceptos



3. Configuración bluetooth como maestro y esclavo.

el bluetooth conectado a la kymo es el maestro y el que va conectado al innobot es el esclavo. En pocas palabras el maestro le da órdenes al esclavo.



4. Comando AT para la configuración de los bluetooth.

- Regresar a opción Fabrica AT+RESET
- ESTE ES MUY IMPORTANTE, SERIA BUENO QUE RESETEE LOS DOS BLUETOOTHS ANTES DE EMPEZAR A CONFIGURAR.**
- Saber la MAC del módulo AT+ADDR?
 - Saber el nombre del módulo: AT+NAME?
 - Cambiarlo a "MINOMBRE": AT+NAME=MINOMBRE
 - Saber configuración: AT+ROLE?
 - Cambiarlo al modo esclavo: AT+ROLE=0
 - Cambiarlo al modo maestro: AT+ROLE=1
 - Saber la contraseña: AT+PSWD?
 - Cambiarlo a 0000: AT+PSWD=1234 según la Contraseña a Conectar (max 4 char.)
 - Saber la velocidad (9600): AT+UART?
 - Cambiarlo a 9600: AT+UART=9600,0,0
 - Saber modo de Conexión: AT+CMODE? (0 Específico 1 General)
 - A cualquier dispositivo: AT+CMODE=1
 - A un dispositivo específico AT+CMODE=0
 - Conectarse a una Mac específica AT+BIND=##,##,##,##
- Nota: **pone la MAC en mayúscula y los ":" los cambia por ",".**

5. Para configurar ambos bluetooths le cargamos en CODE el siguiente programa, este programa nos permitirá configurar el bluetooth que será el esclavo y el maestro:

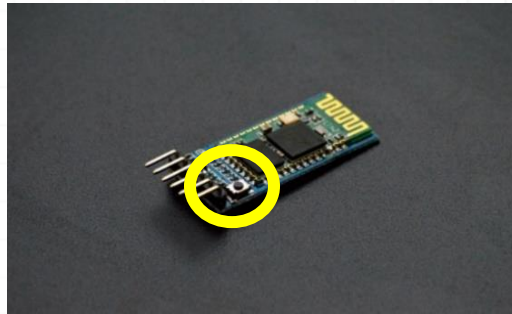
NOTA: los dos bluetooths se configuran con la tarjeta Pygmo

6. Para configurar el bluetooth debemos poner el bluetooth en modo de **comandos AT**.

Para eso hacemos lo siguiente:

- Conecta la tarjeta Pygmo.**
- Carga el programa y NO desconecta la tarjeta la tarjeta para que permanezca alimentada.**

- III. Después insertamos el bluetooth con el botón pequeño oprimido y al ingresar en la tarjeta debe permanecer encendido el led rojo del bluetooth cada dos segundos aproximadamente.



- IV. Si esta intermitentemente el led rojo del bluetooth ya está en modo de comandos AT.
- V. El bluetooth está listo para ingresar los comandos y configurarlo como maestro o como esclavo.

7. Vamos a configurar el bluetooth esclavo, para ello debe estar en modo de comandos AT.

Modulo bluetooth hc-05 como esclavo:

Cuando está configurado de esta forma, se comporta similar a un HC-06, espera que un dispositivo bluetooth maestro se conecte a este, generalmente se utiliza cuando se necesita comunicarse con una PC o Celular, pues estos se comportan como dispositivos maestros.

Modulo bluetooth hc-05 como Maestro:

En este modo, EL HC-05 es el que inicia la conexión. Un dispositivo maestro solo se puede conectarse con un dispositivo esclavo. Generalmente se utiliza este modo para comunicarse entre módulos bluetooth. Pero es necesario antes especificar con que dispositivo se tiene que comunicar, esto se explicará más adelante

El módulo HC-05 viene por defecto configurado de la siguiente forma:

- Modo o role: Esclavo
- Nombre por defeco: HC-05
- Código de emparejamiento por defecto: 1234
- La velocidad por defecto (baud rate): 9600

EL Módulo HC-05 tiene 4 estados los cuales es importante conocer:

Estado Desconectado:

- Entra a este estado tan pronto alimentas el módulo, y cuando no se ha establecido una conexión bluetooth con ningún otro dispositivo
- EL LED del módulo en este estado parpadea rápidamente
- En este estado a diferencia del HC-06, el HC-05 no puede interpretar los comandos AT

Estado Conectado o de comunicación

- Entra a este estado cuando se establece una conexión con otro dispositivo bluetooth.
- El LED hace un doble parpadeo.
- Todos los datos que se ingresen al HC-05 por el Pin RX se transmiten por bluetooth al dispositivo conectado, y los datos recibidos se devuelven por el pin TX. La comunicación es transparente

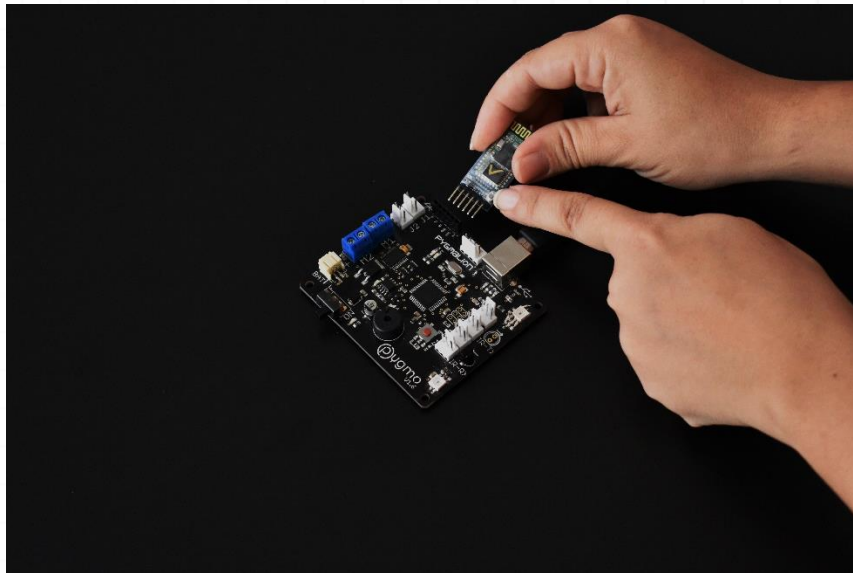
Modo AT 1

- Para entrar a este estado después de conectar y alimentar el módulo es necesario presionar el botón del HC-05.
- En este estado, podemos enviar comandos AT, pero a la misma velocidad con el que está configurado.
- EL LED del módulo en este estado parpadea rápidamente igual que en el estado desconectado.

Modo AT 2

- Para entrar a este estado es necesario tener presionado el botón al momento de alimentar el módulo, es decir el módulo debe encender con el botón presionado, después de haber encendido se puede soltar y permanecerá en este estado.
- En este estado, para enviar comandos AT es necesario hacerlo a la velocidad de 38400 baudios, esto es muy útil cuando nos olvidamos la velocidad con la que hemos dejado configurado nuestro modulo.
- EL LED del módulo en este estado parpadea lentamente.

Entendido lo anterior vamos realizamos las conexiones para configurar el HC-05. Para configurar el módulo necesitamos enviar los comandos AT desde una computadora,





```
//
SLAVE MODULE CONFIGURATION

The slave module is powered up in command mode.

AT+ADDR?
+ADDR:11:3:252002
OK
AT+UART?
+UART:9600,0,0
OK
AT+ROLE?
+ROLE:0
OK
Role=0 is for the slave mode. I keep the default 9600 baud UART speed.

The module Bluetooth address will be useful to establish the connection
from the master module.

MASTER MODULE CONFIGURATION

The master module is powered up in command mode. The role is changed to 1
for master. The UART baudrate is set to the same as the slave module.

AT+UART?
+UART:115200,0,0
OK
AT+UART=9600,0,0
OK
AT+UART?
+UART:9600,0,0
OK
AT+ROLE?
+ROLE:0
OK
AT+ROLE=1
```

```

+ROLE:1
OK
AT+ROLE?
+ROLE:1
OK
To establish an automatic connection to a slave module, the master module
must know the slave address. This is configured with the AT+BIND command
with CMODE=0 (Use a predefined address).

AT+CMODE?
+CMOD:0
OK
AT+BIND=11,3,252002
OK
AT+BIND?
+BIND:11:3:252002
OK
*/

char c = ' ';
boolean NL = true;

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

    //Set to HC-05 default baud rate, found using AT+UART. It is usually
    38400. 115200
    //Serial1.begin(115200);
    //Serial1.begin(9600);
    //Serial1.begin(57600);
    Serial1.begin(38400);
}

void loop()
{
    // Read from the Bluetooth module and send to the Arduino Serial
    Monitor
    if (Serial1.available())
    {
        c = Serial1.read();
        Serial.write(c);
    }

    // Read from the Serial Monitor and send to the Bluetooth module
    if (Serial.available())
    {
        c = Serial.read();
        Serial1.write(c);
    }

    // Echo the user input to the main window. The ">" character
    indicates the user entered text.
    if (NL) {
        Serial.print(">");
    }
}

```

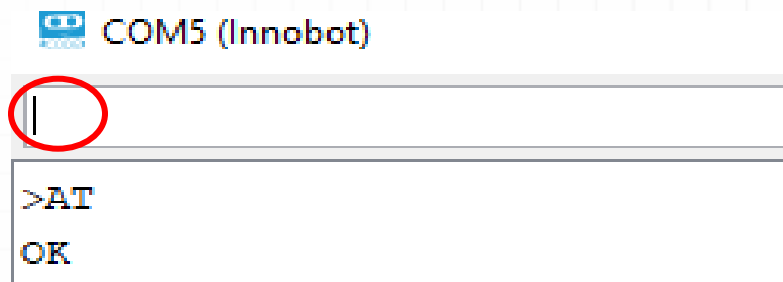
```

    NL = false;
  }
  Serial.write(c);
  if (c == 10) {
    NL = true;
  }
}
}


```

Una vez cargado el programa anterior abrimos el **monitor serie** .

LOS COMANDOS LOS ESCRIBIMOS EN LA SIGUIENTE PARTE:




PASO 1: escribimos “AT” y le damos “enter”.

 COM5 (Innobot)

AT|

Nos debe responder o siguiente:

 COM5 (Innobot)

|

>AT

OK

Si no responde “OK”, cierre y vuelva y abra el monitor serial o verifique el cable de comunicación.

PASO 2: preguntamos por el ROLE para saber si esta configurado como maestro a esclavo.

```
>AT+ROLE?
+ROLE: 1
OK
```

El ROLE esta en 1, es decir que esta como maestro y lo necesitamos como esclavo, entonces le damos el comando para cambiarlo a esclavo.

```
>AT+ROLE=0
OK
```

PASO 3: configuramos la velocidad a 9600 bauds.

```
>AT+UART=9600,0,0
OK
```

PASO 4: le ponemos un nombre al modulo, en este caso digamos que "ESCLAVO1"

```
>AT+NAME=ESCLAVO1
OK
```

PASO 5: preguntamos a ver si la clave es 1234.

```
>AT+PSWD?
+PSWD: 1234
OK
```

Efectivamente la **clave es 1234**.

PASO 6: preguntaremos la MAC.

```
>AT+ADDR?
+ADDR: 98d3:32:708b76
OK
```

Esta MAC la anotamos por que mas adelante se la vamos a ingresar al bluetooth maestro.

98D3:32:708B76

PASO 7: lo configuramos para que se conecte a dispositivo especifico

```
>AT+CMODE=0
OK
```

8. Ahora vamos a configurar el maestro.

Hacemos el mismo proceso de conexión que hicimos en el esclavo.

PASO 1: escribimos “AT” y le damos “enter”.

```
COM5 (Innobot)
```

```
AT
```

Nos debe responder o siguiente:

```
COM5 (Innobot)
```

```
|
```

```
>AT
```

```
OK
```

Si no responde “OK”, cierre y vuelva y abra el monitor serial o verifique el cable de comunicación.

PASO 2: le configuramos el ROLE de maestro.

```
>AT+ROLE=1
```

```
OK
```

PASO 3: configuramos la velocidad a la misma del esclavo 9600 bauds.

```
>AT+UART=9600,0,0
```

```
OK
```

PASO 4: le ponemos un nombre al modulo, en este caso digamos que “MAESTRO1”

```
>AT+NAME=MAESTRO1
```

```
OK
```

PASO 5: preguntamos a ver si la clave es 1234.

```
>AT+PSWD?
```

```
+PSWD:1234
```

```
OK
```

PASO 6: lo configuramos para que se conecte a un dispositivo en específico:

```
>AT+CMODE=0
```

```
OK
```

PASO 7: le damos la orden que se conecte con el esclavo por medio de la MAC.

```
>AT+BIND=98D3,32,708B76
```

```
OK
```


9. Ahora vamos a verificar si los dos bluetooth se están conectados.

PASO 1: conectamos el esclavo a la tarjeta innobot y el maestro a la kymo.

IMPORTANTE:

- La kymo debe estar en opción de teclado.
- Cuando cargue el programa para leer los valores debe tener conectada la Kymo a una fuente de alimentación diferente al pc donde esta conectada la tarjeta innobot.

Cuando hay conectividad ambos parpadean lento al mismo tiempo.

PASO 2: observaremos los valores que envía el maestro la **Kymo** al esclavo la tarjeta **PYGMO**.

Para eso cargamos el siguiente programa:

```
void setup () {  
  Serial1.begin (9600) ;  
}  
void loop () {  
  while (Serial1.available ()) {  
    char dato = Serial1.read ();  
    Serial.print (dato) ;  
  }  
}
```

Después abrimos el monitor serie y tocamos las flechas para ver que letras nos arrojan

En conclusión, tenemos para las flechas de la kymo:

- flecha la derecha= k
- flecha a la izquierda= l
- flecha hacia arriba= h
- flecha hacia abajo= j

10. Por último, le cargamos el programa al innobot para controlarlo con el análogo de la kymo.

```
void setup() {  
  Serial1.begin(9600);  
  motorSpeed(M2, 80);  
  motorSpeed(M3, 80);  
}  
void loop() {  
  while  
  (Serial1.available()) {  
    char dato =  
    Serial1.read();  
    switch(dato) {  
  
      case 'a': {  
        motorOn(M2, FORWARD);  
        motorOn(M3, FORWARD);  
        delay(20);  
        break;  
      }  
      case 'b': {  
        motorOn(M2, REVERSE);  
        motorOn(M3, REVERSE);  
        delay(20);  
        break;  
      }  
      case 'c': {  
        motorOn(M2, FORWARD);  
        motorOn(M3, REVERSE);  
        delay(20);  
        break;  
      }  
      case 'd': {  
        motorOn(M2, REVERSE);  
        motorOn(M3, FORWARD);  
        delay(20);  
        break;  
      }  
      default:  
        motorOff(M2);  
        motorOff(M3);  
    }  
    motorsOff(M2, M3);  
  }  
}
```

Nota: está es la programación para manejar la Kymo con el joystick.

Pygmalion[®]
Piensa • Construye • Aprende

NOTA: el Pygmo siempre va estar andando para que no ande más oprimen tecla “space”.

Valoración y cierre

11. Para finalizar invéntese una carrera para que finalicen la clase.
12. Recoja el material, resuelva dudas y organice el espacio de trabajo.

Tips se la sesión



- ✓ Verifique antes de conectar el bluetooth que los pines coincidan, GND con GND, +5V con VCC
- ✓ Las baterías deben de estar completamente cargadas.
- ✓ que NO haya dudas entre sus estudiantes, es preciso que tengan presente las reglas del reto.
- ✓ Invítelos a que mejoren su programación pensando en lógicas que aporten un mejor desempeño del robot en el reto.

Bibliografía

Módulo bluetooth HC05, tomado de: https://naylorpmechatronics.com/blog/24_configuracion-del-modulo-bluetooth-hc-05-usa.html