CASO PRÁCTICO

LEGO MINDSTORMS-NXT

CASTOR BOT GUIADO POR SONIDO

1.-Descripción general:

Descripción de actuadores y sensores

Elementos de control	Sensores	Actuadores
Señal del micrófono (dB)	Micrófono	Motor A
Potencia del motor A		Motor C
Potencia del motor C		CPU NXT
Sentido del giro de A		
CPU NXT		

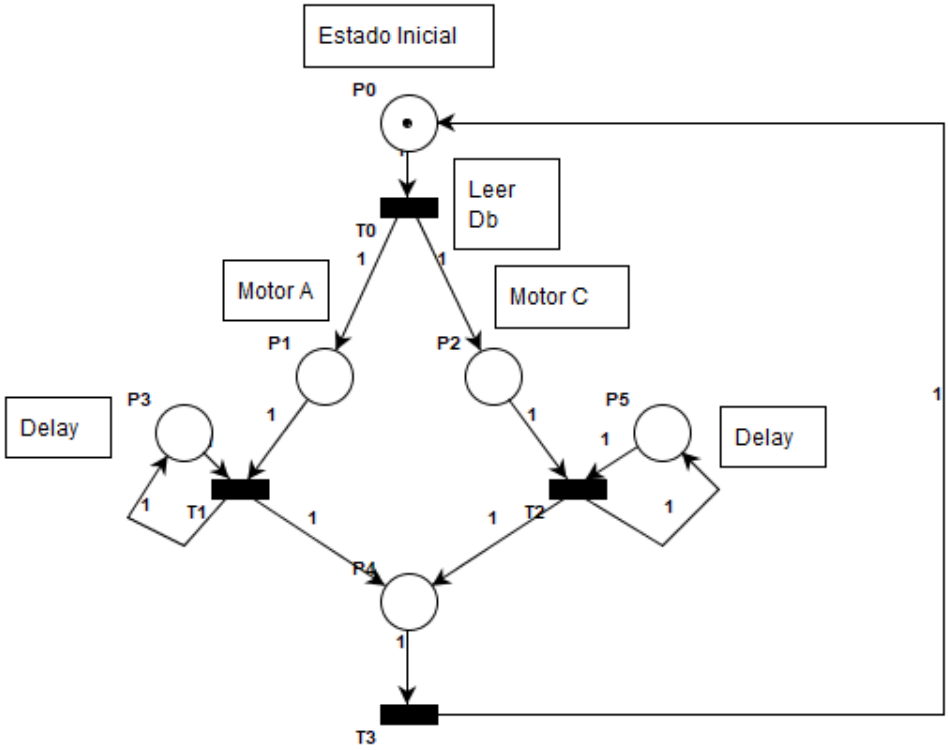
Requisitos Temporales:

Identificador	Descripción
RT1	La toma de medidas del micrófono se hará cada 100 ms
RT2	Cuando detecta un sonido de alta frecuencia e intensidad se realizará una parada de 5000 ms.
RT3	Cuando detecta un sonido de alta frecuencia y una intensidad de <50 dB se realiza una parada de 500 ms

Requisitos Funcionales:

Identificador	Requisito
RF1	Reaccionar a la pulsación de inicio en el NXT
RF2	Lectura del sensor de sonido
RF3	Activación de los motores tras RF1
RF4	Ajuste del sentido de giro de los motores tras RF2

Red de Petri:



Software (Source Code):

```
procedure Tests is
--Procedimiento que ya venía en esta plantilla y que nos permite esperar a la pulsación de una tecla para interactua con el NXT
procedure Wait_Any_Key is
     use         NXT;
begin
     loop
        exit when Current_Button = No_Button;
delay until Clock + Milliseconds (10);       end
loop;
     loop
        exit when Current_Button /= No_Button;
delay until Clock + Milliseconds (10);       end
loop;
   end Wait_Any_Key;

   Micro : Unsigned_16; --Variable Micro para tomar las medidas del sensor     lectura: Unsigned_16 := 0; --Variable auxiliar que nos permitirá actuar con el sensor saltándonos el fuertemente tipado
   tiempo_espera : Time; --Variable que utilizaremos para las esperas
```

```
begin
   Clear_Screen_Noupdate; --Borrado de pantalla del visor de NXT
   Put_Line ("Comenzar"); --Escritura sobre el visto de pantalla del NXT
   Wait_Any_Key;--Espera la pulsación de algún botón del NXT para comenzar
   Clear_Screen_Noupdate; --Borrado de pantalla del visor de NXT
   Put_line ("Lecturas"); --Escritura sobre el visor de pantalla del NXT

  Set_Power(Motor_A, -35, true); --Iniciamos el motor A a una potencia de -35 para realizar la rotación sobre su propio eje
   Set_Power(Motor_C, 25, true);--Iniciamos el motor C a una potencia de 25 tiempo_espera := Clock + Milliseconds(100);--Para realizar el delay necesitamos una variable auxiliar porque ADA es un lenguaje fuertemente tipado delay until (tiempo_espera);--Espera de 100 milisegundos

  loop--Bucle que irá tomando las medidas del sensor y actuando en consecuencia Micro := Raw_Input(Sensor_Id(Sensor_2));--Lectura de la medida del micro que entra por el puerto 2
     tiempo_espera := Clock + Milliseconds(100);--Configuramos la espera a 100 milisegundos, necesaria para tomar datos fiables del sensor
     delay until (tiempo_espera);--Realizamos la espera de 100 milisegundos lectura := lectura + Micro;--Transformamos la lectura en un valor unsigned_16 con el que si podemos trabajar debido al fuerte tipado de ADA
     lectura := lectura / 12;--Reducimos el valor del micro a una valor con el que podamos trabajar, puesto que la lectura da valores demasiado elevados


if (lectura > 50) and (lectura <= 100) then--Si la lectura está entre 50 y 100dB Set_Power(Motor_C, 50, true);--Incrementamos la velocidad del motor C a 50 tiempo_espera := Clock + Milliseconds(500);--Lo dejamos moverse 500 milisegundos
      delay until (tiempo_espera);--Realizamos la espera de 500 milisegundos end if;--Fin del condicional if
    if lectura <= 50 then--Si la lectura es menor o igual a 50 dB
      Set_Power(Motor_A, 25, true);--El motor A se fija a una potencia de 25 y el robot camina en su dirección
      tiempo_espera := Clock + Milliseconds(5000);--Fijamos una espera de 5 segundos, para que el robot no bloquee sus ruedas y le de tiempo a avanzar
delay until (tiempo_espera);--Realizamos la espera de 5 segundos end if;--Fin del condicional if
    --Volvemos al estado inicial de los motores
    Set_Power(Motor_A, -35, true);--Reestablecemos el motor A a una potencia de -35    Set_Power(Motor_C, 25, true);--Reestablecemos el motor C a una potencia de 25   end loop;--Fin del bucle end Tests;--Fin del archivo Tests
```