

ADAPTACIÓN DE UNA SILLA DE RUEDAS ELECTRICA PARA NIÑOS POLIDISCAPACITADOS

Félix Candela Fernández
Tutores :
Romuald Stock
Georges Stentz



Índice

Objetivo del proyecto2

Desarrollo del proyecto.....2

Componentes0

 Sensor de ultrasonidos HC-SR040

Señal del motor1

Programación1

Branchements ¡Error! Marcador no definido.

Objetivo del proyecto

- Reemplazar el joystick de una silla de ruedas eléctrica por un dispositivo que se adapte a las capacidades de niños poli discapacitados.
- Desarrollar una interfaz que muestre los movimientos de la silla de ruedas para hacer posible el aprendizaje de los niños.

Desarrollo del proyecto

El primer problema que había que plantear al empezar el proyecto fue el soporte en que se iba a fijar nuestro dispositivo. Después de una visita al centro con el que trabajábamos, vimos que utilizaban unas tablas de plástico que se fijaban a las sillas de ruedas. Decidimos utilizar esas tablas ya que los niños ya estaban habituados a ella y así cuando aprendiesen a usar el dispositivo, ya estarían familiarizados con ellas. La imagen de debajo muestra una de ellas.



El siguiente paso, fue elegir el dispositivo que hiciese la función para controlar la silla.

De entre muchos dispositivos posibles, por ejemplo, varios sensores de luz, sonido, voz o calor, se eligió un sensor de proximidad para manejar la silla de ruedas.

Para la interfaz que mostrase los movimientos se decidió hacer unas flechas con LEDs que se encendiesen cada vez que se moviese la silla de ruedas. Era importante que cada flecha tuviese un color diferente, y

que mostrase el movimiento por lo que los LEDs con forma de flecha se cubrieron con un plástico del color necesario.

Para ver la montabilidad, el buen funcionamiento de los componentes y que nuestra herramienta se adecuaba a las necesidades de los niños se fabricó un prototipo en una tabla de madera de las mismas dimensiones que el diseño final. Más abajo se pueden ver dos fotos de la tabla prototipo.



De la fabricación del prototipo se sacaron las siguientes conclusiones:

- Se necesitaba una tapa para cubrir todos los circuitos del útil y así evitar que los cables de soltasen o algún daño en los sensores.
- Las flechas y los sensores debían estar en la misma posición para mejorar el aprendizaje. En los ensayos hechos con los niños, iban a pulsar directamente sobre la flecha.
- La batería debe ser accesible para facilitar la recarga y el encendido del sistema.

Con estos tres puntos y además, añadiendo un potenciómetro para regular la distancia a la cual el sensor detectaba los objetos, de cara a el aprendizaje de los niños, se fabricó el diseño final.

Componentes

Sensor de ultrasonidos HC-SR04

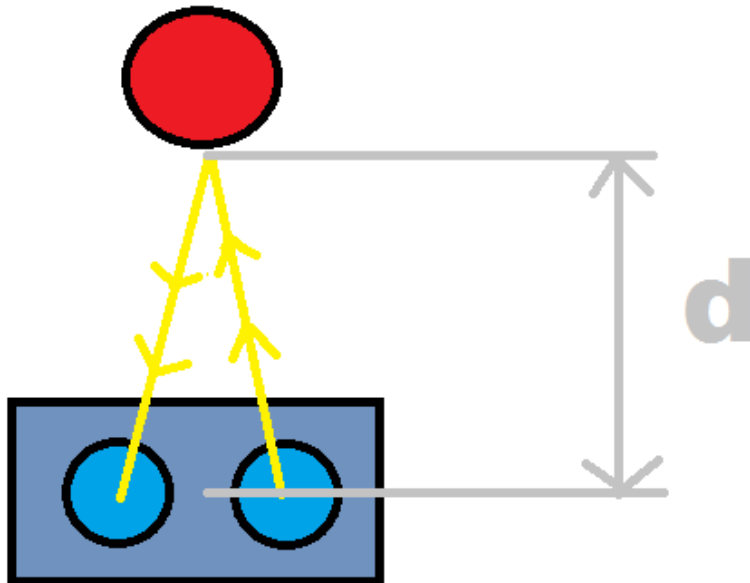


Hemos elegido este sensor por:

- Es fácilmente implementable con el lenguaje de programación de Arduino.
- Tiene un precio contenido, lo que facilitaría utilizarlo en una escala mayor.

- Es pequeño y se puede posicionar en muchos sitios en nuestra tabla.

Este sensor puede detectar objetos a distancias entre 3 cm y 3 m. Consta de dos cilindros, echo y trigger, uno para enviar la señal de ultrasonidos y otro para recibirla. Midiendo el tiempo que tarda la señal en ir y volver obtenemos la distancia utilizando la velocidad del sonido.



Señal del motor

Para conseguir hacer funcionar el motor, lo que se hizo fue conectar el joystick a un voltímetro y medir las tensiones que se enviaban en cada caso. Sustituyendo los cables del joystick por unos provenientes de la placa Arduino y que enviaba las tensiones obtenidas en función del sensor que se activase con la mano se consiguió desplazar la silla de ruedas. Más abajo una imagen que muestra la sustitución de los cables.

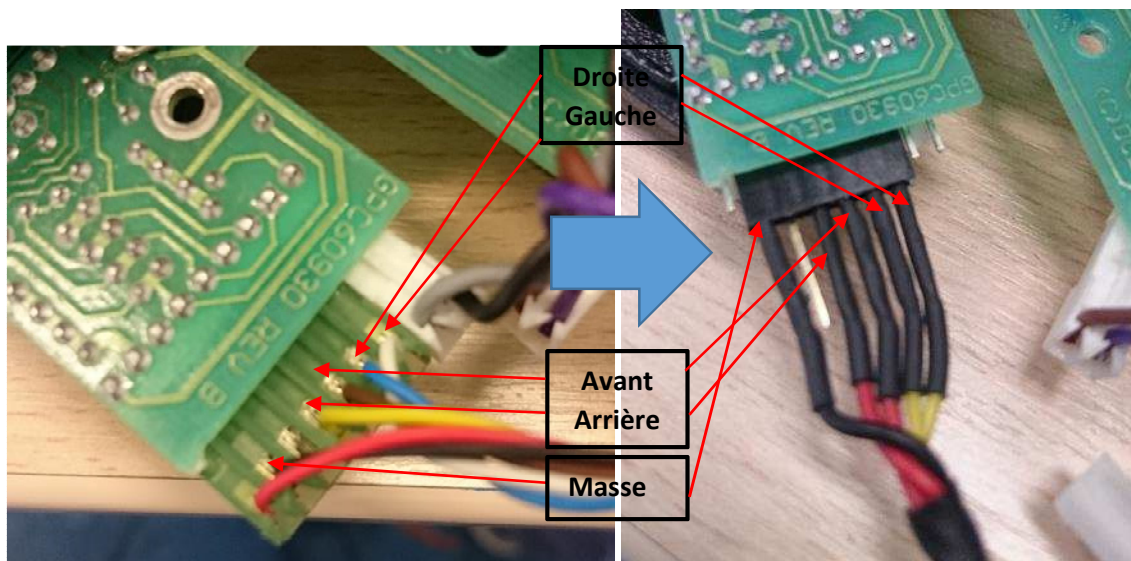
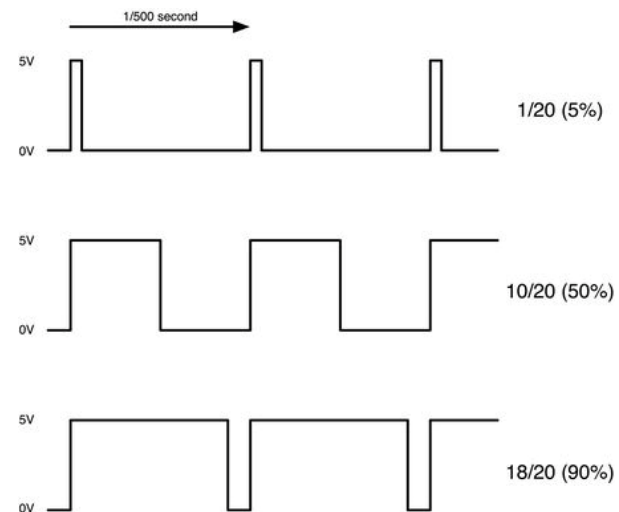


Diagram illustrating the color wheel layout for the test. The center is labeled 'C'. The quadrants are labeled: AV (top), AR (bottom), GA (left), and DR (right).

	Centre	Avant	Arrière	Gauche	Droite
Jaune	2.5 V	4 V	0.8 V	2.5 V	2.5 V
Brun	2.5 V	1.15 V	4 V	2.5 V	2.5 V
Bleu	2.5 V	2.5 V	2.5 V	0.8 V	4 V
Blanc	2.5 V	2.5 V	2.5 V	4 V	0.8 V

The diagram shows a circuit with a resistor in series with a parallel combination of a capacitor and a battery. The input voltage V is a square wave with amplitude 5 and period T , with pulse width a . The output voltage V is shown as a constant value of $\frac{a}{T} 5$.



Programación

En esta sección vamos a ver el programa que utilizado para la tarjeta que controla el funcionamiento de las luces y que da la tensión necesaria para mover la silla de ruedas.

Para empezar la definición de las variables:

```
int ledA = 14, ledG = 15, ledD = 16, ledR = 17, trigG = 4, echoG = 5, trigA = 2, echoA = 3,
trigD = 7, echoD = 8, jaune = 11, brun = 10, bleu = 9, blanc = 6, ledj = 18, pot, h;
long tempsG, distanceG, tempsA, distanceA, tempsD, distanceD;
void setup()
{
  //On define les LEDs comme OUTPUTs.
  pinMode (ledA, OUTPUT);
  pinMode (ledG, OUTPUT);
  pinMode (ledD, OUTPUT);
  pinMode (ledR, OUTPUT);
  pinMode (ledj, OUTPUT);
  //Les TRIGs des capteurs seront sorties puisqu'on envoie l'impulsion avec eux.
  pinMode (trigG, OUTPUT);
  pinMode (trigA, OUTPUT);
  pinMode (trigD, OUTPUT);
  //Les ECHOs des capteurs seront des entrées pour recevoir l'impulsion.
  pinMode (echoG, INPUT);
  pinMode (echoA, INPUT);
  pinMode (echoD, INPUT);
  //Les fils pour la commande.
  pinMode (bleu, OUTPUT);
  pinMode (jaune, OUTPUT);
  pinMode (blanc, OUTPUT);
  pinMode (brun, OUTPUT);
  Serial.begin(9600);
}
```

La définition des variables doit être dans une seule ligne, de cette la visualisation est plus facile.

A continuación, el funcionamiento del potenciómetro:

```
void loop()
{
  //En fonction de la signal qui retourne du potentiometre le valeur de la distance de detection changera
  pot=analogRead(19);
  if (pot>=800)
  {
    h=3;
  }
  else if (pot>=600)
  {
    h=6;
  }
  else if (pot>=400)
  {
    h=9;
  }
  else if (pot>=200)
  {
    h=12;
  }
  else
  {
    h=15;
  }
  delay (100);
  digitalWrite(ledj, HIGH);
}
```

Más abajo el envío de la señal de los sensores y la conversión del tiempo recibido en distancia:

```
//Capteur Ultra son droite
digitalWrite(trigD, LOW);
delayMicroseconds(5);
digitalWrite(trigD, HIGH);
delayMicroseconds(10);
digitalWrite(trigD, LOW);
tempsD = pulseIn(echoD, HIGH);
delayMicroseconds(20);

//Capteur Ultra son gauche
digitalWrite(trigG, LOW);
delayMicroseconds(5);
digitalWrite(trigG, HIGH);
delayMicroseconds(10);
digitalWrite(trigG, LOW);
tempsG = pulseIn(echoG, HIGH);
delayMicroseconds(20);

//Capteur Ultra son Avant
digitalWrite(trigA, LOW);
delayMicroseconds(5);
digitalWrite(trigA, HIGH);
delayMicroseconds(10);
digitalWrite(trigA, LOW);
tempsA = pulseIn(echoA, HIGH);
delayMicroseconds(20);

//On change le temps retourné en distance parmi la vitesse du son.
distanceG = 0.017 * tempsG;
distanceA = 0.017 * tempsA;
distanceD = 0.017 * tempsD;
```


Para terminar, con los datos obtenidos de los sensores, se encienden los LEDs y se da una tensión diferente en cada caso:

```
//Marche arrière
if (distanceG < h && distanceD < h && distanceA > h)
{
//On allume les LEDs necessaires
digitalWrite(ledG, LOW);
digitalWrite(ledA, LOW);
digitalWrite(ledD, LOW);
digitalWrite(ledR, HIGH);
//On envoie les tension precises
analogWrite(jaune, 46);
analogWrite(brun, 201);
analogWrite(bleu, 130);
analogWrite(blanc, 130);
}
//Gauche
else if (distanceG < h && distanceD > h && distanceA > h)
{
digitalWrite(ledG, HIGH);
digitalWrite(ledA, LOW);
digitalWrite(ledD, LOW);
digitalWrite(ledR, LOW);
analogWrite(jaune, 130);
analogWrite(brun, 130);
analogWrite(bleu, 46);
analogWrite(blanc, 201);
}
```

```

//Droite
else if (distanceG > h && distanceD < h && distanceA > h)
{
    digitalWrite(ledG, LOW);
    digitalWrite(ledA, LOW);
    digitalWrite(ledD, HIGH);
    digitalWrite(ledR, LOW);
    analogWrite(jaune, 130);
    analogWrite(brun, 130);
    analogWrite(bleu, 201);
    analogWrite(blanc, 46);
}
//Avant
else if (distanceA < h && distanceG > h && distanceD > h)
{
    digitalWrite(ledG, LOW);
    digitalWrite(ledA, HIGH);
    digitalWrite(ledD, LOW);
    digitalWrite(ledR, LOW);
    analogWrite(jaune, 201);
    analogWrite(brun, 46);
    analogWrite(bleu, 130);
    analogWrite(blanc, 130);
}

```

Pour éviter possibles fautes de programmation et en cas de repos :

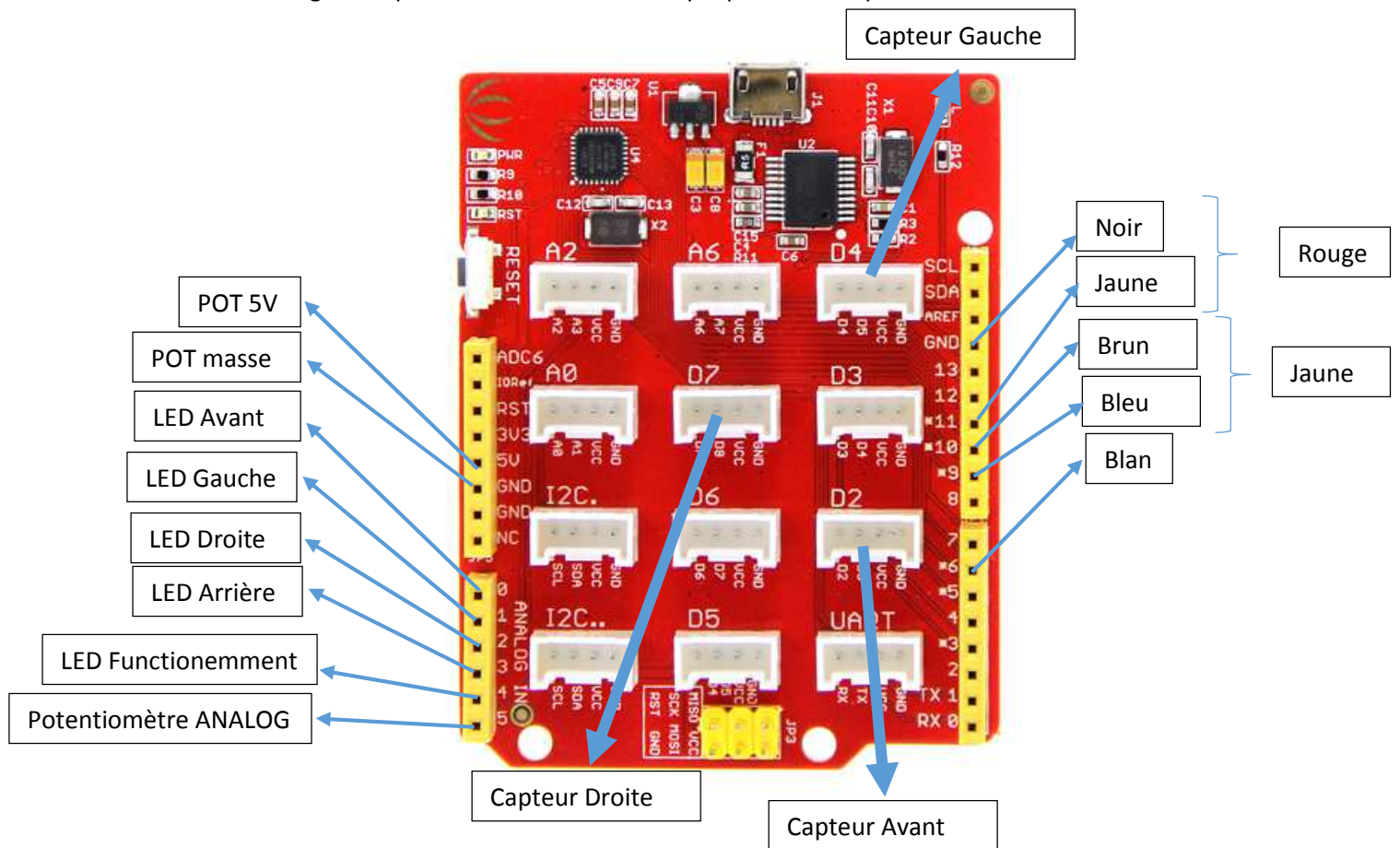
```

//Autres cas
else
{
    digitalWrite(ledG, LOW);
    digitalWrite(ledA, LOW);
    digitalWrite(ledD, LOW);
    digitalWrite(ledR, LOW);
    analogWrite(jaune, 130);
    analogWrite(brun, 130);
    analogWrite(bleu, 130);
    analogWrite(blanc, 130);
}

```

Cableado

En esta figura se presentan las conexiones que parten de la placa arduino:



Producto final

Aquí vemos algunas fotos del producto final:

