**Sensor de orientación automática para un monitor**

# Introducción

El objetivo de este proyecto es crear un sensor para cambiar automáticamente la orientación de la pantalla de un ordenador. Dicho cambio, se realizará una vez el ángulo de la pantalla pase de unos valores en concreto, momento cuando se mandará la señal por UART al ordenador para cambiar de orientación.

Para obtener el ángulo del monitor se usará un sensor de aceleración analógico para medir la dirección de la gravedad. Además, habrá un botón para recalibrar la orientación que se toma como 0 grados.

# Acondicionamiento del acelerómetro

El acelerómetro que se va a usar es el ADXL335, que es un acelerómetro de 3 ejes, el cual puede medir aceleraciones de hasta 3g en las tres direcciones del espacio, tanto en sentido positivo como negativo.

Para determinar el ángulo de la pantalla, se necesita medir la aceleración en dos ejes, en este caso se han elegido X e Y, cuyas direcciones se representan en el propio acelerómetro, tal como se puede ver en Ilustración 1.

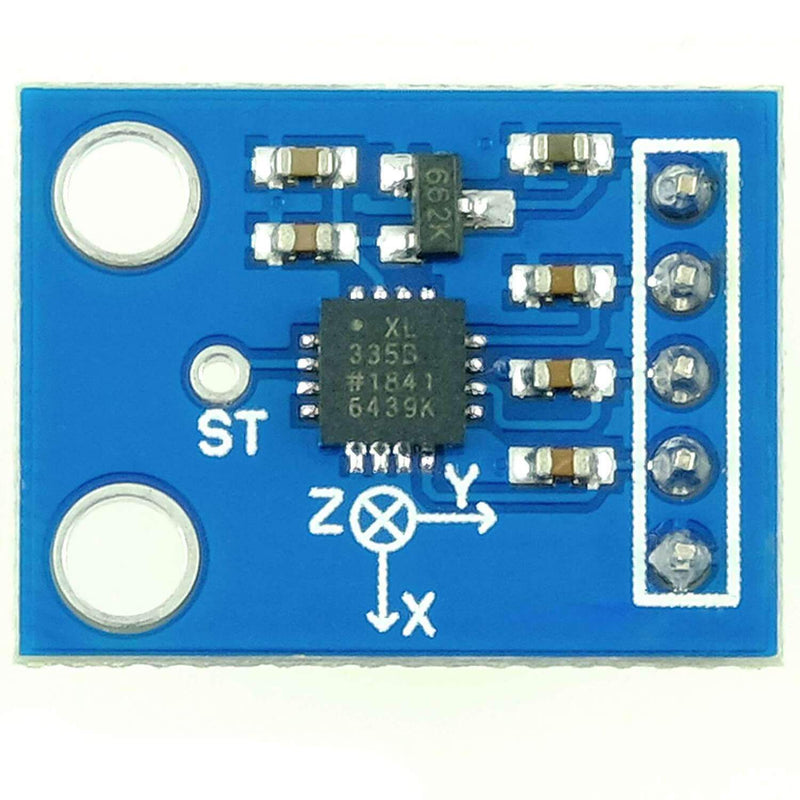


Ilustración 1 ADXL335

En el proyecto, el rango de medida de la aceleración va a ser ±g (± 10 m/s2), y la tensión que dan los pines en función de la aceleración que hay en su eje correspondiente es la que se ve en Ilustración 3, en la cual se ve que la relación entre la entrada y la salida es lineal.

Ilustración 2 Curva de calibración acelerómetro ADXL335

Como la sensibilidad es baja y no se aprovecha todo el rango de tensión del dsPIC, se va a acondicionar el acelerómetro haciendo un ajuste de ganancia utilizando el circuito mostrado en Ilustración 2, donde los valores de los componentes son:

* E = 1.65V
* R1 = 1 kΩ
* R2 = 5.5 kΩ
* Amplificadores operacionales MCP6002 alimentados entre 3.3V-0V.

Diagrama

Descripción generada automáticamente

Ilustración 3 Acondicionamiento acelerómetro

Para poder distinguir entre aceleraciones positivas y negativas, se ha ajustado E para tener Vo=1.65V cuando no hay aceleración, y la ganancia (-R2/R1), se ha ajustado para tener Vo = 0 cuando la aceleración es máxima (1 g) y Vo = 3.3V cuando la aceleración es mínima (-1g), de forma que la ecuación de la tensión de salida del circuito de acondicionamiento es ( 1 ).

( 1 )

La tensión de salida es lineal respecto a la aceleración, pero la pendiente es negativa. Esto no supone ningún problema, ya que se hará la respectiva corrección cuando se obtenga de forma digital la aceleración a partir de la medida de Vo.

El acelerómetro se ha conectado al circuito de acondicionamiento a través de un buffer para aislarlo.

Se necesitan montar dos circuitos de acondicionamiento, uno para el pin correspondiente al eje X y otro para el pin correspondiente al eje Y.

# Parte digital

Una vez acondicionados los sensores, se procede a diseñar la parte digital del proyecto. Esta parte se encargará de leer los sensores, calcular la posición, enviar dicha posición por la UART y representar el cuadrante en el que se encuentra la posición en unos LEDs. Además, se incluirá un botón de posición 0, cada vez que el usuario lo pulse, el dspic tomará el ángulo que dan los sensores en ese instante como el ángulo 0.

Se utilizará el sistema operativo en tiempo real FreeRTOS, en el cual se incluirán 4 tareas:

* LeerAcel: Esta tarea se encarga de leer las entradas analógicas a las cuales están conectados los circuitos de acondicionamiento del sensor (a AN0 se conecta el eje Y y al AN1 el eje X) y se calcula el ángulo, el cual se guarda en una variable global. Si el ángulo cambia más de una cierta tolerancia con respecto a la medida anterior, se libera el semáforo que permite ejecutar la tarea ComputarOri. Meter lo del botón!!!!!!
* ComputarOri: Establece la orientación del sensor a partir del ángulo según Ilustración 4, y la guarda en una variable global. Si la orientación cambia respecto a la orientación de la computación anterior, se liberan los semáforos para ejecutar las tareas MostrarOri y EnviarOri.

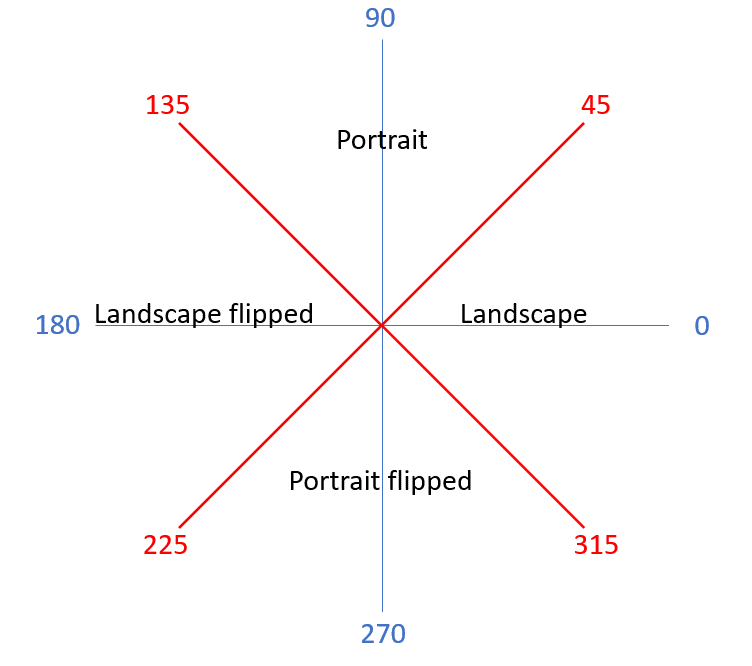


Ilustración 4 Orientaciones

* MostrarOri: Muestra en los LEDs conectados al dspic la orientación en ese momento, siguiendo Tabla 1.

|  |  |
| --- | --- |
| **Orientación** | **LED** |
| Landscape | RB12 |
| Portrait | RB13 |
| Landscape flipped | RB14 |
| Portrait flipped | RB15 |

Tabla 1 Asociación orientaciones con LEDs

* EnviarOri: Envía por la UART la orientación computada por ComputarOri.

En Ilustración 5 se muestra de forma gráfica las relaciones entre las tareas y su nivel de prioridad asociado. A LeerAcel se le ha asociado el nivel de prioridad más bajo porque es una tarea que se va a tener ejecutándose todo el rato, no se bloquea. A las tareas que sí se bloquean se les va asociando niveles de prioridad cada vez más altos para que se ejecuten según se desbloqueen y, cuando terminan de ejecutarse, ceden el semáforo correspondiente y se vuelven a bloquear.

Diagrama

Descripción generada automáticamente

Ilustración 5 Relación entre tareas y niveles de prioridad