

Proyecto III

Computer Engineering

Star Force Remake, May the Force Wireless Control Be With You

Trying to read our DNA is like trying to understand software code - with only 90% of the code riddled with errors. It's very difficult in that case to understand and predict what that software code is going to do.

Elon Musk

To make an embarrassing admission, I like video games. That's what got me into software engineering when I was a kid. I wanted to make money so I could buy a better computer to play better video games - nothing like saving the world.

Elon Musk

My vision is for a fully reusable rocket transport system between Earth and Mars that is able to re-fuel on Mars - this is very important - so you don't have to carry the return fuel when you go there.

Elon Musk

Introduction

The third computer programming project is a video game controller for the Project II based on the game Star Force. A video game controller (joystick, control, steering wheel, guns, etc.) is an input peripheral used to control video games.

Introducción

El tercer proyecto programado es un controlador de videojuegos para el Proyecto II basado en el juego Star Force. Un controlador de videojuegos (palanca de mando, mando, volante, pistolas, etcétera) es un periférico de entrada usado para controlar videojuegos



Figura 1. Versión de Tekhan del juego Star Force, corría en un MP Z80 de 4MHz [3].

Descripción

Objetivo principal:

- Realizar un proyecto que involucre las dos áreas de la carrera, tanto software como hardware, y la interacción entre las mismas.

Objetivos secundarios:

- Poner en práctica los conceptos aprendidos en los talleres brindados.
- Utilizar y adaptarse a un nuevo lenguaje de programación sencillo.
- Poner en práctica las habilidades de construcción de circuitos simples y la soldadura.
- Trabajo en equipo.

Normalmente, un controlador está conectado a una consola de videojuegos (videoconsola) o a un computador. El proyecto III consiste en la integración de un mando de control al proyecto II. En este documento se describe las funcionalidades mínimas con las que deberá contar este control. En este proyecto usted deberá implementar los circuitos que se adjuntan y adaptar el código del proyecto 2 para que acepte las señales que envía el control, como así enviar a su vez las señales necesarias para presentar los datos en el control.

El diagrama 1 muestra a nivel general las conexiones que se deben hacer. Note que el Arduino no se encuentra dentro del control, este solo funciona de interfaz entre el

control y la computadora. De esta manera si se deseara cambiar de microcontrolador, o que otro dispositivo aproveche las características del control no habría ningún inconveniente.

Este documento se basa en este diagrama 1, para ir explicando cada uno de los componentes. Además al final del documento se adjuntan una serie de enlaces que van a ayudar al estudiante en la comprensión y realización del trabajo.

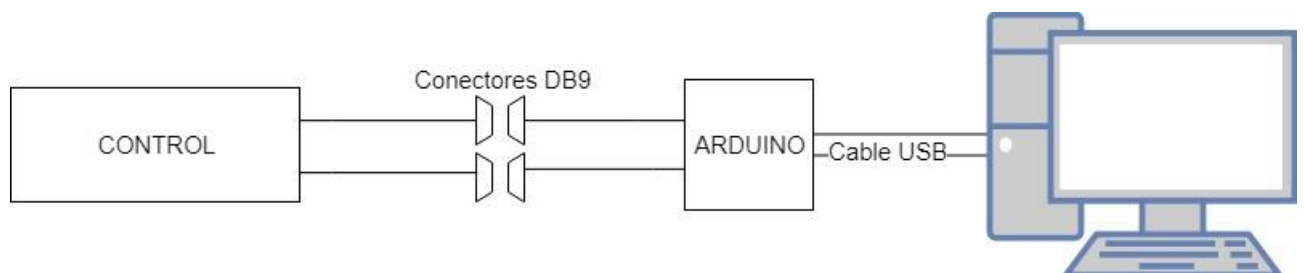


Diagrama 1, modelo general de la conexión del proyecto

Control

En la figura 2 se presenta la distribución de los componentes del control. Recuerde que por dentro debe tener el módulo MPU-9250. Esta distribución puede ser cambiada a conveniencia del estudiante, pero las salidas y las entradas deben ser las mismas.

El control se compone de:

- 2 botones para aceptar y cancelar.
- 6 leds que indican el nivel de energía en el juego
- 1 display 7 segmentos que muestra la cantidad de obstáculos o asteroides que restan en la partida.
- 1 potenciómetro para seleccionar opciones.
- 1 módulo giroscopio, acelerómetro y magnetómetro que permite dirigir la nave del juego en los ejes X y Y.

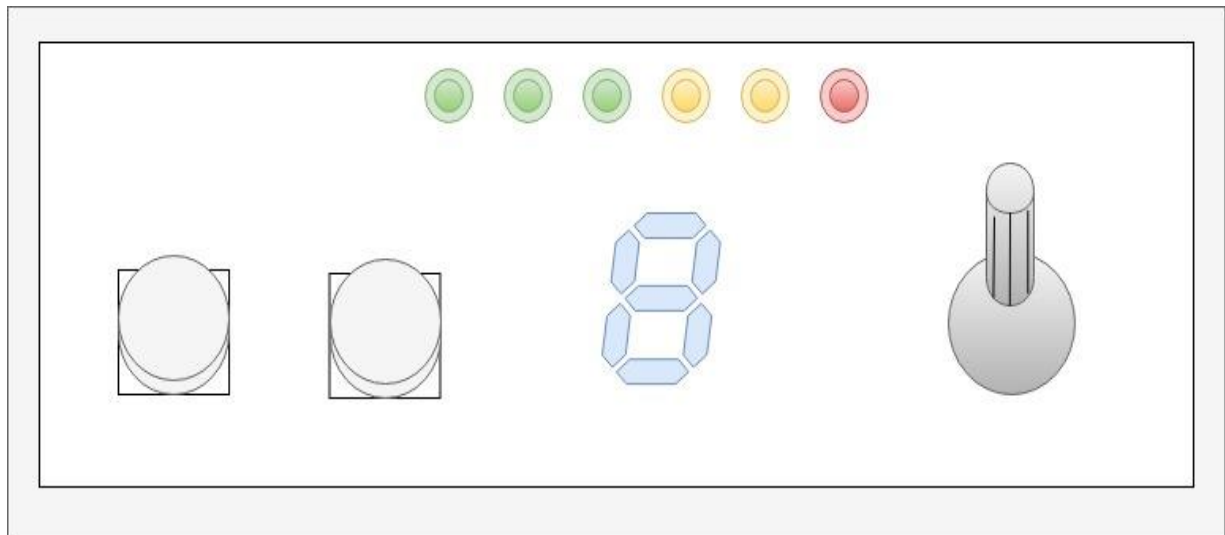


Figura 2, recomendación de la distribución de los componentes.

Módulo MPU-9250

Este módulo permite conocer la posición de un objeto en el espacio, el mismo retorna los valores de los ejes X, Y y Z. En el proyecto usted deberá interpretar estos valores y adecuarlos al movimiento de la nave.

Se recomienda interpretar los valores con rangos y si el rango se cumple llamar a la función que mueve la nave, al igual que se hace cuando se presiona una tecla.

NOTA IMPORTANTE: este módulo trabaja con 3.3V y el Arduino trabaja con 5V, a continuación se presenta una forma de conectarlo, cabe resaltar que esta configuración la permite el Arduino pero NO es la forma más segura de adaptar los diferentes niveles de tensión.

También se especifican los conectores que se deben usar y a cuales pines conectarlos.

Una buena práctica es utilizar un zócalo o base para circuitos integrados o usar varias contiguas y cortarla de ser necesario para que se ajuste a lo largo del módulo y de esta manera no suelda directamente el módulo a la placa preperforada, sino que se suelda el zócalo (o base) y el integrado se puede reutilizar de ser necesario.



Figura 3. Muestra de un zócalo para circuito integrado de 16 pines (8 por lado).

Recuerde que las señales que se enviarán corresponden a 2 de los ejes X, Y, Z. El tercero queda comentado en el código.

La comunicación será por medio de los pines scl y sda, los mismos trabajan con I2C, que es un protocolo de comunicación de dispositivos.

En la Figura 4 se presenta como se debe realizar la conexión entre el Arduino, el módulo MPU 9250, las resistencias de pull up y los pines de comunicación. Así debe permanecer la conexión cuando se pase a placa preperforada.

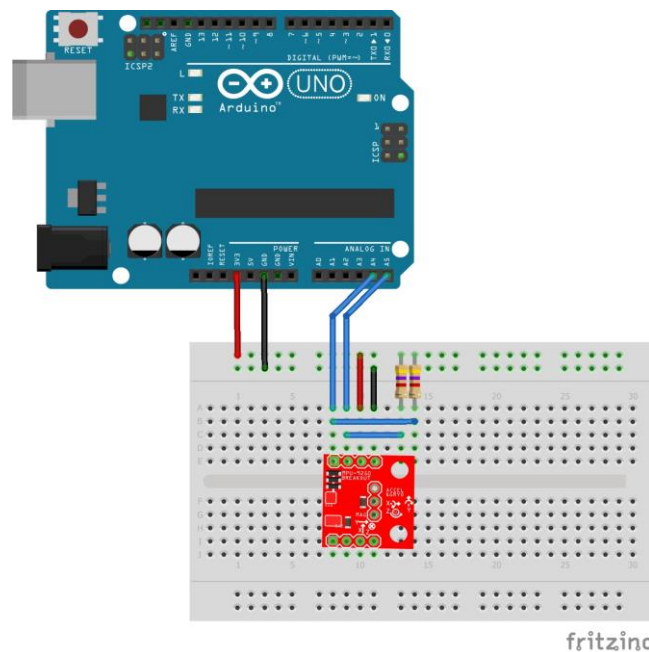


Figura 4. Conexión física del Arduino con el Módulo MPU 9250

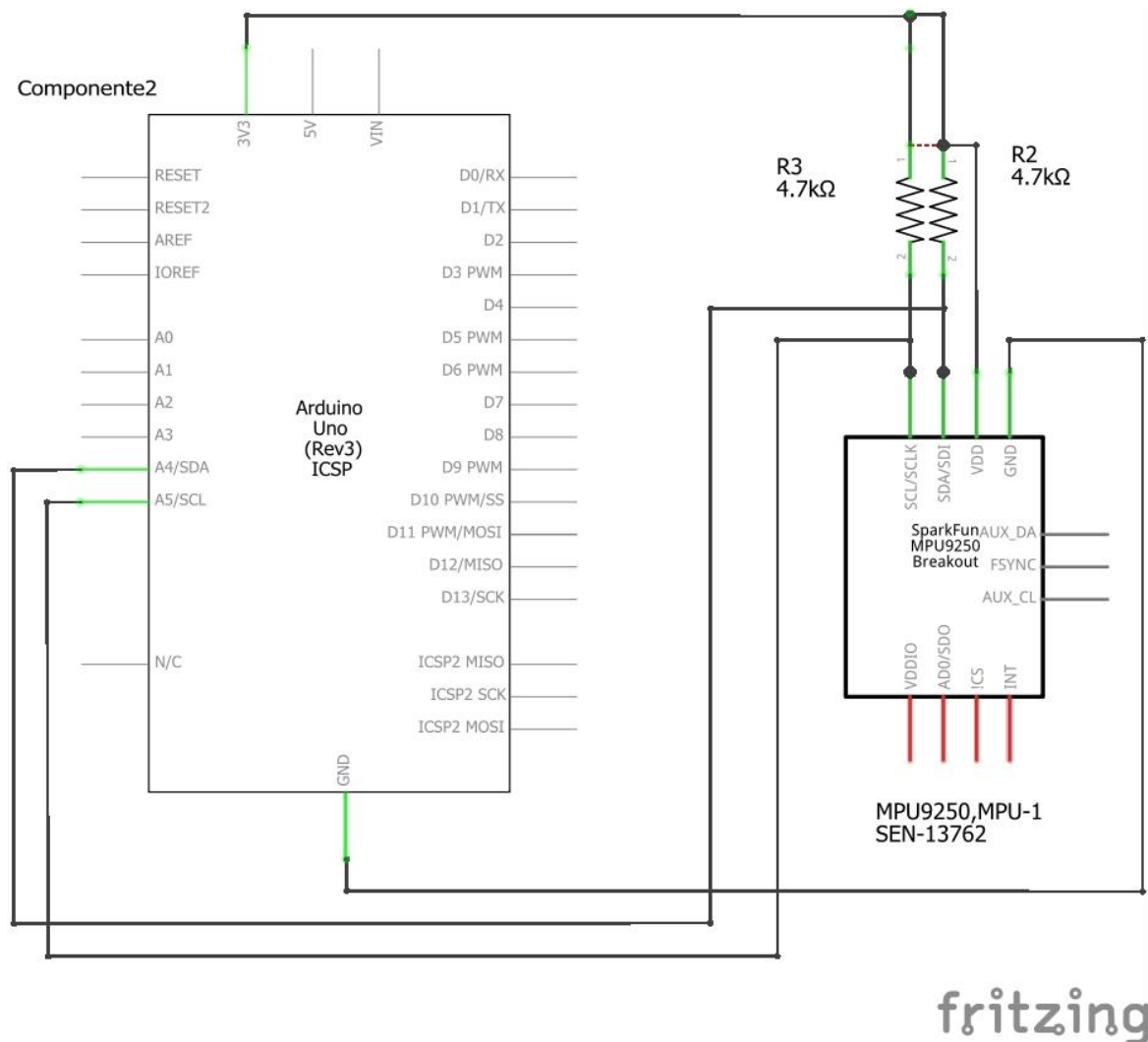


Figura 5. Diagrama electrónico de conexión entre el Arduino y el módulo MPU9250.

Conectores DB9



Figura 6. Conectores DB9 macho y hembra parte anterior y posterior.

Este conector va a permitir agrupar los cables de conexión en el Arduino y su complemento se puede soldar las terminales del control a una estructura fija.

En el proyecto se utilizarán dos conectores hembra para la conexión entre el Arduino y el control. La distribución de pines se muestra en la Figura 7:

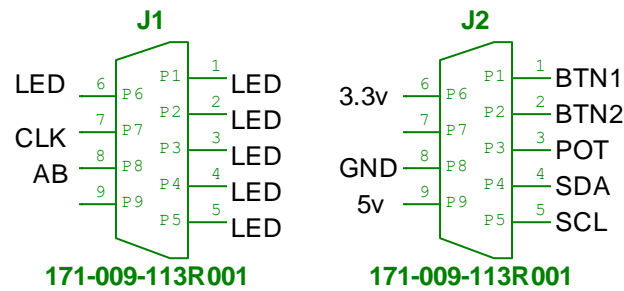


Figura 7. Diagramas de pines de los conectores hembra DB9.

6 leds indicadores.

Para los indicadores de energía se van a utilizar 6 leds, 3 verdes que indican que la energía está por encima del 50%, 2 amarillos para indicar que la energía está alrededor del 30% y 1 rojo cuando se encuentra al 10% o menos. Los colores y porcentajes pueden variar, pero se debe documentar correctamente el cambio.

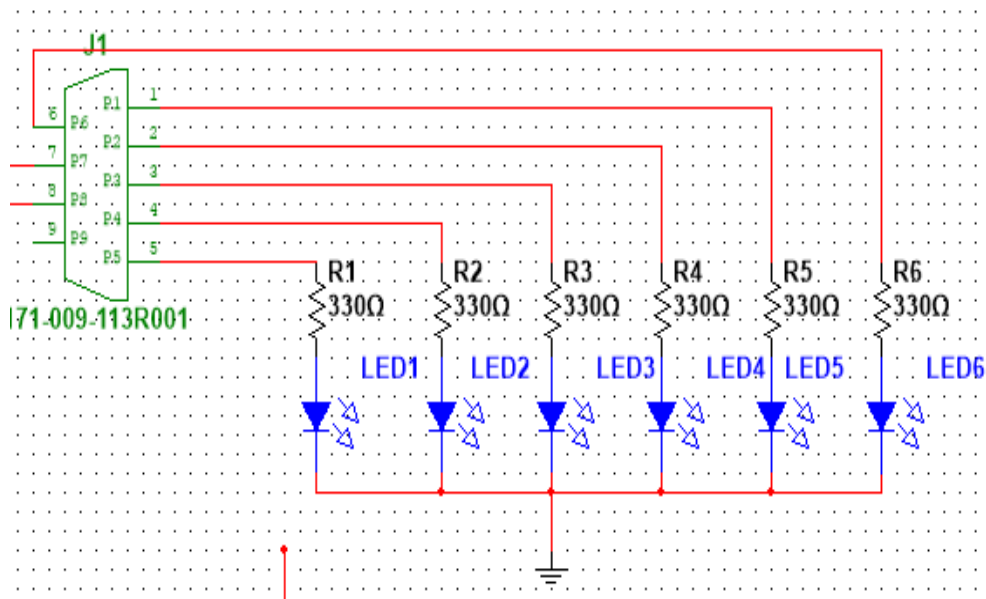


Figura 8. Diagrama electrónico de conexión de los pines del DB9 y los LEDs.

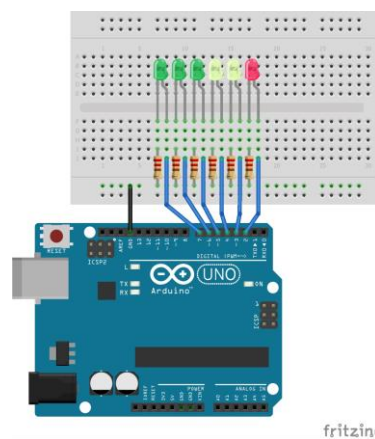
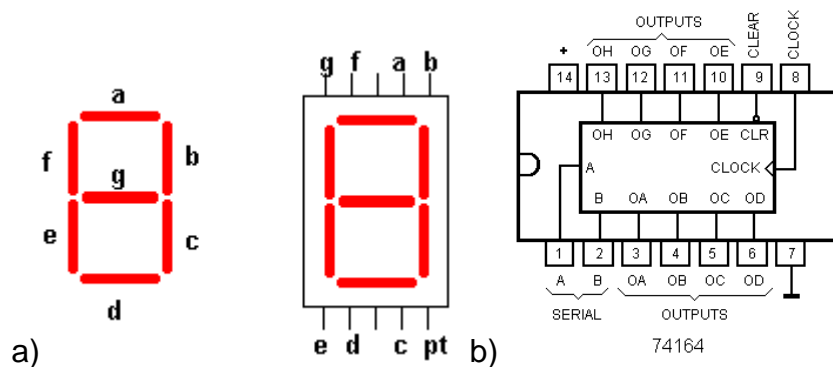


Figura 9. Conexión física entre el Arduino y los 6 LEDs en una *protoboard*

Display 7 segmentos

Este display indica la cantidad de obstáculos o asteroides que restan en la partida. Se debe utilizar un registro de corrimiento (circuito integrado) para controlar los leds ya que no son suficientes los pines del Arduino uno para los 7 segmentos del display. El circuito que se utilizará es el 74LS164.

Importante este circuito trabaja con 5V y no con 3.3V así que en la alimentación se deben usar los pines de 5v.



a) Diagrama de un display 7 segmentos b) Diagrama de un circuito integrado 74LS164

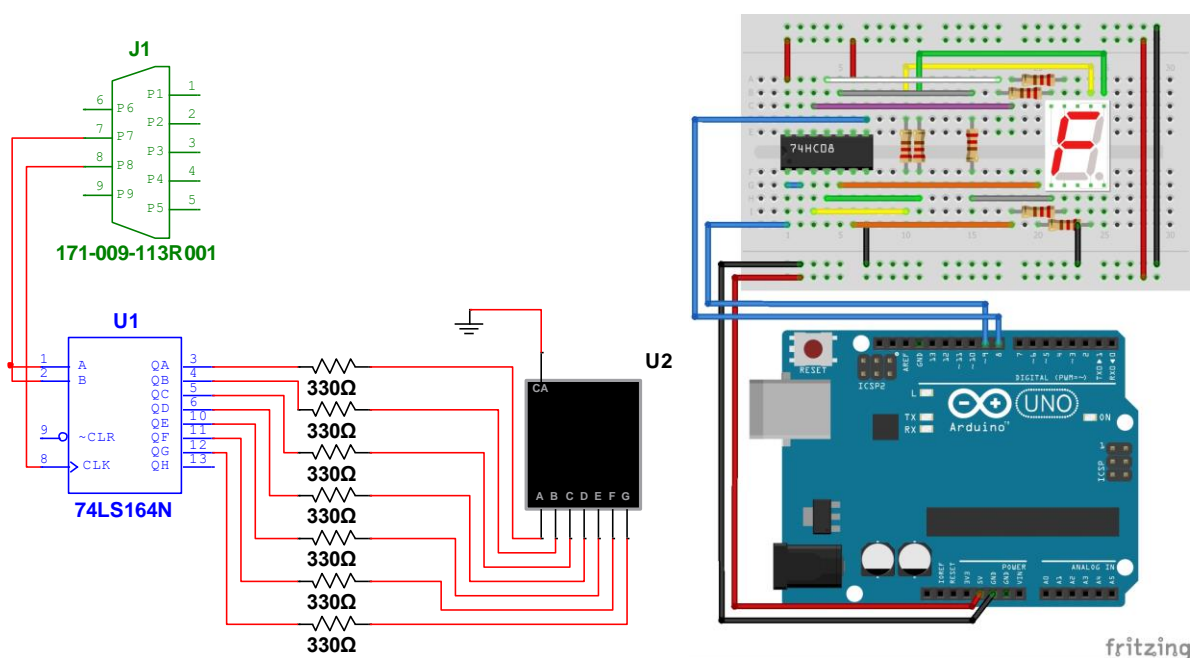


Figura 10. Diagramas electrónico y físico de conexión con el registro de corrimientos.

La Figura 10 muestra el diagrama físico para las pruebas de componentes conexión y el diagrama electrónico para su montaje en la placa preperforada.

Nota: Estos diagramas ejemplifican cómo debe ser la conexión, pero recuerde que usted debe hacer estos circuitos soldados en placa preperforada.

Potenciómetro y botones

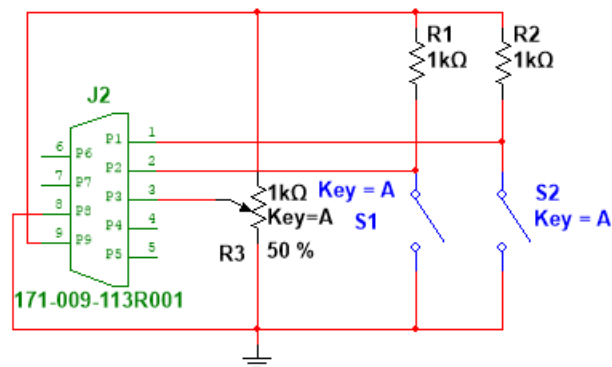


Figura 11. Diagrama electrónico de conexión de los botones y potenciómetro con el DB9

Para el potenciómetro tome en cuenta que la patilla del centro es la que se muestra con una flecha en el diagrama.

Importante los puntos rojos son los que indican que hay conexión. Si 2 líneas se cruzan sin punto quiere decir que no están conectadas entre ellas.

Arduino

Para el código de Arduino usted deberá ir al siguiente enlace y clonar el proyecto. En el mismo encontrará la lectura y escritura del puerto serial, así como el manejo de los componentes del control. Para el mismo se define el siguiente formato, que usted utilizará tanto para leer como escribir en el puerto serial. Estos están dados por nombre "=" valor y separados con "," se finaliza el mensaje con ";"

Recibir códigos para presentar el display y los leds de energía:

- leds=<binario de 6 bits con cada bit pertenece >
- display=<número 0~9>

Ejemplo: leds=111111,display=9;

Enviar códigos de valores de entrada:

- Leer valor de boton 1: btn1=valor,

- Leer valor de botón 2: btn2=valor,
- Leer valor del potenciómetro: pot=<número 0~100>
- Posición X del control: posX=<número con rango>
- Posición Y del control: posY=<número con rango>

Ejemplo: btn1=1,btn2=1,pot=0~100,posX=12,posY=-139;

En la siguiente tabla se especifica la conexión con los pines del Arduino y cada componente:

Pin Arduino	Pin DB9		Función
A2	J2	P3	Patilla del centro del potenciómetro.
A4	J2	P4	SDA del módulo MPU-9250
A5	J2	P5	SCL del módulo MPU-9250
2	J1	P6	LED 1
3	J1	P5	LED 2
4	J1	P4	LED 3
5	J1	P3	LED 4
6	J1	P2	LED 5
7	J1	P1	LED 6
8	J1	P7	CLK del circuito integrado 74LS164
9	J1	P8	AB entrada del circuito integrado 74LS164
10	J2	P1	Entrada del Botón 1
11	J2	P2	Entrada del Botón 2
5 v	J1	P9	Alimentación de 5v
3.3 v	J1	P6	Alimentación de 3,3v
GND	J1	P8	Negativo del circuito.

En los siguientes links usted encontrará documentación y materiales que le pueden ayudar a trabajar en el proyecto:

- [Soldar un conector DB9](#)
- [Soldar componentes a una placa](#)
- [Hoja de datos del 74LS164](#)
- [Distribución pines 7 segmentos](#)
- [Guía del módulo MPU-9250](#)
- [Tutoriales Arduino](#)

- Comunicación serial en python

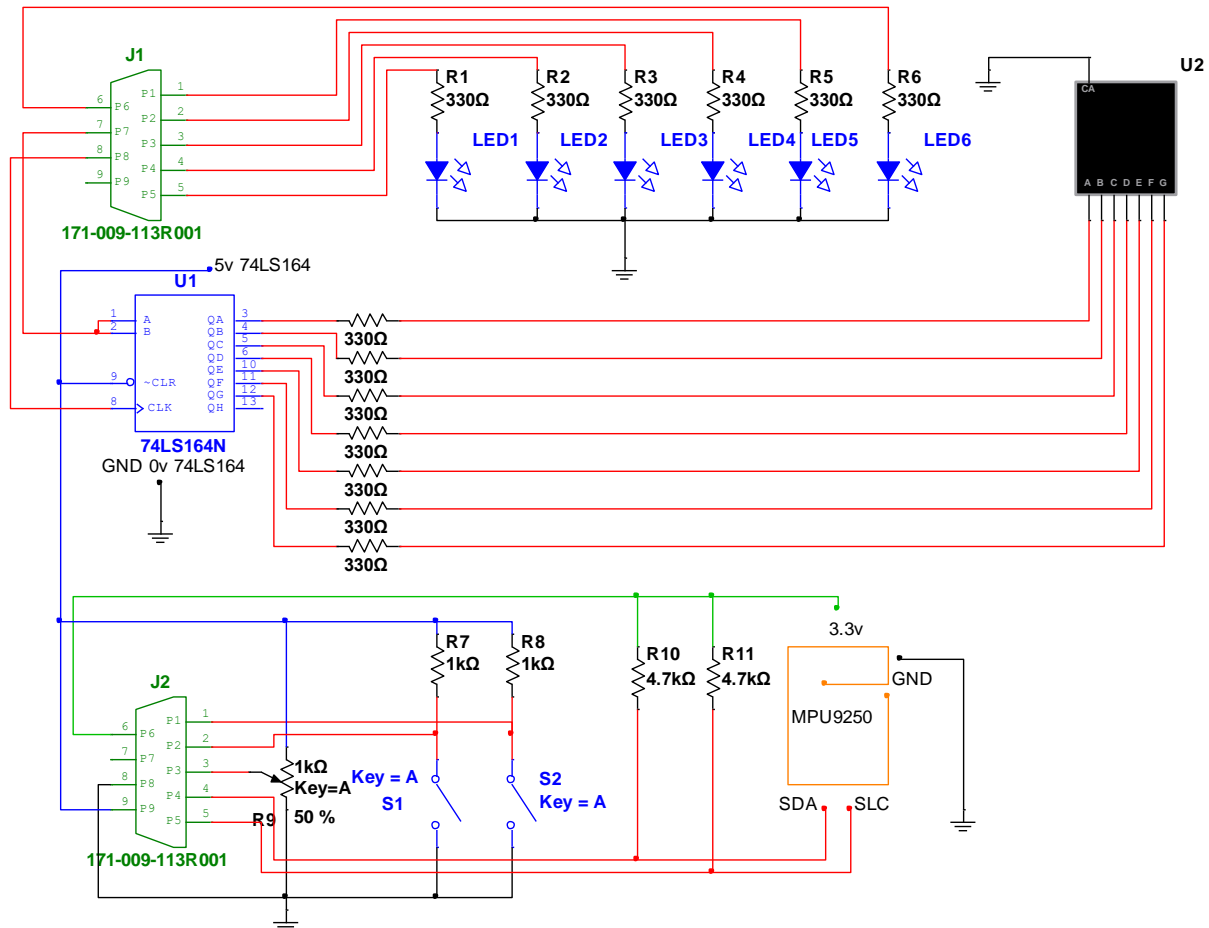


Diagrama eléctrico final de la conexión de componentes del control. Y los DB9

Aspectos generales

- Cualquier duda, omisión o contradicción en la especificación se debe aclarar con el profesor y se difundirá a través del asistente.
- El proyecto es en PAREJAS. Y se deberá entregar el día: Jun 16 2018. Cualquier indicio de copia significará un cero en la nota.
- Se debe imprimir la documentación (archivo .doc) ésta debe contar con lo siguiente: Introducción; Conclusiones; Recomendaciones; Análisis de Resultados que debe incluir: bitácora, resultados; y Literatura o Fuentes Consultadas.
- Para la realización de la parte gráfica se pueden usar bibliotecas complementarias Python (Tkinter, Pygame) u otras que considere de ayuda.
- El código debe estar suficientemente documentado de tal forma que Ud. se pueda orientar en el fácilmente durante la defensa.

- Se debe presentar la tabla de evaluación a la hora de defender el proyecto.
- Se deben definir los requerimientos mínimos si se desea revisar en una computadora de un laboratorio y avisar con antelación para poder ejecutar el programa.
- Se evaluarán el código y la modularización.

Bibliografía

[1] juegos.com. (2018). Star Force. [online] disponible en:
<http://www.juegos.com/juego/star-force>. S. F. Obtenido 17 May 2018.

[2] "Hudson - Shooting game [NES] (Archive)". Hudson. Hudson. S. F. Obtenido el 16 Mayo 2018.

[2] <https://www.arcade-history.com/?n=star-force&page=detail&id=2614> Obtenido May 18 2018.