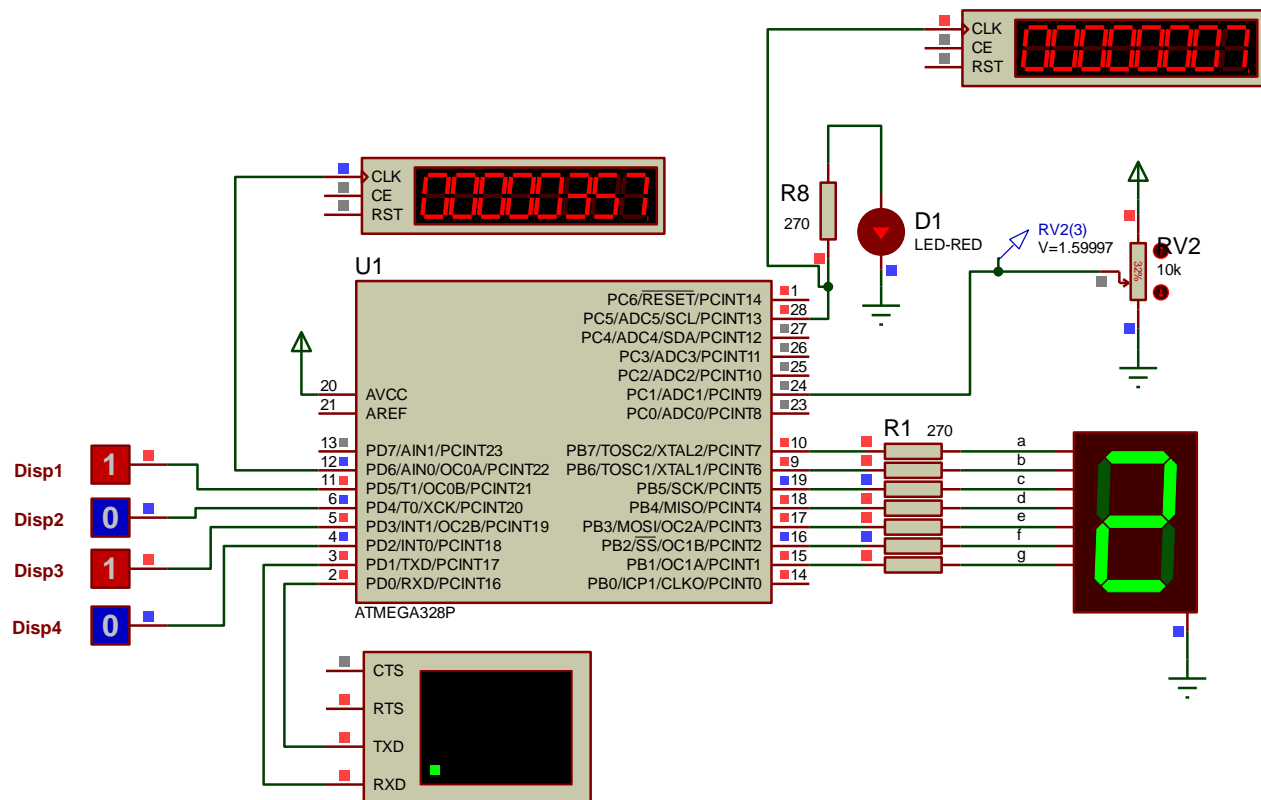


Examen Argumentativo Microcontroladores

Nombre: _____ Matrícula: _____

El examen se deberá entregar por CANVAS a más tardar a las 12:50AM. **Entregar TODA la carpeta del proyecto en un ZIP o RAR.** Si no se entrega toda la carpeta no se podrá calificar el examen. Puedes consultar programas hechos y material del curso. Sin embargo, cualquier comunicación electrónica (Facebook, WhatsApp, correo, Skype, etc) será considerada como una deshonestidad académica.

Se tiene el siguiente hardware que utiliza el oscilador interno a **8MHz** y terminal con un baud rate que dependerá del último dígito de tu matrícula:



4. A través del puerto serial el usuario podrá controlar el encendido o apagado de 4 dispositivos conectados en PD5, PD4, PD3 y PD2. Al inicio deberán estar apagados. El dispositivo 1 se controlará mediante el carácter '1'. Si está apagado y llega un '1' se deberá prender. Si por el contrario está prendido y llega un '1', se apagará el dispositivo 1. Para los dispositivos 2, 3 y 4 será muy similar, pero con los caracteres '2', '3' y '4' respectivamente. (30 puntos).
5. Usando una interrupción periódica con el **Timer 1 Comparador A**, hacer parpadear el LED conectado a PC5 hacerlo parpadear a Y Hz. Donde Y es el dígito menos significativo de tu matrícula diferente de 0. Por ejemplo, si tu matrícula es A01234567, te correspondería una frecuencia de 7Hz. (20 puntos)

En main antes del while(1) tendrás que configurar el timer 1 y su interrupción para lograr la frecuencia requerida. Se calificará también la exactitud de la frecuencia. La función de interrupción debe estar definida de la siguiente manera (se pone antes de main()):

```
interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{
    PORTC.5=~PORTC.5;
}
```

Los parámetros para la comunicación serial serán de: 8 bits de datos, sin paridad y 1 stop bit y el baud rate de acuerdo al último dígito de tu matrícula:

Último dígito matrícula	Baud Rate
0 o 1	1200 bps
2 o 3	2400 bps
4 o 5	4800 bps
6 o 7	19200 bps
8 o 9	38400 bps

Recomendación:

En el ciclo principal (en el while(1)) tienen que escribir los valores actuales del voltaje cada vuelta. Para saber si llegó un carácter del puerto serial pueden utilizar lo siguiente:

```
//Código para configuración inicial (iniciar registros de Puerto Serial, ADC, timer 0, timer 1)

while(1)
{
    //Código para imprimir el valor actual del voltaje)

    if ((UCSR0A&0x80)!=0)          //Instrucción que nos dice si hay un caracter listo
                                   // para leerse (bandera "receive complete")
    {
        dato=getchar();           //Lee caracter del Puerto serie

        // Código para procesar el caracter recibido
    }
}
```

Consideraciones importantes:

- No se puede cambiar el hardware de Proteus. Lo único que se puede hacer es cargar tu cof y variar el voltaje (con el potenciómetro), cambiar el baudrate de la terminal y mandar los caracteres para controlar los 4 dispositivos.
- Si tienes duda de cómo debe funcionar el proyecto carga el archivo "SolucionProfesorMatricula249257.hex" cuya matrícula es 249257 (BaudRate de 19200 bps)
- La tabla de 7 segmentos la puedes consultar del segundero hecho en clase que corresponde a esta:
 unsigned char SieteSeg[10]={0xFC,0x60,0xDB,0xF3,0x66,0xB7,0xBE,0xE1,0xFE,0xF6};