

Programación de Sistemas Embebidos

Facultad de Informática

Universidad Nacional del Comahue

Trabajo Práctico 2

1- Escribir un programa hello world en C, que haga parpadear el led conectado en la placa pro mini. Puede utilizar el código del apunte “El primer programa embebido” aunque este puede contener errores (está advertido).
Utilice el esquemático de la placa para reconocer cuál es el pin en el microcontrolador AVR al que está conectado el led.

Recursos

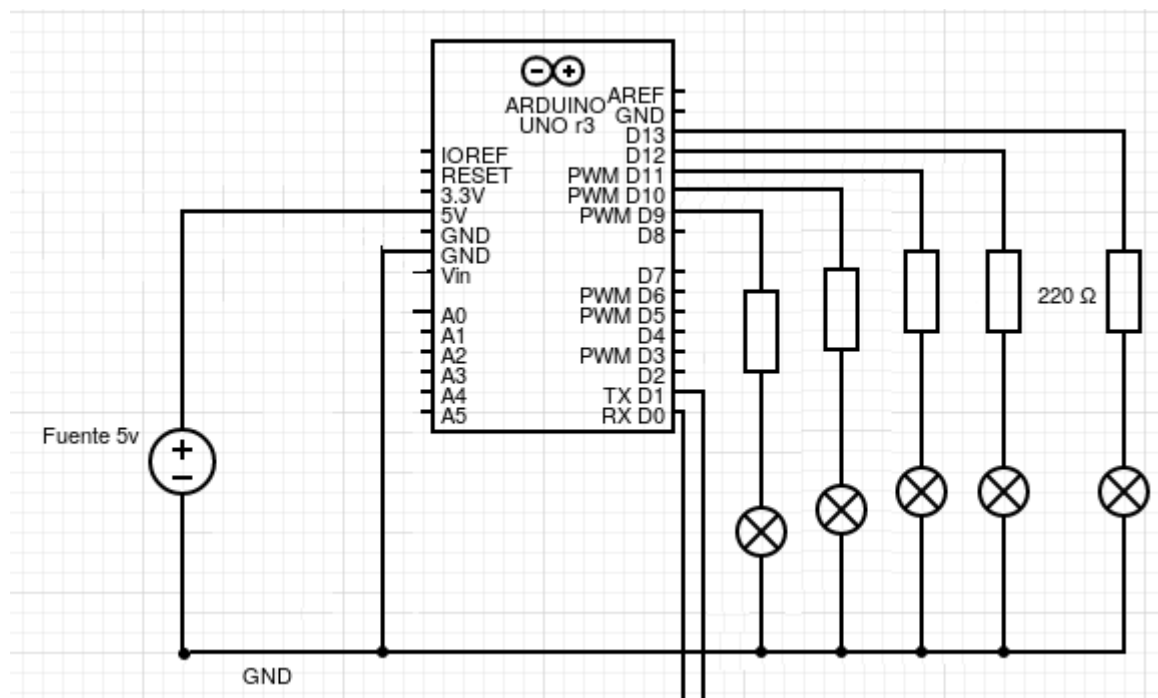
- Utilice el makefile que se provee con este TP en el source code de ejemplo.
- Clase del primer programa embebido

Responda:

1. ¿A que pin de la placa pro mini está conectado el led?
2. ¿A qué pin del microcontrolador AVR está conectado el led?

2- Crear una segunda aplicación para AVR que controle 5 pines de E/S digital paralela del atmega328p (GPIO).

Hardware: utilice el hardware que se detalla a continuación.



Esquemático creado con : <https://www.circuit-diagram.org/>

El hardware a utilizar son 5 LEDs para SALIDA. Todo el hardware debe ser controlado por pines conectados al periférico de E/S GPIO PORTB.

Software: Escriba un programa que realice el efecto del knight-rider, utilizando los 5 leds.

Programación de Sistemas Embebidos
Facultad de Informática
Universidad Nacional del Comahue
Trabajo Práctico 2

3- Utilice avr-objdump para realizar un decodificado del programa binario elf (obtener el código en lenguaje ensamblador AVR a partir del binario elf). Observe ahora que al compilar el programa para un microcontrolador particular aparece al principio del código la tabla de vectores.

Presente en este ejercicio las primeras 10 posiciones de la tabla de vectores.

4- Responder:

- a) ¿Qué hw existe en cada dispositivo de E/S que permite su programación?
- b) ¿Cómo se llaman en el PORT B del AVR?
- c) ¿Cuántos dispositivos de E/S observa conectados al DATABUS del AVR? Mencione sus nombres.
- d) Usted debe controlar un nuevo dispositivo de E/S en un sistema, utilizando un programa escrito en lenguaje C. ¿Que detalles necesita conocer para escribir el software? ¿De dónde obtiene los detalles?
- e) ¿Qué es el voltage? ¿Qué es la corriente?
- f) ¿Por qué se lo clasifica como “arquitectura harvard” a los micros AVR de 8-bits?

Verificación: compilar, vincular y enviar el firmware a cada AVR.

Entrega: subir (push) el trabajo práctico resuelto (o sus versiones intermedias) al repositorio git compartido