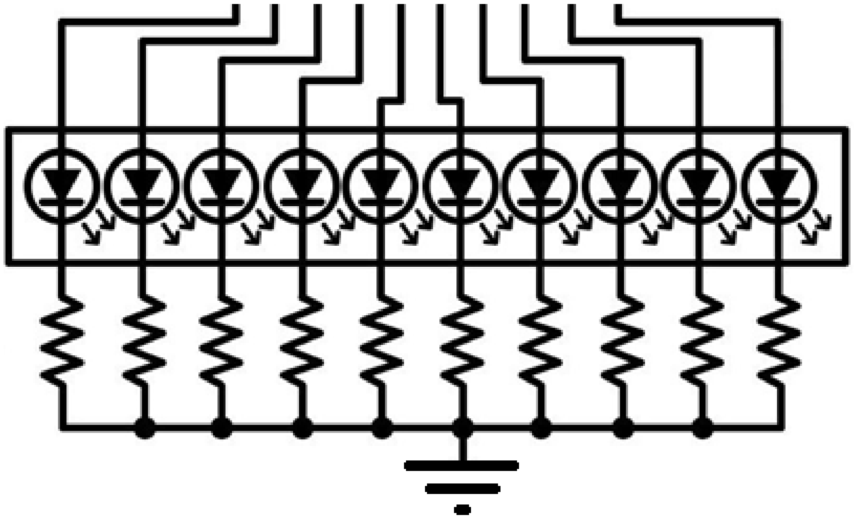
LAB4: Interrupts – ECE433/533/CS456 – Spring 2024

Goals:

- Use interrupts to code a more efficient program
- Use timers, USART and external GPIOs in interrupt mode

Setup:

A 10-segment bar graph LED should be connected to PE0, PE1, PE2, PE3, .. and PE9.



Use the 11 pins resistor array (4611X-101-331LF-ND) provided to limit current into LEDs as shown below.

Requirements:

- Everything must run in interrupt mode.
- The LPUART1 triggers an interrupt whenever a character is received from the PC. The acceptable characters are '0' to '9' where each lights a single LED in the 10-segment bar graph. Any other character should turn off all LEDs.
- LEDs blink using interrupt (free to use any timer(s)):
  - When LED0 is enabled, it should be on all the time (no blinking)
  - When LED1 is enabled, it should blink 1 time per second (500 msec ON, 500 msec OFF, and so on)
  - When LED2 is enabled, it should blink 2 times per second (250 msec ON, 250 msec OFF, and so on)
  - LED3/4/5/6/7/8/9 should be on all the time (no blinking)
- The rising edge of the on-board push button (i.e., button pressed) should trigger an interrupt to send your full name to the PC using the LPUART1 that is running in an interrupt mode.
  - Hint: the push button interrupt should only enable the LPUART1 transmit interrupt flag which will send characters from a character array until it sees a termination character such as '\0'.
- Remember that with everything running in interrupt mode, your while loop in main must be empty. Your program should have no stalls or any polling code.
- Do not use any prebuilt high-level functions but you may use your functions developed in previous labs. Like what you learned in class, write your program in a register level abstraction.
- Add comments to each line in your code.