

Laboratory Task Sheet 08

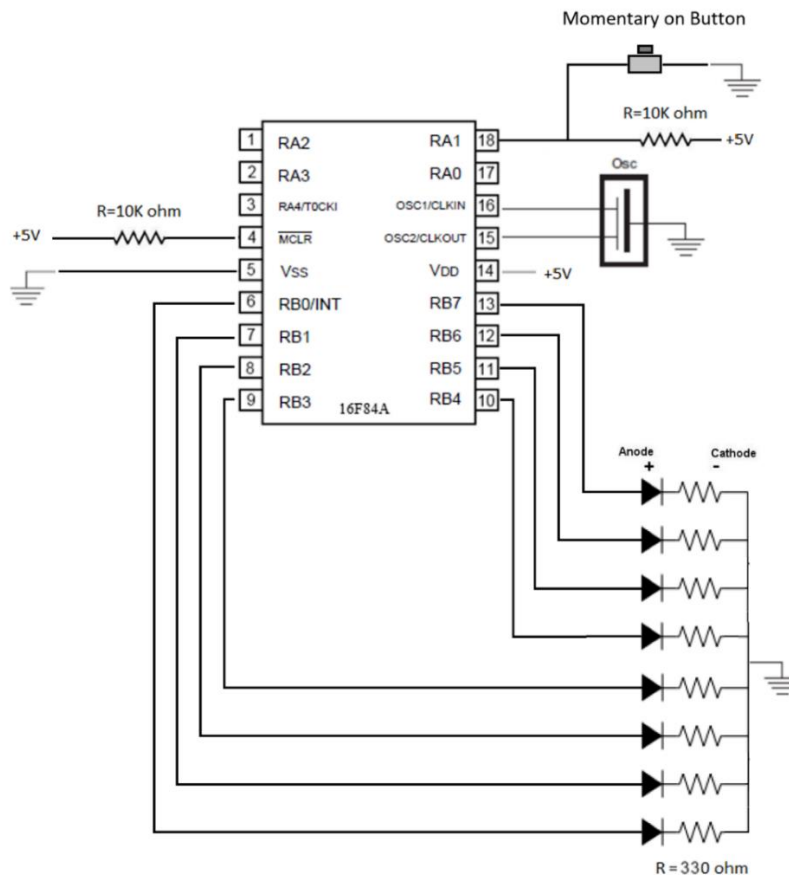
Title: Nightrider with Speed Control

Registers to be learned:

Objective: Program the microcontroller such that the linear array of LEDs displays only one active LED at a time, starting from the rightmost LED (PORTB0), and **every time the Internal Interrupt happens (caused by Timer0)** the active LED shifts one bit to the left. When the active LED reaches the leftmost LED (PORTB7), reverse the direction and consequently when the active LED reaches the rightmost LED reverse the direction again. Each time the button is pressed, the active LED must move faster ($15\ \mu\text{s}$), until it reaches the maximum speed. At this time, pushing the button one more time must reset the LED back to the initial speed.

Tasks

1. Create the circuit below using a linear array of LEDs, a bank of resistors, and a push button.



2. Make a copy of the P16f84A_Template file and name it TASK08Group00. Open the file in MPLAB Software and use the table below to construct the code

Suggested Code Structure
Define Bits and Speed as memory file Define Direction as memory bit
Start
Call Initialization Go to Main
Interrupt Service Routine - ISR Call CheckDirection Call Rotate Move Speed to Timer0 Reset Timer0 Flag
Main To check if the button is pressed, test PORTA1 If it is not pressed, stay here If it is pressed, go ahead To check if the button is not pressed, test PORTA1 If it is pressed, stay here If it is not pressed, go ahead To make LED rotate faster, add decimal 20 to Speed To check if Speed is less than 255, test STATUS C If it is less than 255, go to Main If it is more than 255, go ahead Clear Speed Clear STATUS C go to Main
CheckDirection Check if the most right-hand side LED, connected to PORTB0, is ON If it is ON, clear Direction Check if the most left-hand side LED, connected to PORTB7, is ON If it is ON, set Direction Return
Rotate Check Direction Based on the value of Direction, rotate PORTB to the left or right and then return

Initialization

Bank1

Use TRISA to define PORTA1 as input

Use TRISB to define PORTB (all pins) as output

Use OPTION_REG Register to choose the longest possible pre-scaler rate

Use OPTION_REG Register to assign the pre-scaler to the Timer0 module

Use OPTION_REG Register to define Internal Oscillator as the Timer0 clock source

Bank0

Use INTCON Register to turn on the master switch for all the Interrupts

Use INTCON Register to turn on the Internal Interrupt

Initialize PORTB to turn off all the LEDs except the first one

Clear STATUS C

Clear Speed

end

3. Program the microcontroller and test it on the circuit.
4. Demonstrate the result to the instructor.
5. Upload the code on D2L and save it for yourself.