Bangladesh University of Engineering and Technology
Department of Computer Science and Engineering
CSE 316 July 2022
Microprocessors, Microcontrollers, and Embedded Systems Sessional

Experiment 3
Basic use of ADC using LDR and LCD module in ATmega32

GOAL:
To understand the basic working principle of ADC, LDR and the LCD module.

EXPERIMENTAL TOOLS AND MATERIALS: ATmega32, 16×2 character LCD module, USBASP programmer, Trainer Board, Wires, Potentiometer, LDR.

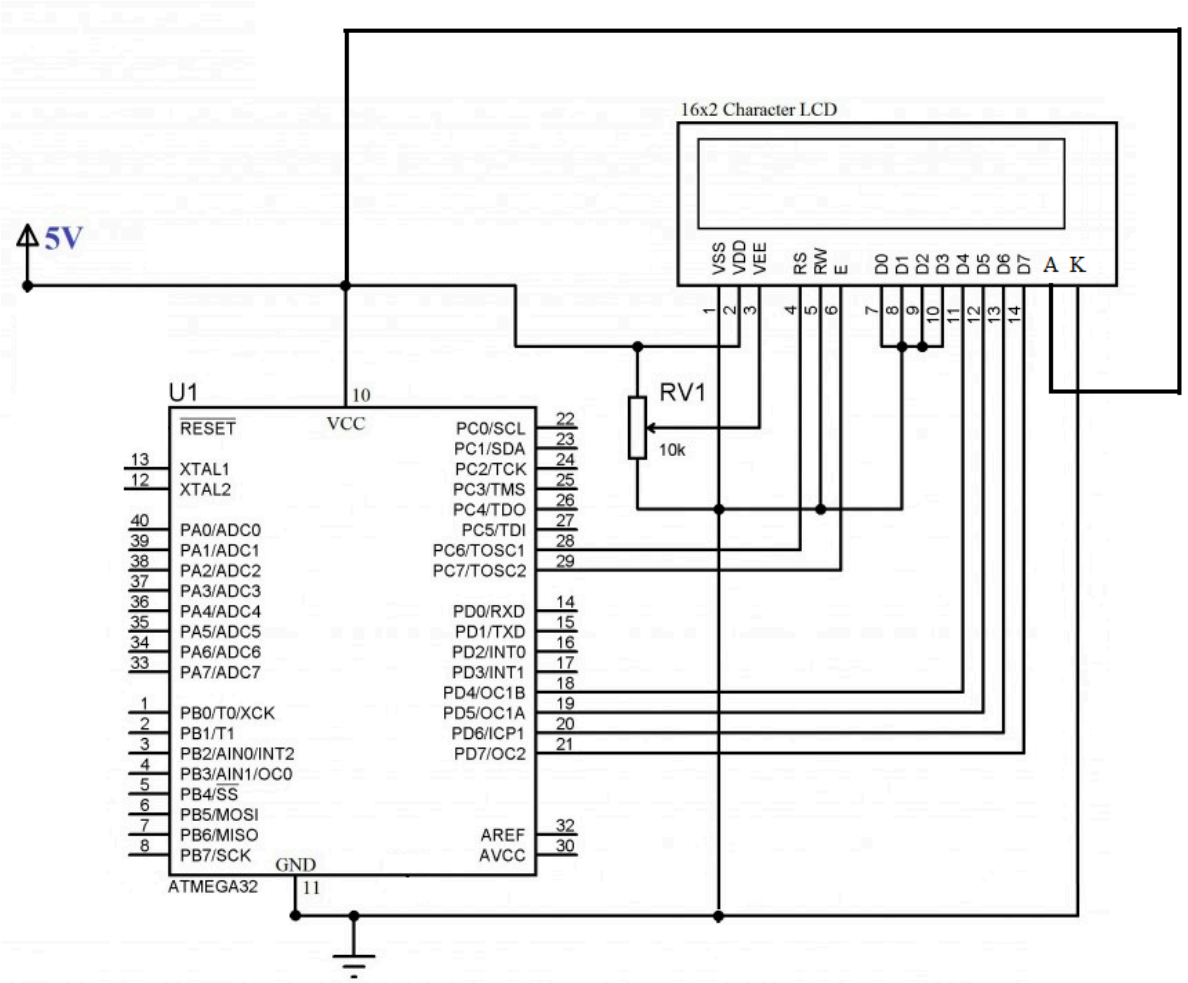
BASIC DESCRIPTION:
In this experiment you will have to turn on a LED depending on light intensity (measured by LDR) and measure voltage difference between two ends of the LDR. You will measure this voltage using ADC and display the voltage in the LCD display. You will also continuously monitor the voltage using a voltmeter. You should use the DVM (digital voltmeter) from the trainer board. We will change the voltage using the artificial light source and shade and the reading of ADC from the LCD segment should be close to the value shown in the DVM.

LCD Module Basics:
You should check out the following tutorial. You will be using the **4 bit mode.**
https://electrosome.com/interfacing-lcd-atmega32-microcontroller-atmel-studio/
Also use the following connection (next page) for LCD Module.

Download the header file from here.
https://gist.github.com/mahmoodtareq/edbdfd3261bb00172a02b5187a8a0095
Place the header file at the save directory as your main.c file.
To add lcd.h to your project: Right click on the solution name > Add > Add Existing Item

ADC Basics:
The basics were covered in the theory class. Do necessary calculations according to your clock speed and configuration. If you use the internal 5V as ADC source. You will have to connect AVCC (PIN30) to 5V and GND (PIN 31 and PIN 11) to 0V. You can connect the AREF pin (PIN32) to a capacitor e.g. 10μF or 1μF, however it is optional.

LCD Module Pin	ATmega32 Pin
VSS	GND
VDD	5V
VEE / V0	Potentiometer
RS	PC6
RW	GND
E	PC7
D0-D3	GND (or, you can leave them open)
D4-D7	D4-D7
A / LED+	5V
K / LED-	GND



PROCEDURE:

1. Measure the resistance of an LDR. Check how the resistance of the LDR varies with light intensity.
2. First connect LDR with a sensitivity resistance. Apply caution when selecting the value of the sensitivity resistance. It should be on par with the resistance of the LDR. Ensure the LDR is working by checking its output in the DVM of the trainer board as you provide artificial shade or light.
3. Connect the LCD module to your microcontroller. **Remember to use the 5V power source of the trainer board.** The USB ASP will not be able to drive the LCD module. Write the necessary code to display a simple string, e.g., "Hello World" to check the LCD module is working. Show it to the lab teachers.
4. Complete the necessary ADC connections and display the variable voltage both in the DVM and your LCD module.
5. Turn on the LED if the light intensity is low. Otherwise, turn the LED off.

MISCELLANEOUS:

1. The USB ASP may not be able to drive the LCD module properly. You should use the trainer board/ArduinoUNO for power source.
2. The connection to the LCD segment is crucial. Without proper connection it will not work. Upon giving A and K and the backlight switch on, the LCD module should lit the backlight on and show a blank screen. Using female to male jumper wires to connect the LCD module can be subject to loose connection. Setting the LCD module directly to the breadboard seemed to work better.
3. There is a pot in the LCD module. You can use it to adjust the contrast.
4. Never input a voltage greater than 5V to your ADC input.
5. Some pins of PORTC can not be used for I/O directly. First the JTAG has to be turned off. One way is to uncheck the JTAG box while writing the fuse bits, the second is to write a 1 to the JTD bit twice consecutively. MCUCSR = (1<<JTD); MCUCSR = (1<<JTD);
   For more details refer to:
   http://www.avrfreaks.net/forum/jtag-enablingdisabling-atmega32-and-fuse-settings-solved
6. You do not need to bother about FUSE bits. By default it is set to: E199
   If the JTAG Interface is disabled it will be set to: E1D9. You can calculate fuse bit from this online tool: http://www.engbedded.com/fusecalc/