

Cairo University

Embedded Final Project SBE403

Delivered to:

Eng/ Hossam Ahmed

Eng/ Alaa Melek

Eng/ Laila Abbas

Submitted by:

Name	Sec	BN
Ibrahim Elsayed	1	1
Bassam Mostafa	1	22
Donia Abd Elslam	1	30
Renad Taher	1	34
Mariem Ahmed	2	29
Mustafa Yehia	2	33

Table of Contents

ATmega32 Microcontroller	1
LM016 Character LCD	1
TC72 SPI to Temperature Convertor	1
(4 * 3) Keypad	1
PWM to Voltage Convertor Module	1
Calibration Resistor	1
Full Schematic (Proteus)	1
Git-hub Repo	1
Link to Code (Dropbox)	1

ATmega32 Microcontroller

In-Code Files

- [DIO.h](#)
- [DIO.c](#)

Source: [ATmega32 AVR Microcontroller](#)

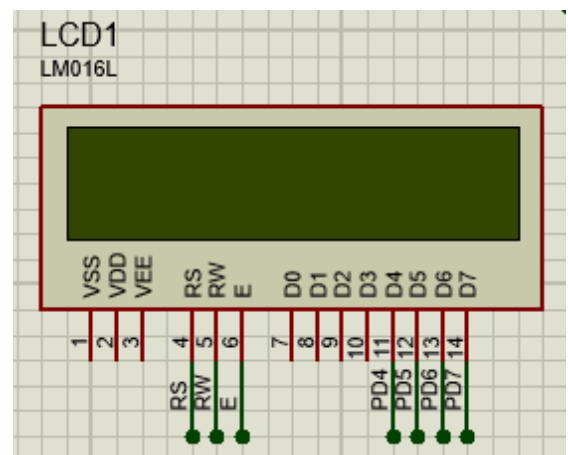
LM016 Character LCD

LCD

LCDs (Liquid Crystal Displays) are used for displaying status or parameters in embedded systems. LCD (16 * 2) is a 16-pin device which has 8 data pins (D0 - D7) and 3 control pins (RS, RW, EN). The remaining 5 pins are for supply and backlight for the LCD.

The control pins help us configure the LCD in command mode or data mode. They also help configure read mode or write mode and when to read or write.

LCD 16x2 can be used in 4-bit mode or 8-bit mode depending on the requirement of the application. To use it, we need to send certain commands to the LCD in command mode and once the LCD is configured according to our need, we can send the required data in data mode.



LCD (4-bit Mode)

- In 4-bit mode, data/command is sent in a 4-bit (nibble) format.
- To do this 1st send a Higher 4-bit and then send a lower 4-bit of data/command.
- Only 4 data (D4 - D7) pins of (16 * 2) of LCD are connected to the microcontroller and other control pins RS (Register Select), RW (Read/Write), E (Enable) is connected to other GPIO Pins of the controller.

Interface of LCD with ATmega32

In-Code Files

- [LCD_config.h](#)

- [LCD.h](#)
- [LCD.c](#)

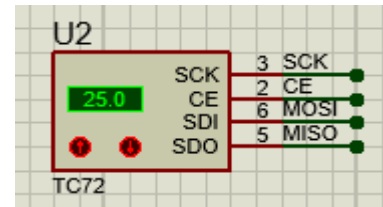
Source: [Interfacing LCD \(16 * 2\)](#)

TC72 SPI to Temperature Convertor

SPI Communication

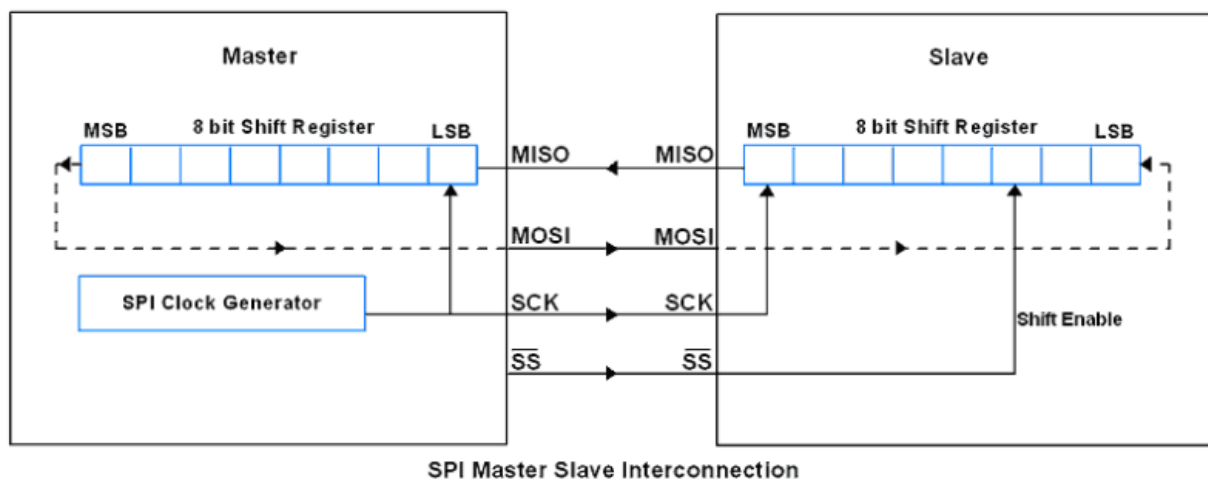
The Serial Peripheral Interface (SPI) is a bus interface connection protocol originally started by Motorola Corp. It uses four pins for communication.

- SDI (Serial Data Input)
- SDO (Serial Data Output)
- SCLK (Serial Clock)
- CS (Chip Select)



ATmega32 SPI Communication

MISO (Master-In-Slave-Out)	MOSI (Master-Out-Slave-In)
The Master receives data, and the slave transmits data.	The master transmits data, and the slave receives data.
SCK (Shift Clock)	SS (Slave Select)
The Master generates this clock for the communication, which is used by the slave.	Master can select slaves through this pin.



Pin Configurations

SPI Pins	Pin on ATmega32	Pin Direction (Master)	Pin Direction (Slave)
MISO	B6	Input	Output
MOSI	B5	Output	Input
SCK	B7	Output	Input
SS	B4	Output	Input

AVR ATmega32 uses three registers to configure SPI communication that are SPI Control Register, SPI status Register and SPI Data Register.

SPCR: SPI Control Register

7	6	5	4	3	2	1	0	
SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR

Illustration:

Bit-No. 7	Bit-No. 6	Bit-No. 5
(SPIE: SPI interrupt Enable bit)	(SPE: SPI Enable bit)	(DORD: Data Order bit)
Bit-No. 4	Bit-No. 3	Bit-No. 2
(MSTR: Master/Slave Select bit)	(CPOL: Clock Polarity Select bit)	(CPHA: Clock Phase Select bit)

Bit-No. 1 & Bit-No. 0
(SPR1 & SPR0: SPI Clock Rate Select bits)

Configuration:

Bit-No. 7	Bit-No. 6	Bit-No. 5
1: Enable SPI Interrupt 0: Disable SPI Interrupt	1: Enable SPI 0: Disable SPI	1: LSB Transmit First 0: MSB Transmit First
Bit-No. 4	Bit-No. 3	Bit-No. 2
1: Master Mode 0: Slave Mode	1: Logic One Clock 0: Logic Zero Clock	1: Data Sample on Training Clock Edge 0: Data Sample on Leading Clock Edge

SPSR: SPI Status Register

7	6	5	4	3	2	1	0	
SPIF	WCOL						SPI2X	SPSR

Bit 7 – SPIF: SPI interrupt flag bit

- This flag gets set when the serial transfer is complete.
- Also gets set when the SS pin is driven low in master mode.
- It can generate an interrupt when SPIE bit in SPCR and a global interrupt is enabled.

Bit 6 – WCOL: Write Collision Flag bit

- This bit gets set when SPI data register writes occurs during previous data transfer.

Bit 5:1 – Reserved Bits

Bit 0 – SPI2X: Double SPI Speed bit

- When set, SPI speed (SCK Frequency) gets doubled.

SPDR: SPI Data Register

7	6	5	4	3	2	1	0	
								SPDR

- SPI Data register used to transfer data between the Register file and SPI Shift Register.
- Writing to the SPDR initiates data transmission.

Programming For TC72

The overall programming interface lists below:

1. Set up the SPI to master mode
2. Select SPI clock and data sampling mode
3. Set up digital output for display
4. Send the command to TC72
5. Read temperature from TC72
6. Display the Result

In-Code Files

- [SPI.h](#)
- [SPI.c](#)

(4 * 3) Keypad

The keypad is used as an input device to read the key pressed by the user and to process it.

4 * 4 keypad consists of 4 rows and 3 columns. Switches are placed between the rows and columns. A keypress establishes a connection between the corresponding row and column between which the switch is placed.

To read the keypress, we need to configure the rows as outputs and columns as inputs.

Columns are read after applying signals to the rows to determine whether a key is pressed and if pressed, which key is pressed.

In-Code Files

- [Keypad.h](#)
- [Keypad.c](#)

Source: [4x3 Keypad Interface](#)

PWM to Voltage Convertor Module

In-Code Files

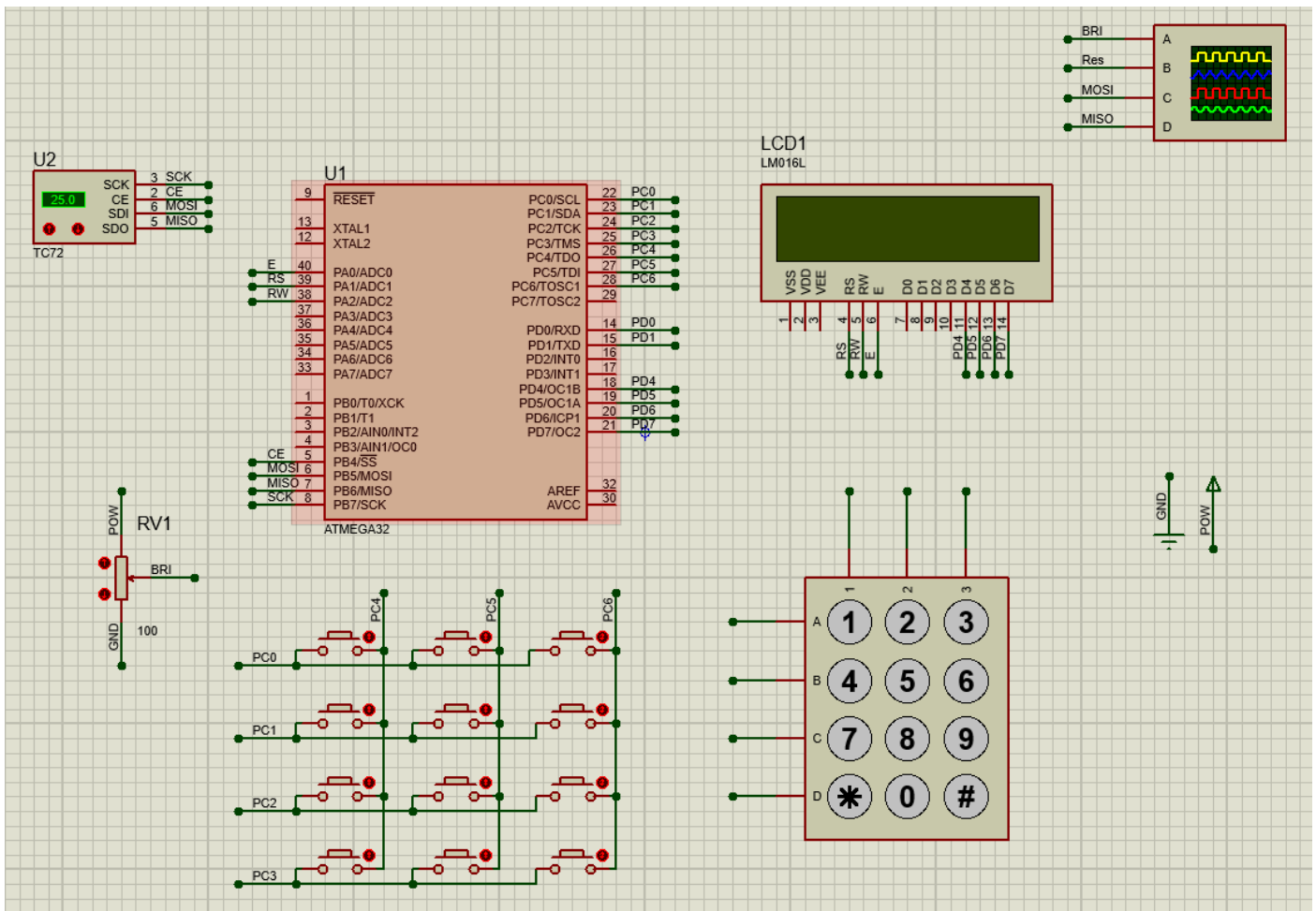
- [Keypad.h](#)
- [Keypad.c](#)

Calibration Resistor

In-Code Files

- [Keypad.h](#)
- [Keypad.c](#)

Full Schematic (Proteus)



Git-hub Repo

- [Embedded Final Project Repo](#)

Link to Code (Dropbox)

-