**Microprocessor Systems Design**

**EEE42101**

**Experiment 3: Displays and Communication**

# Objectives:

* Acquire knowledge about microcontrollers debugging.
* Display the information handled in the microcontroller unit
* Put hand on the communication module

# Tools:

1. PC
2. Arduino Nano board
3. Testing board
4. MiniB-USB cable

Note: all material and sources of this course will be available on:

<https://github.com/ashrafmalraheem/Mircoprocessor_Course>

Feel free to download, study and modify for your own projects.

# Introduction:

When you are working with microcontrollers, you will need to view the information that you are handling. The first and primitive method is by using simple interface (I/O) to view the information as digital info (0 or 1) by blinking and LED. Or another method is by switching the LED ON for a certain amount that is proportional to the value (pulse width). Or the number of blinks is the number you want to view. You can improvise many other methods to view your information inside the microcontroller.

However, in simple application the prementioned suggestions may work. But in real life applications there are many different information to show like integers, float numbers and characters which is inconvenient to translate them into electrical or time series signal. The most common methods to view data inside microcontroller are: displays, debuggers, logging, ...etc.

**Displays** have many types: seven segments, dot matrix and many other. Some could only display characters, graphs and other display only numbers. The draw back of using LCD is it is limited space of display. Therefore, old data should be cleared to display new ones. No record could be saved.

Lcd module

**Logging** is used either by saving the data in the memory or sending them to a server (or any device with more advance capabilities). The data are saved in log files. A timestamp could be created to each log message, so a track of the data could be saved.

**Debuggers** it is a method used while developing the software of the microcontroller. It needs a hardware tools as well as software. Additional instructions are added by the compiler to read/write every register inside the microcontroller. The hardware reads the data in the registers and send them to the debugging application. Most IDEs have special interface to view the variables and registers and many other peripherals inside the microcontroller. Most common methods for debugging are ICSP (in circuit serial programming), JTAG (**J**oint **T**est **A**ction **G**roup), boundary scan, etc...

Debugger and hardware. IDE screenshot.

The debugger holds the execution of the program and wait until it reads the whole registers in the microcontroller unit. Therefore, it has a draw back when debugging Realtime applications and signals.

\*this method is beyond our course reach.