
# برمجة المتحكمات المصغرة

التجارب العملية

## الملحق



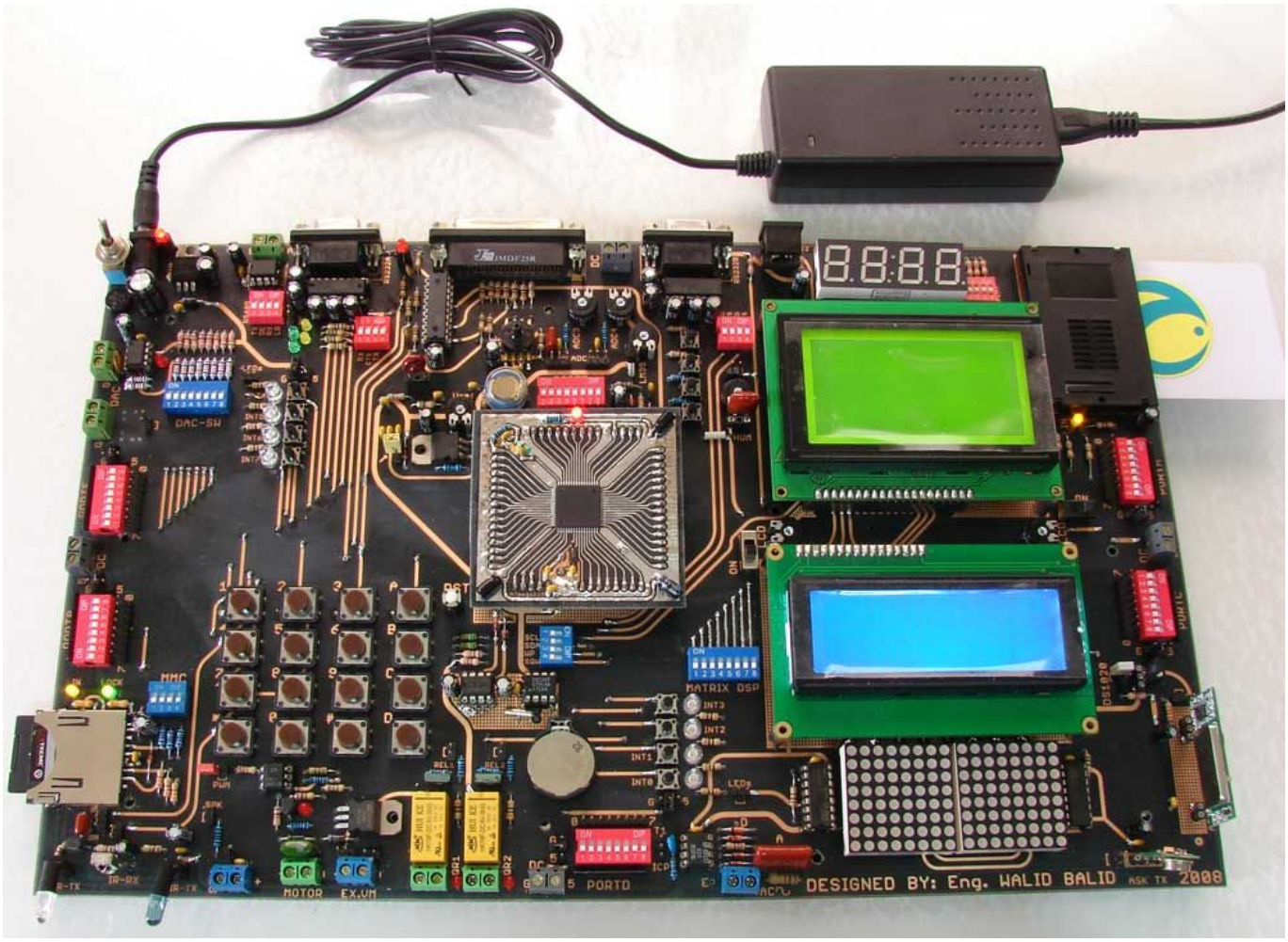
Programming

Embedded Systems Microcontroller

*You Can Practice Microcontroller Programming Easily Now!*

*WALID BALID, Tuesday, December 15, 2009*

# Open-Source multipurpose interactive embedded systems Microcontroller kit for laboratory education




**WALID BALID**

*Faculty of EEE, University of Aleppo*

*Aleppo – Syria*

*2009*

## Acknowledgment :

I would thanks Mr. Mark ALBERTS the programmer of BASCOM-AVR, this kit has been designed and tested with BASCOM.

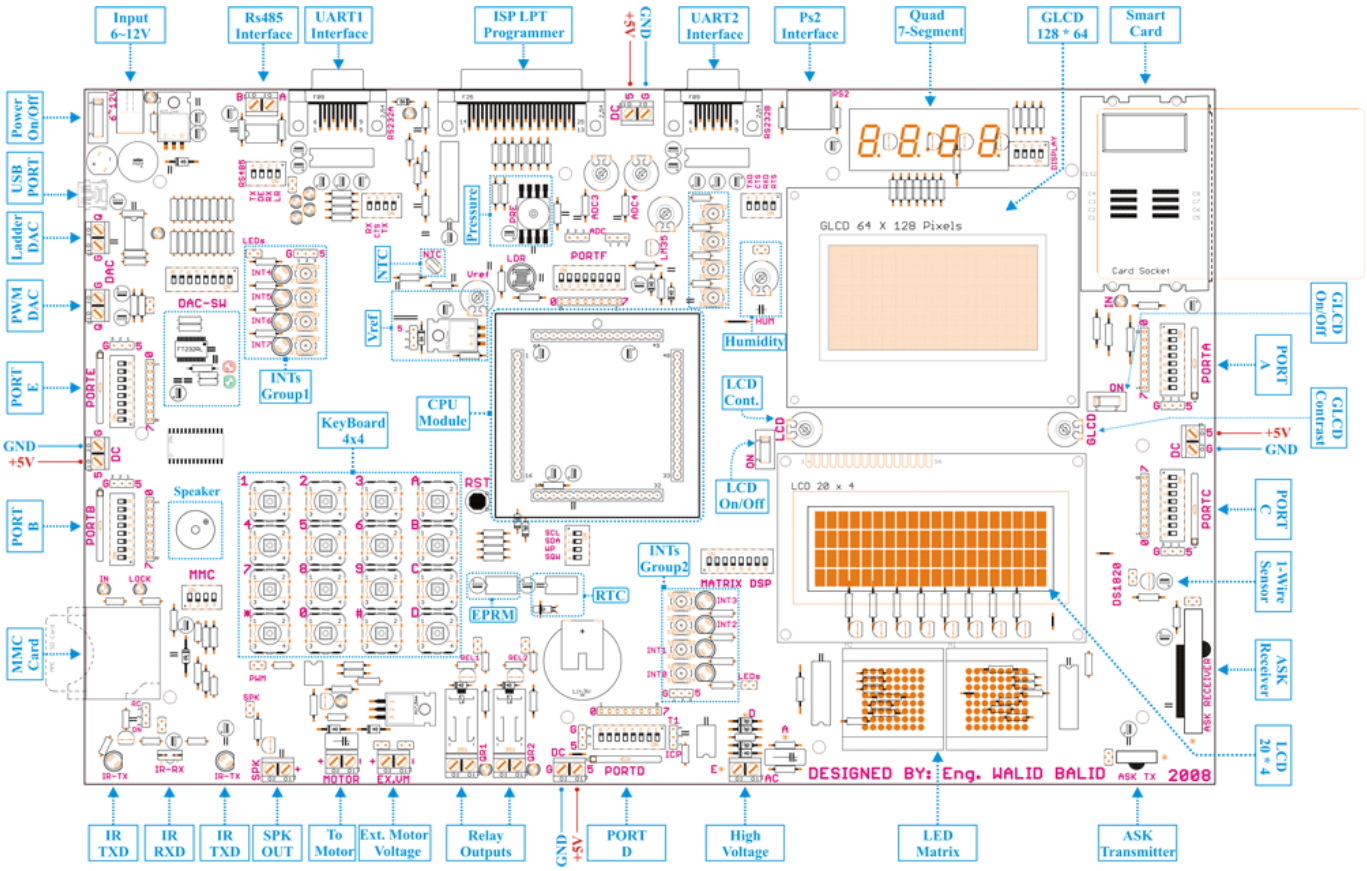
## Introduction:

The kit development aimed at designing a universal training board that can cover a wide range of experiments for the electronics, communication, control, and power departments of the electrical and electronics engineering.

The kit has followed in the designing process all standards, so it can utilize any AVR MCU or any SPI protocol compatible one. The MCU programming can be done by all GNU based compilers with both low-level (ASM) and high-level (Basic, Pascal, C, or C++.) MCUs programming languages.

Many peripherals have been placed on the board to enrich its universality. About 70 different experiments (on basic, intermediate, and advanced level) have been developed.

There has been 47 peripherals unit selected and the units were placed on the board in a way that saves a comfortable working space and separates the elements of each unit from the other units.




Since a good planned student-centric approach has proven effectiveness in engaging students, the experiments manual was designed in a student-centred manner, so that the students can perform proceed with the experiment and develop the aimed skills without a need of teacher supervision.

A detailed description including schematics, experiments, and How to Build recipe have been made available for the students as an open source to enable them to build their own board, and give them even more deeper centric role in this experiential part of engineering.

It is planned to put these resources on the web making the prototype available to implement for any practicing engineering student. It is hoped that the open source approach will attract academic staff and/or students to develop more experiments based on the rich peripheral components of the kit.

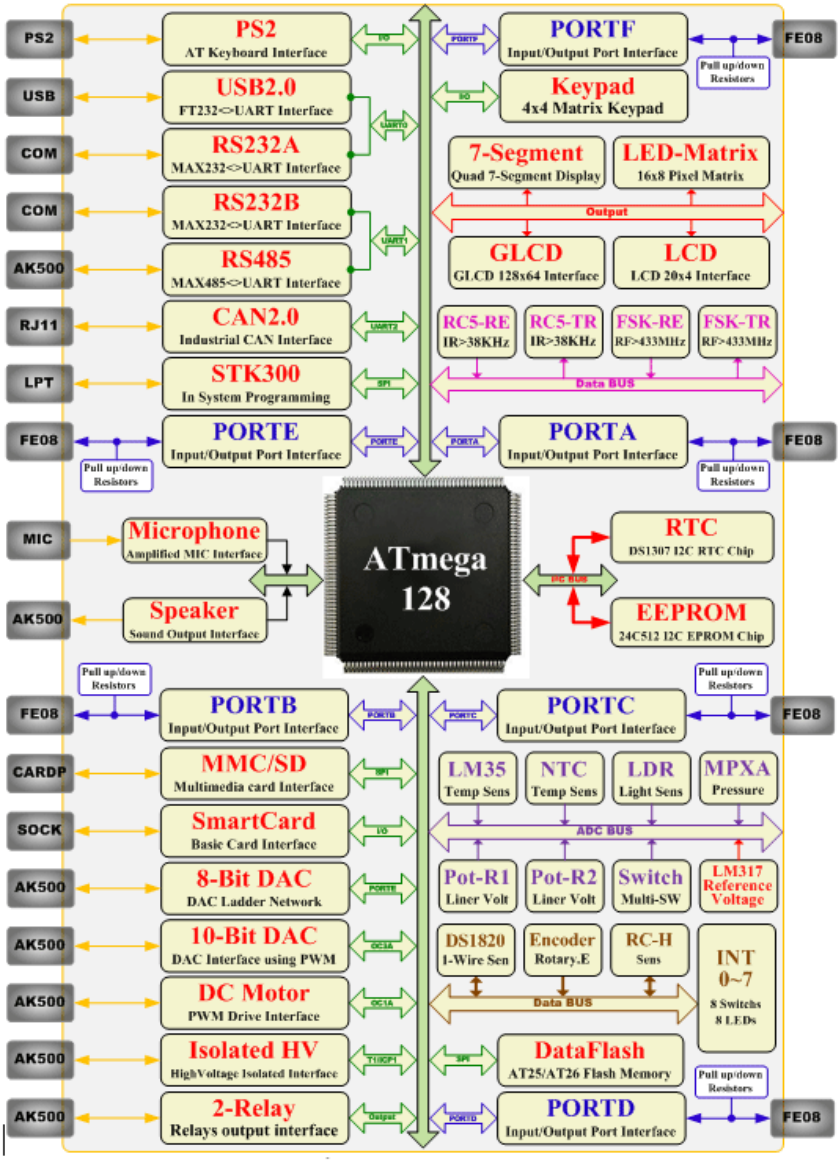
The average cost of building our board is about 120$, while any similar commercial kit will cost more than 700$.

Selection of the designed experiments has been taught in Spring 2008/2009 for about 65 students in the Electrical Engineering faculty at Aleppo University.

So, here we go. I will put the whole system between your hands.

## The Kit Block Diagram:

The MCU ATmega128 is the kit core and it is linked with the kit peripherals such as a Liquid Crystal Display LCD through I/O ports. It is linked with the kit serial communication units, e.g. CAN, RS232, USB, RS485, PS2, I2C, and 1-wire through the serial interfaces. It is linked with the analog sensors, e.g. temperature or pressure, through Analog to Digital (AD) convertors. It utilizes the external interrupts I/O ports to interface with the switch buttons and the rotary encoder. The serial port I2C is connected with the data memory EEPROM (AT24C512 chip), and the real time chip RTC (DS1307 chip).

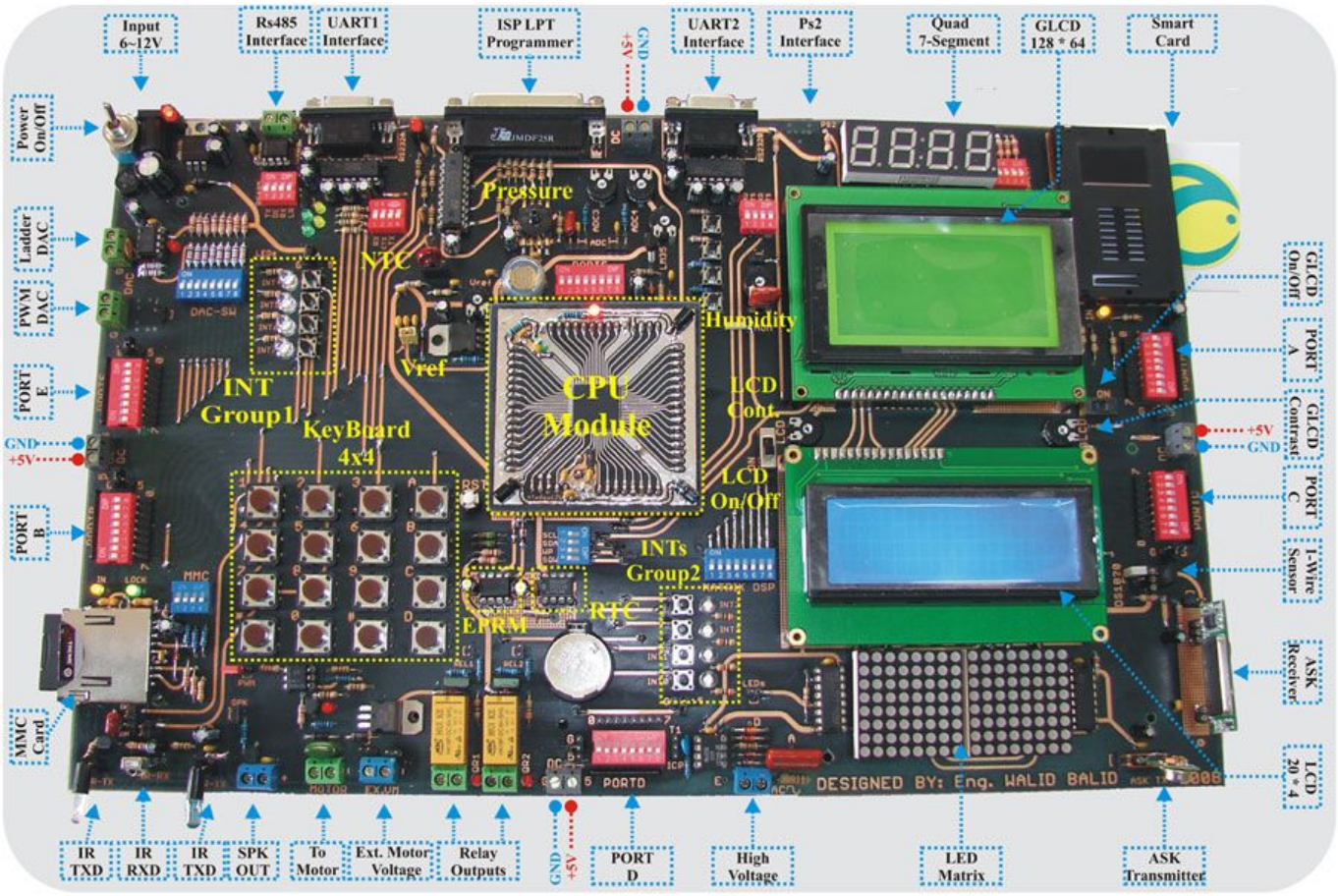


The serial port SPI is connected with the programming unit (STK300) of the MCU and the voice chip APR6016, it is also linked with an MMC card and a data flash memory (AT25xx). The serial port UART0 is connected with USB interface (FT232R chip) and with COM port through RS232 protocol for facilitating PC connection. The serial port UART2 is connected with a CAN port for facilitating industrial communication purposes. The Figure below shows the kit block diagram. All peripherals that needs external communication with the surrounding world are linked to the grey boxes.

**Hands-on Experiments that can be done with the kit:**

1. Programming the MCU ports for displaying LEDs light movements
2. Interfacing switches with the MCU ports
3. Interfacing 4x4 Hexadecimal Array Keypad (16key) with the MCU
4. Interfacing and programming the MCU with a 20x4 LCD
5. Interfacing and programming the MCU with a 128x4 GLCD
6. Interfacing and scanning Quad seven-segment display
7. IR remote control sender/receiver based on RC5 code
8. Data Transfer using RF Transmitter based on FSK modulation
9. Interfacing and programming the MCU with Real-time clock chip
10. DAC by Interfacing 8-bit ladder network with the MCU
11. Programming the MCU analog comparator unit
12. Digital Frequency counter/meter 1HZ – 4MHz
13. Measuring the luminous intensity (Flux) using LDR
14. Measuring the barometric pressure and altitude using a Barometer
15. Interfacing with the LM35DZ analog temperature sensor
16. Storing Data using MMC/SD card in FAT23 format
17. Programming Smart-Card with high-security software algorithms
18. Interfacing and Programming 32x8pixel LED-Matrix scrolling Display
19. Speed control of DC motor using PWM
20. Interfacing the MCU with PC using RS232 protocol
21. wide area data transfer using the industrial CAN protocol
22. Interfacing with RS485 for wide area data transfer Digital Scientific calculator by Interfacing LCD (liquid crystal display) and Hexadecimal Keypad with AVR MCU.
23. Adjustable Signal Generator (Sin, Cos, Smooth, Triangular, square) by Interfacing GLCD (Graphical liquid crystal display), button Keys, and variable resistor with AVR MCU.
24. RC5 code based, IR (Infrared) remote control sender/receiver, by interfacing IR receiver module unit and IR transmitter diode with AVR MCU.
25. Obstacle detection using IR transmitter.
26. Interfacing Barometer with AVR MCU for measuring barometric pressure and altitude.
27. Interfacing RTC (Real Time Clock) chip (DS1307) for real time application.
28. Speed control of DC motor using PWM (Pulse width modulation)
29. Interfacing 8-bit ladder network with AVR MCU for DAC purpose.
30. Interfacing RS485 converter with UART serial interface for long area data transfer
31. Interfacing AVR MCU with PC using RS232 converter and USART serial interface
32. Digital Frequency counter/meter 1HZ – 4MHz
33. Programming Smart-Card with high-security software algorithms using the AES and DES symmetric-key algorithm
34. Storing Data using MMC/SD card in FAT23 format
35. Resistance and Capacitance Digital Meter
36. Wireless data transfer using IR 38KHz (Infrared) based on Ir-Data Protocol
37. Wireless data transfer using RF 433MHz (Radio Frequency) based on FSK modulation
38. wide area data transfer using the industrial CAN protocol for

39. Digital to Analog conversion using 8-bit Ladder network
40. Programming 32x8pixel LED-Matrix scrolling Display
41. Interfacing with LM35 analog temperature sensor (-45ºC ~ +125ºC)




Also, some of the experiments were provided with Simulation using Proteus.