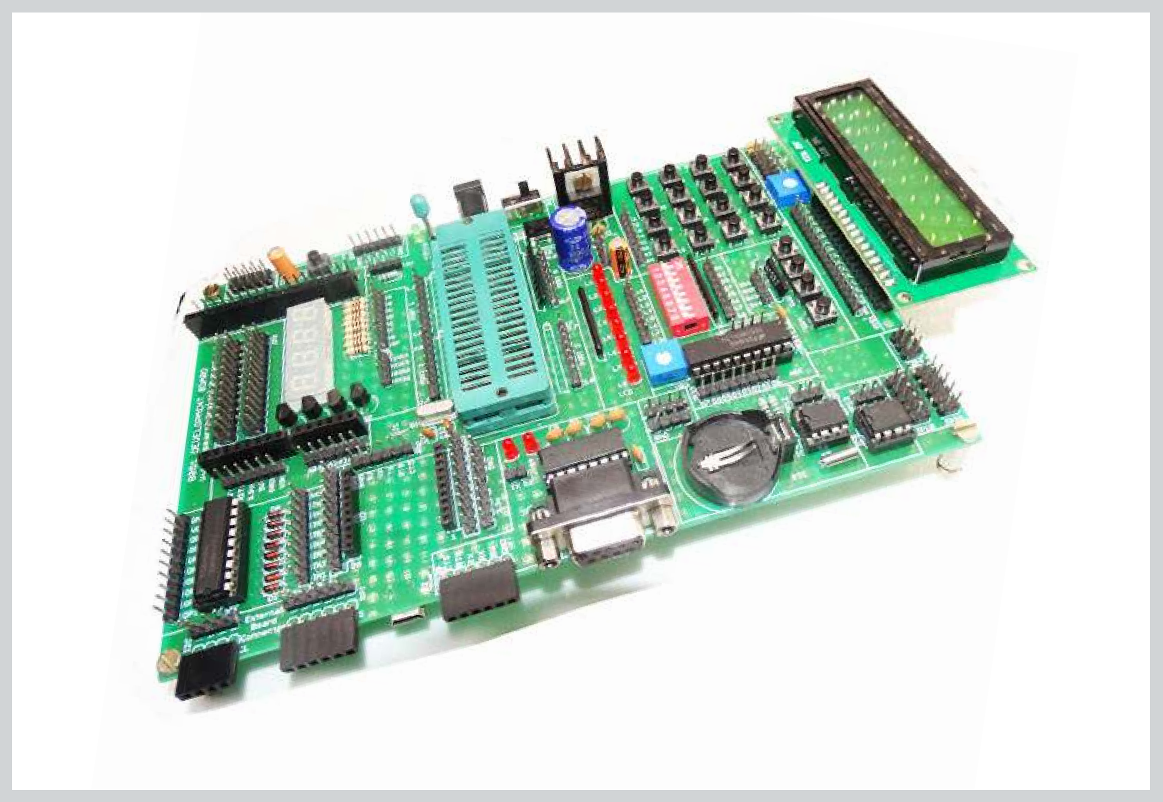
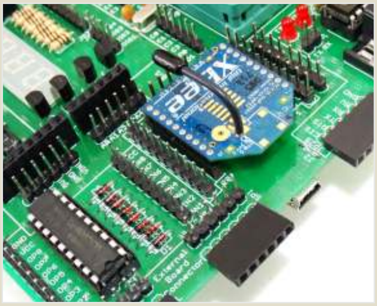
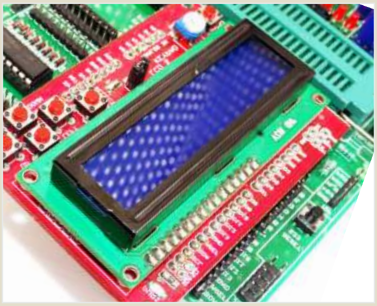
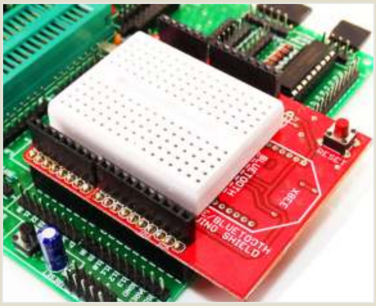
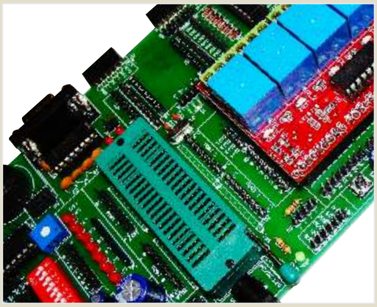
Atmel Development Baord

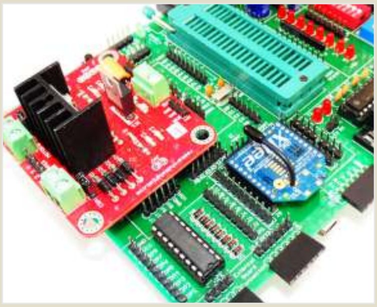










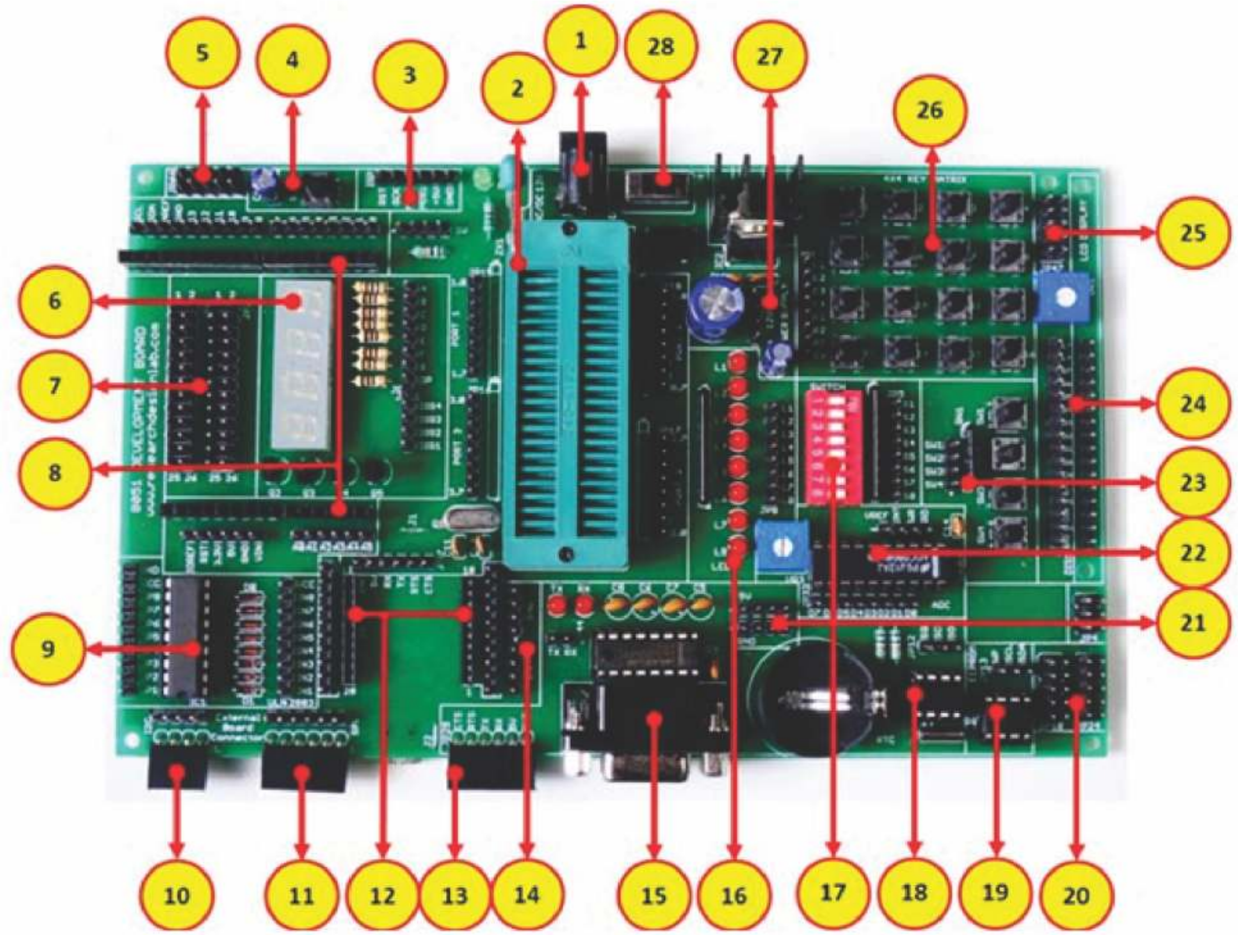


Research Design Lab

www.researchdesignlab.com

Email: sales@researchdesignlab.com I www.researchdesignlab.com

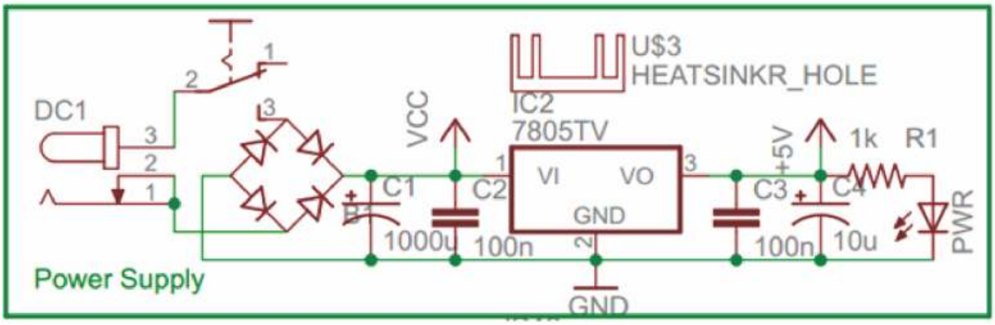
An ISO 9001- 2008 Certified Company



1. Power supply, 5V-12V
2. 40 pin ZIF socket for IC mount.
3. ISP connector*
4. Reset
5. Node connector
6. 4x1 7 segment display
7. 26 pin raspberry connector
8. Arduino Shield footprint
9. ULN 2803 driver
10. I2C bus
11. SPI bus
12. XBEE footprint/XBEE Adaptor module
13. FT232 breakout board connector
14. DC 3.3V connectors
15. DB-9 female connector
16. 8x1 LED's
17. 8 way DIP switch
18. RTC Module
19. EEPROM
20. 2x5x2 jumper node.
21. DC 5V connectors
22. Analog to Digital output
23. 4x1 keypad
24. 16x2 LCD connectors
25. Node connector
26. 4x4 Matrix Keypad
27. DC 12V connectors
28. Power ON switch

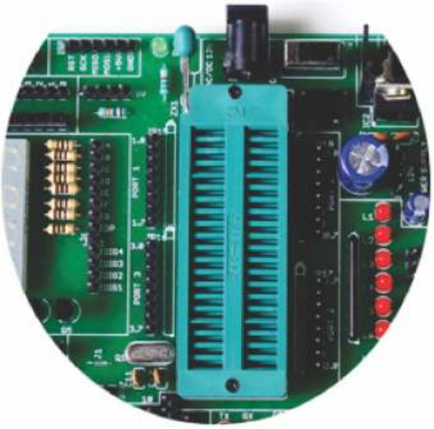
1. Power supply, 5V-12V

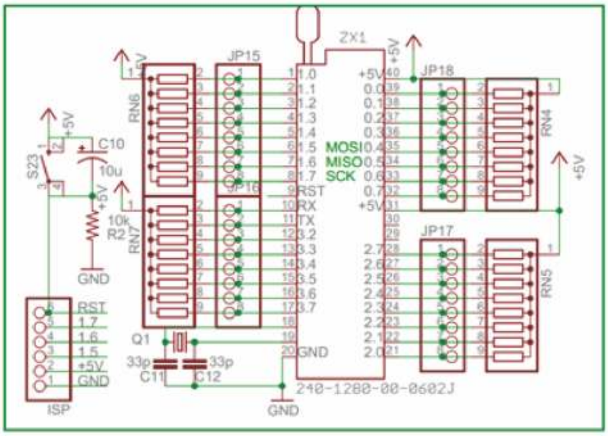
All digital circuits require regulated power supply. Here is a simple power supply circuit diagram used on this board. You can use AC or DC source (12V) which converts into regulated 5V which is required for driving the developmentboard circuit.



1. 40 pin ZIF socket for IC mount & ISP connector*

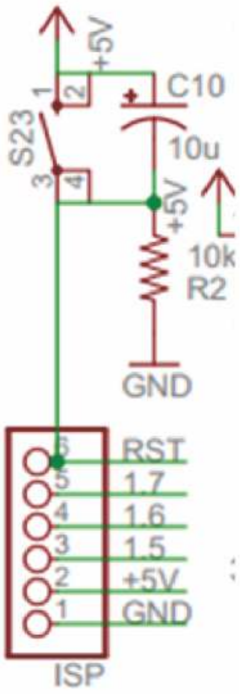
Select the IC's from the given list and mount on the ZIF socket. ZIF socket pin maps out PORT1 PORT2 PORT3 PORT4 for easy making connections for the restof the circuit. Port 1 is enabled with pull up circuit and also connected ISP for easy on board Programming.





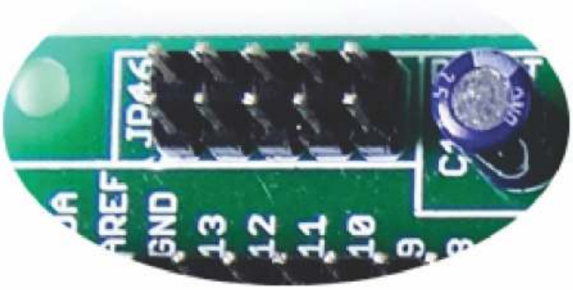
2. Reset

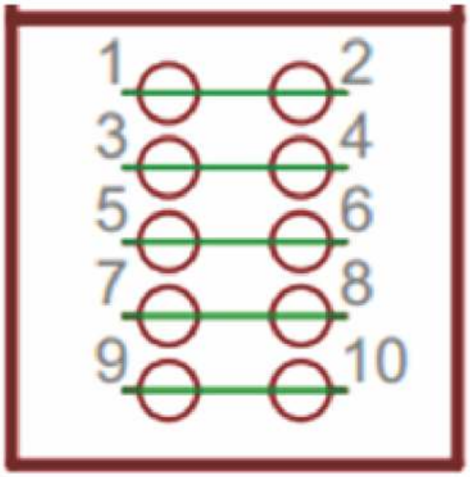
Resets your microcontroller by pressing s1
NOTE: if you are following to on board program, the capacitor c1 should be desoldered and removed from the port. You also need to know, if you remove the capacitor the board has to reset manually by pressing the reset button s1 each time you burn a code.



3. Node connector

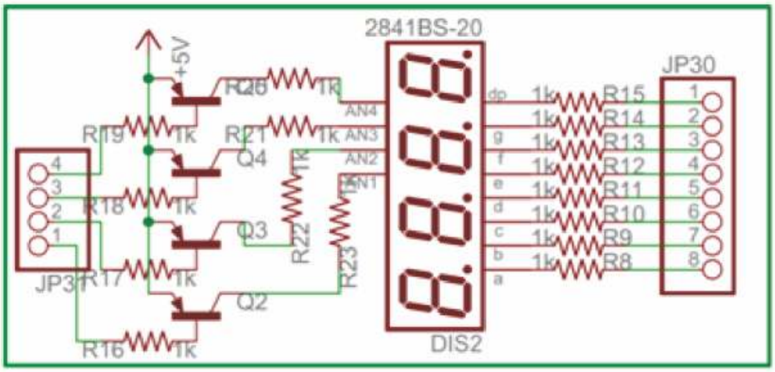
Node connector is an additional on board connection extender or 1 connection IN and 1 connection out





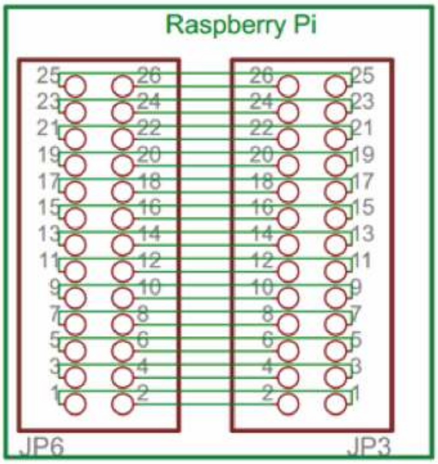
4. 4 digit 7 segment display

One seven segment digit consist of 7+1 LEDs which are arranged in a specific formation which can be used to represent digits from 0 to 9 and even some letters. One additional LED is used for marking the decimal dot, in case you want to write a decimal point in the desired segment.



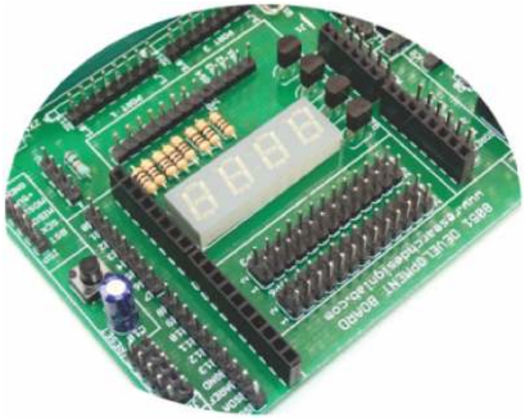
5. 26 pin raspberry connector

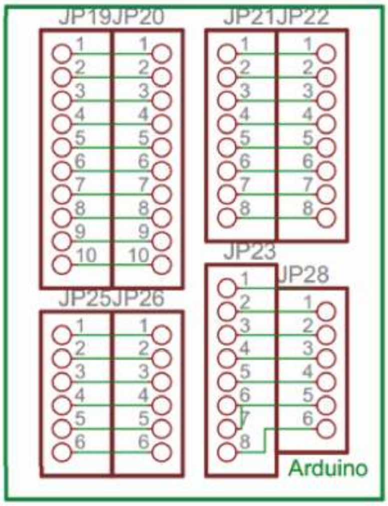
26 Pin raspberry connector is an easy way for making connections with raspberry pi with this development board.



6. Arduino Shield footprint

Arduino Shield footprint is provided in the board to mount different types of Arduino compatible shields on this development board.





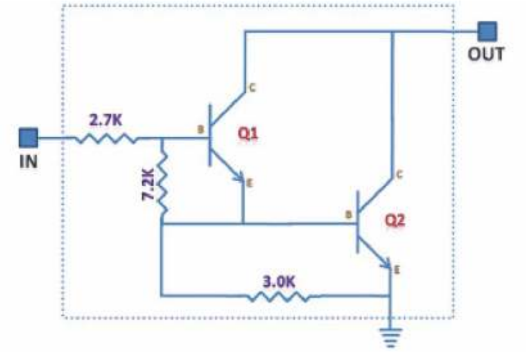
7. ULN 2803 driver

IC ULN2803 consists of octal high voltage, high current darlington transistor arrays. The eight NPN Darlington connected transistors in this family of arrays are ideally suited for interfacing between low logic level digital circuitry (such as TTL, CMOS or PMOS/NMOS) and the higher current/voltage requirements of lamps, relays, printer hammers or other similar loads for a broad range of computer, industrial, and consumer applications.

Features

- Eight Darlingtons with Common Emitter.
- Open-collector outputs.
- Free wheeling clamp diodes for transient suppression.
- Output Current to 500 mA.
- Output Voltage to 50 V.
- Inputs pinned opposite outputs to simplify board layout.

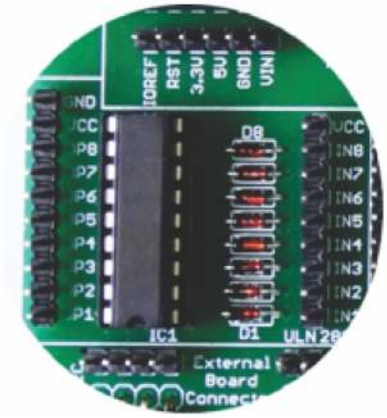
Working

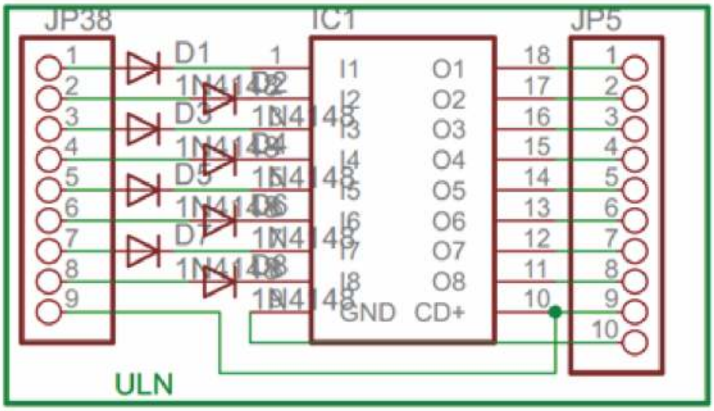


The ULN 2803 IC consists of eight NPN Darlington connected transistors (often called a Darlington pair). Darlington pair consists of two bipolar transistors such that the current amplified by the first is amplified further by the second to get a high current gain ß or hFE. The figure shown below is one of the eight Darlington pairs of ULN 2803 IC.

Now 2 cases arise:-
Case 1: When IN is 0 volts.
Q1 and Q2 both will not conduct as there is no base current provided to them. Thus, nothing will appear at the output (OUT).

Case 2: When IN is 5 volts.
Input current will increase and both transistors Q1 and Q2 will begin to conduct. Now, input current of Q2 is combination of input current and emitter current of Q1, so Q2 will conduct more than Q1 resulting in higher current gain which is very much required to meet the higher current requirements of devices like motors, relays etc. Output current flows through Q2 providing a path (sink) to ground for the external circuit that the output is applied to. Thus, when a 5V input is applied to any of the input pins (1 to 8), output voltage at corresponding output pin (11 to 18) drops down to zero providing GND for the external circuit. Thus, the external circuit gets grounded at one end while it is provided +Vcc at its other end. So, the circuit gets completed and starts operating.





8. I2C bus

One IC that wants to talk to another must: (Protocol)
1) Wait until it sees no activity on the I2C bus. SDA and SCL are both high. The bus is 'free'.

2) Put a message on the bus that says 'its mine' - I have STARTED to use the bus. All other ICs then LISTEN to the bus data to see whether they might be the one who will be called up (addressed).

3) Provide on the CLOCK (SCL) wire a clock signal. It will be used by all the ICs as the reference time at which each bit of DATA on the data (SDA) wire will be correct (valid) and can be used. The data on the data wire (SDA) must be valid at the time the clock wire (SCL) switches from 'low' to 'high' voltage.

4) Put out in serial form the unique binary 'address'(name) of the IC that it wants to communicate with.

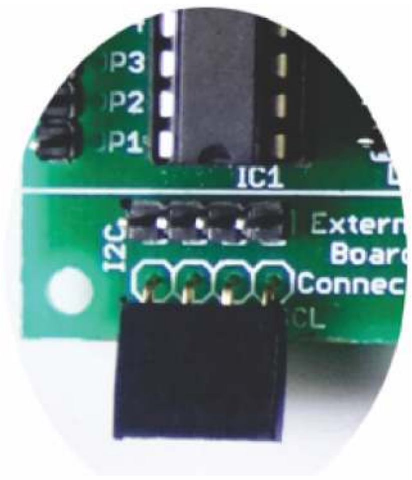
5) Put a message (one bit) on the bus telling whether it wants to SEND or RECEIVE data from the other chip. (The read/write wire is gone!)

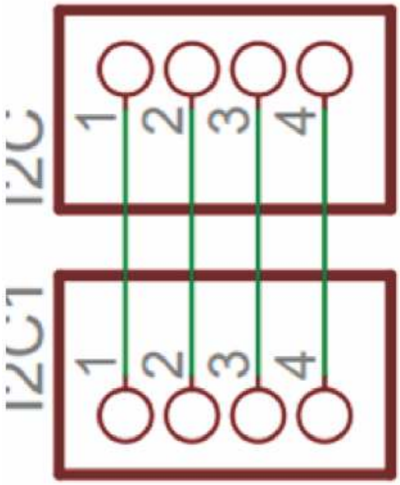
6) Ask the other IC to ACKNOWLEDGE (using one bit) that it recognized its address and is ready to communicate.

7) After the other IC acknowledges all is OK, data can be transferred.

8) The first IC sends or receives as many 8-bit words of data as it wants. After every 8-bit data word the sending IC expects the receiving IC to acknowledge the transfer is going OK.

9) When all the data is finished the first chip must free up the bus and it does that by a special message called 'STOP'. It is just one bit of information transferred by a special 'wiggling' of the SDA/SCL wires of the bus.





9. SPI bus

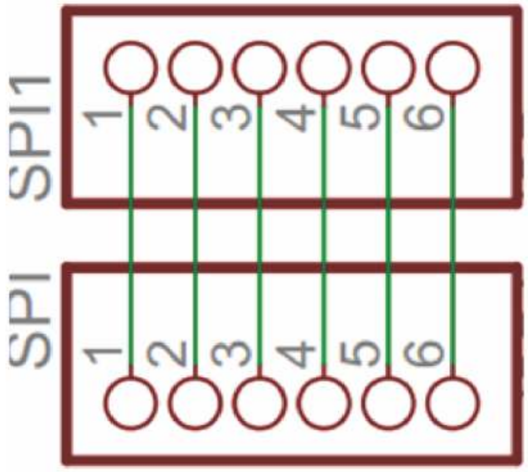
Serial to Peripheral Interface (SPI) is a hardware/firmware communications protocol developed by Motorola and later adopted by others in the industry. Microwire of National Semiconductor is same as SPI. Sometimes SPI is also called a "four wire" serial bus.

The Serial Peripheral Interface or SPI-bus is a simple 4-wire serial communications interface used by many microprocessor/microcontroller peripheral chips that enables the controllers and peripheral devices to communicate each other. Even though it is developed primarily for the communication between host processor and peripherals, a connection of two processors via SPI is just as well possible.

The SPI bus, which operates at full duplex (means, signals carrying data can go in both directions simultaneously), is a synchronous type data link setup with a Master / Slave interface and can support up to 1 megabaud or 10Mbps of speed. Both single-master and multi-master protocols are possible in SPI. But the multi-master bus is rarely used and look awkward, and are usually limited to a single slave.

The SPI Bus is usually used only on the PCB. There are many facts, which prevent us from using it outside the PCB area. The SPI Bus was designed to transfer data between various IC chips, at very high speeds. Due to this high-speed aspect, the bus lines cannot be too long, because their reactance increases too much, and the Bus becomes unusable. However, its possible to use the SPI Bus outside the PCB at low speeds, but this is not quite practical.

The peripherals can be a Real Time Clocks, converters like ADC and DAC, memory modules like EEPROM and FLASH, sensors like temperature sensors and pressure sensors, or some other devices like signal-mixer, potentiometer, LCD controller, UART, CAN controller, USB controller and amplifier.

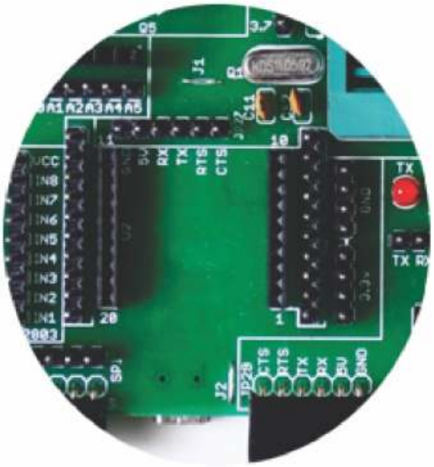


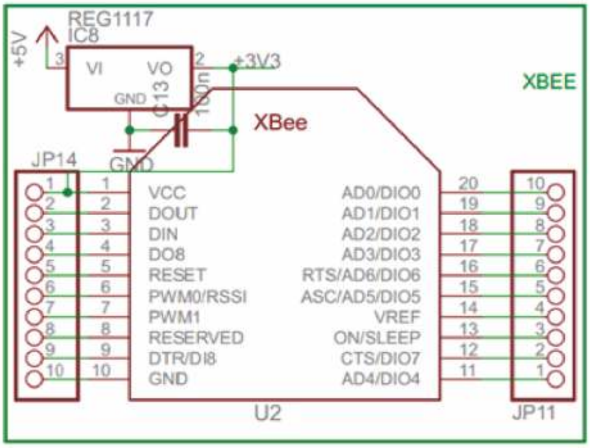
10. XBEE footprint/ XBEE Adaptor module

All XBeeZNet 2.5 modules can be identified by their unique 64-bit addresses or a user-configurable ASCII string identifier The 64-bit address of a module can be read using the SH and SL commands. The ASCII string identifier is configured using the NI command.

To transmit using device addressing, only the destination address must be configured. The destination address can be specified using either the destination device's 64-bit address or its NI-string. The XBee modules also support coordinator and broadcast addressing modes. Device addressing in the AT firmware is configured using the DL, DH, or DN commands. In the API firmware, the ZigBee Transmit Request API frame (0x10) can be used to specify destination addresses.

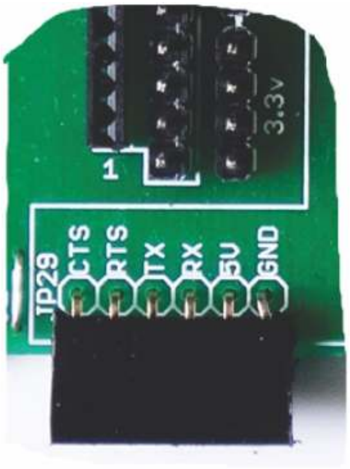
To address a node by its 64-bit address, the destination address must be set to match the 64-bit address of the remote. In the AT firmware, the DH and DL commands set the destination 64-bit address. In the API firmware, the destination 64-bit address is set in the ZigBee Transmit Request frame. ZigBee end devices rely on a parent (router or coordinator) to remain awake and receive any data packets destined for the end device. When the end device wakes from sleep, it sends a transmission (poll request) to its parent asking if the parent has received any RF data destined for the end device. The parent, upon receipt of the poll request, will send an RF response and the buffered data (if present). If the parent has no data for the end device, the end device may return to sleep, depending on its sleep mode configuration settings. The following figure demonstrates how the end device uses polling to receive RF data through its parent.

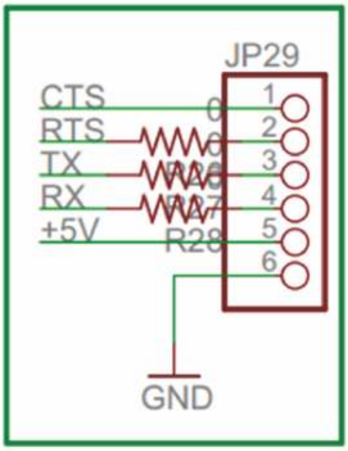




11. FT232 breakout board connector

A standard FT232 breakout board from researchdesignlab.com could be used to interface on these connectors, whose other end is connected to a USB.



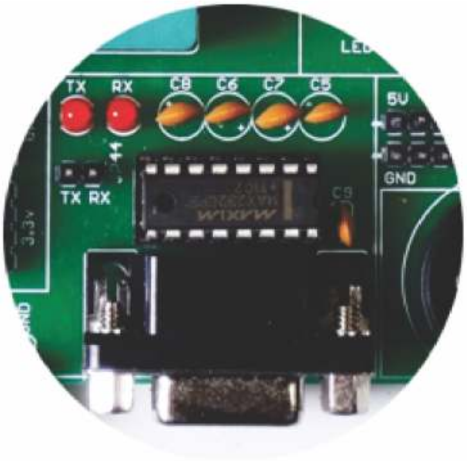


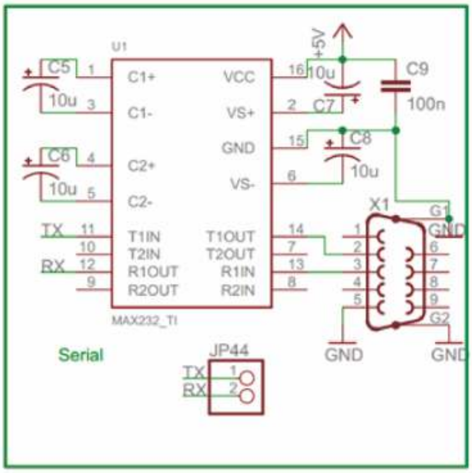
12. DC 3.3V connectors

These connectors provide on board 3.3V DC connections.

13. DB-9 female connector

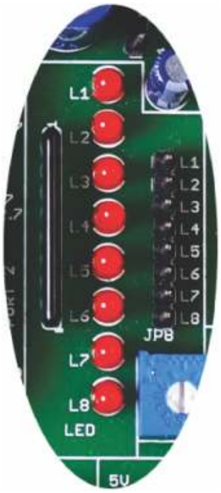
RS-232 is a standard communication protocol for linking computer and its peripheral devices to allow serial data exchange. In simple terms RS232 defines the voltage for the path used for data exchange between the devices. It specifies common voltage and signal level, common pin wire configuration and minimum, amount of control signals.

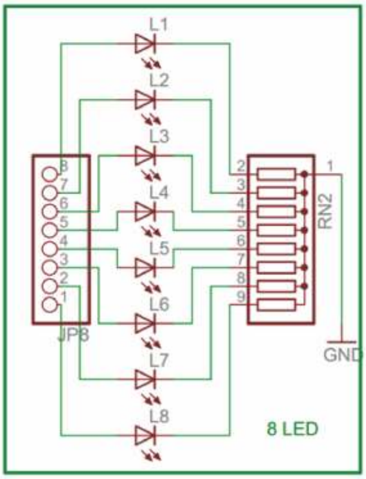




14. 8x1 LED's

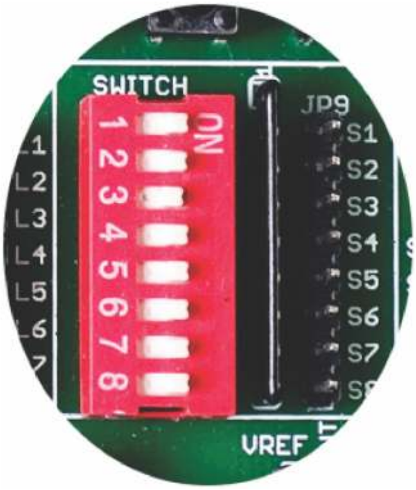
LED's are used to indicate something, whether any pin is high or indicating the output for many purposes like indicating I/O status or program debugging running state. We have four led outputs on board which can be used by the programmer as per the requirement for testing and development.

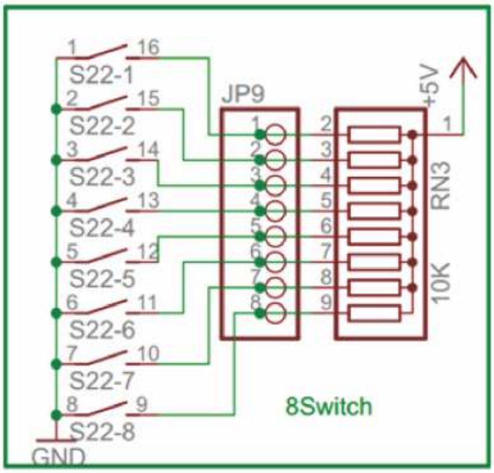




15. 8 way DIP switch

DIP switches are an alternative to jumper blocks. Their main advantages are that they are quicker to change and there are no parts to lose.





16.RTC Module

The DS1307 Serial Real Time Clock is a low power, full BCD clock/calendar plus 56 bytes of nonvolatile SRAM. Address and data are transferred serially via a 2-wire bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with less than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit which detects power failures and automatically switches to the battery supply.

Operation

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When VCC falls below 1.25 x VBAT the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When VCC falls below VBAT the device switches into a low current battery backup mode. Upon power up, the device switches from battery to VCC when VCC is greater than VBAT +0.2V and recognizes inputs.

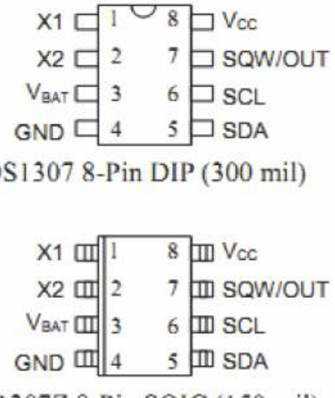
Features:

1. 56 byte nonvolatile RAM for data storage
2. 2-wire serial interface
3. Programmable square wave output signal
4. Automatic power-fail detect and switch circuitry
5. Consumes less than 500 nA in battery backup mode with oscillator running
6. Optional industrial temperature range -40°C to +85°C
7. Available in 8-pin DIP or SOIC
8. Recognized by Underwriters Laboratory

PIN DESCRIPTION

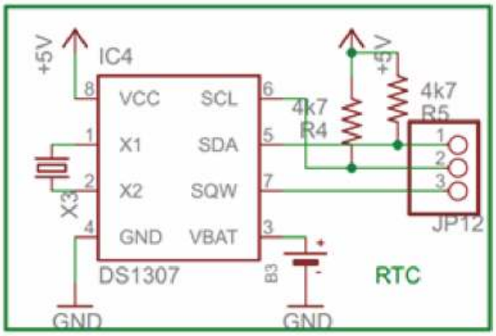
1. VCC - Primary Power Supply
2. X1, X2 - 32.768 kHz Crystal Connection
3. VBAT - +3V Battery Input
4. GND - Ground
5. SDA - Serial Data
6. SCL - Serial Clock
7. SQW/OUT - Square wave/Output Driver

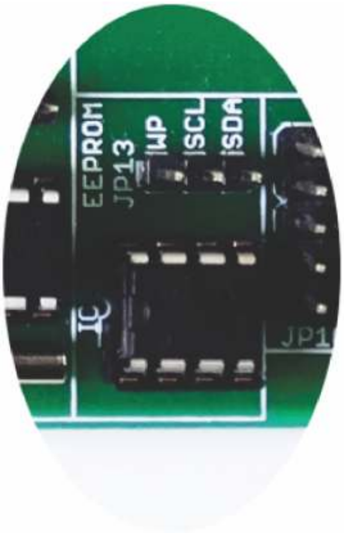
PIN ASSIGNMENT

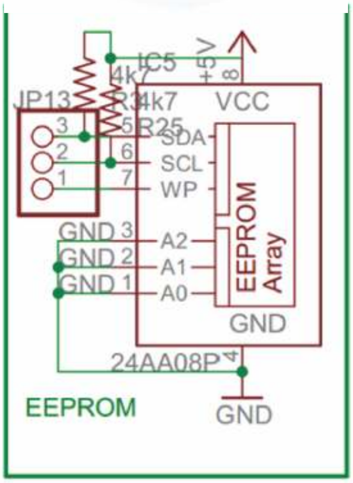


DS1307 8-Pin DIP (300 mil)

DS1307Z 8-Pin SOIC (150 mil)





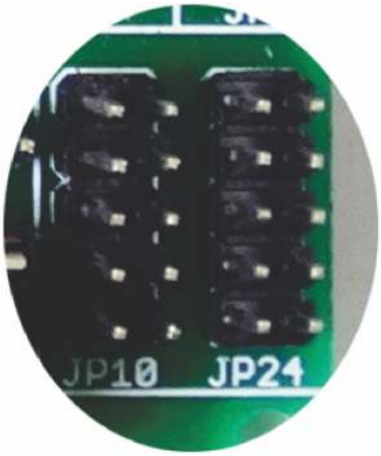


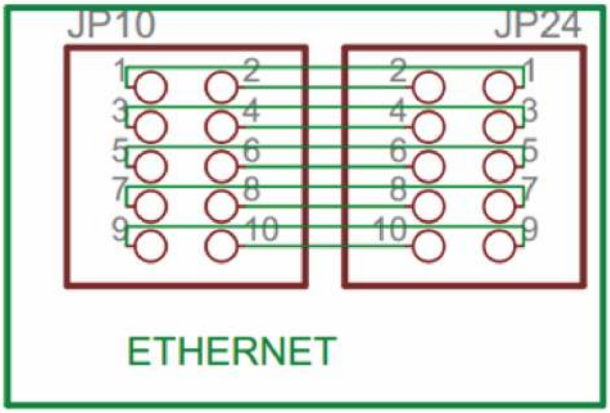
17. EEPROM

IC, EEPROM I2C 4K, 24C04, DIP8
Memory Size: 4Kbit
Memory Configuration: 512 x 8
Interface Type: I2C, Serial
Clock Frequency: 400kHz
Supply Voltage Range: 2.5V to 5.5V
Memory Case Style: DIP
No. of Pins: 8
Operating Temperature Range: -40°C to +85°C
SVHC: No SVHC (19-Dec-2011)
Base Number: 24
Device Marking: M24C04
IC Generic Number: 24C04
Interface: I2C
Interface Type: Serial, I2C
Logic Function Number: 24C04
Memory Configuration: 512 x 8
Memory Size: 4Kbit
Memory Type: EEPROM
Memory Voltage Vcc: 2.5V
Operating Temperature Max: +85°C
Operating Temperature Min: -40°C
Package / Case: DIP
Supply Voltage Max: 5.5V
Supply Voltage Min: 2.5V
Termination Type: Through Hole
Voltage Vcc: 2.5V

18. 2x5x2 jumper node

Node connector is an additional on board connection extender or 1 connection IN and 1 connection out





19. DC 5V connectors

These connectors provide on board 5V DC connections.

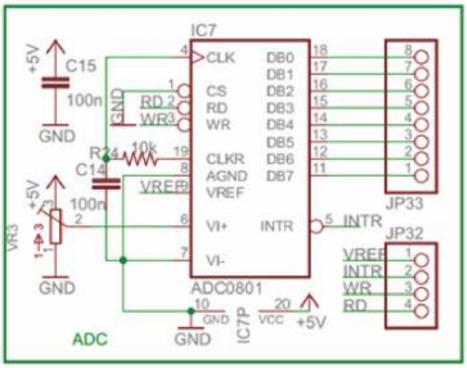
20. Analog to Digital output

The ADC0804 is CMOS 8-bit successiveapproximation A/Dconverters that use a differential potentiometric ladder similar to the 256R products. This converteris designed to allow operation with the NSC800 and INS8080Aderivative control bus with TRI-STATEoutput latches directly driving the data bus. These A/Ds appear like memory locations or I/O ports to the microprocessor and no interfacing logic is needed.

Differential analog voltage inputs allow increasing the common-mode rejection and offsetting the analog zero inputvoltage value.. In addition, the voltage reference input canbe adjusted to allow encoding any smaller analog voltagespan to the full 8 bits of resolution.

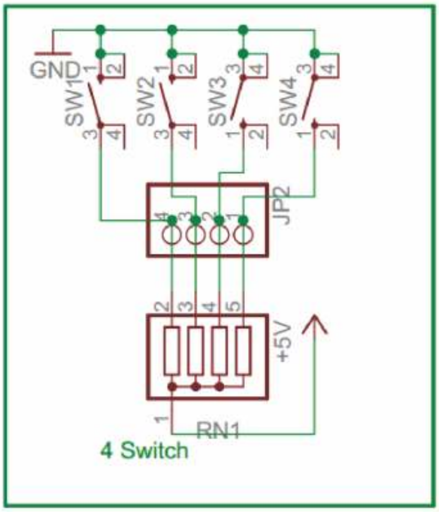
Features

- Compatible with 8080 µP derivatives no interfacing logic needed - access time - 135 ns
- Easy interface to all microprocessors, or operates ``stand-alone"
- Differential analog voltage inputs
- Differential analog voltage inputs
- Works with 2.5V (LM336) voltage reference
- On-chip clock generator
- 0V to 5V analog input voltage range with single 5V supply
- No zero adjust required
- 0.3× standard width 20-pin DIP package
- 20-pin molded chip carrier or small outline package
- Operates ratio metrically or with 5 VDC, 2.5 VDC, or analog span adjusted voltage reference



21. 4x1 keypad

Switches are mainly used to switch the controls of a module. We have four switches on board which can be used by the programmer as per the requirement for testing and development.



22. 16x2 LCD connectors

LCD screen consists of two lines with 16 characters each. Each character consists of 5x7 dot matrix. Contrast on display depends on the power supply voltage and whether messages are displayed in one or two lines. For that reason, variable voltage 0-Vdd is applied on pin marked as Vee. Trimmer potentiometer is usually used for that purpose. Some versions of displays have built in backlight (blue or green diodes). When used during operating, a resistor for current limitation should be used (like with any LE diode). LCD Connection Depending on how many lines are used for connection to the microcontroller, there are 8-bit and 4-bit LCD modes. The appropriate mode is determined at the beginning of the process in a phase called “initialization”. In the first case, the data are transferred through outputs D0-D7 as it has been already explained. In case of 4-bit LED mode, for the sake of saving valuable I/O pins of the microcontroller, there are only 4 higher bits (D4-D7) used for communication, while other may be left unconnected.

Consequently, each data is sent to LCD in two steps: four higher bits are sent first (that normally would be sent through lines D4-D7), four lower bits are sent afterwards. With the help of initialization, LCD will correctly connect and interpret each data received. Besides, with regards to the fact that data are rarely read from LCD (data mainly are transferred from microcontroller to LCD) one more I/O pin may be saved by simple connecting R/W pin to the Ground. Such saving has its price. Even though message displaying will be normally performed, it will not be possible to read from busy flag since it is not possible to read from display.

Features:

1. Can display 224 different symbols.
2. Low power consumption.
3. 5x7 dot matrix format.
4. Powerful command set and user produced characters.

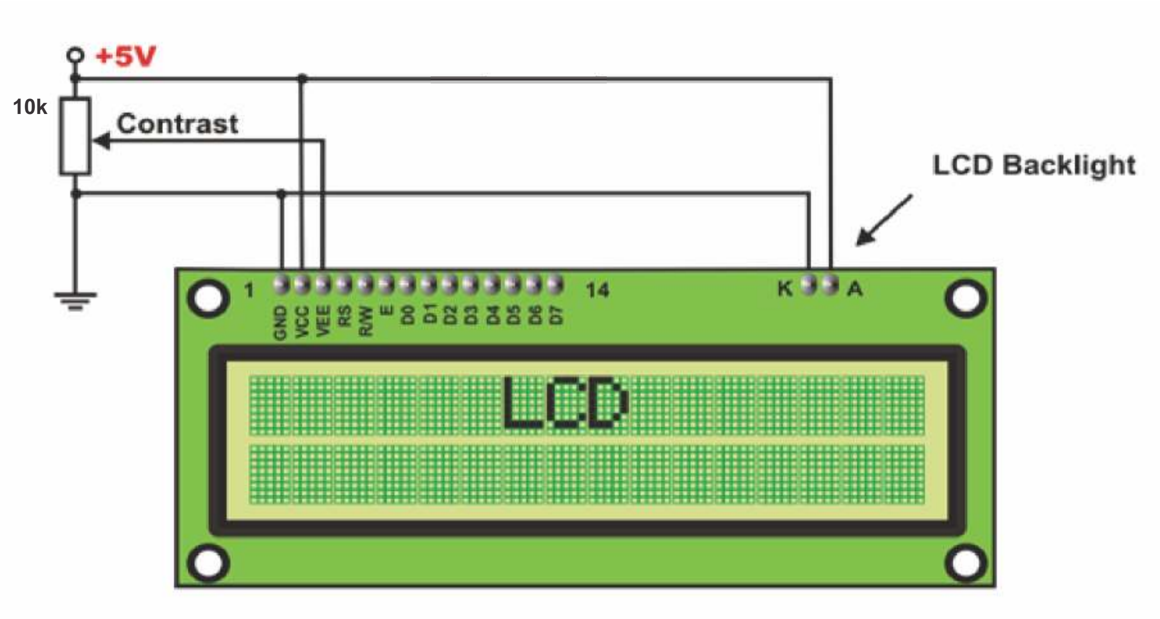
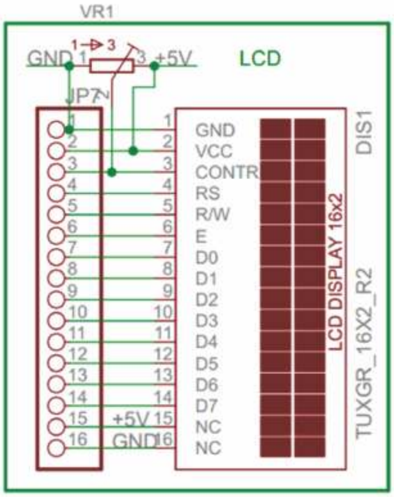
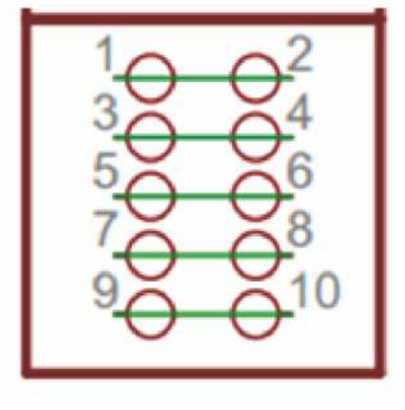


Fig: Circuit connections of LCD



PIN DESCRIPTION

1. Gnd:- Power supply ground
2. VCC:-+5v Power supply input
3. RS:- Reset pin
4. R/W:- Read/Write pin
5. En:-Enable pin
6. D0-D7:- Data lines



23. Node connector

Node connector is an additional on board connection extender or 1 connection IN and 1 connection out

24. 4x4 Matrix Keypad

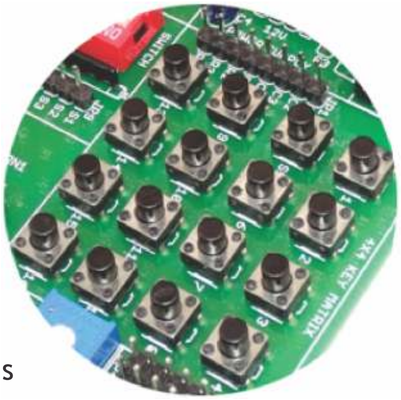
In a 4x4 matrix keypad eight Input/Output ports are used for interfacing with any microcontrollers. Rows are connected to Peripheral Input/Output (PIO) pins configured as output. Columns are connected to PIO pins configured as input with interrupts. In this configuration, four pull-up resistors must be added in order to apply a high level on the corresponding input pins as shown in below Figure. The corresponding hexadecimal value of the pressed key is sent on four LEDs.

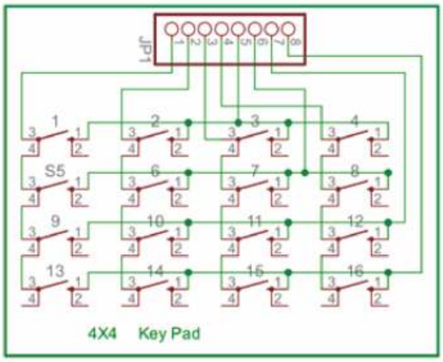
WORKING

This Application Note describes programming techniques implemented on the AT91 ARM-based microcontroller for scanning a 4x4 Keyboard matrix usually found in both consumer and industrial applications for numeric data entry.AT91 Keyboard interface In this application, a 4x4 matrix keypad requiring eight Input/Output ports for interfacing is used as an example. Rows are connected to Peripheral Input/Output (PIO) pins configured as output. Columns are connected to PIO pins configured as input with interrupts. In this configuration, four pull-up resistors must be added in order to apply a high level on the corresponding input pins as shown in Figure 1. The corresponding hexadecimal value of the pressed key is sent on four LEDs.

FEATURES

1. Contact debouncing.
2. Easy to interface.
3. Interfaces to any microcontroller or microprocessor.
4. Data valid output signal for interrupt activation.





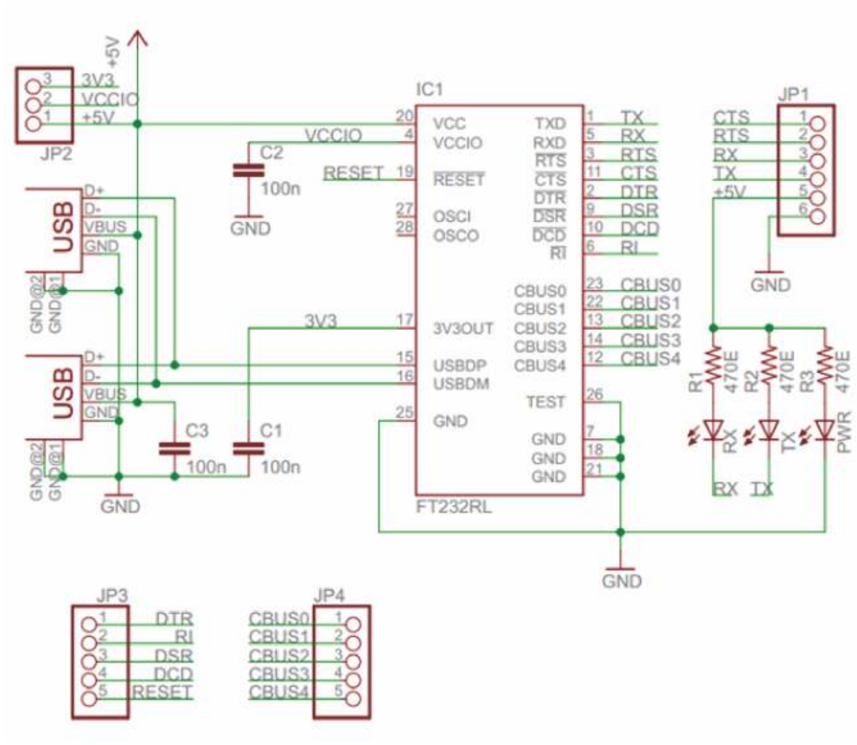
PIN DETAILS
pin 1-4: R0-R3:- Rows
pin 5-8: C0-C3:- Columns

25.DC 12V connectors

These connectors provide on board 12V DC connections.

26.USB to serial converter (optional)

In modern computers you cannot find serial port (DB-9). But most of the basic controllers work with this protocol. To connect your system to such controllers we require USB to serial converters. This board has the facility to be connected directly to USB using a USB cable (A to B).



Programming Codes:

- LED BLINK
http://researchdesignlab.com/8051-i/o-code
- LCD
http://researchdesignlab.com/8051-lcd-code
- KEYPAD
http://researchdesignlab.com/8051-keypad-code
- UART
http://researchdesignlab.com/8051-uart-code
- RTC
http://researchdesignlab.com/8051-rtc-code
- EEPROM
http://researchdesignlab.com/8051-eeprom-code
- ADC
http://researchdesignlab.com/8051-adc-code.html
- 7 Segment Display
http://researchdesignlab.com/7-segment-atmel-code.html