Project Report Titled

**Automated Seed Sower**

Submitted

in partial fulfillment for

the requirements of the diploma in

**Electronics Engineering**

(Semester V)

By

**Mr. Amanshu S Tiwari ID 164120024**

**Mr. Antariksh Mohandas ID 164160017**

**Mr. Nishant G Bangar ID 164130020**

**Mr. Omkar A Bhilare ID 164130047**

**Mr. Om V Chaurasia ID 164130013**

**Mr. Saheel S Patil ID 164130037**

**Mr. Swapnil V Gore ID 164130037**

**Prof. Ami Dapkawala**

**Guide**



**Electrical Engineering Department**

**Veermata Jijabai Technological Institute**

Mumbai 400 019

2018-19

**CERTIFICATE**

This is to certify that

**Mr. Amanshu S Tiwari ID 164120024**

**Mr. Antariksh Mohandas ID 164160017**

**Mr. Nishant G Bangar ID 164130020**

**Mr. Omkar A Bhilare ID 164130047**

**Mr. Om V Chaurasia ID 164130013**

**Mr. Saheel S Patil ID 164130037**

**Mr. Swapnil V Gore ID 164130053**

has successfully completed and submitted the project entitled

**Automated Seed Sower**

for the partial fulfillment of

**Diploma**

in

**Electronics Engineering**

**(Semester VI)**

during the academic year 2018-2019 as prescribed by VJTI Mumbai.

Head Diploma (Electronics)

Prof. Ami Dapkawala

Guide External Examiner

Date:-

Place:- Mumbai

**ACKNOWLEDGEMENT**

In performing our project, we had been assisted by our respected guide ,who deserve our greatest gratitude. The completion of this project gives us much pleasure. We would like to show our gratitude to **Ms. Ami Dapkawala (Professor, Electronics Department, VJTI)** for giving us good guideline for project throughout numerous consultations. They never left a single stone unturned for us providing with all the possible resources, when we were in dire need of them, we would also like to expand our deepest gratitude to all those directly and indirectly guided us in this project. We take this opportunity to thank **Mr. Head of Department (Head of Diploma Electronics Engineering Department, VJTI)** and **Dr. Dhiren Patel (Director, VJTI)** for having permitted us to carry out this project.

We are also grateful to our college for providing us with the internet lab and it’s co-operative employees where we were allowed browsing, learning and hunting every this we could think about studying this project. We would like to offer thanks to the Mechanical Department for allowing us to carry out our project work in the Mechanical workshop and the laboratory assistants for their co-operation for providing us with all the possible resources.

Lastly, we thank our parents, friends and well-wishers who always looked for a chance to help us in whatever means came forth and for their constant encouragement, without which the project would be a distant reality. Above all, we express our deep sense of gratitude to the almighty God who gave us immense strength and showed us path to make this project victorious.

**Abstract**

India, the fourth largest agricultural country in the world has 60% of its total area as agricultural land, with 144.3 million farm laborers (as per 2011 data) working for feeding the nation having the same ordinary techniques for farming. For the progress in the farming technology, large production with minimum amount of labor is made possible with the Automated Seed Sower. This machinery would help the farmers to control the human friendly robot by just putting up the required land data into it. With this the controller will be able to farm whole of its land by just sitting in one place and watching the robot do everything. This would reduce the complexities in hiring manual labor to do the job and jumping up to child labor for increase in mass production. No country has ever succeeded without it’s techno-enthusiasts empowering the nation and bringing about the growth required for a change and to economically compete with the other nations. The automated seed sower would bring about a drastic change in the income of the farmers and would yield the economic graph of the nation above and above as time passes. The working of the robot has been designed in such a way that an automated digging mechanism and a seed mechanism would sow the seeds as per the data given by the user. Almost many farmers across country have been showing keen interest in the robotic farming technology.

**1. PROJECT REPORT**

**1.1 SELECTION OF GROUP**

Collection of people with an ultimate aim of working for the project with full dedication and hard work giving their full potential for making this project a grand success.

**1.2 FACTORS FOR SELECTION OF PROJECT**

* It should be widely useful for the main aim for which it is designed for.
* It should be in the economic range of the customers.
* Knowledge skills must be improved.
* Everyone must approve their decision for designing the same.
* It should not harm the environmental conditions in any way.

**1.3 OBJECTIVE OF THE PROJECT**

* To bring about the change in the farming technology.
* To increase the economic growth of the nation.
* To minimize labour and bring about large production in minimum time.

**2. INTRODUCTION**

There hasn’t been a lot of improvements in the farming technology over past three to four years. Humankind has experienced a lot of advancements leading to maximum production in minimum amount of time. The automated seed sower focuses on the aim of reducing the manual labor and replacing it with a programmed robot which can be controlled by the user easily. The automated bot works on the program uploaded by the engineer working on the development of the bot. The farming land dimensions are to be entered by the user which would be processed by the microcontroller to run the program. The seed sowing mechanism would be running in sync with the bot so that the seeds are sowed efficiently at the exact place.

As the bot moves in the farm, a digging mechanism at the frame work would dig the farming land for the seeds. In this way, the digging, moving and the seed sowing mechanism would help in the overall working of the robot. A levelling mechanism is used in the pillion of the chassis of the framework which would help in the final soil covering for the seed. The Automated seed sower aims at the economic growth in the rural areas where the farmers are deprived of the latest farming technology. This would surely bring about the required advancement in the efficient small as well as large scale farming for a cost effective and user friendly solution.

**3. CONSTRUCTION**

**3.1 CHASSIS:**

The framework of the Automated seed sower has a rigid rectangular wooden chassis which would provide a strong mounting base to the robot All of the mechanisms. such as the seed sowing mechanism, the digging mechanism and the levelling mechanism are mounted on this wooden rectangular frame of the robot.

The dimensions of the chassis are (“”).

**3.2 WHEELS:**

The stability of the bot is depended totally on the wheels which are mounted on the chassis. The wheels are of a diameter of (\*\*). The wheels have over them which helps the bot to sustain on any land structure. The wheels are of hard fiber with a grip bound over it.

**3.2 SEED SOWING ASSEMBLY:**

The seeding mechanism comprises of the seed holder, the seed container, the rotating disc and the outlet pipe. The seed container is placed on to the holder which acts a support to the seed container. The disc is mounted on a servo motor at the base of the holder.

The rotating disc rotates at specific intervals set by the engineer. In this way the seeding mechanism assembly is mounted on the frame work and the outlet pipe is fed in the downward direction through the chassis.

**3.3 MECHANICAL DIGGER**

The digging process is carried out by a metallic mixer roller which is mounted via a motor to the chassis in front of the seeding mechanism. The dig to be made in the farm land must be of a major depth. Hence two parallel diggers on a metal pole are mounted to the face of the bot.

**3.4 GROUND LEVELLER**

A 360o rolling grip attested leveler is mounted on the pillion which has a hardbound grip over it to give a clean level to the farm land after the seed is sowed. The leveler performs the final covering of the seed with the same soil dug by the mechanical digger.

**4. DESCRIPTION OF COMPONENTS**

**4.1 I2C MODULE:**

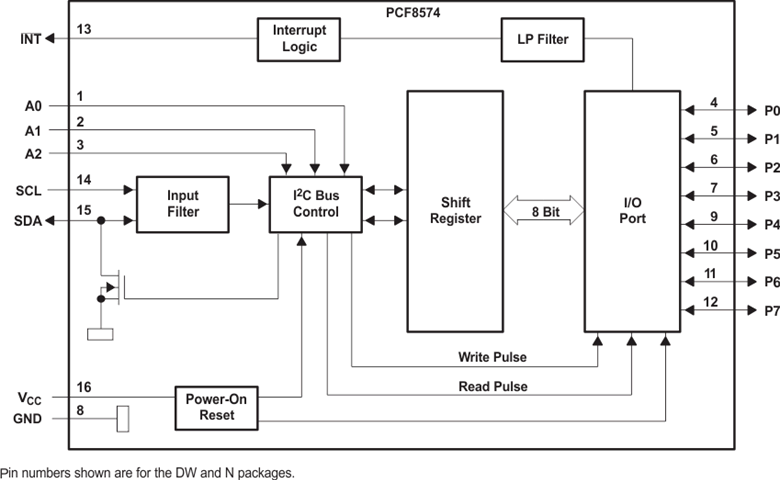


Fig (1.0)

This is a RoHS compliant I2C Serial LCD Daughter board that can be connected to a standard 16×2 or 20×4 Character Display Module that supports 4-bit mode. All Character Modules support 4-bit mode, and nearly all commercially available 16×2 and 20×4 line character modules support it too.

This board has a PCF8574 I2C chip that converts I2C serial data to parallel data for the LCD display. The I2C address is 0x3F by default, but this can be changed via 3 solder jumpers provided on the board. This allows up to 3 LCD displays to be controlled via a single I2C bus. For the description of the pin is given in table ().

The I2C module works efficiently when the user does not exceed the given limit by the manufacturer. These characteristics are called as Electrical characteristics of the component being used. The electrical characteristics of I2C module are shown in the table ()

Refer to the table below for the pin description of I2C module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PIN** | | | | | **TYPE** | **DESCRIPTION** |
| **NAME** | **RGT** | **RGY** | **DGV or PW** | **DW or N** |
| A [0.2] | 2, 3, 4 | 6, 7, 9 | 6, 7, 9 | 1, 2, 3 | I | Address inputs 0 through 2. Connect directly to VCC or ground. Pullup resistors are not needed. |
| GND | 9 | 15 | 15 | 8 | — | Ground |
| INT | 14 | 1 | 1 | 13 | O | Interrupt output. Connect to VCC through a pullup resistor. |
| NC | - | 3, 8, 13, 18 | 3, 8, 13, 18 | - | — | Do not connect |
| P[0..7] | 5, 6, 7, 8,  10, 11, 12,  13 | 10, 11, 12,  14, 16, 17,  19, 20 | 10, 11, 12,  14, 16, 17,  19, 20 | 4, 5, 6, 7,  9, 10, 11,  12 | I/O | P-port input/output. Push-pull design structure. |
| SCL | 15 | 2 | 2 | 14 | I | Serial clock line. Connect to VCC through a pullup resistor |
| SDA | 16 | 4 | 4 | 15 | I/O | Serial data line. Connect to VCC through a pullup resistor. |
| VCC | 1 | 5 | 5 | 16 | — | Voltage supply |

Table (1.0)

The specifications of the I2C module along with the features are

1. Serial I2C control of LCD display using PCF8574.
2. 5V power supply.
3. Backlight can be enabled or disabled via a jumper on the board.
4. Contrast control via a potentiometer.
5. Can have 8 modules on a single I2C bus (change address via solder jumpers) address, allowing.
6. Size ：41.6 x 19.2 mm.

Refer to the table below for the absolute maximum ratings of the I2C module

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **MIN MAX** | **UNIT** |
| VCC Supply voltage range | | –0.5 7 | V |
| VI Input voltage range | | –0.5 VCC + 0.5 | V |
| VO Output voltage range | | –0.5 VCC + 0.5 | V |
| IIK Input clamp current | VI < 0 | –20 | mA |
| IOK Output clamp current | VO < 0 | –20 | mA |
| IOK Input/output clamp current | VO < 0 or VO > VCC | ±400 | μA |
| IOL Continuous output low current | VO = 0 to VCC | 50 | mA |
| IOH Continuous output high current | VO = 0 to VCC | –4 | mA |
| Continuous current through VCC or GND | | ±100 | mA |
| TJ Junction temperature | | 150 | °C |
| Tstg Storage temperature range | | –65 150 | °C |

Table (1.1)

###### The recommended operating conditions are given in the table ()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PARAMETERS** | | **MIN** | **MAX** | **UNIT** |
| VCC | Supply voltage | 2.5 6 | | V |
| VIH | High-level input voltage | 0.7 × VCC | VCC + 0.5 | V |
| VIL | Low-level input voltage | –0.5 | 0.3 × VCC | V |
| IOH | High-level output current | –1 | | mA |
| IOL | Low-level output current | 25 | | mA |
| TA | Operating free-air temperature | –40 | 85 | °C |

Table (1.2)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PARAMETER** | | **TEST CONDITIONS** | | | **VCC** | **MIN** | **TYP(1)** | **MAX** | **UNIT** |
| VIK | Input diode clamp voltage | II = –18 mA | | | 2.5 V to 6 V | –1.2 | | | V |
| VPOR | Power-on reset voltage( | VI = VCC or GND, | IO = 0 |  | 6 V |  | 1.3 | 2.4 | V |
| IOH | P port | VO = GND | | | 2.5 V to 6 V | 30 |  | 300 | μA |
| IOHT | P port transient pullup current | High during acknowledge, VOH = GND | | | 2.5 V | –1 | | | mA |
| IOL | SDA | VO = 0.4 V | | | 2.5 V to 6 V | 3 | | | mA |
| P port | VO = 1 V | | | 5 V | 10 | 25 |  |
| INT | VO = 0.4 V | | | 2.5 V to 6 V | 1.6 | | |
| II | SCL, SDA | VI = VCC or GND | | | 2.5 V to 6 V | ±5 | | | μA |
| INT | ±5 | | |
| A0, A1, A2 | ±5 | | |
| IIHL | P port | VI ≥ VCC or VI ≤ GND | | | 2.5 V to 6 V | ±400 | | | μA |
| ICC | Operating mode | VI = VCC or GND, | IO = 0, | fSCL = 100 kHz | 6 V |  | 40 | 100 | μA |
| Standby mode | VI = VCC or GND, | IO = 0 |  |  | 2.5 | 10 |
| Ci | SCL | VI = VCC or GND | | | 2.5 V to 6 V | 1.5 7 | | | pF |
| Cio | SDA | VIO = VCC or GND | | | 2.5 V to 6 V |  | 3 | 7 | pF |
| P port |  | 4 | 10 |

Table (1.3)

**4.2 Servomotor(to be replaced)**

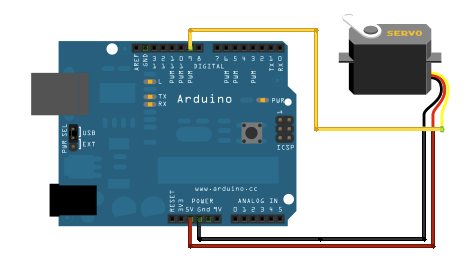


Fig (1.1)

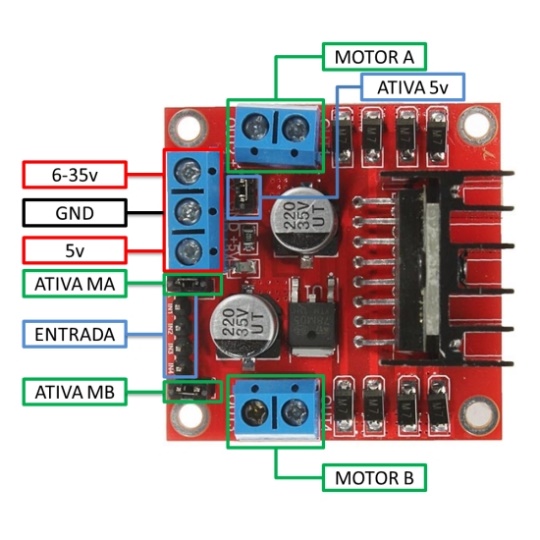
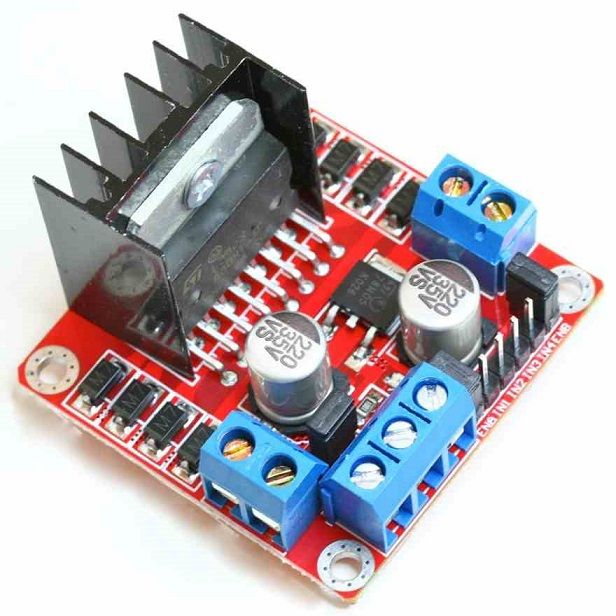
A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors.

A servomotor is a closed loop servo mechanism that uses position feedback to control its motion and final position. The input to its control is a signal (either analogue or digital) representing the position commanded for the output shaft.

The motor is paired with some type of encoder to provide position and speed feedback. In the simplest case, only the position is measured. The measured position of the output is compared to the command position, the external input to the controller. If the output position differs from that required, an error signal is generated which then causes the motor to rotate in either direction, as needed to bring the output shaft to the appropriate position. As the positions approach, the error signal reduces to zero and the motor stops.

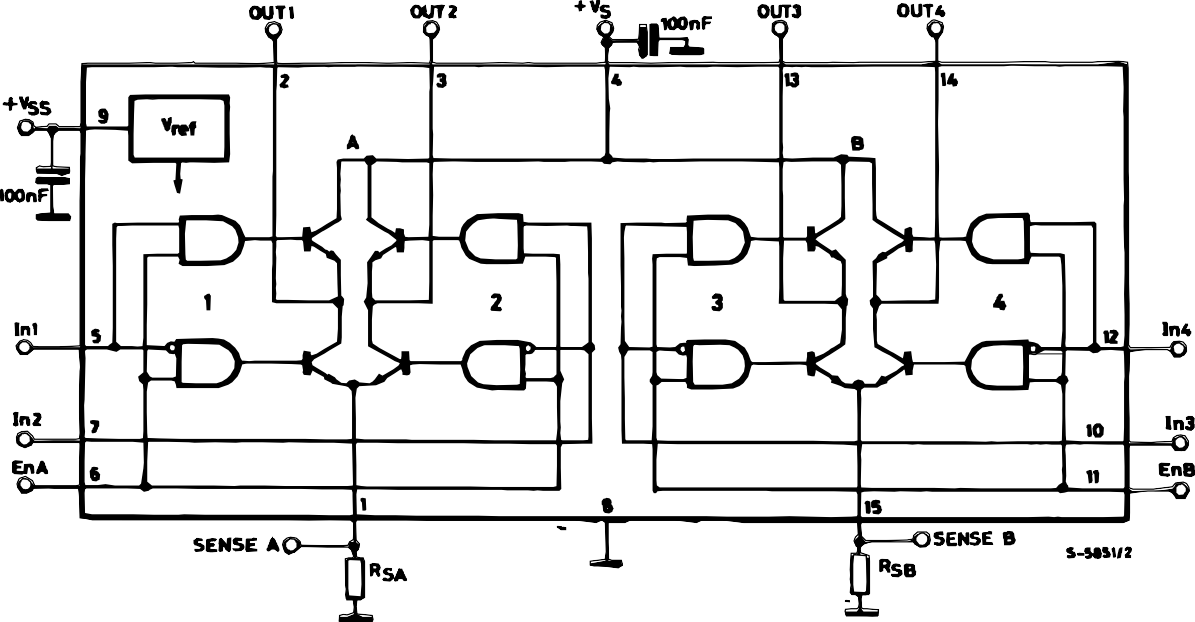
The very simplest servomotors use position-only sensing via a potentiometer and bang-bang control of their motor; the motor always rotates at full speed (or is stopped). This type of servomotor is not widely used in industrial modern control, but it forms the basis of the simple and cheap servos used for radio- controlled models.

**4.3 Motor driver L298n:**



Fig(1.2) Fig(1.3)

The L298 is an integrated monolithic circuit in a 15-lead Multi watt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.



Fig(1.4)

Driver specifications

* Driver: L298N
* Driver power supply: +5V~+46V
* Driver Io: 2A
* Logic power output Vss: +5~+7V (internal supply +5V)
* Logic current: 0~36mA
* Controlling level: Low -0.3V~1.5V, high: 2.3V~Vss
* Enable signal level: Low -0.3V~1.5V, high: 2.3V~Vss
* Max power: 25W (Temperature 75 cesus)
* Working temperature: -25C~+130C
* Dimension: 60mm\*54mm
* Driver weight: ~48g
* Other extensions: current probe, controlling direction indicator, pull-up resistoer switch, logic part power supply.

**ABSOLUTE MAXIMUM RATINGS**

|  |  |  |  |
| --- | --- | --- | --- |
| **S ymb ol** | **P arameter** | **Value** | **Uni t** |
| VS | Power Supply | 50 | V |
| VSS | Logic Supply Voltage | 7 | V |
| VI,Ven | Input and Enable Voltage | ±0.3 to 7 | V |
| IO | Peak Output Current (each Channel) | 3 | A |
|  | ± Non Repetitive (t = 100s) |
|  | ±Repetitive (80% on ±20% off; ton = 10ms) | 2.5 | A |
|  | ±DC Operation | 2 | A |
| Vsens | Sensing Voltage | ±1 to 2.3 | V |
| Ptot | Total Power Dissipation (Tcase = 75C) | 25 | W |
| Top | Junction Operating Temperature | ±25 to 130 | C |
| Tstg, Tj | Storage and Junction Temperature | ±40 to 150 | C |

Table()

**PIN FUNCTIONS**

|  |  |  |  |
| --- | --- | --- | --- |
| **M W. 15** | **Power SO** | **Name** | **Function** |
| 1;15 | 2;19 | Sense A; Sense B | Between this pin and ground is connected the sense resistor to |
|  |  |  | control the current of the load. |
| 2;3 | 4;5 | Out 1; Out 2 | Outputs of the Bridge A; the current that flows through the load |
|  |  |  | connected between these two pins is monitored at pin 1. |
| 4 | 6 | VS | Supply Voltage for the Power Output Stages. |
|  |  |  | A non-inductive 100nF capacitor must be connected between this |
|  |  |  | pin and ground. |
| 5;7 | 7;9 | Input 1; Input 2 | TTL Compatible Inputs of the Bridge A. |
| 6;11 | 8;14 | Enable A; Enable B | TTL Compatible Enable Input: the L state disables the bridge A |
|  |  |  | (enable A) and/or the bridge B (enable B). |
| 8 | 1,10,11,20 | GND | Ground. |
| 9 | 12 | VSS | Supply Voltage for the Logic Blocks. A100nF capacitor must be |
|  |  |  | connected between this pin and ground. |
| 10; 12 | 13;15 | Input 3; Input 4 | TTL Compatible Inputs of the Bridge B. |
| 13; 14 | 16;17 | Out 3; Out 4 | Outputs of the Bridge B. The current that flows through the load |
|  |  |  | connected between these two pins is monitored at pin 15. |
| ± | 3;18 | N.C. | Not Connected |

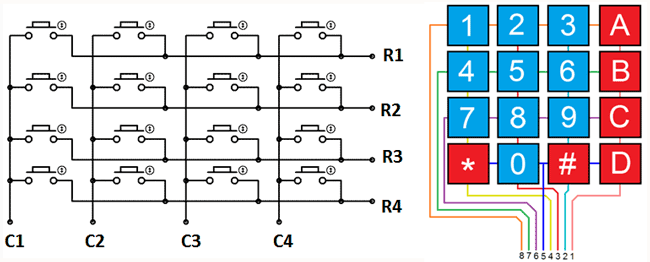
Table()

**ELECTRICAL CHARACTERISTICS** (VS= 42V; VSS= 5V, Tj= 25C; unless otherwise specified)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Symbol** | **Parameter** | **Test Co ndi tions** | | | **M in .** | **Typ .** | **Max.** | **Uni t** |
| VS | Supply Voltage (pin 4) | Operative Condition | |  | VIH +2.5 |  | 46 | V |
| VSS | Logic Supply Voltage (pin 9) |  |  |  | 4.5 | 5 | 7 | V |
| IS | Quiescent Supply Current (pin 4) | Ven = H; | IL = 0 | Vi = L |  | 13 | 22 | mA |
|  |  |  |  | Vi = H |  | 50 | 70 | mA |
|  |  | Ven = L |  | Vi = X |  |  | 4 | mA |
| ISS | Quiescent Current from VSS (pin 9) | Ven = H; | IL = 0 | Vi = L |  | 24 | 36 | mA |
|  |  |  |  | Vi = H |  | 7 | 12 | mA |
|  |  | Ven = L |  | Vi = X |  |  | 6 | mA |
| ViL | Input Low Voltage |  |  |  | ±0.3 |  | 1.5 | V |
|  | (pins 5, 7, 10, 12) |  |  |  |  |  |  |  |
| ViH | Input High Voltage |  |  |  | 2.3 |  | VSS | V |
|  | (pins 5, 7, 10, 12) |  |  |  |  |  |  |  |
| IiL | Low Voltage Input Current | Vi = L |  |  |  |  | ±10 | A |
|  | (pins 5, 7, 10, 12) |  |  |  |  |  |  |  |
| IiH | High Voltage Input Current | Vi = H ≤ VSS ±0.6V | |  |  | 30 | 100 | A |
|  | (pins 5, 7, 10, 12) |  |  |  |  |  |  |  |
| Ven = L | Enable Low Voltage (pins 6, 11) |  |  |  | ±0.3 |  | 1.5 | V |
| Ven = H | Enable High Voltage (pins 6, 11) |  |  |  | 2.3 |  | VSS | V |
| Ien = L | Low Voltage Enable Current | Ven = L |  |  |  |  | ±10 | A |
|  | (pins 6, 11) |  |  |  |  |  |  |  |
| Ien = H | High Voltage Enable Current | Ven = H ≤ VSS ±0.6V | |  |  | 30 | 100 | A |
|  | (pins 6, 11) |  |  |  |  |  |  |  |
| VCEsat (H) | Source Saturation Voltage | IL = 1A |  |  | 0.95 | 1.35 | 1.7 | V |
|  |  | IL = 2A |  |  |  | 2 | 2.7 | V |
| VCEsat (L) | Sink Saturation Voltage | IL = 1A | (5) |  | 0.85 | 1.2 | 1.6 | V |
|  |  | IL = 2A | (5) |  |  | 1.7 | 2.3 | V |
| VCEsat | Total Drop | IL = 1A | (5) |  | 1.80 |  | 3.2 | V |
|  |  | IL = 2A | (5) |  |  |  | 4.9 | V |
| Vsens | Sensing Voltage (pins 1, 15) |  |  |  | ±1 (1) |  | 2 | V |

**4.3 4x4 Matrix Keypad**

This 16-button keypad provides a useful human interface component for microcontroller projects. Convenient adhesive backing provides a simple way to mount the keypad in a variety of applications. The Keypad 4x4 features a total of 16 buttons in Matrix form. This is a membrane keypad with no moving parts. It has a nice overlay depicting a telephone type keypad with additional four functional buttons. A female 8-pin berg connector is provided for interfacing it with your microcontroller circuits.



Fig()

**Working**

First test whether any key is pressed down. Connect power to rows, so they are High level. Then set all the rows Y1-Y4 as Low and then detect the status of the columns. Any column of Low indicates there is key pressing and that the key is among the 4 keys of the column. If all columns are high, it means no key is pressed down.  
Next, locate the key. Since the column in which the pressed key lies is identified, knowing the line would finalize the testing. Thus, set the rows as Low in turns until any is unveiled accordingly – other rows will still be High.   
Now the row can be identified. Detect the status of each column in turns. The column tested Low is the one intersecting with the line – their cross point is just the key pressed.

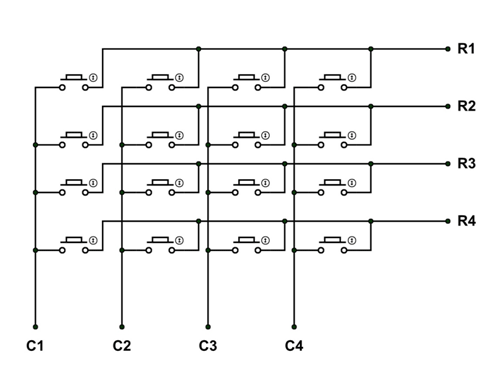
The description of the is as shown below in the table

|  |  |
| --- | --- |
| **Pin Number** | **Description** |
| ROWS | |
| 1 | PIN1 is taken out from 1st  ROW |
| 2 | PIN2 is taken out from 2nd  ROW |
| 3 | PIN3 is taken out from 3rd  ROW |
| 4 | PIN4 is taken out from  4th ROW |
| COLUMN | |
| 5 | PIN5 is taken out from 1st  COLUMN |
| 6 | PIN6 is taken out from 2nd  COLUMN |
| 7 | PIN7 is taken out from 3rd  COLUMN |
| 8 | PIN8 is taken out from 4th COLUMN |

Table()

As given in above table a 4X4 KEYPAD will have EIGHT TERMINALS. In them four are ROWS of MATRIX and four are COLUMNS **of** MATRIX. These 8 PINS are driven out from 16 buttons present in the MODULE. Those 16 alphanumeric digits on the MODULE surface are the 16 buttons arranged in MATRIX formation.

The internal structure of 4X4 KEYPAD MODULE is shown below.



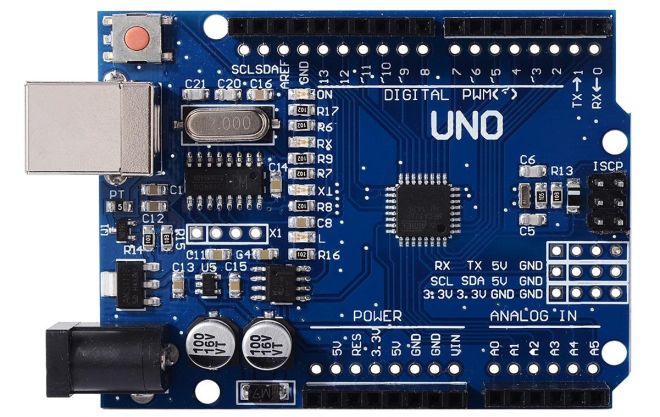
Fig()

4X4 KEYPAD MODULE Features and Specifications

Features and specifications

* Ultra-thin design
* Adhesive backing
* Excellent price/performance ratio
* Easy interface to any microcontroller
* Maximum Voltage across EACH SEGMENT or BUTTON: 24V
* Maximum Current through  EACH SEGMENT or BUTTON: 30mA
* Maximum operating temperature: 0°C to + 50°C
* Ultra-thin design
* Adhesive backing
* Easy interface
* Long life.

**4.5 ARDUINO UNO**

****

Fig()

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

**Revision 2** of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

**Revision 3** of the board has the following new features:

* 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin that is reserved for future purposes.
* Stronger RESET circuit.
* Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

|  |  |
| --- | --- |
| **Summary** Microcontroller | ATmega328 |
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |

|  |  |
| --- | --- |
| Input Voltage (limits) | 6-20V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 40 mA |
| DC Current for 3.3V Pin | 50 mA |
| Flash Memory | 32 KB (ATmega328) of which 0.5 KB used by boot loader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |
| Clock Speed | 16 MHz |

Table()

**Schematic & Reference Design:**

Note:The Arduino reference design can use an Atmega8, 168, or 328, Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board’s power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

* VIN.The input voltage to the Arduino board when it’s using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
* 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 – 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don’t advise it.
* 3V3**.** A 3.3volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
* GND.Ground pins.

**Memory:**

The Atmega328 has 32 KB (with 0.5 KB used for the boot loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output:

Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode (), digital Write (), and digital Read () functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

* Serial: 0 (RX) and 1 (TX).Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the Atmega8U2 USB-to-TTL Serial chip.
* External Interrupts: 2 and 3**:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt () function for details.

**Pin Description**

|  |  |  |
| --- | --- | --- |
| **Pin Category** | **Pin Name** | **Details** |
| Power | Vin, 3.3V, 5V, GND | Vin: Input voltage to Arduino when using an external power source.  5V: Regulated power supply used to power microcontroller and other components on the board.  3.3V: 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA.  GND: ground pins. |
| Reset | Reset | Resets the microcontroller. |
| Analog Pins | A0 – A5 | Used to provide analog input in the range of 0-5V |
| Input/Output Pins | Digital Pins 0 – 13 | Can be used as input or output pins. |
| Serial | 0(Rx), 1(Tx) | Used to receive and transmit TTL serial data. |
| External Interrupts | 2, 3 | To trigger an interrupt. |
| PWM | 3, 5, 6, 9, 11 | Provides 8-bit PWM output. |
| SPI | 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK) | Used for SPI communication. |
| Inbuilt LED | 13 | To turn on the inbuilt LED. |
| TWI | A4 (SDA), A5 (SCA) | Used for TWI communication. |
| AREF | AREF | To provide reference voltage for input voltage. |

Table()

**Arduino Uno Technical Specifications**

|  |  |
| --- | --- |
| Microcontroller | ATMEGA – 8 bit AVR family microcontroller |
| Operating Voltage | 5V |
| Recommended Input Voltage | 7-12V |
| Input Voltage Limits | 6-20V |
| Analog Input Pins | 6 (A0 – A5) |
| Digital I/O Pins | 14 (Out of which 6 provide PWM output) |
| DC Current on I/O Pins | 40 mA |
| DC Current on 3.3V Pin | 50 mA |
| Flash Memory | 32 KB (0.5 KB is used for Bootloader) |
| SRAM | 2 KB |
| EEPROM | 1 KB |
| Frequency (Clock Speed) | 16 MHz |

Table()

### Overview

**Arduino Uno** is a microcontroller board based on 8-bit ATmega328P microcontroller. Along with ATmega328P, it consists other components such as crystal oscillator, serial communication, voltage regulator, etc. to support the microcontroller. Arduino Uno has 14 digital input/output pins (out of which 6 can be used as PWM outputs), 6 analog input pins, a USB connection, A Power barrel jack, an ICSP header and a reset button.

### How to use Arduino Board

The 14 digital input/output pins can be used as input or output pins by using pinMode(), digitalRead() and digitalWrite() functions in arduino programming. Each pin operate at 5V and can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 KOhms which are disconnected by default.  Out of these 14 pins, some pins have specific functions as listed below:

* **Serial Pins 0 (Rx) and 1 (Tx):** Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.
* **External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
* **PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analogWrite() function.
* **SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.
* **In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.

Along with 14 Digital pins, there are 6 analog input pins, each of which provide 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts but this limit can be increased by using AREF pin with analog Reference() function.

* Analog pin 4 (SDA) and pin 5 (SCA) also used for TWI communication using Wire library.

Arduino Uno has a couple of other pins as explained below:

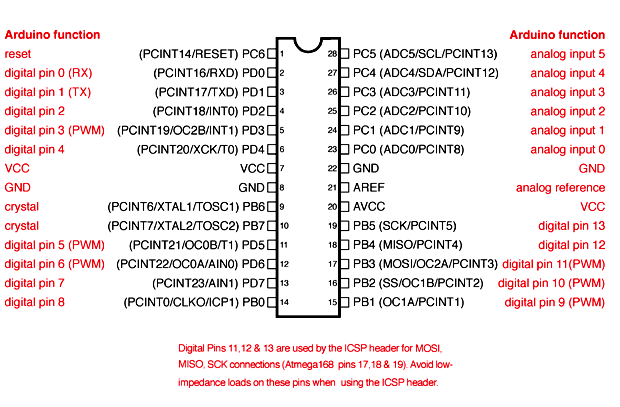
* **AREF:** Used to provide reference voltage for analog inputs with analogReference() function.
* **Reset Pin:**Making this pin LOW, resets the microcontroller.

### Communication

Arduino can be used to communicate with a computer, another Arduino board or other microcontrollers. The ATmega328P microcontroller provides UART TTL (5V) serial communication which can be done using digital pin 0 (Rx) and digital pin 1 (Tx). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The ATmega16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. There are two RX and TX LEDs on the arduino board which will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (not for serial communication on pins 0 and 1). A SoftwareSerial library allows for serial communication on any of the Uno's digital pins. The ATmega328P also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus.

### Arduino Uno to ATmega328 Pin Mapping

When ATmega328 chip is used in place of Arduino Uno, or vice versa, the image below shows the pin mapping between the two.



### Fig()

### Programming Arduino

Once arduino IDE is installed on the computer, connect the board with computer using USB cable. Now open the arduino IDE and choose the correct board by selecting Tools>Boards>Arduino/Genuino Uno, and choose the correct Port by selecting Tools>Port. Arduino Uno is programmed using Arduino programming language based on Wiring. To get it started with Arduino Uno board and blink the built-in LED, load the example code by selecting Files>Examples>Basics>Blink. Once the example code (also shown below) is loaded into your IDE, click on the ‘upload’ button given on the top bar. Once the upload is finished, you should see the Arduino’s built-in LED blinking.  Below is the example code for blinking:

### ****Applications****

* Prototyping of Electronics Products and Systems
* Multiple DIY Projects.
* Easy to use for beginner level DIYers and makers.
* Projects requiring Multiple I/O interfaces and communications.

**BIBILIOGRAPHY**

* <https://components101.com/microcontrollers/arduino-uno>
* <https://www.arduino.cc/>
* <https://www.tinkercad.com/>
* <https://www.youtube.com/watch?v=ZySGP4AwGCY&feature=youtu.be>
* <https://youtu.be/q9YC_GVHy5A>
* <https://youtu.be/GyprsoB887Y>
* <https://youtu.be/kVw6kPSJJfw>
* <https://youtu.be/RGkIHFQf8-w>
* <https://youtu.be/I1ocRfZ_Zis>