**MICROCONTROLLER BASED ALPHANUMERIC TO MORSE CODE ENCODER**

by

CHENNA HEMANTH 19BEC1108

KAILASH NATARAJAN 19BEC1120

VISHWA RAJA 19BEC1355

B. ASWIN 19BEC1433

A project report submitted to

**Dr. M. Jagannath**

**SCHOOL OF ELECTRONICS ENGINEERING**

in partial fulfilment of the requirements for the course of

**ECE3003 – Microcontroller and its Applications**

in

**B.Tech. Electronics and Communication Engineering**



**VIT CHENNAI**

**Vandalur-Kelambakkam Road**

**Chennai – 600127**

**JUNE 2021**

**BONAFIDE CERTIFICATE**

Certified that this project report entitled “**MICROCONTROLLER BASED ALPHANUMERIC TO MORSE CODE ENCODER”** is a bonafide work of **CHENNA HEMANTH (19BEC1108), KAILASH NATARAJAN (19BEC1120), VISHWA RAJA (19BEC1355), B. ASWIN (19BEC1433)** who carried out the Project work under my supervision and guidance.

**Dr. M. Jagannath**

Associate Professor Senior

School of Electronics Engineering (SENSE),

VIT Chennai

Chennai-600127.

**ABSTRACT**

Morse code[1] is a method used in telecommunication to encode text characters as standardized sequences of two different signal durations, called dots and dashes. Morse code is named after Samuel Morse, one of the inventors of the telegraph. Our project mainly aims to ease the process of morse code generation for numbers and English alphabets.

This project uses, keypad for input and LED for Morse code output. In this device when a key is pressed in the keypad, the keypad grounds all the rows and checks for zero in the column by using the code and the send it to the accumulator and reads the three-position rotary switch to understand the exact key pressed. Then the corresponding Morse code is displayed.

Morse code can be sent in a variety of ways with improvised devices that can be switched easily on and off, such as flashlights. These can be performed using this device easily. This device can help everyone translate alphabets and numbers to morse codes.

Today, Morse code remains popular with amateur radio operators around the world. It is also commonly used for emergency signals.

In software simulation, we have connected keypad, output LEDs and other required components. After uploading the code, when the user gave input through keypad. The output LEDs were able to show the respective morse code for the inputs.

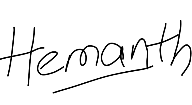
**ACKNOWLEDGEMENT**

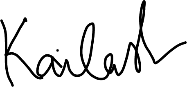
We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. M. Jagannath,** Associate Professor Senior, School of Electronics Engineering, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

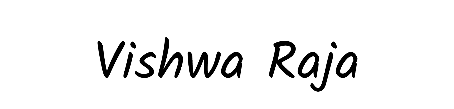
We are extremely grateful to **Dr. A. Sivasubramanian,** Dean of the School of Electronics Engineering, VIT Chennai, for extending the facilities of the School towards our project and for her unstinting support.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

**CHENNA HEMANT 19BEC1108 **

**KAILASH NATARAJAN** **19BEC1120** 

**VISHWA RAJA**  **19BEC1355 **

**B. ASWIN 19BEC1433** 

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** |  | **TOPIC** | **PAGE NO.** |
| **1** |  | **INTRODUCTION** | **6** |
|  | **1.1** | **OBJECTIVE** | **6** |
|  | **1.2** | **BENEFITS** | **6** |
|  | **1.3** | **FEATURES** | **6** |
| **2** |  | **MICROCONTROLLER BASED ALPHANUMERIC TO MORSE CODE ENCODER** | **7** |
|  | **2.1** | **WORKING PRINCIPLE** | **7** |
|  | **2.2** | **BLOCK DIAGRAM** | **8** |
|  | **2.3** | **REQUIRED EQUIPMENTS** | **8** |
|  | **2.4** | **HARDWARE ANALYSIS** | **9** |
|  | **2.5** | **SOFTWARE ANALYSIS** | **16** |
|  | **2.6** | **CIRCUIT AND BLOCK DIAGRAM** | **19** |
|  | **2.7** | **ALGORITHM** | **20** |
|  | **2.8** | **PROGRAM CODE** | **20** |
|  | **2.9** | **WORKING** | **23** |
| **3** |  | **LIMITATIONS** | **26** |
| **4** |  | **APPLICATIONS AND ADVANTAGES** | **26** |
| **5** |  | **CONCLUSION** | **26** |
| **6** |  | **RELATED WORKS** | **27** |
| **7** |  | **REFERENCES** | **27** |

**1.INTRODUCTION**

Morse code method used in telecommunication to encode text characters as standardized sequences of two different signal durations, called dots and dashes. In this microcontroller based alphanumeric to morse code encoder, when a key in the keypad is pressed, we get the morse code of the respective input through output LEDs. The 4x4 keypad used for this project can support only 16 unique inputs. This project needs input of all alphabets and decimal digits totalling 36 unique inputs. To take this input from one keypad, a three-positional rotary switch is used to give different mappings to the keypad buttons.

**1.1** **OBJECTIVES**

* Design a morse code generator.
* Detect the input given (numbers and alphabets) in the keypad.
* Design a circuit so as to give an equivalent morse code for the respective inputs given using keypad, using 8051 microcontroller.

**1.2 BENEFITS**

* This device can help everyone translate alphabets and numbers to morse codes.
* This device is fast and efficient in operation.
* Secret communications can be done easily through this device.
* It can be sent in a variety of ways with improvised devices that can be switched easily on and off, such as flashlights. These can be performed using this device easily.
* This device is designed in a simplified fashion and is easy to use.

**1.3 FEATURES**

* Keypads are used so as to send information with ease.
* LEDs are used in the output. Thus, device is cost efficient.
* Efficient coding is done in software tool called as Keil μvision.
* The chosen AT89S52 is a Low-power, high-performance CMOS 8-bit microcontroller with 8KB of ISP flash memory. The device uses Microchip high-density, nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout[2].

**2. MICROCONTROLLER BASED ALPHANUMERIC TO MORSE CODE ENCODER.**

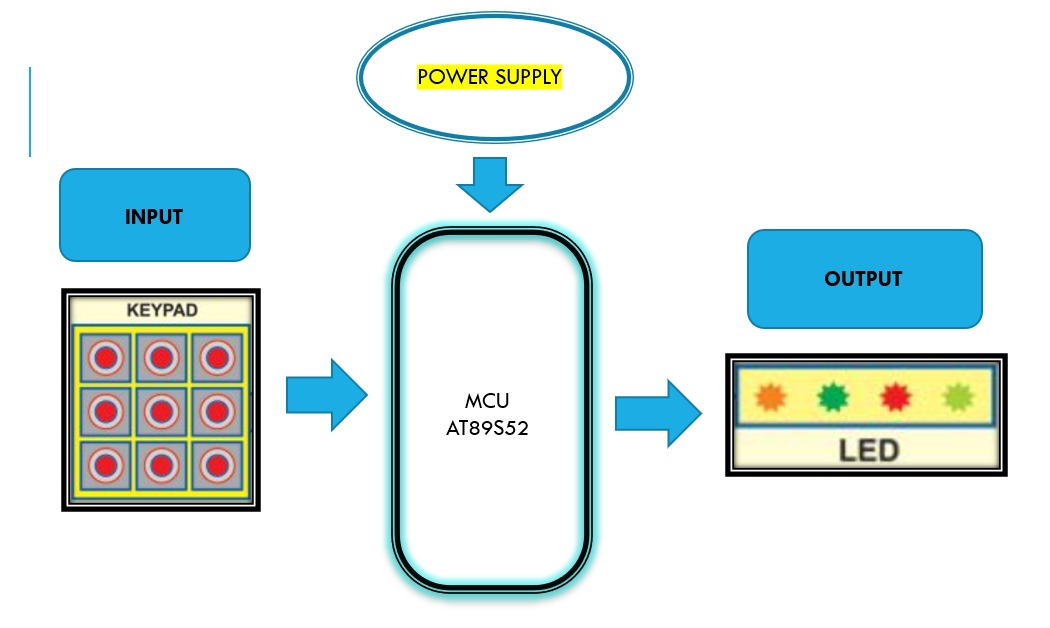
**2.1 WORKING PRINCIPLE**

This project is alphanumeric to morse code encoder where keypad is used as the input device, 8051 microcontroller equivalent AT89S52 as the processing unit and three LEDs to indicate the generated morse code output. One LED is used to output Morse code where dot is determined by glowing for one time unit and dash is determined by glowing for three time units[1]. The other two LEDs correspond to dots and dashes respectively. These two LEDs glow to signify if the first LED is glowing for dot or dash. The 4x4 keypad used supports inputs for only 16 unique characters. To be able to get input for all alphabets and numbers, a three-position rotary switch had to be used. The status of this switch determines the mapping of the keypad buttons as shown in section 2.9. When any button on the keypad is pressed, the status of the switch is checked and the microcontroller understands the character pressed. Then the corresponding Morse code is output using the first LED and the other two LEDs glow to signify dots and dashes respectively.

**2.2 BLOCK DIAGRAM**

The four main features of the basic block diagram (given below) are

* The MCU used – AT89S52 in the processing unit.
* Keypad for the alphanumeric input phase.
* Three LEDs in the output phase for Morse, dot and dash respectively.
* Power supply.



**FIGURE 1:** Block Diagram

Figure 1 shows the block diagram for Microcontroller based alphanumeric to morse code encoder.

**2.3 REQUIRED EQUIPMENTS**

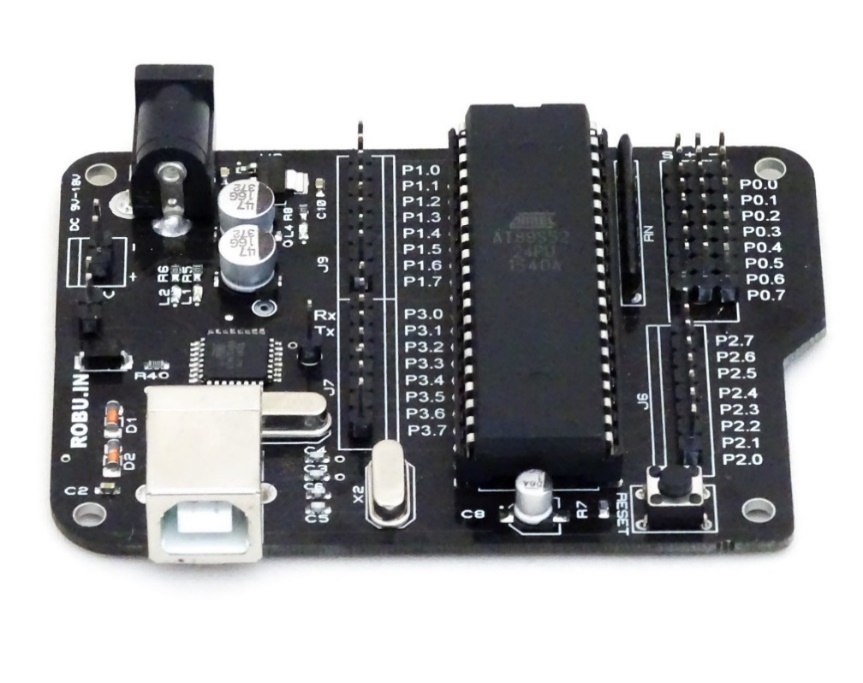
* Keypad (4X4 matrix)
* MCU (AT89S52)
* LEDs (3 pieces).
* 8051 Development board
* Male/Female jumper Wires for connection.
* Breadboard
* Power Supply
* Three-position rotary switch

**2.4** **HARDWARE ANALYSIS**

The hardware components consist of AT89S52, power supply, 4x4 keypad, three-position rotary switch and three LEDs. The keypad is used to give the input as an alphabet or as a decimal number. The AT89S52 Microcontroller unit acts as a processing unit and produces the morse code using the output unit that is three LEDs, one for Morse, one for dots and the other for dashes.

**2.4.1 SMARTELEX ARYABHATTA 8051 MICROCONTROLLER DEVELOPMENT BOARD AT89S52**

This 8051 Development Board is aimed at providing the microcontroller newbies with a quick way into the development of embedded projects. The Aryabhatta 8051 Development Board AT89S52 with Onboard USB Programmer[3] is very easy to use but also at the same time offering high functionality and yet it is available at a price everyone can afford!!!



**FIGURE 2:** SmartElex’s Aryabhatta 8051 Development Board AT89S52[3]

With SmartElex’s Aryabhatta 8051 Development Board AT89S52 – on-board USB Programmer you can develop and prototype 40 pins 89S52 and 89s51 microcontrollers. The onboard programmer allows easy connection with PC using USB type B cable for Programming. The Operating Voltage is 9V to 15V DC.

**AT89S52**

The development board runs this AT89S52 Microcontroller unit(Fig. 3).

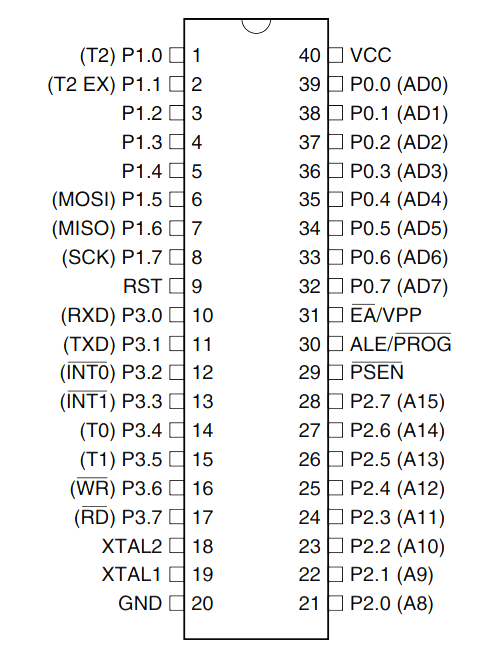


**FIGURE 3**: AT89S52 Microcontroller Unit[4]

The AT89S52[4] comes from the popular 8051 family of Atmel Microcontrollers. It is an 8-bit CMOS microcontroller with 8K as Flash memory and 256 bytes of RAM. Since it is similar to the trust worthy 8051 architecture these microcontrollers are as per industry standard. It has 32 I/O pins comprising of three 16-bit timers, external interrupts, full-duplex serial port, on-chip oscillator and clock circuitry.

The Microcontroller also has Operating mode, Idle Mode and Power down mode which makes it suitable for battery operated applications. Few considerable drawbacks of the microcontroller are that it does not have in-built ADC and does not support SPI or I2C protocols. However, you can utilise external modules for the same.

**PIN DESCRIPTION**



**FIGURE 4**: Pin diagram of AT89S52[4]

**Pins 1 to 8** − These pins are known as Port 1. This port doesn’t serve any other functions. It is internally pulled up, bi-directional I/O port.

**Pin 9** − It is a RESET pin, which is used to reset the microcontroller to its initial values.

**Pins 10 to 17** − These pins are known as Port 3. This port serves some functions like interrupts, timer input, control signals, serial communication signals RxD and TxD, etc.

**Pins 18 & 19** − These pins are used for interfacing an external crystal to get the system clock.

**Pin 20** − This pin provides the power supply to the circuit.

**Pins 21 to 28** − These pins are known as Port 2. It serves as I/O port. Higher order address bus signals are also multiplexed using this port.

**Pin 29** − This is PSEN pin which stands for Program Store Enable. It is used to read a signal from the external program memory.

**Pin 30** − This is EA pin which stands for External Access input. It is used to enable/disable the external memory interfacing.

**Pin 31** − This is ALE pin which stands for Address Latch Enable. It is used to demultiplex the address-data signal of port.

**Pins 32 to 39** − These pins are known as Port 0. It serves as I/O port. Lower order address and data bus signals are multiplexed using this port.

**Pin 40** − This pin is used to provide power supply to the circuit.

8051 microcontrollers have 4 I/O ports each of 8-bit, which can be configured as input or output. Hence, total 32 input/output pins allow the microcontroller to be connected with the peripheral devices.

**Port 0**

The P0 (zero) port is characterized by two functions −

* When the external memory is used then the lower address byte (addresses A0A7) is applied on it, else all bits of this port are configured as input/output.
* When P0 port is configured as an output then other ports consisting of pins with built-in pull-up resistor connected by its end to 5V power supply, the pins of this port have this resistor left out.

**Port 1**

P1 is a true I/O port as it doesn’t have any alternative functions as in P0, but this port can be configured as general I/O only. It has a built-in pull-up resistor and is completely compatible with TTL circuits.

**Port 2**

P2 is similar to P0 when the external memory is used. Pins of this port occupy addresses intended for the external memory chip. This port can be used for higher address byte with addresses A8-A15. When no memory is added then this port can be used as a general input/output port similar to Port 1.

**Port 3**

In this port, functions are similar to other ports except that the logic 1 must be applied to appropriate bit of the P3 register.

**2.4.2. KEYPAD (4x4 MATRIX)**



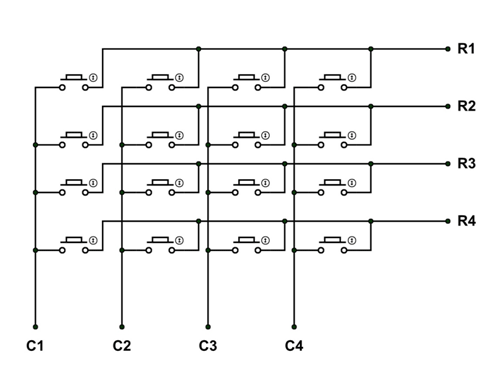
**FIGURE 5:** 4x4 Keypad module[5]

4X4 Keypad Modules(Fig. 5) are available in different sizes and shapes. But they all have same pin configuration. It is easy to make 4X4 KEYPAD by arranging 16 buttons in matrix formation by yourself.

|  |  |
| --- | --- |
| **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 |

As given in above table a **4X4 KEYPAD[5]** 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 in Fig. 6.



**FIGURE 6:** 4x4 Keypad Internal Structure[5]

**4X4 KEYPAD MODULE Features and Specifications**

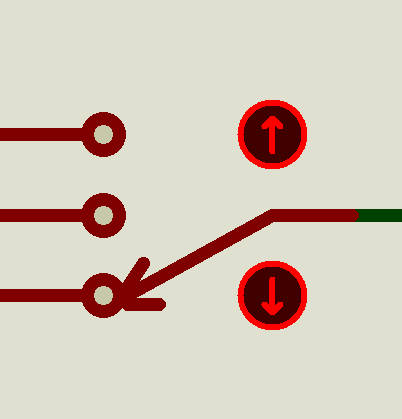
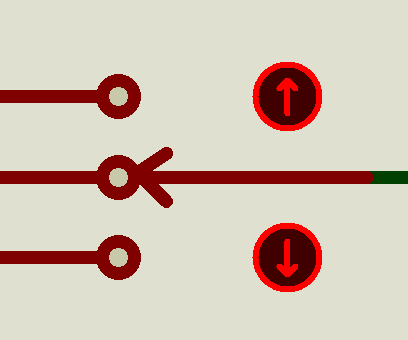
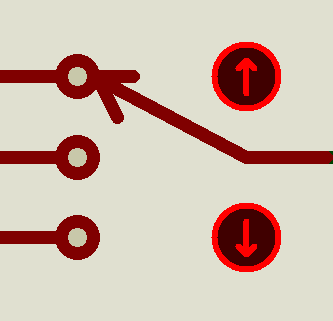
* Maximum Voltage across EACH SEGMENT or BUTTON: 24V
* Maximum Current through EACH SEGMENT or BUTTON: 30mA
* Maximum operating temperature: 0°C to + 50°C
* Ultra-thin design
* Adhesive backing
* Easy interface
* Long life.

**2.4.3. THREE-POSITION ROTARY SWITCH**



**FIGURE 7:** Three-Position Rotary switch [6]

A three-position rotary switch(Fig. 7) is a type of switch that can take three positions.

1. (**b**) (**c)**

**FIGURE 8:** Three-position rotary switch in Proteus

The switch can be used to open or close two circuits at a time depending on the design of the circuit.

**2.4.4. POWEROWER SUPPLY**

The power supply will be 9V batteries and the voltage eliminator to supply power to different devices. The power supply will connect to all the other components through the breadboard and wires.

**2.5 SOFTWARE ANALYSIS**

Simulation software is based on the process of modelling a real phenomenon with a set of mathematical formulas. It is, essentially, a program that allows the user to observe an operation through simulation without actually performing that operation.

**2.5.1 CIRCUIT SOFTWARE REQUIREMENTS**

* **Proteus Design Suite.**
* **Keil μVision 5.**
* **ProgISP software.**

**2.5.2. PROTEUS DESIGN SUITE.**

The **Proteus Design Suite**[7] is a proprietary software tool suite used primarily for [electronic design automation](https://en.wikipedia.org/wiki/Electronic_design_automation). The software is used mainly by electronic [design engineers](https://en.wikipedia.org/wiki/Design_engineer) and technicians to create [schematics](https://en.wikipedia.org/wiki/Schematic) and electronic prints for manufacturing [printed circuit boards](https://en.wikipedia.org/wiki/Printed_circuit_board).

It was developed in [Yorkshire](https://en.wikipedia.org/wiki/Yorkshire), England by Labcenter Electronics Ltd and is available in English, French, Spanish and Chinese languages.

History

The first version of what is now the Proteus Design Suite was called PC-B and was written by the company chairman, John Jameson, for DOS in 1988. Schematic Capture support followed in 1990, with a port to the Windows environment shortly thereafter. Mixed mode SPICE Simulation was first integrated into Proteus in 1996 and microcontroller simulation then arrived in Proteus in 1998. Shape based autorouting was added in 2002 and 2006 saw another major product update with 3D Board Visualisation. More recently, a dedicated IDE for simulation was added in 2011 and MCAD import/export was included in 2015. Support for high speed design was added in 2017.  Feature led product releases are typically biannual, while maintenance based service packs are released as it is required.

Product Modules

The Proteus Design Suite is a Windows application for schematic capture, simulation, and PCB (Printed Circuit Board) layout design. It can be purchased in many configurations, depending on the size of designs being produced and the requirements for microcontroller simulation. All PCB Design products include an autorouter and basic mixed mode SPICE simulation capabilities.

**Schematic Capture**

Schematic capture in the Proteus Design Suite is used for both the simulation of designs and as the design phase of a PCB layout project. It is therefore a core component and is included with all product configurations.

**Microcontroller Simulation**

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a training or teaching tool. Support is available for co-simulation of:

* [Microchip Technologies](https://en.wikipedia.org/wiki/Microchip_Technology) PIC10, PIC12, PIC16,PIC18,PIC24,dsPIC33 Microcontrollers.
* [Atmel](https://en.wikipedia.org/wiki/Atmel) AVR (and [Arduino](https://en.wikipedia.org/wiki/Arduino)), 8051 and [ARM Cortex-M3](https://en.wikipedia.org/wiki/ARM_Cortex-M#Cortex-M3) Microcontrollers
* [NXP](https://en.wikipedia.org/wiki/NXP_Semiconductors) 8051, ARM7, [ARM Cortex-M0](https://en.wikipedia.org/wiki/ARM_Cortex-M#Cortex-M0) and ARM Cortex-M3 Microcontrollers.
* [Texas Instruments](https://en.wikipedia.org/wiki/Texas_Instruments) MSP430, PICCOLO DSP and ARM Cortex-M3 Microcontrollers.
* Parallax Basic Stamp, Freescale HC11, 8086 Microcontrollers.

**PCB Design**

The PCB Layout module is automatically given connectivity information in the form of a [netlist](https://en.wikipedia.org/wiki/Netlist) from the schematic capture module. It applies this information, together with the user specified [design rules](https://en.wikipedia.org/wiki/Design_rule_checking) and various design automation tools, to assist with error free board design. PCB's of up to 16 copper layers can be produced with design size limited by product configuration.

**3D Verification**

The 3D Viewer module allows the board under development to be viewed in 3D together with a semi-transparent height plane that represents the boards enclosure. [STEP](https://en.wikipedia.org/wiki/ISO_10303-21) output can then be used to transfer to mechanical CAD software such as [Solidworks](https://en.wikipedia.org/wiki/Solidworks) or [Autodesk](https://en.wikipedia.org/wiki/Autodesk) for accurate mounting and positioning of the board.

**2.5.3. KEIL μVISION 5**

The Keil µVision IDE[8] combines project management, run-time environment, build facilities, source code editing, and program debugging in a single powerful environment. µVision is easy-to-use and accelerates your embedded software development.

The Keil 8051 Development Tools are designed to solve the complex problems facing embedded software developers. When starting a new project, simply select the microcontroller you use from the Device Database and the µVision IDE sets all compiler, assembler, linker, and memory options for you.

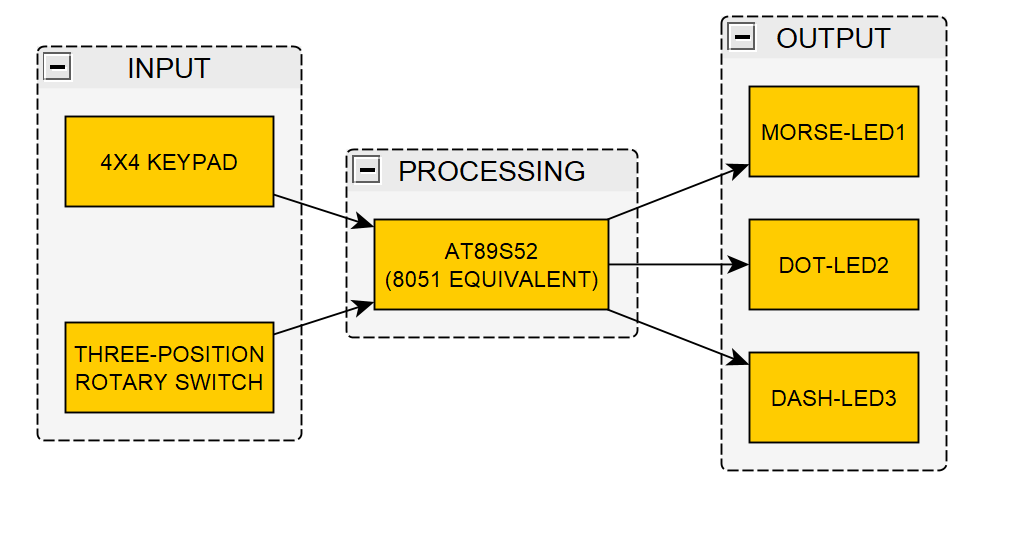
**Key Features**

* The Project window shows application source files and selected software components. Below the components you will find corresponding library and configuration files.
* Projects support multiple targets. They ease configuration management and may be used to generate debug and release builds or adoptions for different hardware platforms.
* The Manage Run-Time Environment window shows all software components that are compatible with the selected device. Inter-dependencies of software components are clearly identified with validation messages.
* The Configuration Wizard is an integrated editor utility for generating GUI-like configuration controls in assembler, C/C++, or initialization files.
* The Functions window gives fast access to the functions in each C/C++ source code module.
* The Code Completion list and Function Parameter information helps you to keep track of symbols, functions, and parameters.
* Dynamic Syntax Checking validates the program syntax while you are typing and provides real-time alerts to potential code violations before compilation.

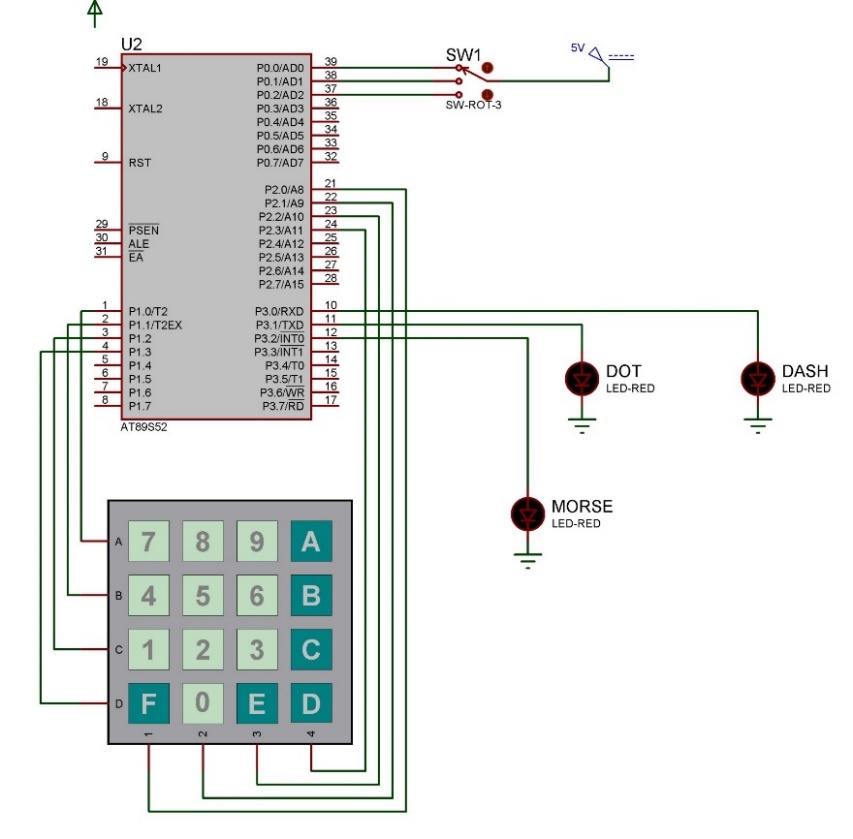
**2.5.4. PROGISP**

This is a programmer software that is used to load the .hex file of our code into the SmartElex Aryabhatta 8051 Microcontroller development board.

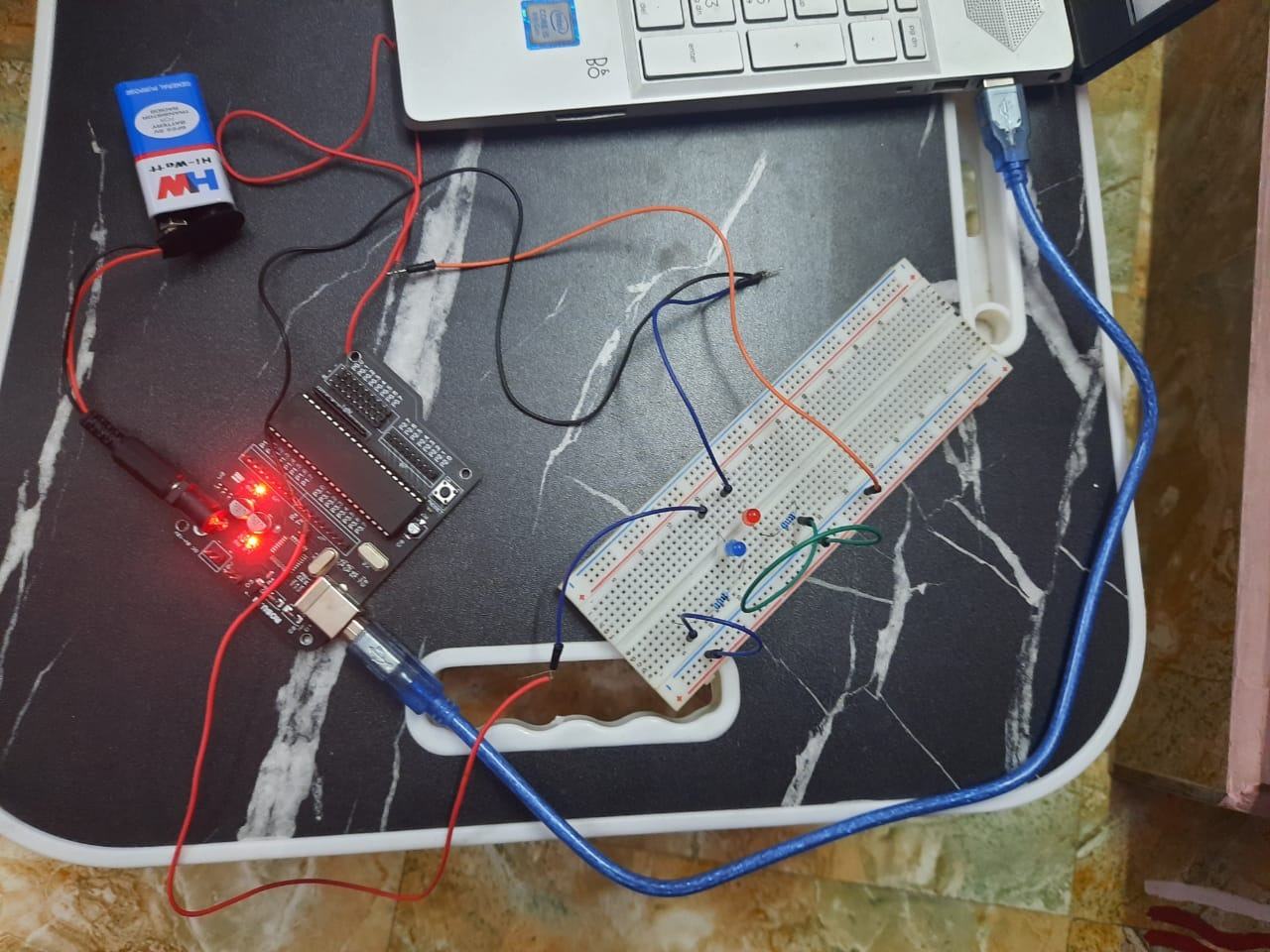
**2.6 CIRCUIT AND BLOCK DIAGRAM**



**FIGURE 9:** Circuit Block Diagram



**FIGURE 10:** Circuit Diagram in Proteus

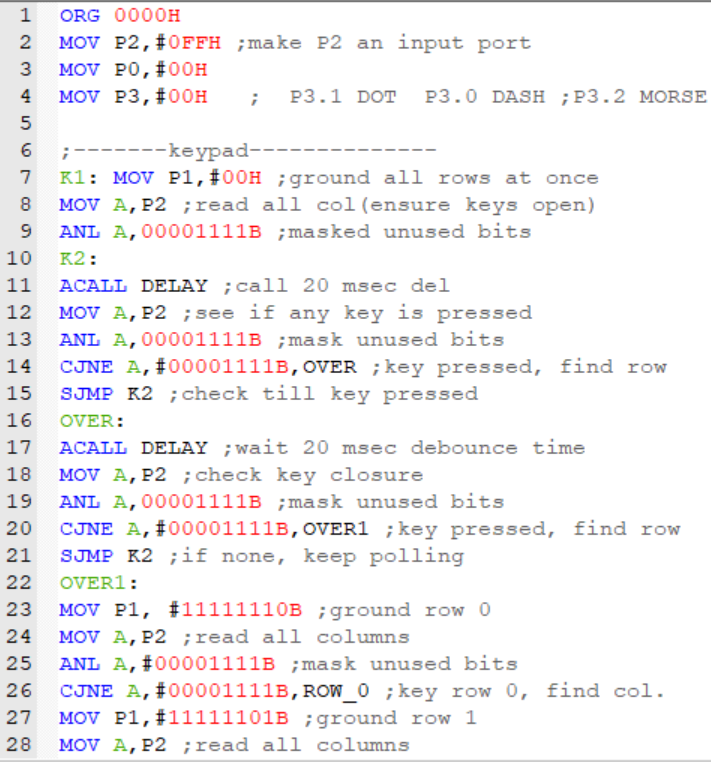


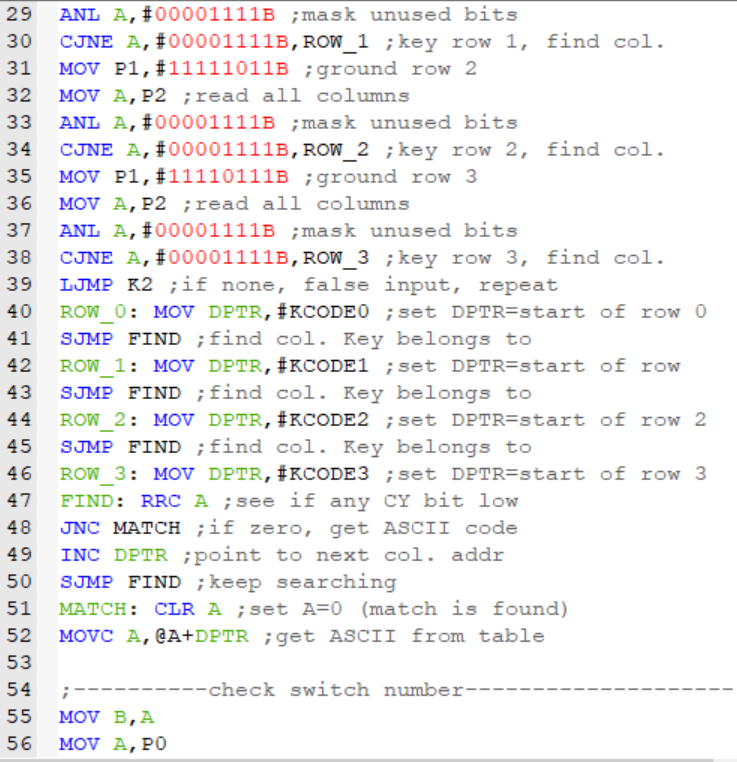
**FIGURE 11:** Hardware Circuit

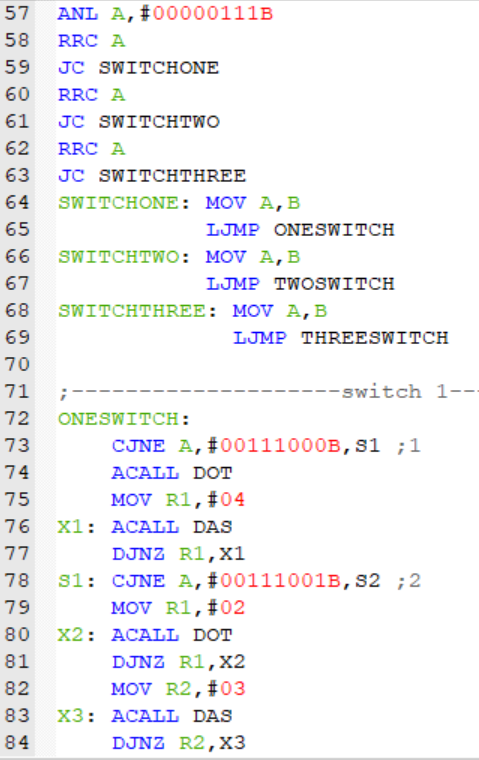
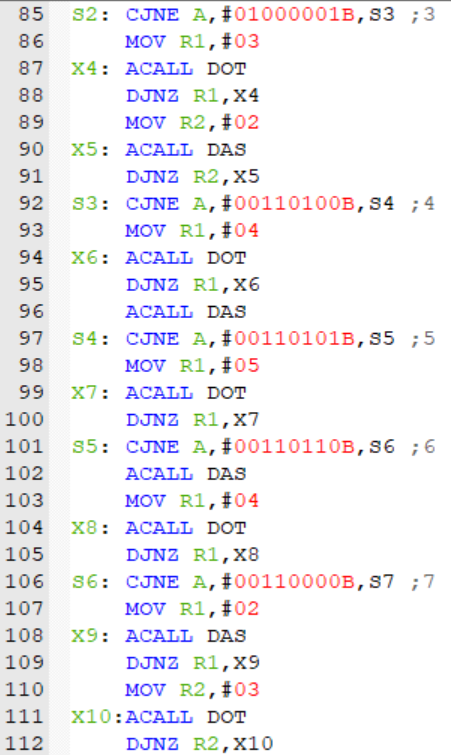
**2.7 ALGORITHM**

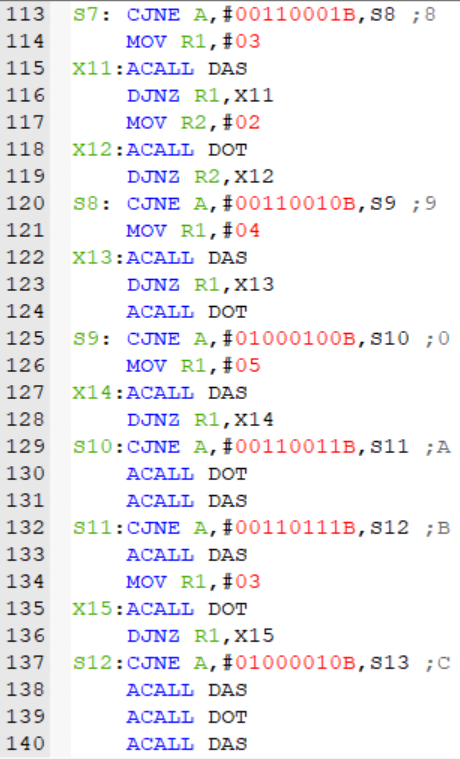
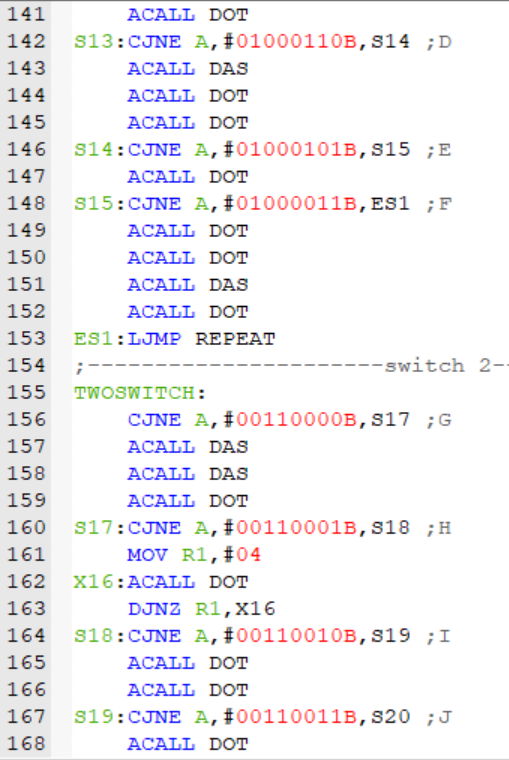
1. Initialize the input and output ports.
2. Read the input and find the pressed key.
3. Depending on the three-position rotary switch position compare the output value of pressed key.
4. If same display the respective morse code in led with delay.
5. Delay of dash is three times the delay of dot.

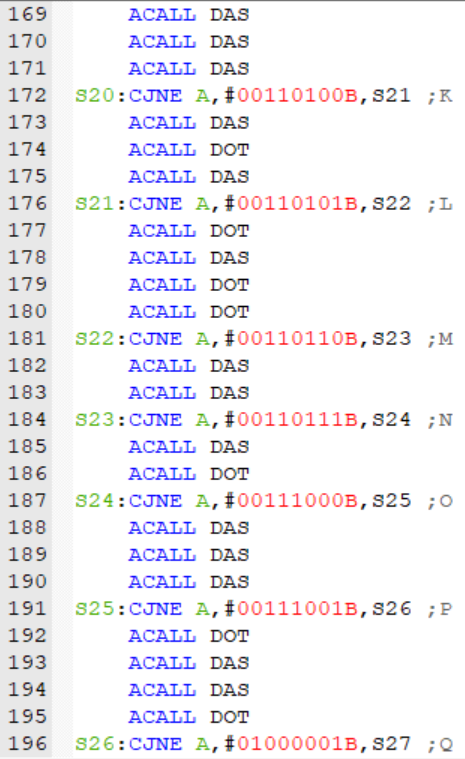
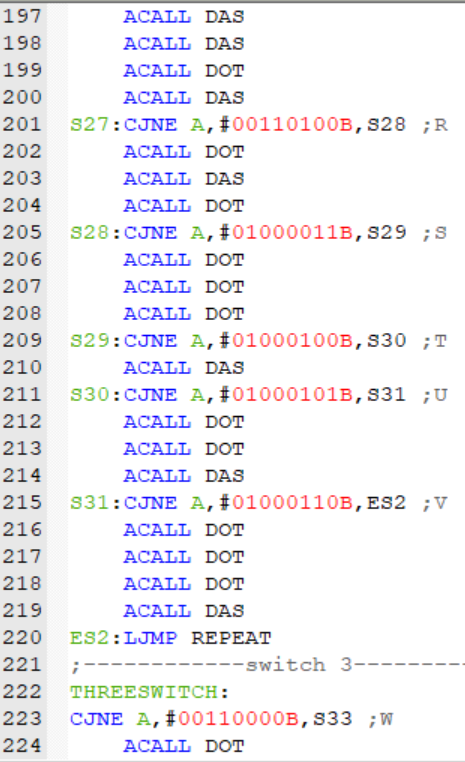
**2.8. PROGRAM CODE**

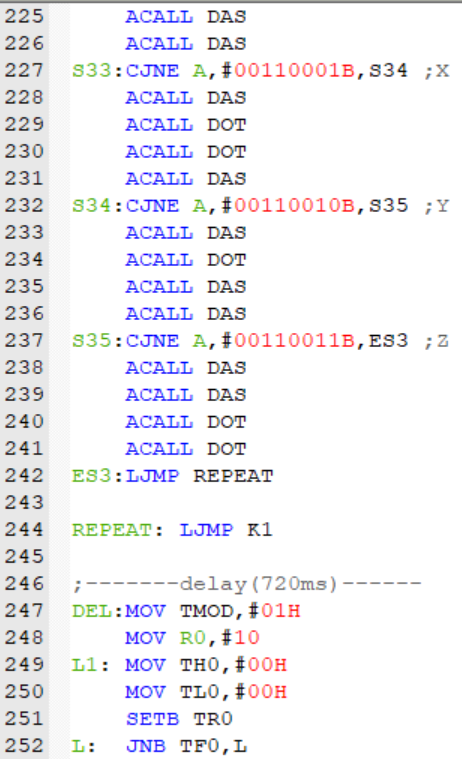
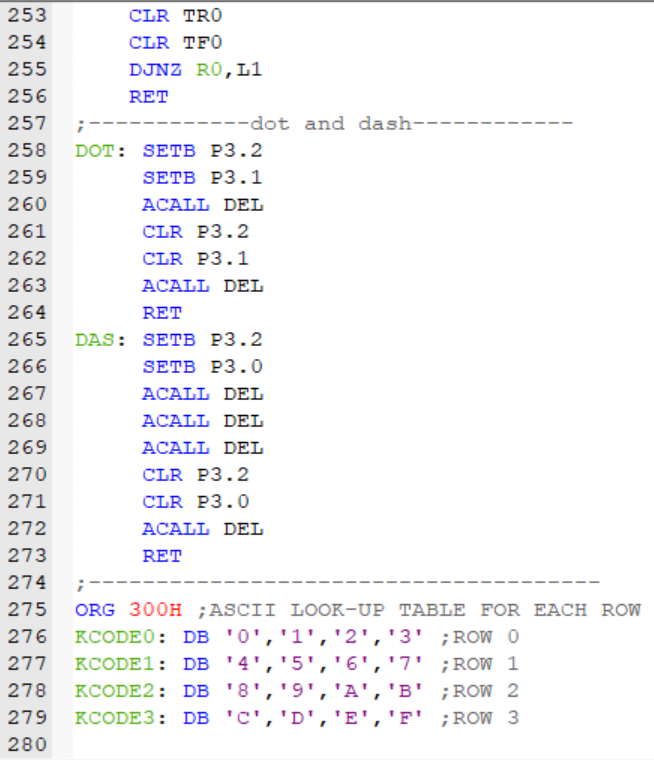


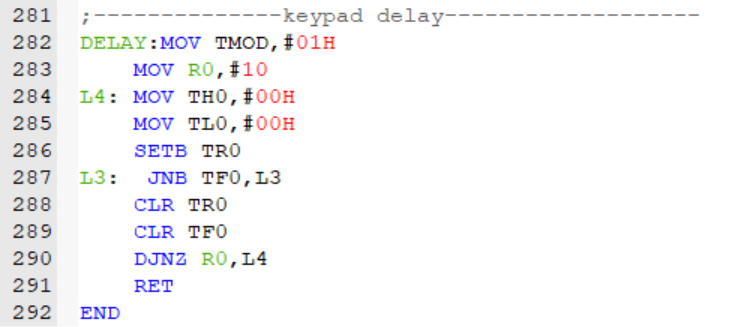


**** ****

** **

** **

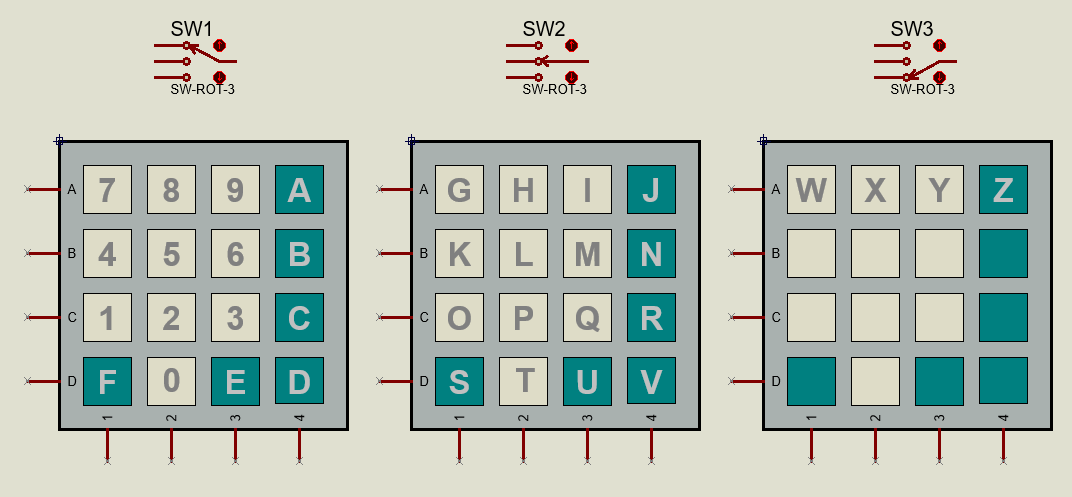
** **

****

**2.9 WORKING**

Due to the COVID-19 pandemic, we can’t get some required parts for keypad interfacing like jumper wires. So, in hardware, we directly include the input into the code and get the output through the output LEDs (Dot-LED and Dash-LED).

1. Write the program in Keil software and create .hex file.
2. Open a project in Proteus software and make the circuit (or) make the circuit using hardware components
3. Double click on the AT89S52 in proteus and select the created .hex file for program file (or) load the firmware onto the development board
4. Run the simulation (proteus only).
5. Select the position of the three-way switch for different mappings of the keypad buttons and then press a button for output.

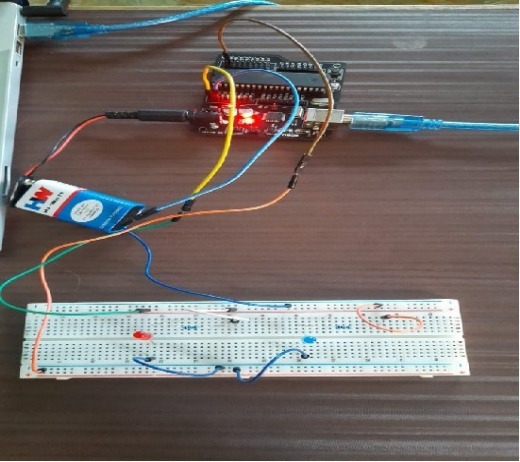


**FIGURE 12:** Various Keypad Mapping with respective switch positions

The figure represents the keypad button mappings for various positions of the three-way switch.

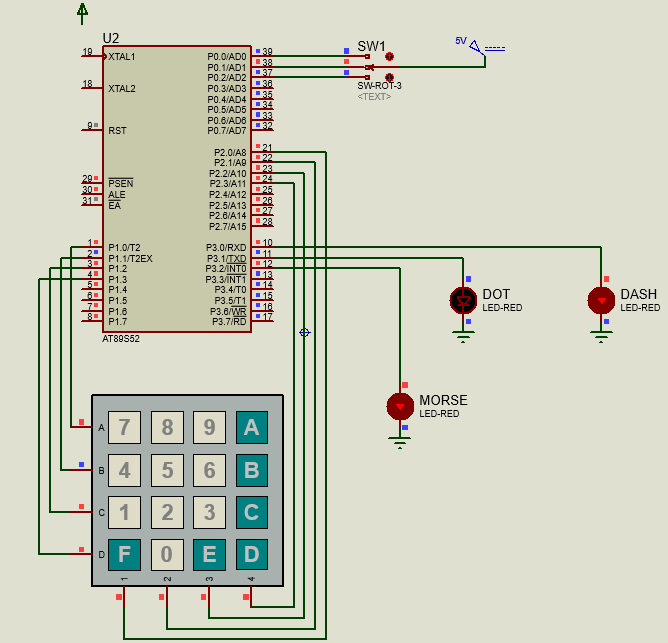
**2.9.1. SIMULATION**

For hardware working, input is given in code itself without using keypad. Due to COVID-19 leading to lack of accessibility of components, 2 LEDs were used for output where red LED represents dot and blue represents dash respectively.

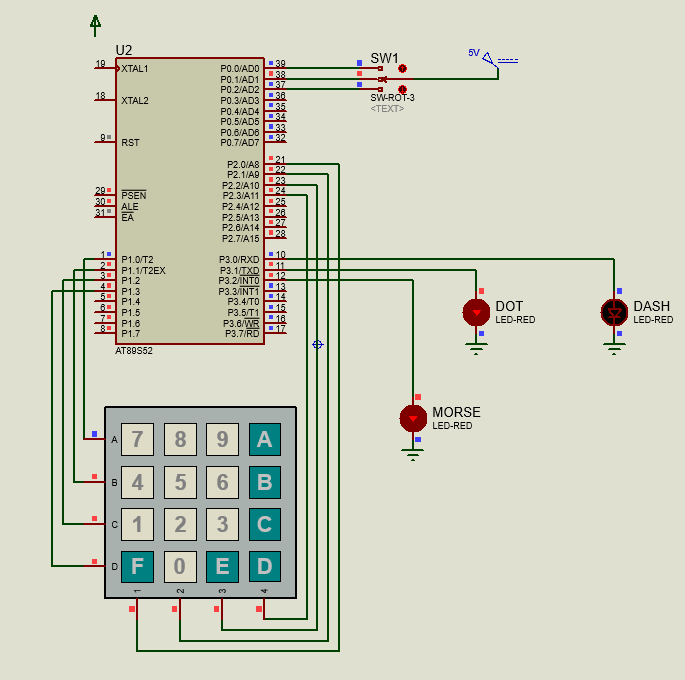


**FIGURE 13:** Hardware Working

For software simulation, Proteus Design Suite[7] was used. The corresponding components were connected. When any key is pressed, the LEDs start glowing in the respective order.



**FIGURE 14:** Proteus Simulation representing Dash



**FIGURE 15:** Proteus Simulation representing Dot.

**3. LIMITATIONS**

The LEDs may not be powerful enough to provide good visibility in long ranges. This can be overcome by using 20Watt LEDs, but the power consumption would increase.

**4. APPLICATION AND ADVANTAGES**:

1. This project can be used for educational purposes to learn Morse code.
2. The Morse code is used in aviation, it is used to continue communication between flights and air traffic control. Present days. Morse code is used to give the navigation to help the pilots and check the stations. Where all the names of the station can be transferred in a short set of identification letters. Usually, a 2 to 5 letter version of the name of the station in Morse code.
3. This is a very efficient way of digital communication system.
4. A unique implementation of this using powerful LEDs can be used by people to send SOS signals[9].
5. The Morse code is used in communication systems like radios. But, mainly used for radio operators to interact with each other. Most of the radio amateur uses this code to transmit radio.
6. The Morse code is used in Navy for security purpose. Navy people will signal the other ship people by switching ON/ OFF in the form of Morse code. This code is still used in many applications to send the signals such as aviation, military and government applications.

**5. CONCLUSION**

In conclusion, we have successfully implemented the alphanumeric to morse code encoder by using keypad (4x4 matrix) as input interface, AT89S52 as the Microcontroller unit for generating the morse code after comparing the keypad button and three-position rotary switch status and then indicates via the three LEDs connected to P3.0, P3.1, P3.2 where Morse code is displayed by LED 1, dots and dashes are indicated by LED 2 and LED 3. Therefore, when key is pressed the keypad grounds all the rows and checks for zero in the column by using the code and the send it to the accumulator and reads the three-position rotary switch to understand the exact key pressed. Then the corresponding Morse code is displayed. Future works include using 20Watt LEDs to have more practical applications. Others forms of output such as Radio, Buzzers and LCD can also be included to increase applications. Additional speed-dial type buttons can also be included that can output prefixed codes like SOS.

**6. RELATED WORKS:**

1. Microcontroller based Morse-code encoder using keypad with LCD and buzzer output.
2. Morse code Generator Using Microcontroller with Alphanumeric Keypad input and LCD output.

**7. REFERENCES:**

[1]https://en.wikipedia.org/wiki/Morse\_code

[2]https://www.microchip.com/wwwproducts/en/AT89S52.

[3]https://robu.in/product/smartelex-aryabhatta-8051-microcontroller-development-board-at89s52-with-onboard-usb-programmer/

[4]https://components101.com/microcontrollers/at89s52-microcontroller-pinout-features-datasheet

[5]https://components101.com/misc/4x4-keypad-module-pinout-configuration-features-datasheet

[6]https://tubedepot.com/products/three-position-rotary-switch

[7]https://en.wikipedia.org/wiki/Proteus\_Design\_Suite

[8]https://www2.keil.com/mdk5/uvision/#:~:text=%C2%B5Vision%C2%AE%20IDE,accelerates%20your%20embedded%20software%20development.

[9]https://www.efxkits.co.uk/morse-code-generator-circuit-with-application