

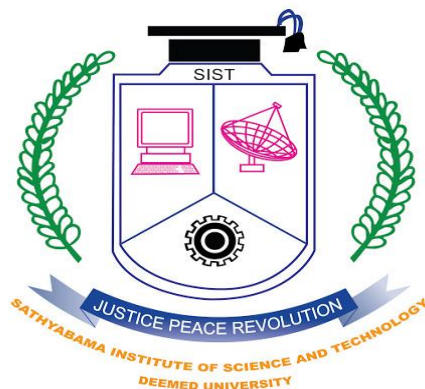
Self-Learning Braille Application for Visually Impaired People

Submitted in partial fulfilment of the requirements for the award of Bachelor of
Engineering Degree in Electronics and Communication Engineering

by

L.VAMSI (37130428)

Y.V.YASWANTH KUMAR (37130450)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL – 2021



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with “A” grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Y.V .YASWANTH KUMAR (37130450)
And L.VAMSI (37130428) who carried out the project entitled “**Self-Learning Braille Application for Visually Impaired People**” under my supervision from November 2020 to March 2021.

Project Guide
Dr.T.Vino M.E.Ph.D

Head of the Department
Dr.T.RAVI,M.E.Ph.D

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

We, L VAMSI (Reg.No.37130428) and Y. V. YASWANTH KUMAR (Reg.No.37130450) hereby declare that the Project Report entitled “**Self-Learning Braille Application for Visually Impaired People**” done by us under the guidance of Dr.T.Vino M.E.Ph.D. is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

1.

2.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of We are pleased to acknowledge our sincere thanks to the Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey our thanks to Electronics And **Dr. N. M. Nandhitha, M.E, Ph.D.**, Dean, School of Electrical and Electronics and **Dr.T.RAVI,M.E.Ph.D** Head of the Department, Dept. of Electronics and Communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to my Project Guide **Dr.T.Vino M.E.Ph.D** Electronics and Communication Engineering of Sathyabama Institute of Science and Technology for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

We wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication who were helpful in many ways for the completion of the project.

ABSTRACT

Braille system is the main method for the blind and visually impaired people for acquiring the reading and writing skills. Considering their disability, efforts were dedicated to facilitating the learning with Braille. Studying Braille system at the specialized centers and schools is not easy for all the involved blind and visually impaired people. Moreover, making the learning process independent of others is very useful. This project presents the design and implementation of a Self-learning Braille application for the blind and visually impaired people on the smart devices.

This project describes an Arduino based Hardware Implementation of the Braille teaching device, where the problem of existing method can be abridged. This project implements a Braille Script Teaching Aid in which the presence of combination of six vibration motors is operated using the input got from the Arduino. All these vibration motors are controlled using Relay. This method provides an easy way of teaching Braille Script. Some self-learning programs are developed to make students to learn without the help of a teacher. It is well suited for first time learners. We can teach different languages with the same setup by selecting the language based on the requirement.

TABLE OF CONTENTS

Chapter No.	TITLE	PAGE NO
1	INTRODUCTION	1
2	LITERATURE SURVEY	6
3	INVESTIGATION	9
4	HARDWARE COMPONENTS DETAILS	12
4.1	Arduino Uno	12
4.1.1	Power	13
4.1.2	Memory	14
4.1.3	Automatic Software Download	15
4.1.4	Pin Configuration	18
4.1.5	Pin Description	18
4.1.6	Usb Overcurrent Protection	20
4.1.7	Applications	20
4.1.8	Physical Characteristics	20
4.2	IC7805	20
4.3	Bluetooth Module	21
4.4	Hardware Features	22
4.5	Software Features	22
4.6	Relay	24
4.7	Power Supply Unit	26
4.8	Vibration Motors	29
5	SOFTWARE REQUIREMENTS DETAILS	
5.1	Embedded C	30
5.2	Arduino Ide	34
6	Results and Discussion	39
6.1	Braille method	39
6.2	User interface	39
6.3	Blind person's user interface	40
6.4	OUTPUT	40
7	CONCLUSION AND REFERENCES	41

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
1.1	Embedded System	2
1.2	Classification Of Embedded System	3
1.3	Block Diagram Of A Typical Embedded System	4
3.1	Block Diagram	10
4.1	Arduino Uno Diagram	13
4.2	Pin Diagram Of Arduino Uno	16
4.3	Bluetooth Module	17
4.4	Internal Pin Configuration	23
4.5	555 Timer	25
4.6	Block Diagram Of Power Supply	26
4.7	Processing Of Power Supply	26
4.8	Vibration Motors	29
5.1	Block Diagram Of Embedded C	30
5.2	Embedded C Program With Examples	33
5.3	Arduino Window	35
5.4	Button Bar	36
5.5	Source Code Window	37
6.1	UI (Bluetooth App)	39
6.2	Six Vibrating Motors	40
6.3	Output	40

CHAPTER 1

INTRODUCTION

EMBEDDED SYSTEM

An embedded system is an electronic/electro-mechanical system designed to perform a specific function and is a combination of both hardware and firmware (software). Every embedded system is unique, and the hardware as well as the firmware is highly specialised to the application domain. Embedded systems are becoming an inevitable part of any product or equipment in all fields including household appliances, telecommunications, medical equipment, industrial control, consumer products, etc. However, by building intelligent mechanisms on top of the hardware, taking advantage of possible existing sensors and the existence of a network of embedded units, one can both optimally manage available resources at the unit and network levels as well as provide augmented functions, well beyond those available. For example, intelligent techniques can be designed to manage power.

Embedded systems are designed for a specific application from the characteristics of the embedded system for the mobile embedded device category. However, smart phones come with a built-in operating system and it is not modifiable by the end user. It makes the clause: “The firmware of the embedded system is unalterable by the end user”, still a valid clause in the mobile embedded device category

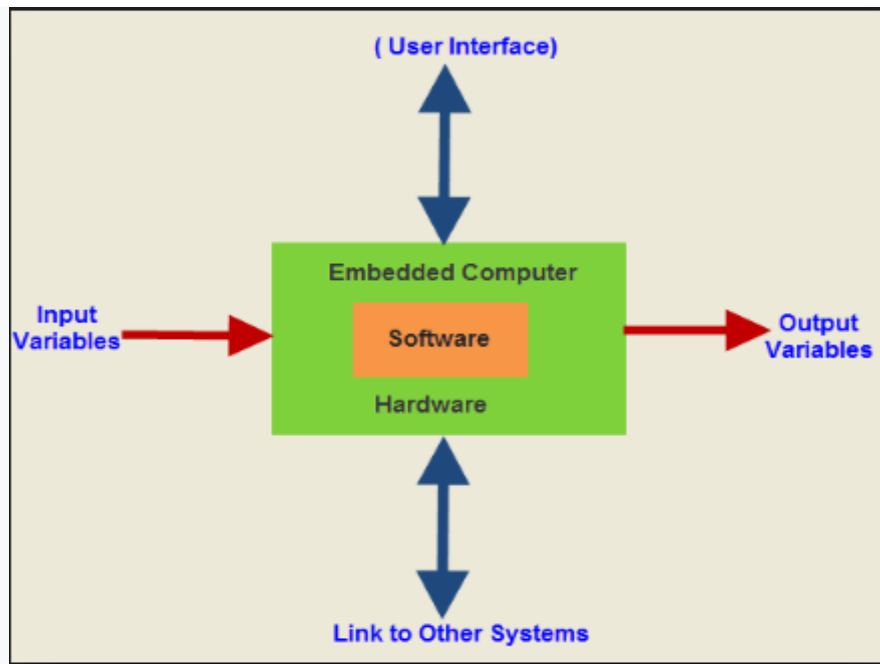


Fig 1.1 Embedded system

Embedded hardware/software systems are basically designed to regulate a physical variable or to manipulate the state of some devices by sending some control signals to the Actuators or devices connected to the O/p ports of the system, in response to the input signals provided by the end users or Sensors which are connected to the input ports. Hence an embedded system can be viewed as a reactive system. The control is achieved by processing the information coming from the sensors and user interfaces, and controlling some actuators that regulate the physical variable.

CLASSIFICATIONS OF EMBDDED SYSTEMS:

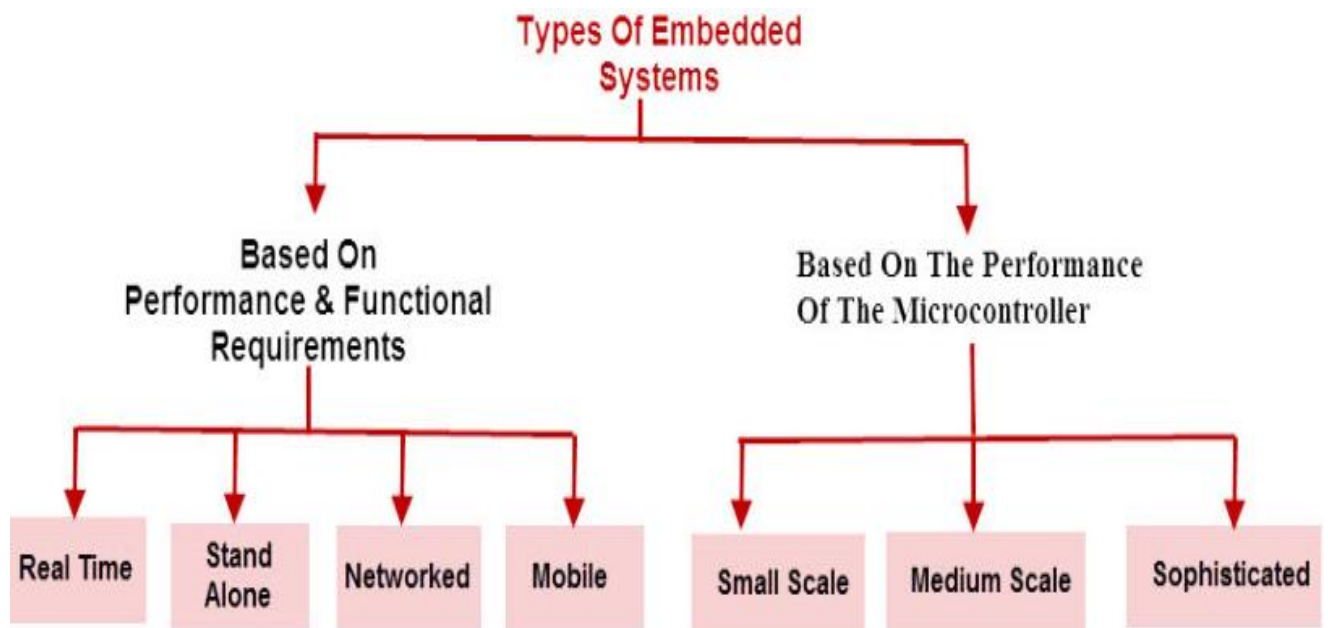


Fig 1.2 Classification of Embedded System

Advanced HVAC systems use networked controls to more accurately and efficiently control temperature that can change by time of day and season. Home automation uses wired- and wireless-networking that can be used to control lights, climate, security, audio/visual, surveillance, etc., all of which use embedded devices for sensing and controlling.

Like traffic lights, factory controllers, and largely complex systems like MRI, and avionics Embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations. Complexity varies from low, with a single microcontroller

BLOCK DIAGRAM OF AN EMBEDDED SYSTEM

An embedded system usually contains an embedded processor. Many appliances that have a digital interface microwaves, VCRs, cars utilize embedded systems. Some embedded systems include an operating system. Others are very specialized resulting in the entire logic being implemented as a single program. These systems are embedded into some device for some specific purpose other than to provide general purpose computing.

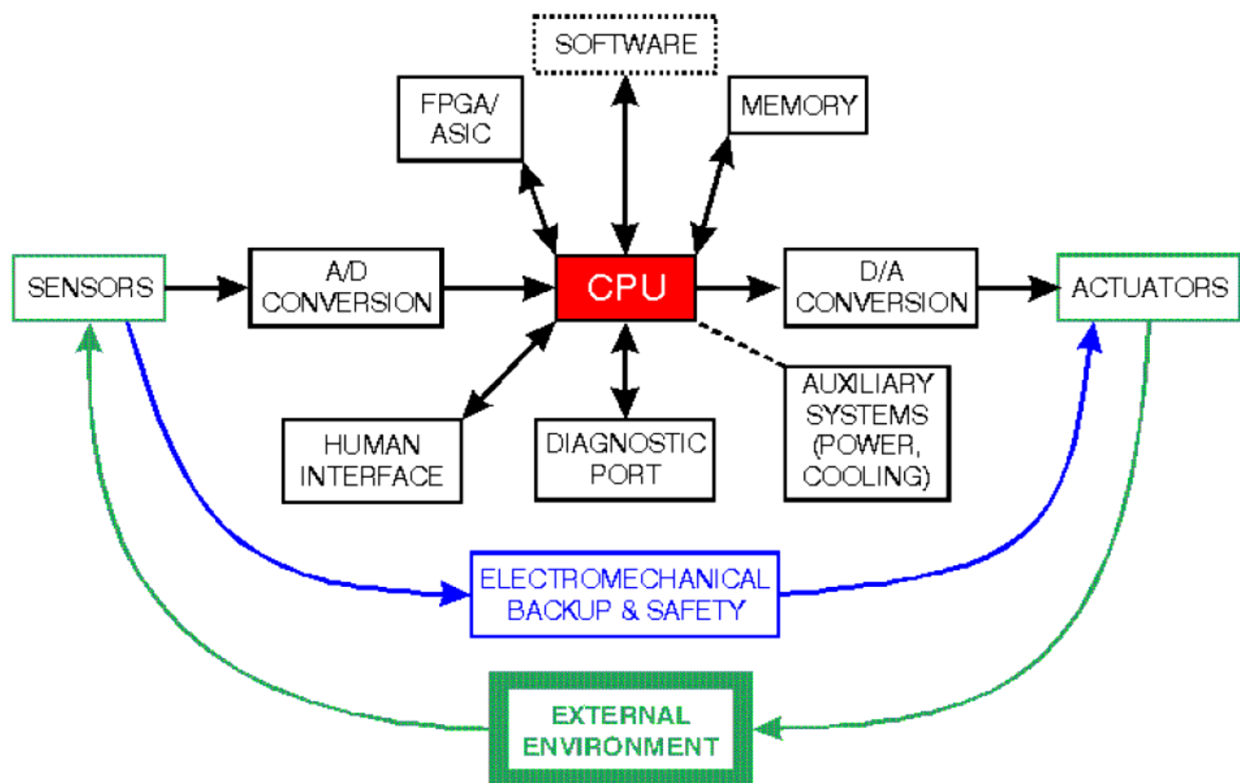


Fig 1.3 Block diagram of a typical embedded system

EMBEDDED SYSTEMS APPLICATIONS

Embedded systems in automobiles include motor control, cruise control, body safety, engine safety, robotics in an assembly line, car multimedia, car entertainment, E-com access, mobiles etc.

1. Embedded systems in telecommunications include networking, mobile computing, and wireless communications, etc.
2. Embedded system in smart cards include banking, telephone and security systems.
3. Embedded Systems in satellites and missiles include defence, communication, and aerospace
4. Embedded systems in computer networking & peripherals include image processing, networking systems, printers, network cards, monitors and displays
5. Embedded Systems in digital consumer electronics include set-top boxes, DVDs, high definition TVs and digital cameras

CHAPTER 2

LITERATURE SURVEY

Braille Character Recognition Based on Neural Networks

Braille is the most popular system used for interaction between visually-impaired and sighted people using tactile means. Optical Braille character recognition (OBCR) includes two main steps: Braille cells' recognition (image acquisition, pre-processing, Braille dots' recognition, Braille cells' recognition and segmentation) and Braille cells' transcription to corresponding natural language characters. System example has been created using image processing methods and artificial neural networks approach. These methods allow to achieve high speed and recognition accuracy level. System can adapt to factors like quality of input patterns and differences between them dynamically. In this paper, artificial neural network is developed to identify letter's images of Cyrillic alphabet in Braille representation system. Network will be trained and tested for identifying of scanned Cyrillic letters in Braille. Some of the letters are noised with some type of noise to simulate the real-world environment.

BrailleÉcran: A Braille Approach to Text Entry on Smartphones

Nowadays, due to touchscreen, mobile devices have become much more dynamic than in the past. However, due the same reason, those devices are less accessible for blind people. It is expected that in 2018, over half of mobile phone users will have a smartphone, therefore researches about mobile accessibility are very important. So, the aim of this systematic review is to cooperate with new research about methods for braille text entry on smartphones. The systematic searches in 5 databases, resulted in 11 papers that answered the research questions that grounded this work.

Braille Image Recognition for Beginners

Braille is a tactile that consists of dots which is used by visually impaired people in reading. Braille pattern of alphabet consists of its own pattern in which some of it does not even relate to the alphabet. Thus, it is difficult for normal people to detect and recognize the braille pattern. This paper presents a study of braille image recognition for beginners. The outcome of this study is expected to translate a braille image patterns into a readable alphabet text. A technique of Bag of Features (BOF) is proposed for the recognition of the braille image. On the other hand, the image classification is done using a Support Vector Machine (SVM) technique. Seventy-eight of braille images is tested. From the testing performed, it is found that 97.44% of correct recognition accuracy is achieved which revealed that the proposed techniques are applicable for braille image recognition.

A Customized Transportation Intervention for Persons with Visual Impairments

Transportation can be an employment barrier for persons with disabilities, particularly those with visual impairments. A customized transportation intervention for people with visual impairments, based on concepts associated with customized employment, was devised, implemented, and evaluated. Methods: A pretest and post-test intervention and comparison group design evaluated changes in participants' social problem-solving skills, transportation self-efficacy, and transportation knowledge. Participants worked with a transportation coordinator to plan and secure work-related transportation. Surveys assessed participant satisfaction with the intervention. Results: The intervention group had significantly higher scores for social problem-solving

skills at post-test after considering protest scores. The intervention group responded favourably to measures of satisfaction with the intervention.

Discussion: The customized transportation

intervention had a positive effect on social problem-solving skills and the participants were satisfied with the intervention, although the impact of the intervention on employment outcomes remains unclear. Attrition in the study and lack of a random sample mean the results must be interpreted with caution.

Implications for practitioners: Engaging individuals with visual impairments in discussions about transportation to work may educate them about new options or assist them in thinking of unexplored options. Persons who appear to have good problem-solving skills and confidence in their abilities may still benefit from these discussions

CHAPTER 3

INVESTIGATION

3.1 EXISTING SYSTEM

Education is the key to success in life, normal children learn from teachers in schools seeing and communicating with each other. For Visually challenged students learning to read and write is much more difficult because of visual impairment.

The teaching aids required to teach such blind pupil are special and require lot of teacher's attention.

For a first time learning blind student, character set a number are taught with a help of marbles and slotted slate.

Different arrangement patterns of marbles on slotted slate represent different characters. For each character representation, teacher has to reach each student and change the arrangement of marbles. This is harder and time-consuming exercise.

3.2 PROPOSED SYSTEM

The projected system provides translation for visually impaired people to read English characters.

The system uses the characters for vibrating the vibration motors for the visually impaired people in the form of braille language which is communicated to the users through mini vibration motors representing each braille glyph. Suitable hardware is chosen for implementing the system. The system comprises, a single braille cell and mobile for voice recognition to get the characters as the input. STM32 microcontroller is used as the processing device in this project.

In this projected system also consists of an spare switch. In case of any emergency situation the blind people use this switch and the alert message is sent to the corresponding person through GSM.

3.3 BLOCK DIAGRAM

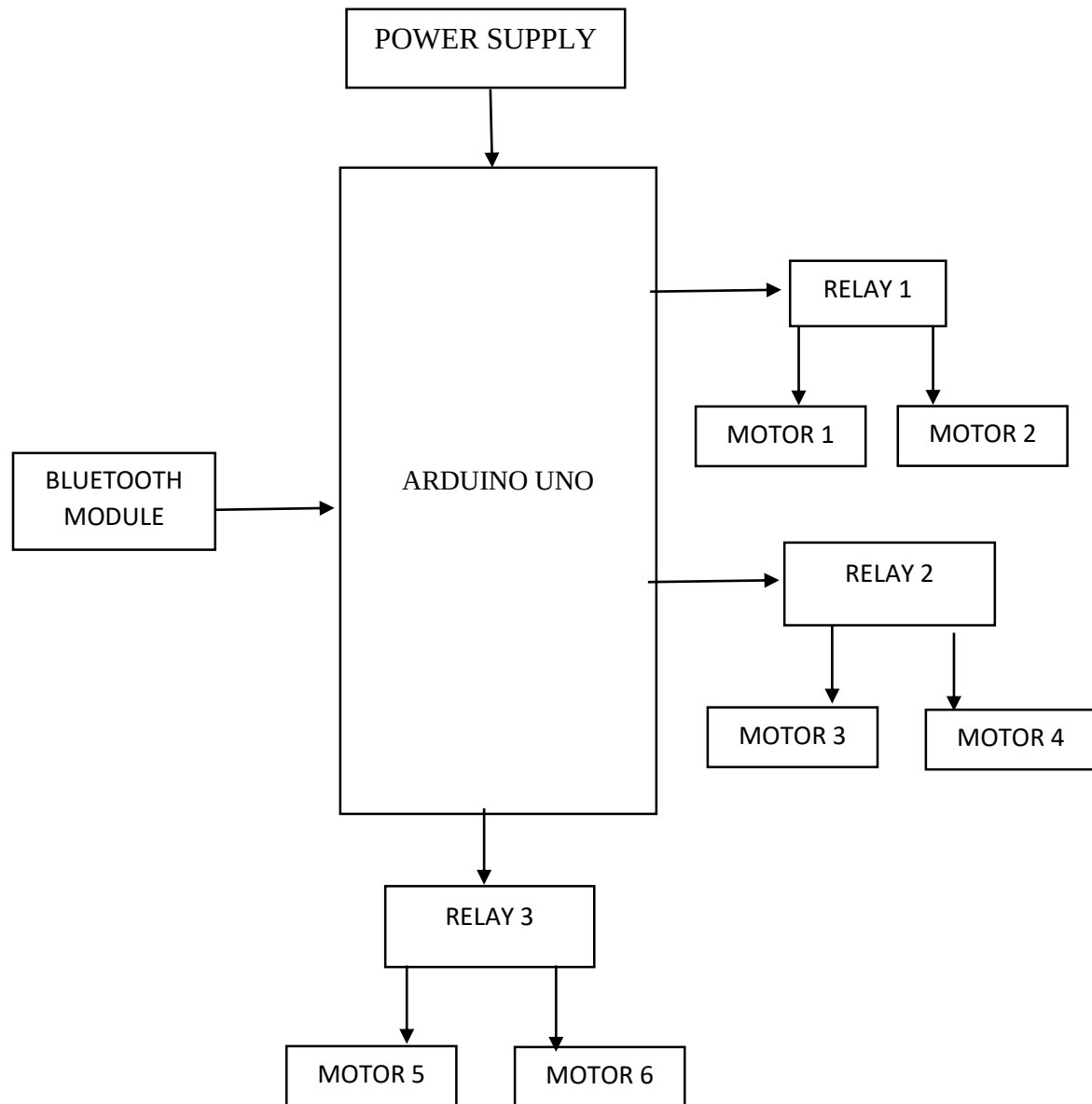


Fig 3.1 Block Diagram

3.4 HARDWARE REQUIREMENTS

1. Arduino uno
2. Bluetooth Module (HC-05)
3. Relay
4. Vibration motors.

3.5 SOFTWARE REQUIREMENTS

1. Arduino IDE,
2. Embedded C.

Chapter 4

HARDWARE COMPONENTS DETAILS

4.1 ARDUINO UNO

The microcontroller Arduino Uno board based on the ATmega328 (datasheet). It has 14 digital i/o pins, 6 analog i/p, a USB connection, a power jack, an ICSP header, a reset button, and a 16 MHz ceramic resonator. It contains everything needed to support the microcontroller; very easy to connect to a computer by using USB cable or power it with an AC-to-Direct Current 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 programmed as a USB-to-serial converter. 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.

At mega 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.

4.1.1 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: The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

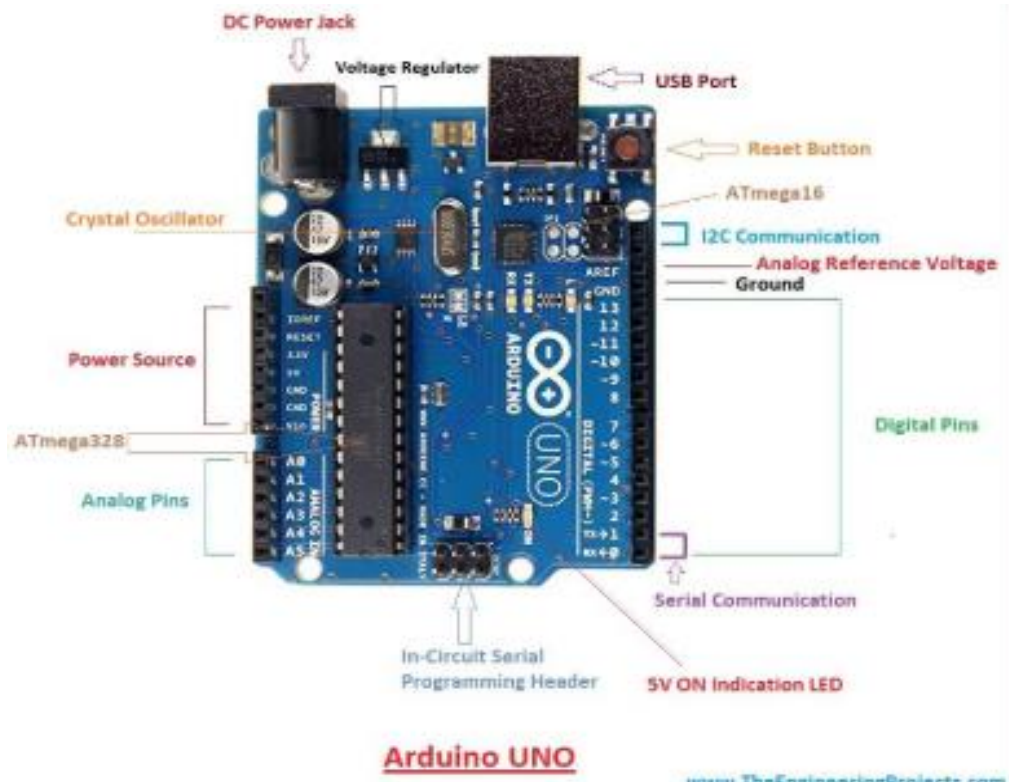


Fig 4.1 Arduino Uno diagram

3V 3. A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

GND. Ground pins.

4.1.2 MEMORY

The ATmega328 has 32 KB. It also has 2 KB of SRAM and 1 KB of EEPROM .i/o Each of the 14 digital pins on the Uno can be used as an input or output, using digital Write (),pin Mode () and digital Read () functions. They operate at 5 volts. Every pin can provide or receive a max 40 mA and has an internal pull-up resistor of 20-50 kohms.

4.1.3 AUTOMATIC (SOFTWARE) DOWNLOAD

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can

be soldered together to re-enable it. It's labelled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line

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 attachInterrupt () function for details.

PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite () function.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off. The Uno has 6 analog inputs, labelled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default, they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference () function. Additionally, some pins have specialized functionality:

TWI: A5 or SCL pin. Support TWI communication using the Wire library. There are a couple of other pins on the board:

AREF. Reference voltage for the analog inputs. Used with analogReference ().

Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board. See also the mapping

between Arduino pins and ATmega328 ports. The mapping for the Atmega8, 168, and 328 is identical. Communication the Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.

The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 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 '16U2 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. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

4.1.4 PIN CONFIGURATION

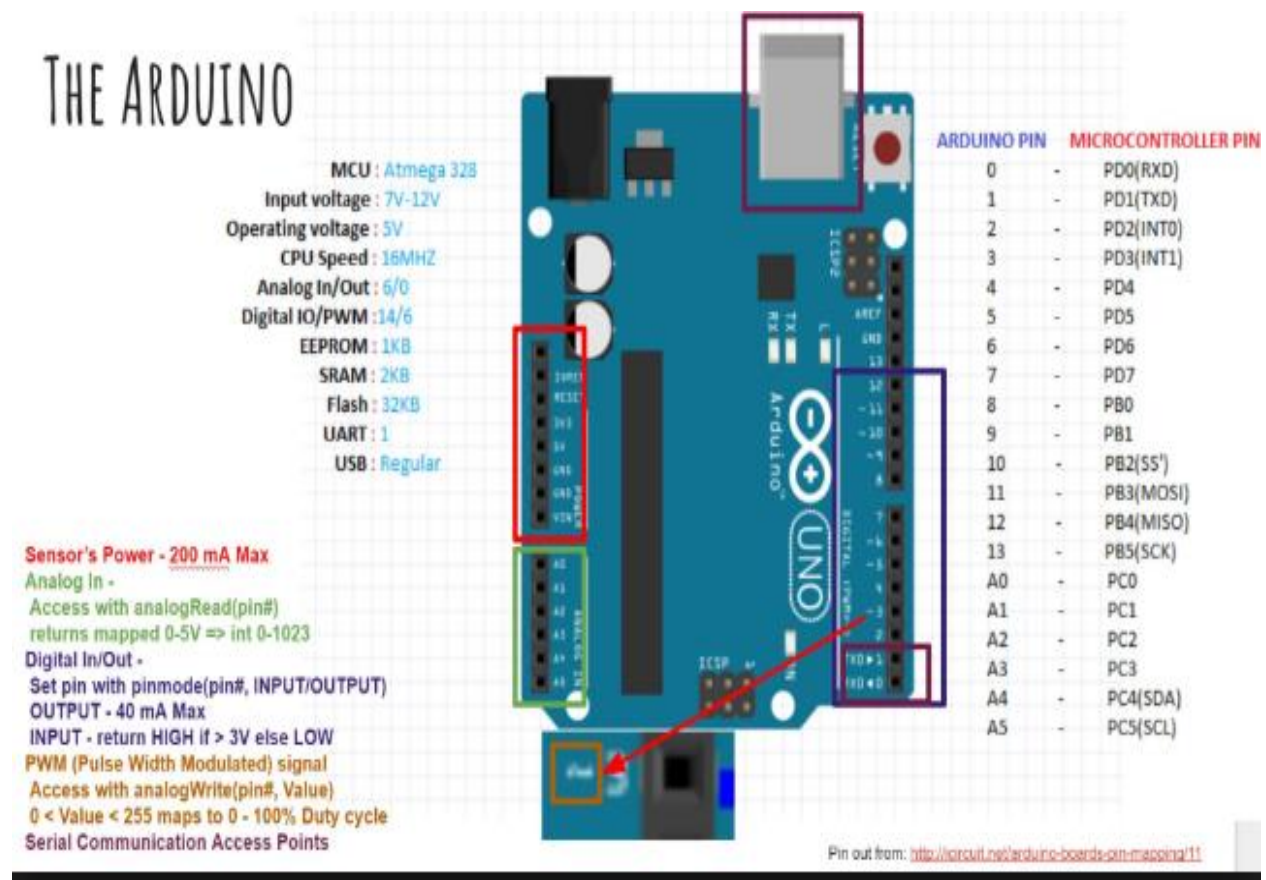


Fig 4.2 Pin diagram of Arduino Uno

4.1.5 PIN DESCRIPTION

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used

as output from the inverting Oscillator amplifier. If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set. Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5..0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running.

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

AVCC is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. AREF is the analog reference pin for the A/D Converter. In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

4.1.6 USB OVERCURRENT PROTECTION

The Arduino Uno has a resettable poly fuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

4.1.7. APPLICATIONS

Today the ATmega328 is commonly used in many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino Uno and Arduino Nano models.

4.1.8 PHYSICAL CHARACTERISTICS

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100-mil spacing of the other pins.

4.2 IC 7805 (VOLTAGE REGULATOR IC)

Voltage regulators are very common in electronic circuits. They provide a constant output voltage for a varied input voltage. In our case the 7805 IC is an iconic regulator IC that finds its application in most of the projects. The name 7805 signifies two meaning, —78|| means that it is a positive voltage regulator and —05|| means that it provides 5V as output. So our 7805 will provide a +5V output voltage 7805 is a voltage regulator integrated circuit. It is a member of

78xx series of fixed linear voltage regulator ICs. The voltage source in a circuit may have fluctuations and would not give the fixed voltage output. The voltage regulator IC maintains the output voltage at a constant value. The xx in 78xx indicates the fixed output voltage it is designed to provide. 7805 provides +5V regulated power supply. Capacitors of suitable values can be connected at input and output pins depending upon the respective voltage levels.

4.3 BLUETOOTH MODULE

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for translucent wireless serial connection setup.

Serial port Bluetooth module is entirely competent Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

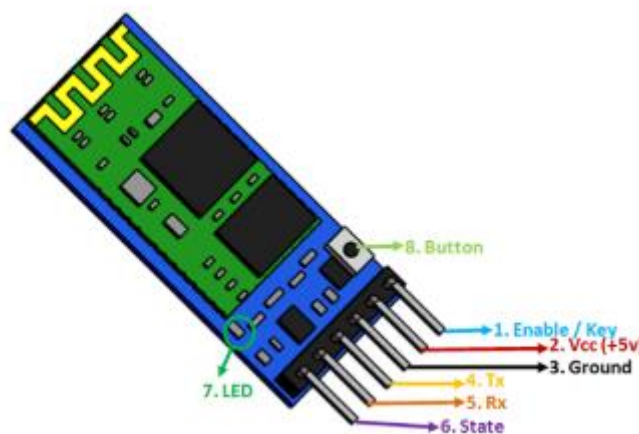


Fig 4.3 Bluetooth Module

4.4 HARDWARE FEATURES

Typical -80dBm sensitivity

Up to +4dBm RF transmit power

Low Power 1.8V Operation ,1.8 to 3.6V I/O

PIO control

UART interface with programmable baud rate

With integrated antenna

With edge connector

4.5 SOFTWARE FEATURES

Default Baud rate: 38400, Data bits:8, Stop bit:1,Parity:No parity, Data control: has.

Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.

Given a rising pulse in PIO0, device will be disconnected.

Status instruction port PIO1: low-disconnected, high-connected;

PIO10 and PIO11 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s

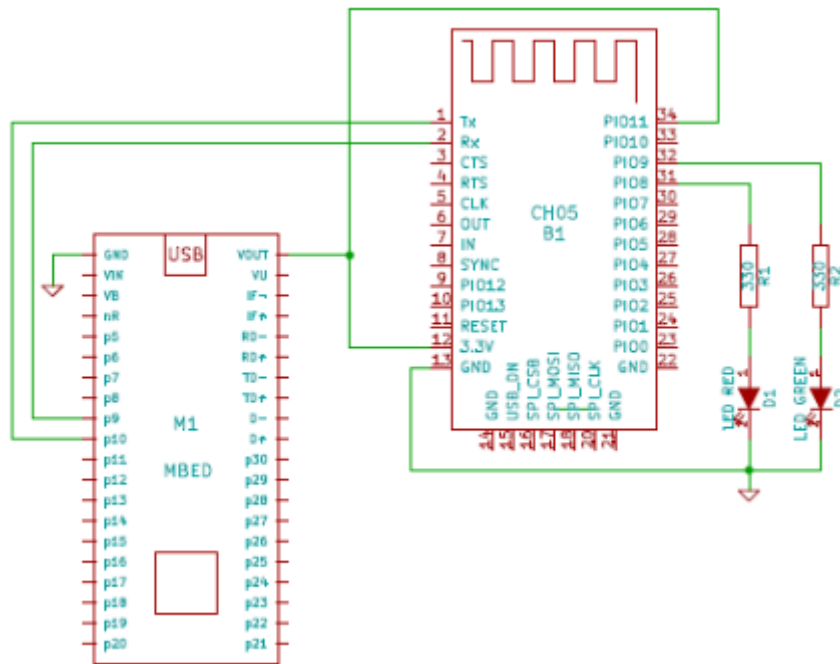


Fig: 4.4 Internal Pin Configuration

Auto-connect to the last device on power as default.

Permit pairing device to connect as default.

Auto-pairing PINCODE: "0000" as default

Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

The HC-05 has 2-operating modes, one is the Data mode in which can send and receive data from other devices and the other is the AT Command mode where the default device settings can be changed. We can operate the device in either of these two modes by using the key pin as explained in the pin description.

This one is very easy to pair the HC-05 module with microcontrollers because works using the SPP. Simply power the module with +5V and connect the Rx pin of the module to Tx of microcontrol unit and Tx pin of module to Rx of MCU

During power up the key pin can be grounded to enter into Command mode, if left free it will by default enter into the data mode. this module is powered it you should be able to discover the Bluetooth device as HC-05 then connect with it by using 1234 as password and start communicating with your device .

Applications

Wireless communication between two microcontrollers

Communicate with Laptop, Desktops and mobile phones

Data Logging application

Consumer applications

Wireless Robots

Home Automation

4.6 RELAY

A relay is an electrically mechanically operated switch. Current flowing through the coil of the relay generates a magnetic field that changes NC to NO switch contacts. The coil current can be switched on or off, so that relays have two switching positions. Relays allow a ckt to another circuit, which can be completely separated from the first. For example, a low voltage battery circuit can use a relay to switch a 230V AC circuit. There is no electrical connection within the relay between the two circuits; the connection is magnetic and mechanical. The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be up to 100mA for relays designed for operation with lower voltages. Most ICs (chips) cannot provide this current and a transistor is usually used to convert the small IC current to the larger The maximum output

current for the popular 555 timer IC is 200mA so these devices can supply relay coils directly without amplification.



Fig 4.4 555 timer

Relays usually SPoleDT or DPDT, but they can have much more sets of switch contacts, ex relays with 4 sets of switch associates are fervently available. Most relays are designed for PCB bump, but you can solder wires directly to the pins, provided you take care of melting, plastic housing of the relay to avoid. The energetic image shows a working relay with its coil and switch contacts. You can see that a lever on the left side is attracted by magnetism when the coil is turned on. This lever moves the switch contacts. There are a number of contacts (SPDT) at the top and another behind them, so the dpDT relay.



The relay's switch connections are usually labelled COM, NC and NO:

COM = Common, always connect to this, it is the moving part of the switch.

NC = Normally Closed, COM is connected to this when the relay coil is **off**.

NO = Normally Open, COM is connected to this when the relay coil is **on**.

4.7 POWER SUPPLY UNIT

power supply is a device or asystem that supplies electrical or other types of energy to an output load or group of loads. The term is most commonly applied to electrical energy supplies, less often to motorised ones, and rarely to others.

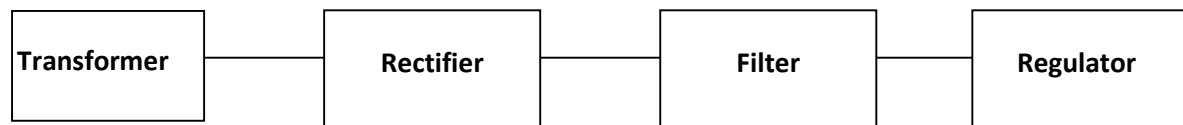


Fig 4.6 Block diagram of power supply

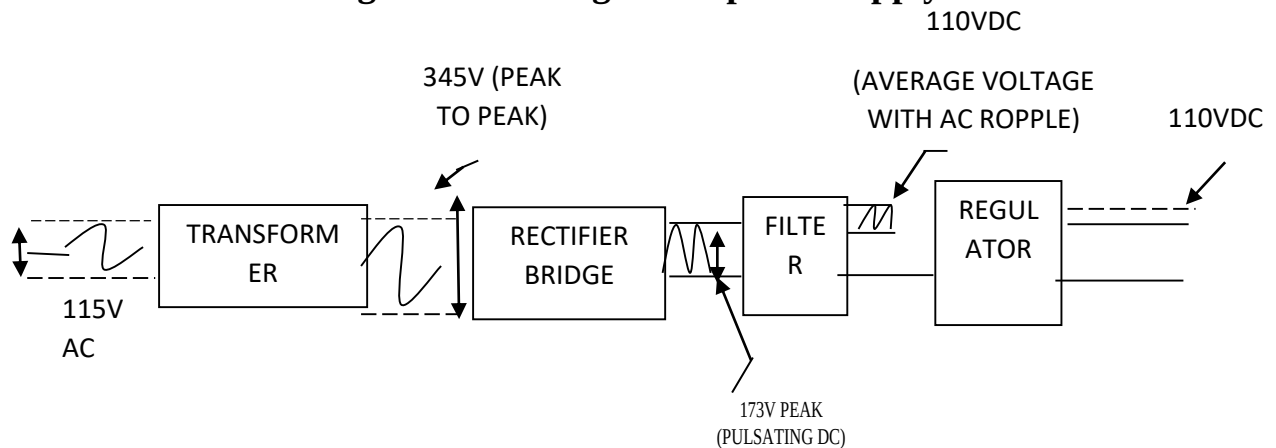


Fig 4.7 Processing Of Power Supply

The transformer steps up or steps down the input line voltage and isolates the power supply from the power line. The RECTIFIER section converts the alternating current input signal to a pulsating direct current. However, as you proceed in this chapter you will learn that pulsating dc is not desirable. For this reason a FILTER section is used to convert pulsating dc to a purer, more desirable form of dc voltage.

At last, the REGULATOR is doing exactly what the name suggests. Supports the o/p of the power supply to a persistent level despite large changes in the load current. Let's locate an AC signal through the power supply. At this point you need to see how this signal changes in each area of power supply. Later in the chapter you will see how these variations take place. An input signal of 115 volts ac is applied to the transformer's primary. Usually it is a step-up transformer with it rotating ratio of 1:3. Find the output for this Xer by swelling the input voltage around the ratio of the revolutions in the primary to the ratio of the revolutions in the secondary; therefore $115 \text{ volts ac} \times 3 = 345 \text{ volts ac}$ at the output. Since each diode in the rectifier section leads to 180 degrees of 360-degree input, the output of the rectifier is 11.2 or about 173 v pulsating DC. The filter section, a network of resistors, capacitors or inductors, controls the rise and fall time of the different signal; accordingly, the signal remains at a more constant DC level. You will see the filtering process more clearly in the discussion of the actual filter circuits. The output of the filter is a signal of 110 voltdc, with AC ripple drive on the dc. The reason for the low voltage (average voltage) is explained later in this chapter. The regulator maintains its output at a constant 110-volt dc level, which is used by the electronic equipment .

Simple 5Volt PS for digital circuits Short image of operation: Gives outwell controlled 5Volt output, Output current capacity of 100 mA Circuit safety: Integral overheat protection switches output when RegulatorIC gets too hot Circuit density: Very easy and easy to build circuit power: Very stable +5V output voltage, reliable operation Availability of components: Easy to get, uses only very common basic components design tests : Based on data sheet sample circuit, I have used this circuit successfully as part of many electronic projects : Part of Electronic Devices, Small Laboratory Power Supply Voltage: Unregulated DC 8-18V Power Supply Power Power Power: Required Output Current + 5 mA

Component Cost: Few Dollars for The Electronic Components + The Input Transformer Cost.

4.8 VIBRATION MOTORS

There are two basic types of vibrating motor. A linear resonance actuator (LRA) contains a small internal mass near a spring, generating a force during the drive. A mass vibration motor (ERM) that rotates unusually uses a small asymmetric mass in a DC motor, when it rotates, generating a force that results in vibrations. As specialists in the supply and construction of vibration motors, you will find our equipped engines in our product catalog. You'll find a lot of additional information and instructions to help you understand how vibration motors work



Fig 4.8 Vibration Motors

Chapter 5

SOFTWARE REQUIREMENTS DETAILS

5.1 EMBEDDED C

The most popular programming language in software field is developing electronic gadgets is Embedded C. Each processor used in electronic system is associated with embedded software.

It plays a important role in performing specific task by the processor. In usual life we used many ED'S such as mobile phone digital camera, etc. These all device working is based on MC that are programmed by emb C.

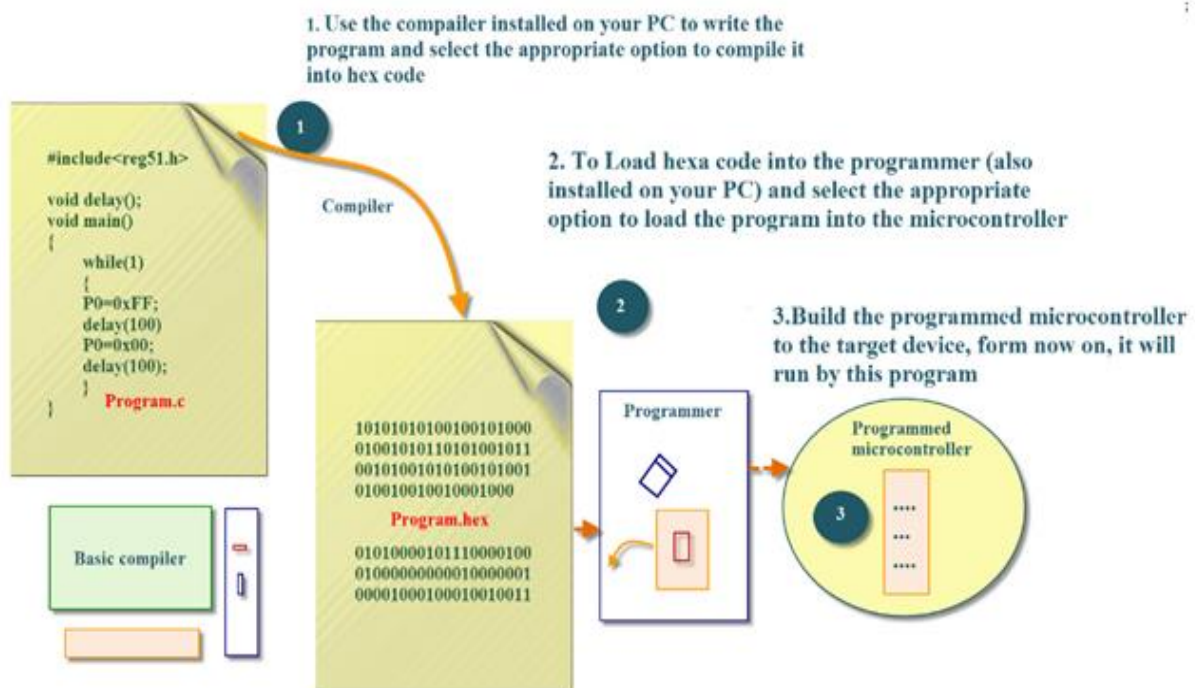


Fig 5.1 Block diagram of Embedded C

you are going to repeat the LED blink sketch that you used during your testing stage. This time, however, you are going to connect an LED to one of the

digital pins rather than using LED13, which is soldered to the board. You will also learn exactly how the hardware and the software for this project works, learning a bit about electronics and coding in the Arduino language (which is a variant of C) at the same time.

In E- system programming C code is preferred over other language. Due to the following reasons:

1. Informal to understand
2. Great Reliability
3. Manageability
4. Scalability

Function is a collection of statements that is used for performing a specific task and a collection of one or more functions is called a programming language.

Most patrons are familiar with application software that provide functionality on a computer. Embedded software however is often less visible, but no less complicated. Distinct application software, embedded software has fixed hardware requirements and abilities, and addition of third-party hardware or software is strictly controlled.

Embedded software wants to include all desirable device drivers at manufacturing time, and the device drivers are written for the specific hardware. The software is highly dependent on the CPU and specific chips chosen. Most embedded software engineers have at least a passing knowledge of reading schematics, and reading data sheets for components to determine usage between decimal, hexadecimal and binary is useful as well as using bit manipulation.

Web applications are rarely used, while XML files and other output may be passed to a computer for display. File systems with folders are classically absent as are mySQL databases.

Software development requires use of a cross compiler, which runs on a computer but produces executable code for the target device. Debugging requires use of an in-circuit emulator, JTAG or SWD. Software developers often have access to the complete kernel (OS) source code.

Size of the storage memory and RAM can vary significantly. Some systems run in 16 KB of Flash and 4 KB of RAM with a CPU operating at 8 MHz, other systems can rival contemporary computers. These space requirements lead to more work being done in C. Interpreted languages like BASIC and Java is older ARM11 used in Raspberry Pi and Intel Galileo Gen.

are not commonly used; while an implementation of the interpreted Python 3 language – MicroPython – is however available expressly for microcontroller use, e.g. 32-bit ARM-based (such as BBC micro:bit) and 16-bit PIC microcontrollers.

Communications between processors and between one processor and other components are essential. Besides direct memory addressing, common protocols include I²C, SPI, serial ports, and USB.

Communications protocols designed for use in embedded systems are available as closed source from companies including InterNiche Technologies and CMX Systems. Open-source protocols stem from uIP, lwip, and others

This program explains how to use structure within structure in C using normal variable. “student_School_detail” structure is declared inside “student_detail”

structure in this program. Both structure variables Please note that members of “student_college_detail” structure are retrieved by 2 dot(.) operator and members of “student are normal structure variables.detail” structure are accessed by single dot(.) operator.

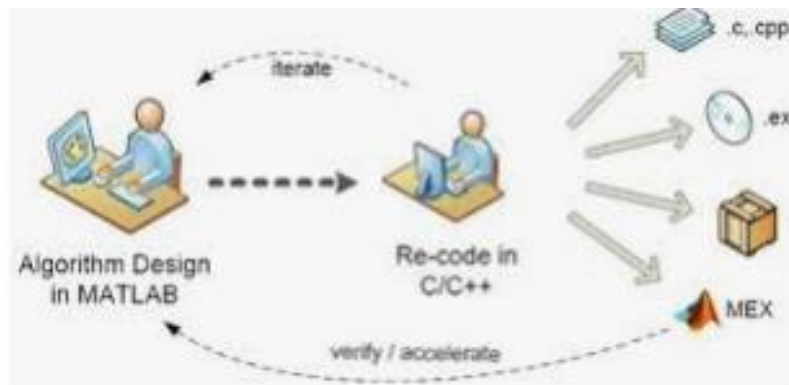


Fig 5.2 Embedded C program with examples

A Keyword is a special word with a special import to the compiler (a C Compiler for example, is a software that is used to convert program written in C to Machine Code). For example, if we take the Keil’s Cx51 Compiler the following are some of the keywords:

1. bit
2. sbit
3. sfr
4. small
5. large

These are few of the many keywords associated with the Cx51 C Compiler along with the standard C Keywords.

5.2 ARDUINO IDE

When you open up the Arduino IDE, it will look very similar to the image in Figure 1-13. If you are using Windows or Linux, there may be some slight differences but the IDE is pretty much the same no matter what OS you use

The IDE is split into three parts: the Toolbar across the top, the code or Sketch Window in the center, and the messages window in the bottom. The Toolbar consists of seven buttons. Underneath the Toolbar is a tab, or set of tabs, with the filename of the sketch within the tab. There is also one button on the far right hand side. Along the top is the file menu with drop down menus labeled File, Edit, Sketch, Tools and Help. The buttons in the Toolbar provide convenient access to the most commonly used functions within this file menu.. The Toolbar buttons and their functions are listed in Table 1-1. Table 1-1. The Toolbar button functions

Toolbar button	functions
Verify/Compile	Checks the code for errors
Stop	Stops the serial monitor, or un-highlights the other buttons
New	Creates a new blank sketch
Open	Shows a list of sketches in your Sketchbook to open
Save	Saves the current Sketch to your Sketchbook
Upload	Uploads the current Sketch to the Arduino
Serial Monitor	Displays serial data being sent from the Arduino

The Verify/Compile button is used to check that your code is correct and error free before you upload it to your Arduino board. The Stop button stops the serial monitor from operating. It also un-highlights other selected buttons. While the serial monitor is operating, you can press the Stop button to obtain a snapshot of the serial data so far to examine it. This is particularly useful if you are sending data out to the Serial Monitor quicker than you can read it. The New button creates a new and blank sketch ready for you to enter your code into.. The Arduino hardware and software were designed for artists, designers, hobbyists, hackers, newbies, and anyone interested in creating interactive objects or environments. Arduino can interact with buttons, LEDs, motors, speakers, GPS units, cameras,

the internet, and even your smart-phone or your TV! This flexibility combined with the fact that the Arduino software is free, the hardware boards are pretty cheap, and both the software and hardware are easy to learn has led to a large community of users who have contributed code and released instructions for a **huge** variety of Arduino-based projects

As shown in the figure 5.2 The Arduino IDE is incredibly minimalistic, yet it provides a near-complete environment for most Arduino-based projects. The top menu bar has the standard options, including “File” (new, load save, etc.), “Edit” (font, copy, paste, etc.), “Sketch” (for compiling and programming), “Tools” (useful options for testing projects), and “Help”. The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that were found in the program, and various other useful messages.



Fig 5.2 Arduino Window

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from

programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an “Arduino language.” However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

The 6 Buttons

While more advanced projects will take advantage of the built-in tools in the IDE, most projects will rely on the six buttons found below the menu bar.



Fig 5.3 Button Bar

The button bars

As shown in the figure 5.3

1. The **check mark** is used to verify your code. Click this once you have written your code.
2. The **arrow** uploads your code to the Arduino to run.
3. The **dotted paper** will create a new file.
4. The **upward arrow** is used to open an existing Arduino project.
5. The **downward arrow** is used to save the current file.
6. The far-right button is a **serial monitor**, which is useful for sending data from the Arduino to the PC for debugging purposes.



Fig 5.4 Source Code Window

As shown in the figure 3.19 the source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures.

User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub `main()` into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program `avrdude` to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an “Arduino

language.” However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

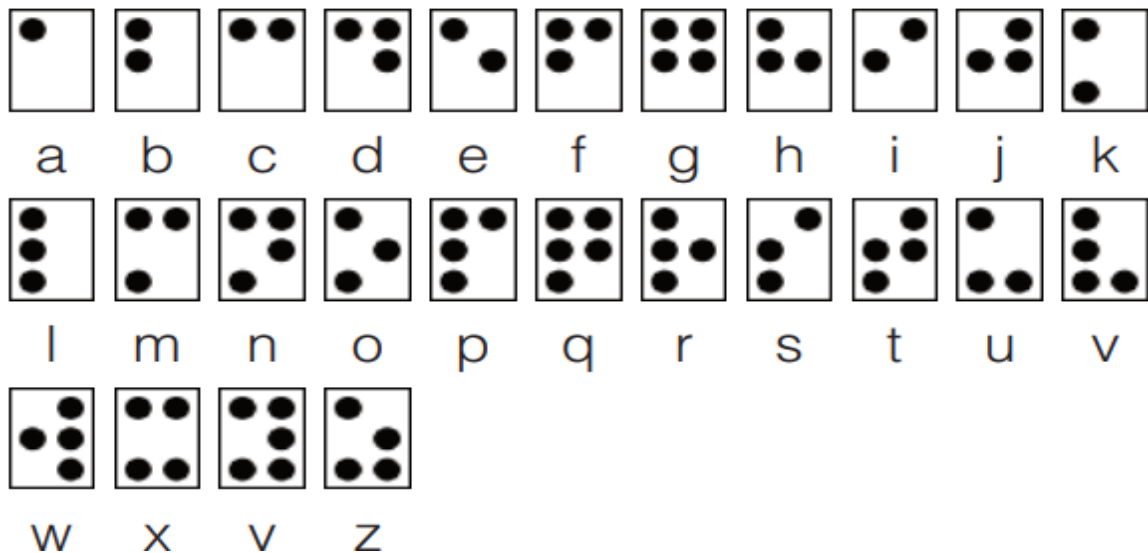
The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub `main()` into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution.

The Arduino IDE employs the program `avrdude` to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

CHAPTER 6

Results and Discussion

6.1 BRAILLE METHOD



The motor vibrating positions are identified by numbers from one to six. There are 64 possible combinations, including no dots at all for a word space. A cell can be used to represent a letter, digit, punctuation mark, or even a word. Braille literacy is a social-justice issue.

6.2 USER INTERFACE

When a user touches buttons or any menu options, the phone will vibrate to help them identify the buttons or the menu options. Additionally, when a user presses any button and performs any of the four general tasks, it will provide vibrational feedback to confirm users' input in the application.

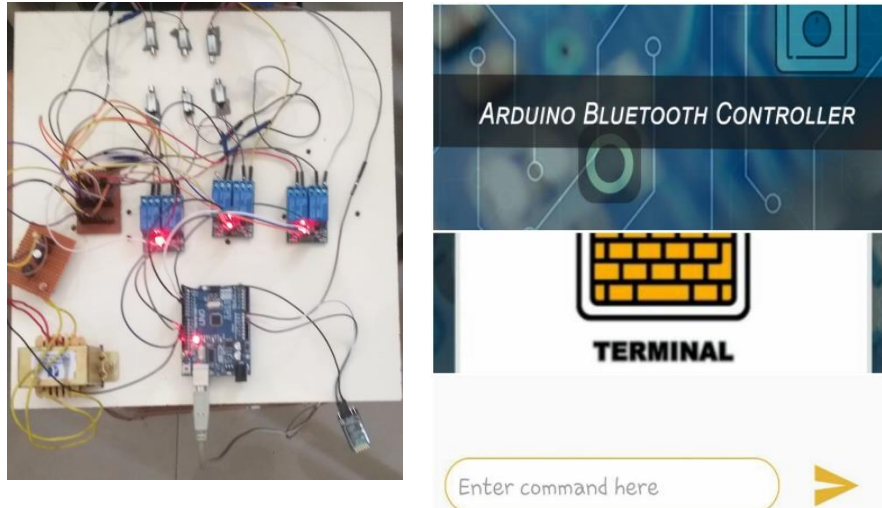


Fig 6.1 UI (Bluetooth app)

6.3 BLIND PERSON'S USER INTERFACE

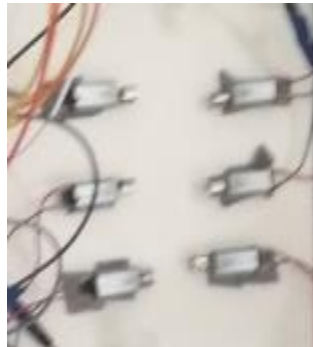


Fig 6.2 Six Vibrating Motors

According to the 6-vibrating motors positions the end user can identify the letters from A to Z this device only Applicable for English Language.

6.4 OUTPUT

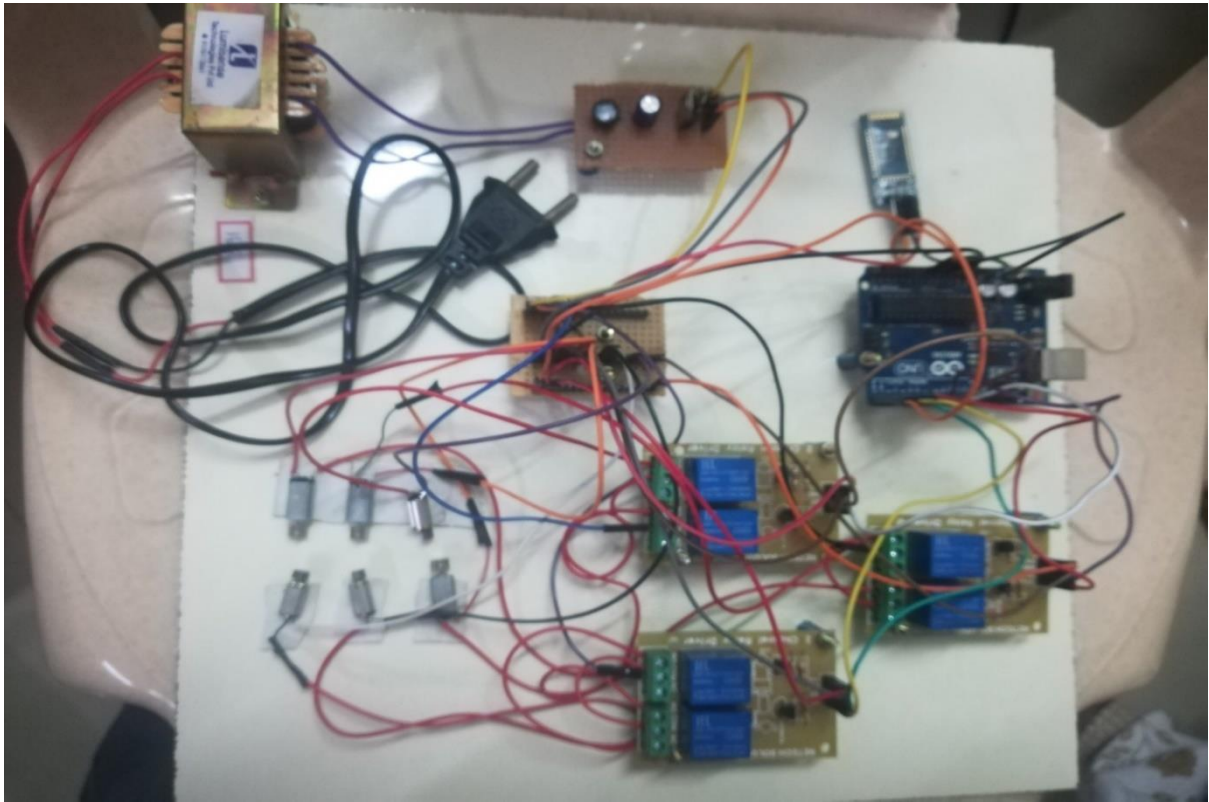


Fig.no:6.3 Output

Chapter 7

CONCLUSION

In our project, we have designed the prototype model for blind, deaf and dumb people by employing a single compact device. The important key factor of this project to facilitate these people and to fix them more confident to manage their sites by themselves. The primary advantage is that the device can be taken away easily and is of about less weight.

REFERENCES

- [1] Ascolini.D and Mariotti S. P “Global estimates of visual ent: ” Brit. mology J, vol. 96, no. 5, pp. 614–618, 2012.
- [2] Busfield J , Frediani G and Carpi F “Enabling portable multiple-line refreshable braille displays with electroactive elastomers,” Med. 60, pp. 86–93, 2018.
- [3] Bicchi, A. E. P. Scilingo, E. Ricciardi, and P. Pietrini, “Tactile flow explains haptic counterparts of common visual illusions,” Brain Res Bull.,vol. 75, no. 6, pp. 737–741, 2008.
- [4] BZarate J. J. and H. Shea, “Using pot-magnets to enable stable and electromagnetic tactile displays,” IEEE Trans. Haptics, vol. 10, no. 112, Jan.–Mar. 2016.
- [5] Brayda L, Campus C, M. Memeo, and Lucagrossi “The importance of visual experience, gender, and emotion in the assessment of an

assistive tactile mouse,” IEEE Trans. Haptics, vol. 8, no. 3, pp. 279–286, Jul-15

- [6] Carriere J. S, Varao T. L Sousa, , and Smilek D, “The way encount
-material influences how frequently we mind wander,” Psychol vol
4, 2013, Art. no. 892.
- [7] Cussomanno A. S. O’Modhrain, R. B. Gillespie, and M. W.
Rodger, refreshable braille displays,” IEEE Trans. Haptics, vol.8,n
o. 3, pp. 287–297, Jul.–Sep. 2015.
- [8] Das S , Sarkar R ,“Analysis of different braille devices for imple-
menting a cost-effective and portable braille system for the visually
impaired people,” Int. J. Comput. Appl., vol. 60, no. 9, pp. 1–5,
2012.
- [9] Eonardis D L. Claudio, and A. Frisoli, “A survey on innovative-
able braille display technologies,” in Proc. Int. Conf. Appl. Human
Ergonom., 2017, pp. 488–498.
- [10] Gillespie B ,Russomanno.A , R., S. O’Modhrain, M.Burns,“The
pressure-controlled valves for a refreshable tactiledisplay,” in Proc.
World Haptics Conf., 2015, pp. 177–182
- [11] Haga Makishi W,. Iwami, K. Totsu, K. Nakamura, and Esashi M.
Dynamic braille display using sma coil actuator and magnetic
latch,” Sensors Actuators A, Phys., vol. 119, no. 2, pp. 316–322,
2005.

- [12] Liang S H., “Mechanism design of the flapper actuator in chinese braille display,” *Sensors Actuators A: Phys.*, vol. 135, no. 2, pp. 680–689, 2007.
- [13] Loconsole C., Leonardis M Gabardi D A. Frisoli, “Braillecursor: An innovative refreshable braille display based on a single slidin actuator and simple passive pins,” in *Proc. IEEE World Haptics Conf.*, 2019, pp. 139–144.
- [14] Karastoyaov V, Kotev D and Yatchev I, “Development of a braille tactile device driven by linear magnet actuators,” *Int. J. E Innov. Technol.*, 2014, pp. 1–4.
- [15] Vidal-Verdu F and Navas-Gonzalez R., “Thermopneumatic actuator for tactile displays,” in *Proc. 18th Conf. Des. Circuits Integr. Syst.*, 2003, pp. 629–633.



Plagiarism Checker X Originality Report

Similarity Found: 19%

Date: Friday, April 16, 2021

Statistics: 898 words Plagiarized / 4698 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

CHAPTER 1 INTRODUCTION EMBEDDED SYSTEM: An implanted framework is a regulator modified and constrained by an ongoing working framework (RTOS) with a committed capacity inside a bigger mechanical or electrical framework, frequently with continuous utilization of installed frameworks processing limitations. It is implanted as a feature of a total gadget frequently including equipment and mechanical parts.

Inserted frameworks control numerous gadgets in like manner use today. 98% of all chip are fabricated to fill in as installed framework segment. Instances of properties of common implanted PCs when contrasted and universally useful partners are low force utilization, little size, tough working reaches, and low per- unit cost. This comes at the cost of restricted preparing assets, which make them altogether more hard to program and to connect with.

Notwithstanding, by building knowledge components on top of the equipment, exploiting conceivable existing sensors and the presence of an organization of inserted. An installed framework is a regulator modified and controlled by an ongoing framework (RTOS) with a dedicated capacity within a larger mechanical or electrical framework, regularly with the continuous use of inserted frames that record imperatives.

It is installed as a component of a total gadget that regularly includes equipment and mechanical parts. The implanted frames control numerous devices in a similar way today. 98% of all microchips are manufactured to be filled as an installed frame segment./ Fig 1.1 Embedded system Instances of properties of average implanted PCs when contrasted and universally useful partners are low force utilization, little size, rough working reaches, and low per-unit cost. This comes at the cost of restricted handling assets, which make them essentially more hard to program and to cooperate

with.

In any case, by building knowledge systems on top of the equipment, exploiting conceivable existing sensors and the presence of an organization of implanted units, one can both ideally oversee accessible assets at the unit and organization levels just as give increased capacities, past those accessible. For instance, clever methods can be intended to oversee power

Classification of Embedded system Advanced air conditioning systems use mesh thermostat systems to control temperature more accurately and efficiently, which can change depending on the time of day and season. Home automation uses telecommunications networks that can be used to control lights, climate, security, sound/visual surveillance, etc.,

all of which use built-in sensors and control. Like traffic lights, works controllers, principally composite systems such as mixture vehicles, MRI and united flight systems range from mobile devices such as digital clocks and MP3 players to large fixed fittings. Difficulty varies from low, with an true controller

BLOCK DIAGRAM OF AN EMBEDDED SYSTEM: The built-in system usually has an embedded processor. Many devices that have microwave digital interface, VCRs, and auto take advantage of embedded systems. Some built-in systems include an operating system. Others are very specialized, leading to the entire logic being implemented as a single programme.

These systems are included in some devices for a specific purpose other than providing computing for general purposes. / Fig 1.3 **Block diagram of a typical embedded system**

EMBEDDED SYSTEMS APPLICATIONS: Systems embedded in auto car control, cruise control, body safety, engine safety, assembly line robots, car multimedia, car entertainment, E-com access, mobile phones, etc.

5 (Systems integrated into telecommunications include networks, mobile computing, wireless communications, etc. Smart card systems include banking, telephone and security systems. Built-in satellite and rocket systems include defense, communications and space systems embedded in networks and peripherals including image processing, network systems, printers, network cards, screens, and displays systems integrated into digital consumer electronics including set-top boxes, DVDs, highdefinition TVs, and digital cameras.

CHAPTER 2 LITERATURE SURVEY Braille Character Credit Based on Neural Networks

Braille is that the preferred system wont to act between visually impaired and clear-sighted folks exploitation tactile suggests that. Optical Braille Recognition (OBCR) includes 2 main steps: Braille cell recognition (image acquisition, pre-processing, braille purpose recognition, braille cell recognition and segmentation) and a duplicate of Braille cells akin to linguistic communication symbols.

the instance of the system is formed exploitation imaging techniques associated an approach to artificial neural networks. These ways permit you to realize a high level of speed and accuracy of recognition. The system will adapt to factors like the standard of input models and also the variations between them dynamically.

this text developed a synthetic neural network to spot pictures of the Cyrillic of the letter within the Braille illustration system. The network are going to be trained and tested to spot scanned Cyrillic letters . Braille Image Recognition for Beginners Braille pattern of alphabet consists of its own pattern within which a number of it doesn't relate equally to the alphabet.

Thus, it's tough for traditional folks to find and acknowledge the braille pattern. This paper presents a study of braille image recognition for beginners. the end result of this study is predicted to translate a braille image patterns into a clear alphabet text. a way of Bag of options (BOF) is projected for the popularity of the braille image.

On the opposite hand, the image classification is finished employing a Support Vector Machine (SVM) technique. Ixxviii or braille pictures has been tested. From the testing performed, it's found that ninety seven.44% of correct recognition accuracy has been achieved that discovered that the projected techniques square measure applicable for braille image recognition. Transportation are often associate employment barrier for persons with disabilities, significantly those with visual impairments.

A custom transportation intervention for folks with visual impairments, supported ideas related to custom employment, was devised, enforced, and evaluated. Methods: A protest and post-test intervention and comparison cluster style evaluated changes in participants' social problem-solving skills, transportation self-efficacy, and transportation data. Participants worked with a transportation organiser to set up and secure work-related transportation.

Surveys assessed participant satisfaction with the intervention. Results: The intervention cluster had considerably higher scores for social problem-solving skills at post-test when considering protest scores. The intervention cluster responded favorably to measures of satisfaction with the intervention.

Discussion: The custom transportation intervention had a positive result on **social problem-solving skills** and also the participants were glad with the intervention, though **the impact of** the intervention on employment outcomes remains unclear. CHAPTER 3 INVESTIGATION 3.1 EXISTING SYSTEM **Education is the** key to success in life, normal children learn from school teachers who see and communicate with each other.

For the visually impaired, student **reading and writing** is much more difficult due to visual impairment. The teaching tools **needed to teach** this blind student are special and require **a lot of** attention from the teacher. For the first time, learning a blind student, the character game number is taught with marbles and grooved **with the help of a** whiteboard.

Different models of marble arrangement for a grooved whiteboard represent different brands. For each character position, the teacher must reach each student and change the arrangement of marbles. This is harder and it's going to take a long time. 3.2 PROPOSED SYSTEM The projected system provides translation **for visually impaired people to read English characters.**

The system uses characters to vibrate vibration motors **for visually impaired people in the form of braille language** that communicates with users **through mini vibration motors representing each braille** glyph. The appropriate hardware is chosen to deploy the system. The system **consists of a single** braille and a mobile cell for speech recognition to get the characters as input.

The STM32 microcontroller is used as a processing device in this **project. In this** projected system it also **consists of a** replacement switch. **In case of** any emergency situation, blind people use this switch and **the alert message** is sent to the appropriate person through gsm 3.3 BLOCK DIAGRAM: 3.4 HARDWARE REQUIREMENTS: Arduino uno Bluetooth Module (HC-05) Relay Vibration motors. 3.5 SOFTWARE REQUIREMENTS: Arduino IDE, Embedded C.

Chapter 4 HARDWARE COMPONENTS DETAILS 4.1 Arduino Uno Arduino Uno 'r ett exakt styrkort baserat p. ATmega328 Den har 14 digital I/O-stift (varav 6 kan anv'ndas som PWM-utg'ngar), 6 analoga ing'ngar, 16 MHz keramiska grodor, USB-anslutning, eluttag, ICSP-huvud och . Inneh.ller allt du beh.ver f.r att st.dja den exakta drivrutinen.

Anslut den bara till en dator **med en USB-kabel** eller ge den en ACto-DC-adapter eller batteri f'r att komma ig'ng. In skiljer sig fr'n alla paneler ovan, eftersom den inte anv'nder USB-styrenhetschips i FTDI-serien. Ist-lllet har den Atmega16U2 (Atmega8U2

upp till R2-version) programmerad som i USB-adapter seriesII. Ono Council Review 2 har ett motstånd som drar HWB 8U2-linjen till marken, vilket gör det enkelt att montera i DFU-läge.

Sammanfattning 3 av styrelsen har följande nya egenskaper: In Mega 16U2 replace 8U2. "Ono" means one in Italian and is invited to celebrate the upcoming release of Arduino1.0. Uno and version 1.0 will have reference versions of Arduino and will proceed. The Ono is the latest in a series of USB Arduino panels and reference model for the Arduino platform.

For comparison with previous versions, see the index in the Arduino table.. POWER: The information voltage to the Arduino board when you are using an external power supply (instead of 5 volts of USB affiliation or other controlled power supply). You can supply voltage through this pin or, if you give voltage through the power outlet, access it through this pin. • 3V 3. A 3.3

volt supply supplied by the on-board regulator. Most of the limit current draw is 50 mA. • GND. Ground pins. / Fig 4.1 Arduino Uno diagram The ATmega328 has thirty two K (with zero.5 K used for the boot loader). It furthermore has two K of SRAM and one K of EEPROM knowledge and output every of the fourteen progressive pins in One are often used as knowledge or performance, victimization the Mode1() pin, the write(modernized), and also the progressive read() limits. They work on five volts.

every pin will offer or acquire a restriction of forty mA AND has an inner extraction resistance (disconnected consistent with traditional procedure) of 20-50kOhms. • Series: zero (RX) and one (TX). you'll not get (RX) and communicate (TX) TTL ordered data. These pins square measure associated with the corresponding pins on the ATmega8U2 USB-to-TTL serial chip. • External interruptions: two and 3 .

These pins square measure typically organized to trigger an issue during a occasional price, AN increasing or down edge, or AN adjustment within the estimate. See the append Interrupt() job for subtleties. • PWM: 3, 5, 6, 9, 10 and 11. Provides 8-digit PWM performance with the straightforward Write() job. • SPI: ten (SS), 11 (MOS) ... 13 (SCK). These pins support SPI utili map SPI • LED: 13.

there is AN inherent light-emitting diode associated with computerised pin thirteen. At the aim once the pin is HIGH price, the light-emitting diode is on, once the pin is LOW, it's off. The Uno has six straightforward data sources, named A0 through A5, all of that offer ten items of goal As a matter in reality , they live from ground to 5 volts, but it's conceivable to vary the higher end of their reach utilizing the AREF pin and so the

straightforward Reference () work.

Moreover, one or two of pins have specific usefulness: • TWI: A4 or SDA pin and A5 or SCL pin. Backing TWI correspondence utilizing the Wire library. There one or two of various pins on the board: • AREF. Reference voltage for direct sources of knowledge. Used with straightforward reference (). • Reset. Bring this LOW line to reset the microcontroller. usually you can't add a restart capture to the safeguards that block the one on the whiteboard.

confer the planning between Arduino pins and ATmega328 ports. the planning for the Atmega8, 168 and 328 is indistinguishable. Communication the Arduino One has many offices to speak to a laptop, another Arduino, or different microcontrollers. The ATmega328 provides consecutive UART TTL (5V) correspondence, that is offered on machined pins zero (RX) and one (TX).

AN ATmega16U2 on the board channels this consecutive correspondence via USB and appears as a virtual port to programming on the portable computer. The '16U2 code uses the quality USB COM drivers, and no external driver is required. In any case, on Windows, a .inf record is required. The Arduino programming joins associate current screen that grants direct content primarily based data to be delivered off and from the Arduino board.

The RX and state LEDs on the board will streak once data is being sent through the USB-to-consecutive chip and USB relationship with the portable computer (yet not for correspondence consecutive on pins zero and 1). A code Serial library worries consecutive correspondence on any of the Uno's high level pins. The ATmega328 what is more maintains I2C (TWI) and SPI correspondence.

The Arduino programming fuses a Wire library to develop usage of the I2C transport; see the documentation for nuances. For SPI correspondence, use the SPI library .PINCONFIGURATION: / Fig 4.2 Pin diagram of Arduino Uno 4.1.46 USB Overcurrent Protection: The Arduino Uno encompasses a resettable poly fuse that protects your all USB ports Mostly pc's from shorts and overcurrent.

even most computer give have their own internal safety like fuse provides an additional layer of protection, encase bigger than five hundred mA is applied to the USB port of computer, the fuse can break mechanically the overload is detached. 4.1.7. Applications Today the ATmega328 is usually employed in projects some automated systems wherever an easy with less-power, terribly low-priced Mc is required.

maybe the foremost used common operation for chip is on the popular is at 328 development, particularly the Arduino Uno and Arduino Nano models. 4.1.8 Physical Characteristics: Uno PCB's most length and breadth square measure a pair of .7 and 2.1 inches severally, with USB connectors and power jacks that transcend previous dimensions. Four screw holes permit the board to be connected to the surface or case.

Note that the space between digital pins seven and eight is a hundred and sixty miles (0.16"), not even a multiple of a distance of one hundred miles from different pins. 4.2 IC 7805 Voltage regulators square measure terribly communal in electronic circuits. it afford a relentless output voltage for a distinct input voltage.

In our project I used 7805 IC is associate painting IC regulator that results its application in most of the comes square measure been used. The name 7805 means that 2 that means, seventy eight within the sense +ve voltage regu and 05 within the sense it provides 5v as output, therefore ic 7805 can give a +5V output voltage 7805 could be a transformer ic. it's a member of 78xx series of mounted linear transformer ICs.

The voltage supply in an exceedingly circuit could have fluctuations and wouldn't offer the mounted voltage output. The transformer IC maintains the output voltage at a relentless worth. The name 7805 means that 2 that means, seventy eight within the sense +ve voltage regu and 05 within the sense it provides 5v as output, therefore ic 7805 can give a +5V output power provide. Capacitors of appropriate values are often connected at input and output pins relying upon the several voltage levels. 4.3

BLUETOOTH MODULE The BT module it's a v4 BT by default its thereforeme countersign by victimization that solely we'd like to attach primarily this wireless device we will use terribly simply each can't connect at a time one device inly doable there's limit speed and distance if u cross it device shut down so its speed is around a pair of .5 Gc it's a radio transiever information speed is 3mbps by basicall bT could be a cmos technology It modify development cycle.. / Fig 4.3

BLUETOOTH MODULE 4.4 Hardware Features Typical -80dBm sensitivity Up to +4dBm RF transmit power Low Power one.8V Operation ,1.8 to 3.6V I/O PIO management UART interface with programmable baud With integrated antenna With edge connective 4.5 code options Default baud rate rate: 38400, information bits:8, Stop bit:1,Parity:No parity, information control: has. Supported baud rate rate: 9600,19200,38400,57600,115200,230400,460800.

Given a rising pulse in PIO0, device are disconnected. Status instruction port PIO1: low-disconnected, high-connected; PIO10 and PIO11 are often connected to red and

blue light-emitting diode severally. once master and slave square measure paired, red and blue light-emitting diode blinks 1time/2s in interval, whereas disconnected solely blue light-emitting diode blinks 2times/s/ Fig: 4.4 INTERNAL PIN CONFIGUATION Automatically connects to the last device that is turned on by default.

Allow pairing devices to connect by default. PIN CODE:Automatic pairing of "0000" is the default auto-connection in 30 minutes if the connection area causes the connection to be lost. The HC-05 has two operating modes, one for transferring and receiving data from other devices, and the other for AT command mode where you can change the default device settings. This works with SPP, so it is very easy to pair the hc module with the microcontroller.

Wireless communication between the two microcontrollers communicates with laptop, desktop and mobile phone data logging applications...Consumer applications Wireless Robots Home Automation 4.6 Relay: A relay is an electrical mechanically switch by default it is in normally open basicall relay have 2 modes anamata the number one is normally open that means by default by apply sply coil of relay generate magnetic field so it comes no anamata like vise versa happen automatically operating voltages are 5 and 12 I used I my project single phase double throw relay with op voltage 5v. .

The maximum output current for the popular 555 timer IC is 200mA so these devices can supply relay coils directly without amplification. FigNo: 4.5 555 timer

The relay's switch connections are usually labelled COM, NC and NO: . 4.7 Power supply unit power supply is a device or asystem that supplies electrical or other types of energy to an output load or group of loads.

The term is most commonly applied to electrical energy supplies, less often to motorised ones, and rarely to others. Fig 4.6 Block diagram of power supply Fig 4.7 processing of power supply the power line from the ability line. The wein bridge section conv the alternating currentto DC. However, as you proceed during this chapter, you may learn that rhythmical dc isn't fascinating.

For this reason, a MESH section is employed to convert dc rhythmical into a purer and a lot of fascinating sort of dc voltage. Finally, the REGULATOR, will specifically what the name implies. It keeps the ability offer o/p at a persistent level despite major changes within the charge current. let's trace AN ac signal through the ability offer. At this time, you would like to visualize however this signal is modified in every space of the ability offer.

Later within the chapter, you will see however these variations unfold. A 115-volt ac

input is applied to the transformer's primary. Generally it's a transformer with it turns quantitative relation of 1:3. realize the output for this Xer by swelling the input voltage by the quantitative relation of turns within the primary to the quantitative relation of turns within the secondary the output of the rectifier are $11\sqrt{2}$, or approximately 173 v of rhythmic dc.

The filter section, a network of resistors, capacitors, or inductors, controls the increase and fall time of the varied signal; consequently, the signal remains at a constant dc level. you may see the filter method a lot of clearly within the discussion of the particular filter circuits. The output of the filter could be a signal of one hundred ten volts dc, with ac ripple riding on the dc.

the rationale for the low voltage (average voltage) are explained later during this chapter. The regulator maintains its output at a relentless 110-volt dc level, that is employed by the equipment. Simple 5Volts annotation for digital circuits Brief image of operation: offers outwell regulated 5Volts output, output current capability of a hundred mA Applications: physics devices, tiny laboratory annotation Power offer voltage: Unregulated DC 8-18V power offer Power offer current: required output current + five mA Component costs: Few bucks for the electronics components + the input electrical device price. 4.8

Vibration Motors: vibration motor I used LRA however primarily two vibratory motors area unit there. A linear resonant mechanism (LRA) internally tiny mass getting ready to a spring, that creates a force once driven. AN uncommon rotating mass vibration motor (ERM) uses a little unbalanced mass on a DC motor once it rotates it creates a force that interprets to vibrations.

specialists within the offer and style of vibration motors, you'll be able to realize our equipped motors in our product catalogue. you may realize innumerable extra info and guides to assist you perceive however vibration motors work / Fig no. 4.8 Vibration Motors Chapter 5 SOFTWARE REQUIREMENTS DETAILS 5.1

EMBEDDED C: Inserted C is most documented artificial language in programming field for making electronic contraptions. each processor utilised in electronic framework is expounded with put in programming. Implanted C programming assumes a vital half in acting specific capability by the processor.

In way of life we have a tendency to utilised varied electronic gadgets like telephone, garments washer, advanced camera, and then on These all-gadget operating depends on microcontroller that ar changed by inserted C. / Fig 3.19 Process of compilation The

Embedded C code written in higher than diagram is employed for blinking the junction rectifier connected with Port0 of microcontroller.

In embedded system programming C code is most well-liked over different language. because of the subsequent reasons: o Easy to know o High responsibility o Portability o Scalability . put in programming must incorporate all needed gismo drivers at aggregation time, and therefore the gismo drivers ar composed for the actual instrumentation.

the merchandise is exceptionally dependent on the central processing unit and specific chips picked. Most put in laptop programmers have at any rate a passing data on understanding schematics, and browsing data sheets for components to choose utilization of registers and correspondence framework. modification between decimal, hex and parallel is effective even as utilizing bit management.

net applications ar once in an exceedingly whereas utilised, despite the very fact that XML documents and different yield could be passed to a laptop for show. Document frameworks with organizers ar commonly missing as ar SQL data bases. Software development entails use of a cross compiler, that activates a comp. however produces workable code for the top device. Debugging needs use of associate degree soul, SWD.

code developers typically have access to the whole kernel (OS) ASCII text file . Size of the capability memory and RAM will shift basically. many frameworks run in sixteen KB of Flash and four KB of RAM with a central processing unit engaging at eight megacycle, completely different frameworks will equal up to date computers.[8] These house conditions result in additional work being exhausted C or put in C++, instead of C++.

deciphered dialects like BASIC (while e.g., optical phenomenon propellor will utilize ordered BASIC) and (Java Pine Tree State Embedded eight.3[9] is accessible for e.g., ARM Cortex-M4, Cortex-M7 microcontrollers and capable ARM11 used in Raspberry Pi and Intel stargazer info. 2) don't seem to be commonly utilized; whereas associate degree execution of the deciphered Python three language – small Python – is anyway accessible expressly for microcontroller use, e.g., 32-cycle ARM-based, (for example, BBC miniature: piece) and 16-bit PIC microcontrollers.

Correspondences among processors and between one processor and completely different components ar elementary. apart from direct memory tending to, regular conventions incorporate I²C, SPI, consecutive ports, and USB. Interchanges conventions meant to be used in inserted frameworks ar accessible as shut supply from organizations together with inhome Niche Technologies and CMX Systems.

ASCII text file conventions come back from uip, lwip, and others This program discloses the way to utilize structure within structure in C utilizing typical variable. understudy faculty detail structure is declared within understudy detail structure during this program. each style factors Please note that people from understudy faculty subtleties structure ar accessed by two dot operator and members of "student ar traditional structure variables. • _detail" structure is accessed by single dot(.) operator./ Fig 5.2

Embedded C program with examples A Keyword **could be a** uncommon word with a weird import to the compiler for instance, if we have a tendency to take the Keil's Compiler Cx51 the subsequent ar **a number of** the keywords: 1. bit 2. sbit 3. sfr 4. small 5. large These ar few **of the various** keywords related to the Cx51 compiling program in conjunction with the quality C Keywords. 5.2

ARDUINO IDE: The Arduino stage has gotten fine glorious with people merely starting with gadgets, and in light-weight of current circumstances. Not in the least like most past programmable circuit sheets, the Arduino needn't hassle with a distinct piece of apparatus (called a developer) to stack new code onto the board – **you'll be able to** basically utilize a USB link.

Moreover, the Arduino IDE utilizes associate degree improved on variety of C++, creating it less complicated to work out the way to program. At last, Arduino provides a customary structure issue that breaks out the weather of the miniature regulator into a additional open bundle. The Arduino instrumentation and programming were meant for craftsmen, architects, specialists, programmers, novices, and anybody keen on establishing intuitive things or conditions.

Arduino will collaborate with catches, LEDs, engines, speakers, GPS units, cameras, the web, and astonishingly your advanced cell or your TV! This ability joined with the manner that **the Arduino programming** is free, the instrumentation sheets ar extremely modest, associate degreeed each the merchandise and instrumentation don't seem to be tough to be told **has prompted a** vast native space of shoppers UN agency have contributed code and delivered tips for an vast assortment of Arduino-based undertakings There ar varied assortments of Arduino sheets (clarified on the subsequent page) **that may be** utilised for numerous functions.

many sheets look somewhat not an equivalent because the one at a lower place, but most Arduinos share most of those segments for all intents and purpose: As shown within the figure five.3 The Arduino IDE is unbelievably minimalistic, nevertheless it provides a near-complete setting for many Arduino-based comes. the highest menu bar

has the quality choices, together with "File" (new, load save, etc.), "Edit" (font, copy, paste, etc.), "Sketch" (for collection and programming), "Tools" (useful choices for testing projects), and "Help".

the center section of the IDE could be a easy text editor that wherever you'll be able to enter the program code. very cheap section of the IDE is devoted to associate degree output window that's accustomed see the standing of the compilation, what proportion memory has been used, any errors that were found within the program, and numerous different helpful messages./ Figure 5.3

ARDUINO WINDOW The 6 Buttons While more advanced projects will take advantage of the built-in tools in the IDE, most projects will rely on the six buttons found below the menu bar. /Figure 5.4 BUTTON BAR / Figure 5.5 SOURCE CODE WINDOW . CHAPTER 6
6.1 Braille method/The motor vibrating positions are identified by numbers from one to six There are 64 possible combinations, including no dots at all for a word space.

A cell can be used to represent a letter, digit, punctuation mark, or even a word. Braille literacy is a social-justice issue. 6.2 User interface When a user touch buttons or any menu options the phone will vibrate to help them to identify the buttons or the menu options. Additionally, when a user presses any button and perform any of the four general tasks, it will provide vibrational feedback to confirm users' input in the application.

/ Fig no 6.1 UI (Bluetooth Fig no 6.1 UI (Bluetooth app) 6.3 Blind person's user interface:
Fig no 6.2 six vibrating motors According to the 6-vibrating motors positions the end user can identify the letters from A to Z this device only Applicable for English Language.
6.4 OUTPUT/ Fig.no:6.3

output Chapter 7 CONCLUSION In our project, we have designed the prototype model for blind, deaf and dumb people by employing a single compact device. The important key factor of this project to facilitate these people and to fix them more confident to manage their sites by themselves. The primary advantage is that the device can be taken away easily and is of about less weight.

INTERNET SOURCES:

0% - Empty

1% - <https://www.sciencedirect.com/science/ar>
0% - <http://www.e2matrix.com/blog/2018/09/28/>
0% - <https://www.researchgate.net/publication>
0% - https://en.wikipedia.org/wiki/Embedded_d
1% - <https://www.sciencedirect.com/science/ar>
0% - <https://www.gangboard.com/blog/iot-inter>
1% - <https://www.sciencedirect.com/science/ar>
1% - <https://www.sciencedirect.com/science/ar>
0% - <https://www.electronicshub.org/temperatu>
0% - https://en.wikipedia.org/wiki/Daylight_s
0% - <https://vissicompocoder.wordpress.com/ta>
0% - <https://www.bcu.ac.uk/engineering/resear>
0% - <https://patents.google.com/patent/US2009>
0% - <https://www.slideshare.net/Rajendraprssd>
0% - <https://www.quora.com/What-is-the-import>
0% - https://en.wikipedia.org/wiki/Embedded_s
0% - <https://bmcmmedinformdecismak.biomedcentr>
0% - <https://www.slideshare.net/vishalbabugar>
0% - <https://patents.google.com/patent/US2002>
0% - <https://www.ijcsi.org/papers/IJCSI-10-6->
0% - <https://www.researchgate.net/publication>
0% - <https://www.investopedia.com/terms/n/neu>
0% - <https://www.researchgate.net/publication>
0% - <https://citeseerx.ist.psu.edu/viewdoc/do>
0% - <https://scips.worc.ac.uk/subjects-and-ch>
0% - <https://www.researchgate.net/publication>
0% - <https://hcis-journal.springeropen.com/ar>
0% - <https://www.researchgate.net/publication>
0% - <https://iisc.academia.edu/RamakrishnanAn>
0% - <http://www.diva-portal.org/smash/get/div>
0% - <https://www.un.org/development/desa/disa>
0% - <https://dokumen.pub/hci-international-20>
0% - <https://trialsjournal.biomedcentral.com/>
0% - <https://www.hackerearth.com/community-ha>
0% - <https://www.questionpro.com/blog/program>
0% - <https://www.researchgate.net/publication>
0% - <https://files.eric.ed.gov/fulltext/EJ114>
0% - <https://academic.oup.com/her/article/27/>
0% - <https://bmcpublikealth.biomedcentral.co>
0% - <https://www.yourarticlelibrary.com/educa>

0% - <https://xbsoftware.com/blog/assistive-te>
0% - <https://www.accreditedschoolsonline.org/>
0% - <https://blogs.cornell.edu/rosescholarssp>
0% - <http://www.ijsrp.org/research-paper-1217>
0% - <https://acadpubl.eu/hub/2018-119-15/1/13>
0% - <https://www.loc.gov/nls/resources/blindn>
0% - <https://projectiot123.com/2019/12/25/dow>
0% - <https://www.slideshare.net/SolomonMutwir>
0% - <https://www.pantechsolutions.net/iot-pro>
0% - <http://www.seeab.se/SkolSamv/skolsamverk>
0% - <https://library.e.abb.com/public/5a20807>
0% - <https://blog.adafruit.com/2010/09/27/ard>
0% - <https://www.instructables.com/Arduino-Tu>
0% - <https://uploads.strikinglycdn.com/files/>
0% - <https://www.tomsonelectronics.com/blogs/>
0% - <https://www.ijariit.com/manuscripts/v5i2>
0% - <https://www.academia.edu/29713576/Rotary>
0% - <https://components101.com/microcontrolle>
0% - <https://www.slideshare.net/saibhaskar/de>
0% - <https://www.researchgate.net/publication>
0% - <https://www.electricaltechnology.org/201>
0% - <https://www2.mvcc.edu/users/faculty/jfio>
0% - <https://ukdiss.com/examples/wireless-ges>
0% - <https://www.slideshare.net/saibhaskar/de>
0% - <http://www.robotpark.com/Arduino-LEONARD>
0% - https://www.academia.edu/23159272/GPS_Tr
0% - <https://www.bing.com/aclk?ld=e8Gy0saWs4p>
0% - <https://www.slideshare.net/AshfaquulHaque>
0% - <https://www.ijets.in/Downloads/Published>
0% - <https://www.arduino.cc/en/Reference/Wire>
0% - <https://ukdiss.com/examples/blind-spot-w>
0% - <https://core-electronics.co.ke/product/a>
0% - <https://turbobobbicycleblog.wordpress.co>
0% - <https://dokumen.pub/advances-in-decision>
0% - <https://colinkarpfinger.com/blog/2010/th>
0% - <https://www.geeksforgeeks.org/program-to>
0% - <https://shelleysackier.com/tag/robin-got>
0% - <https://www.rroij.com/open-access/develo>
0% - <https://www.slideshare.net/sauravcool/th>
0% - https://en.wikipedia.org/wiki/Load_bank

0% - <https://www.instructables.com/Password-B>
0% - <https://www.feasycom.net/uploads/2017130>
0% - https://wiki.seeedstudio.com/Bluetooth_B
0% - <https://www.astanadigital.com/products/H>
0% - <https://www.slideshare.net/shashikumarve>
0% - <https://manualzz.com/doc/28443946/5-degr>
0% - <https://www.techlicious.com/how-to/how-t>
0% - <https://www.instructables.com/AT-command>
0% - <https://www.edgefxkits.com/blog/interfac>
0% - <https://www.slideshare.net/Aakashkumar27>
0% - <https://www.amazon.com/Shelly-Wireless-A>
0% - <http://www.learningaboutelectronics.com/>
0% - <http://ijettjournal.org/2015/volume-29/n>
0% - <https://www.coursehero.com/file/88389841>
0% - <https://www.academia.edu/36071211/CHALLE>
0% - <https://www.ecosophia.net/the-cosmic-doc>
0% - <https://www.hemponcape.com/weed-mississa>
0% - https://www.academia.edu/35453082/AC_Gen
0% - <https://www.researchgate.net/publication>
0% - <https://www.mdpi.com/rss/journal/process>
0% - <https://www.ti.com/lit/an/slue121/slue12>
0% - <http://www.lbenchindia.com/finalprojects>
0% - <https://in.yahoo.com/>
0% - https://en.wikipedia.org/wiki/Lemon_batt
0% - <https://www.allaboutcircuits.com/textboo>
0% - <https://blog.zorbtouch.com/how-devices-p>
0% - <http://www.leader-w.com/news/small-vibra>
0% - https://issuu.com/husqvarna_construction
0% - <https://journals.plos.org/plosone/articl>
0% - <https://www.slideshare.net/sumitvarshney>
0% - <https://iot.electronicsforu.com/content/>
0% - <https://www.researchgate.net/publication>
0% - <https://www.slideshare.net/digitalinasia>
0% - <http://lwn.net/Articles/829733/>
0% - <http://www.the8bitguy.com/2576/what-is-m>
0% - <https://link.springer.com/chapter/10.100>
0% - <https://www.electronicshub.org/basics-of>
0% - <https://arstechnica.com/civis/viewtopic>
0% - <http://www.ownwatches.co.uk/>
0% - <https://www.codeinwp.com/blog/sketch-alt>

0% - <https://ksvi.mff.cuni.cz/~dingle/2017/pr>
0% - <https://qualitysafety.bmj.com/content/12>
0% - <https://www.digikey.ca/en/maker/blogs/20>
0% - <https://sightscotland.org.uk/articles/in>
0% - <https://handwiki.org/wiki/Braille>
0% - https://www.wireshark.org/docs/wsug_html
0% - <https://latesttechnology-world.blogspot>
0% - <https://www.slideshare.net/RoshmiSarmah/>
0% - <https://www.slideshare.net/deresemelssie>
0% - <https://teachjewelry.com/rings/keep-ring>