**TITLE**

**TABLE OF CONTENTS**

**CHAPTER . NO** **CONTENTS** **PAGE NO**

**ABSTRACT**

**1 INTRODUCTION**

**2 LITERATURE SURVEY**

**3 EXISTING SYSTEM & PROPOSED SYSTEM**

**4 SYSTEM FUNCTION**

**5 HARDWARE REQUIERMENTS & SPECIFICATION**

**6**  **SYSTEM SPECIFICATION**

**7 RESULTS**

**8 CONCLUSION**

**9 REFERENCES**

**ABSTRACT**

* Skin diseases like melanoma are traditionally screened by a visual analysis of key features, such as the pigmentation and vascularity of the region of interest. Monitoring the changes of these features during follow-up imaging sessions is critical towards a correct medical diagnosis. This project proposes a framework to monitor these changes on the skin over time. The proposed framework utilizes the automated data collection from skin displacement flow implementation to detect the severity of spatial changes in the skin. These spatial changes are captured via the magnitude and direction of the vectors in the resultant displacement field. This change monitoring is tested for surface and sub-surface skin image data and color view analysis. if any changes in skin detected means solution recommended medicine can be sent to medical shop . The patient do not even need to visit the stores once they can access the home where the druggists upload their drug lists data to server. Smart phones are becoming cheaper and are most popular around the globe. Thus this environment friendly 'pen-paperless' Android based application can be an efficient solution to save the druggists' and patient' time by reducing the extra time for drug requisition. The solution has been designed, implemented and testing to show the effectiveness of the solution

**CHAPTER 1**

**INTRODUCTION**

* The average annual number of adults treated for any skin cancer (non-melanoma skin cancer and melanoma) increased from 3.4 to 4.9 million between 2010–2006 and 2017–2018 with the largest proportion representing keratinocytic tumors, while the average number treated for all other cancers increased from 7.8 to 10.3 million.
* Cutaneous melanoma and non-melanoma skin cancers are initially diagnosed based on visual attributes of a lesion or mole. Cutaneous melanocytic lesions metastasize aggressively and the tumor thickness becomes a critically important prognostic factor.
* The visual examination of such suspicious lesions leads to a clinical biopsy of the affected skin and the cross-section of the skin tissue is examined by pathological assessment. However, this diagnostic process is time-consuming, invasive and subjective to medical experience.

**CHAPTER 2**

**LITERATURE SURVEY**

**[1]. B.V.Dhandra, Shridevi Soma, Shweta Reddy, “Color Histogram Approach For Analysis Of Psoriasis Skin Disease”.**

In this paper, we present Image analysis system to diagnosis multiple skin disease using statistical parameter analysis. Statistical analysis is concerned with analysis of random data. This system is combo-model which is to be used to diagnosis multiple skin diseases at a time. The target skin diseases are skin cancer, psoriasis and dermatophytosis. The disease diagnosis and classification is built on statistical parameter analysis. Statistical parameters includes: Entropy, Texture index, Standard deviation, Correlation fact Depending on standard range of parameters skin disease is going to be diagnosis and classified.

**[2]. R. Siegel, D. Naishadhama, A. Jemal, “Cancer Statistics, 2017”CA: a cancer journal for clinicians.**

Skin image identification has become one of the most demanding and attractive research areas in the past few years. Colour histogram based characteristics are used to analyse and classify the psoriasis infected skin images in sequence to take the diagnostic measures . Skin images for cancers of different types are obtained from, of these images for BCC (Basal cell carcinoma), SCC (squamous cell carcinoma) and normal or harmless skin lesions are collected and database is created for testing purpose Next step in image processing is de-noising using wavelet tool. To remove low frequency or background noise from image, filtering is used. After de-noising the image, median filtering is applied to remove some hair like material from skin image, if present Thresholding is the simplest and most commonly used method of segmentation.

**[3]. T. Wadhawan, N. Situ, K. Lancaster, X. Yuan, G. Zouridakis, “SkinScan: A Portable Library for Melanoma Detection On Handheld devices.**

A support Vector Machine with RBF kernel is used for the classification of images. The experimental results gave the encouraging results in an initial attempt for identification of psoriasis infected skin images. Classification of psoriasis skin diseases and their severity will be carried out. Image processing method is implemented in MATLAB, for skin cancer detection. In this paper, online database of skin cancer images is used for testing the method

**[4]. Omar Abuzaghleh, Buket D. Barkana, Miad Faezipour, “SKIN cure: A Real Time Image Analysis System to Aid in the malignant melanoma Prevention and Early Detection.**

Buket D. have worked on real time system for the malignant melanoma prevention and early detection . In this system user is able to analyse captured image. System process the image and shows the notification for medical help. This system shows convincing results and accuracy.

**[5]. Ho Tak Lau, Adel Al-Jumaily, “Automatically early detection of skin cance**

Ho Tak Lau. Have worked on an automatic skin cancer classification. Available image is given to the system and it goes through different image processing procedure. Use full information is extracted from the image and then with the help of training and testing system classifies the image. Recognition accuracy of the neural network classifier is 90% Image analysis system to detect skin diseases. Our system capture image from standard database and put in to the system to inform the user for preventing the threats linked to skin diseases. More briefly, we present the Image analysis system to detect different skin diseases, where user will able to take images of different moles or skin patches. Our system will analyze and process the image and classifies the image to normal, melanoma, psoriasis or dermo case based extracting the image features.

**CHAPTER 3**

**EXISTING SYSTEM AND PROPOSED SYSTEM**

**3.1 EXISTING SYSTEM**

* In this system using image processing texture analysis of skin disease has been carried out.
* Using conventional camera the images are recorded and then developed.
* It requires lots of process for developing an image.
* Sometimes it may ended up with poor photo captures or images taken unintentionally.
* It also prompts too longing process.

**3.2 PROPOSED SYSTEM**

* Skin modeling aims to develop adaptive margins of skin detector. Fast detection using AI and microcontroller .
* The patient do not even need to visit the stores once they can access the drug to there home where the our project kit upload their drug lists data to server and they detect GPS location co-ordinates. Delivery to there patient home. Smart phones are becoming cheaper and are most popular around the globe.
* Here two MQ2 sensor is used for detect the wound smell for different stages . And ultrasonic patch used for healing the skin diseases and relay is used to activate the blue light , wireless device used for pair the physical device to android app. The wound smell is detected the SMS notification send to medical shop. And GPS live location send to the shop. They deliver the drug to the patient.

**CHAPTER 4**

**SYATEM FUNCTION**

**BLOCKDIAGRAM**

**HARDWARE SECTION:**



APP STRUCTURE :



**CIRCUIT DIAGRAM**

**CHAPTER 5**

**HARDWARE REQUIERMENTS & SPECIFICATION**

**5.1 ATMEGA 8A MICROPROCESSOR**

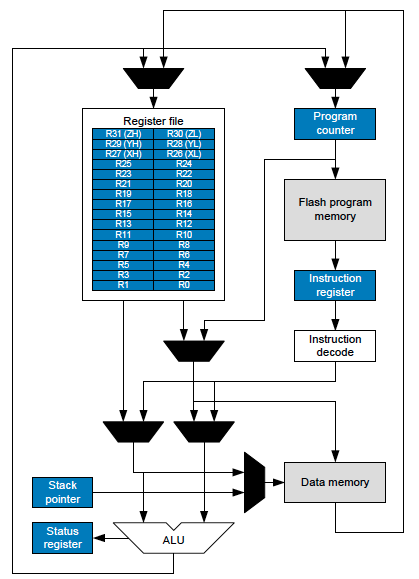
**INTRODUCTION**

The ATMEGA is a low-power, high-performance CMOS 8-bit Micro Controller with 4K bytes of in-system programmable Flash memory. The device is manufactured using high-density nonvolatile memory technology and is compatible with the industry-standard 8C51 instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the atmega is a powerful Micro Controller which provides a highly-flexible and cost-effective solution to many embedded control applications. The ATMEGA provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five vector two-level interrupts architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the ATMEGA is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

**THE MAJOR FEATURES OF 8-BIT MICRO CONTROLLER ATMEGA**

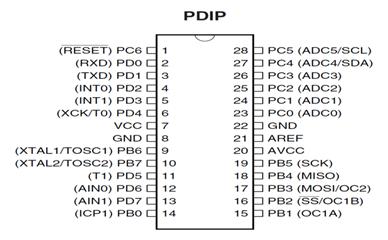
* 8 Bit CPU optimized for control applications
* Extensive Boolean processing (Single - bit Logic) Capabilities.
* On - Chip Flash Program Memory
* On - Chip Data RAM
* Bi-directional and Individually Addressable I/O Lines
* Multiple 16-Bit Timer/Counters
* Full Duplex UART
* Multiple Source / Vector / Priority Interrupt Structure
* On - Chip Oscillator and Clock circuitry.
* On - Chip EEPROM
* SPI Serial Bus Interface
* Watch Dog Timer

**ARCHITECTURE**



**Fig 3.3 Architecture of ATMEGA Microcontroller**

**PIN DESCRIPTION**

****

**Pin out diagram**

|  |  |
| --- | --- |
| **VCC** | Supply voltage. |
| **GND** | Ground |
| **Port 0** | Port 0 is an 8-bit open drain bidirectional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.  Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups.  Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification. |
| **Port 1** | Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current because of the internal pull-ups. Port 1 also receives the low-order address bytes during Flash programming and verification. |
| **Port 2** | Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current because of the internal pull-ups. Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that uses 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memories that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some control signals during Flash programming and verification. |

|  |  |
| --- | --- |
| **Port Pin** | **Alternate Functions** |
| P3.0 | RXD (serial input port) |
| P3.1 | TXD (serial output port) |
| P3.2 | INT0 (external interrupt 0) |
| P3.3 | INT1 (external interrupt 1) |
| P3.4 | T0 (timer 0 external input) |
| P3.5 | T1 (timer 1 external input) |
| P3.6 | WR (external data memory write strobe) |
| P3.7 | RD (external data memory read strobe) |

**RST**

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives High for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled**.**

**ALE/PROG**

Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory. If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high.

**PSEN**

Program Store Enable is the read strobe to external program memory. When the AT89S51 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

**EA/VPP**

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions. This pin also receives the 12-volt programming enables voltage (VPP) during Flash programming.

**XTAL1**

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

**XTAL2**

Output from the inverting oscillator amplifier.

**ADDRESSING MODES**

**DIRECT ADDRESSING**

In direct addressing, the operand specified by an 8-bit address field in the instruction. Only internal data RAM and SFR’s can be directly addressed.

**INDIRECT ADDRESSING**

In Indirect addressing, the instruction specifies a register that contains the address of the operand. Both internal and external RAM can indirectly address. The address register for 8-bit addresses can be either the Stack Pointer or R0 or R1 of the selected register Bank. The address register for 16-bit addresses can be only the 16-bit data pointer register, DPTR.

**INDEXED ADDRESSING**

Program memory can only be accessed via indexed addressing this addressing mode is intended for reading look-up tables in program memory. A 16 bit base register (Either DPTR or the Program Counter) points to the base of the table, and the accumulator is set up with the table entry number. Adding the Accumulator data to the base pointer forms the address of the table entry in program memory. Another type of indexed addressing is used in the“case jump” instructions. In this case the destination address of a jump instruction is computed as the sum of the base pointer and the Accumulator data.

**REGISTER INSTRUCTION**

The register banks, which contains registers R0 through R7, can be accessed by instructions whose op codes carry a 3-bit register specification. Instructions that access the registers this way make efficient use of code, since this mode eliminates an address byte. When the instruction is executed, one of four banks is selected at execution time by the row bank select bits in the PSW.

**REGISTER - SPECIFIC INSTRUCTION**

Some Instructions are specific to a certain register. For example some instruction always operates on the Accumulator, so no address byte is needed to point It. In these cases, the op code it points to the correct register. Instruction that refer to Accumulator as A assemble as Accumulator - specific op codes.

**IMMEDIATE CONSTANTS**

The value of a constant can follow the op code in program memory For example. MOV A, #100 loads the Accumulator with the decimal number 100. The same number could be specified in hex digit as 64h.

**PROGRAM STATUS WORD**

**Program Status Word Register in Atmega Flash Micro controller**

**CY AC F0 RS1 RS0 OV --- P**

PSW 7 PSW 0

PSW 6 PSW 1

PSW 5 PSW 2

PSW 4 PSW 3

**Fig 3.5 Formate of Program Status Word Register**

**PSW 0**

Parity of Accumulator Set by Hardware to 1 if it contains an Odd number of 1s, Otherwise it is reset to 0.

**PSW1**

User Definable Flag

**PSW2**

Overflow Flag Set By Arithmetic Operations

**PSW3**

Register Bank Select

**PSW4**

Register Bank Select

**PSW5**

General Purpose Flag.

**PSW6**

Auxiliary Carry Flag Receives Carry Out Form

**PSW7**

Carry Flag Receives Carry Out From Bit 1 of ALU Operands. The Program Status Word contains Status bits that reflect the current state of the CPU. The PSW has shown if Fig resides in SFR space. The PSW contains the Carry Bit, The auxiliary Carry (For BCD Operations) the two - register bank select bits, the Overflow flag, a Parity bit and two user Definable status Flags.

The Carry Bit, in addition to serving as a Carry bit in arithmetic operations also serves the as the “Accumulator” for a number of Boolean Operations .The bits RS0 and RS1 select one of the four register banks. A number of the instruction's register to these RAM locations as R0 through R7.The status of the RS0 and RS1 bits at execution time determines which of the four banks are selected.

The Parity bit reflects the Number of 1s in the Accumulator .P=1 if the Accumulator contains an even number of 1s, and P=0 if the Accumulator contains an even number of 1s. Thus, the number of 1s in the Accumulator plus P is always even. Two bits in the PSW are uncommitted and can be used as general-purpose status flags.

**INTERRUPTS**

The ATMEGA provides 5 interrupt sources: Two External interrupts, two-timer interrupts and a serial port interrupts. The External Interrupts INT0 and INT1 can each either level activated or transition - activated, depending on bits IT0 and IT1 in Register TCON. The Flags that actually generate these interrupts are the IE0 and IE1 bits in TCON. When the service routine is vectored to hardware clears the flag that generated an external interrupt only if the interrupt transition - activated. If the interrupt was level - activated, then the external requesting source (rather than the on-chip hardware) controls the requested flag. Tf0 and Tf1 generate the Timer 0 and Timer 1 Interrupts, which are set by a rollover in their respective Timer/Counter Register (except for Timer 0 in Mode 3). When a timer interrupt is generated, the on-chip hardware clears the flag that generated it when the service routine is vectored to.

The logical OR of RI and TI generate the Serial Port Interrupt. Neither of this flag is cleared by hardware when the service routine is vectored to. In fact, the service routine normally must determine whether RI or TI generated the interrupt the bit must be cleared in software. In the Serial Port Interrupt is generated by the logical OR of RI and TI. Neither of this flag is cleared by hardware when the service Routine is vectored in. In fact, the service routine normally must determine whether RI to TI generated the interrupt and the bit must be cleared in software the device.

**OSCILLATOR AND CLOCK CIRCUIT**

XTAL1 and XTAL2 are the input and output respectively of an inverting amplifier which is intended for use as a crystal oscillator in the pierce configuration, in the frequency range of 1.2 MHz to 12 MHz. XTAL2 also the input to the internal clock generator. To drive the chip with an internal oscillator, one would ground XTAL1 and XTAL2. Since the input to the clock generator is dividing by two flip flop there are no requirements on the duty cycle of the external oscillator signal. However, minimum high and low times must be observed. The clock generator divides the oscillator frequency by 2 and provides a tow phase clock signal to the chip.

The phase 1 signal is active during the first half of each clock period and the phase 2 signals are active during the second half of each clock period.

**OSCILLATOR CHARACTERISTICS**

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2.There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be

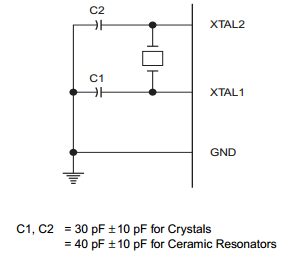
observed.

**Idle Mode**

In idle mode, the CPU puts itself to sleep while all the on chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset. It should be noted that when idle is terminated by a hard ware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to

the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle

should not be one that writes to a port pin or to external memory.

****

**Oscillator Connections**

**Power-down Mode**

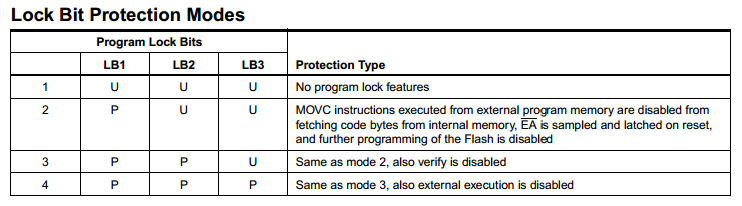
In the power-down mode, the oscillator is stopped, and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. Reset redefines the SFRs but does not change the

on-chip RAM. The reset should not be activated before VCC is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and

stabilize.

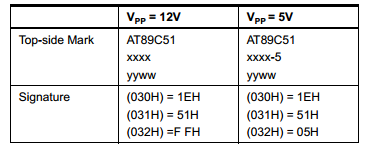
**Program Memory Lock Bits**

On the chip are three lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below.When lock bit 1 is programmed, the logic level at the EA pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value, and holds that value until reset is activated. It is necessary that the latched value of EA be in agreement with the current logic level at that pin in order for the device to function properly



**Programming the Flash**

The ATMEGA is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (VCC) program enable signal. The low-voltage programming mode provides a convenient way to program the ATMEGA inside the user’s system, while the high-voltage programming mode is compatible with conventional third party Flash or EPROM programmers. The ATMEGA is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

****

The ATMEGA code memory array is programmed byte-by byte in either programming mode. To program any nonblank byte in the on-chip Flash Memory, the entire memory

must be erased using the Chip Erase Mode.

**Programming Algorithm:**

Before programming the ATMEGA, the address, data and control signals should be set up according to the Flash programming mode table and Figure 3 and Figure 4. To program the ATMEGA, take the following steps.

1. Input the desired memory location on the address lines.

2. Input the appropriate data byte on the data lines.

3. Activate the correct combination of control signals.

4. Raise EA/VPP to 12V for the high-voltage programming mode.

5. Pulse ALE/PROG once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

**Data Polling:**

The ATMEGA features Data Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on PO.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

**Ready/Busy:**

The progress of byte programming can also be monitored by the RDY/BSY output signal. P3.4 is pulled low after ALE goes high during programming to indicate BUSY. P3.4 is pulled high again when programming is done to indicate READY .Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

**Chip Erase:**  The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/PROG low for 10 ms. The code array is written with all “1”s. The chip erase operation must be executed before the code memory can be re-programmed. Reading the Signature Bytes: The signature bytes are

read by the same procedure as a normal verification of locations 030H, 031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned areas follows.

(030H) = 1EH indicates manufactured by Atmega

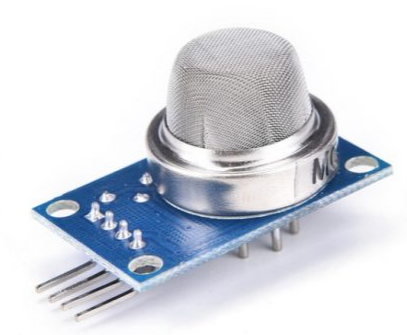
(031H) = 51H indicates 89C51

(032H) = FFH indicates 12V programming

(032H) = 05H indicates 5V programming.

**5.2 MQ2 SENSOR**

An MQ2 detector is a device which detects the presence of various gases within an area, usually as part of a safety system. This type of equipment is used to detect a gas leak and interface with a control system so a process can be automatically shut down. A gas detector can also sound an alarm to operators in the area where the leak is occurring, giving them the opportunity to leave the area.

****

**FEATURES**

* High sensitivity .
* Small sensitivity to MQ2, smoke.
* Fast response .
* Stable and long life
* Simple drive circuit.

**APPLICATION**

* They are used in gas leakage detecting equipments in family and industry, are suitable for detecting of LPG, natural gas , town gas, avoid the noise of MQ2 and cooking fumes and cigarette smoke.

**5.3 ULTRASOUND PATCH**

**INTRODUCTION**

With 2.5V - 5.5V power the LV-Max Sonar provides very short to long-range detection and ranging, in an incredibly small package. The LV-MaxSonarEZ1™detects objects from 0-inches to 254-inches.

(6.45-meters) and provides sonar range information from 6-inches out to 254-incheswith 1-inch resolution. Objects from 0-inches to 6-inches range as 6-inches. The interface output formats included are pulse width output, analog voltage output, and serial digital output.

**Features**

* Continuously variable gain for beam control and side lobe suppression
* Object detection includes zero range objects
* 2.5V to 5.5V supply with2mA typical current draw
* Readings can occur up to every 50mS, (20-Hz rate)
* Free run operation can continually measure and output range information
* Triggered operation provides the range reading as desired
* All interfaces are active simultaneously
* Serial, 0 to Vcc
* 9600Baud, 81N
* Analog, (Vcc/512) / inch
* Pulse width, (147uS/inch)
* Learns ring down pattern when commanded to start ranging
* Designed for protected indoor environments
* Sensor operates at 42KHz
* High output square wave sensor drive (double Vcc)

**Benefits**

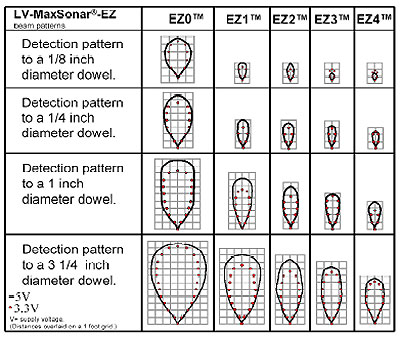
* Very low cost sonar ranger

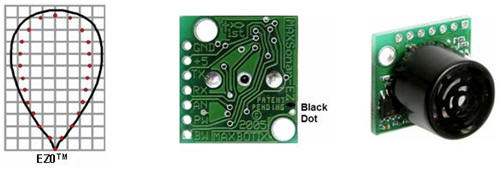
Reliable and stable range data

* Sensor dead zone virtually gone

Lowest power ranger

* Quality beam characteristics
* Mounting holes provided on the circuit board
* Very low power ranger, excellent for multiple sensor or battery based systems
* Can be triggered externally or internally
* Sensor reports the range reading directly, frees up user processor
* Fast measurement cycle
* User can choose any of the three sensor outputs



**5.4 BLUETOOTH**

**Introduction**

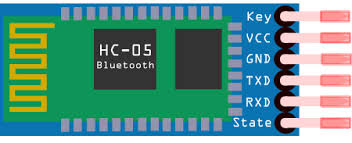
* It is used for many applications like wireless headset, game controllers, wireless mouse, wireless keyboard and many more consumer applications.
* It has range up to <100m which depends upon transmitter and receiver, atmosphere, geographic & urban conditions.
* It is IEEE 802.15.1 standardized protocol, through which one can build wireless Personal Area Network ([PAN](https://en.wikipedia.org/wiki/Personal_area_network)). It uses frequency-hopping spread spectrum ([FHSS](https://en.wikipedia.org/wiki/Frequency-hopping_spread_spectrum)) radio technology to send data over air.
* It uses serial communication to communicate with devices. It communicates with microcontroller using serial port (USART).

**HC-05 Bluetooth Module**

* HC-05 is a Bluetooth module which is designed for wireless communication. This module can be used in a master or slave configuration.

****

**Pin Description**



Bluetooth serial modules allow all serial enabled devices to communicate with each other using Bluetooth.

It has 6 pins,

1.  **Key/EN:** It is used to bring Bluetooth module in AT commands mode. If Key/EN pin is set to high, then this module will work in command mode. Otherwise by default it is in data mode. The default baud rate of HC-05 in command mode is 38400bps and 9600 in data mode.

HC-05 module has two modes,

          1.  **Data mode:**Exchange of data between devices.

          2.  **Command mode:**It uses AT commands which are used to change setting of HC-05. To send these commands to module serial (USART) port is used.

2.  **VCC:**Connect 5 V or 3.3 V to this Pin.

3.  **GND:**Ground Pin of module.

4.  **TXD:**Transmit Serial data (wirelessly received data by Bluetooth module transmitted out serially on TXD pin)

5.  **RXD:** Receive data serially (received data will be transmitted wirelessly by Bluetooth module).

6.  **State:**It tells whether module is connected or not.

**HC-05 module Information**

* HC-05 has red LED which indicates connection status, whether the Bluetooth is connected or not. Before connecting to HC-05 module this red LED blinks continuously in a periodic manner. When it gets connected to any other Bluetooth device, its blinking slows down to two seconds.
* This module works on 3.3 V. We can connect 5V supply voltage as well since the module has on board 5 to 3.3 V regulator.
* As HC-05 Bluetooth module has 3.3 V level for RX/TX and microcontroller can detect 3.3 V level, so, no need to shift transmit level of HC-05 module. But we need to shift the transmit voltage level from microcontroller to RX of HC-05 module.

**Bluetooth communication between Devices**

 E.g. Send data from Smartphone terminal to HC-05 Bluetooth module and see this data on PC serial terminal and vice versa.

To communicate smartphone with HC-05 Bluetooth module, smartphone requires Bluetooth terminal application for transmitting and receiving data. You can find Bluetooth terminal applications for android and windows in respective app. store.

**Bluetooth Module Serial Interface**

So, when we want to communicate through smartphone with HC-05 Bluetooth module, connect this HC-05 module to the PC via serial to USB converter.

Before establishing communication between two Bluetooth devices, 1st we need to pair HC-05 module to smartphone for communication.

**Pair HC-05 and smartphone**:

1. Search for new Bluetooth device from your phone. You will find Bluetooth device with “HC-05” name.
2. Click on connect/pair device option; default pin for HC-05 is 1234 or 0000.

After pairing two Bluetooth devices, open terminal software (e.g. Teraterm, Realterm etc.) in PC, and select the port where we have connected USB to serial module. Also select default baud rate of 9600 bps.

In smart phone, open Bluetooth terminal application and connect to paired device HC-05.

It is simple to communicate, we just have to type in the Bluetooth terminal application of smartphone. Characters will get sent wirelessly to Bluetooth module HC-05. HC-05 will automatically transmit it serially to the PC, which will appear on terminal. Same way we can send data from PC to smartphone.

|  |  |  |
| --- | --- | --- |
| **Command** | **Description** | **Response** |
| AT | Checking communication | OK |
| AT+PSWD=XXXX | Set Password  e.g. AT+PSWD=4567 | OK |
| AT+NAME=XXXX | Set Bluetooth Device Name  e.g. AT+NAME=MyHC-05 | OK |
| AT+UART=Baud rate, stop bit, parity bit | Change Baud rate  e.g. AT+UART=9600,1,0 | OK |
| AT+VERSION? | Respond version no. of Bluetooth module | +Version: XX OK  e.g. +Version: 2.0 20130107   OK |
| AT+ORGL | Send detail of setting done by manufacturer | Parameters: device type, module mode, serial parameter, passkey,etc. |

**Command Mode**

* When we want to change settings of HC-05 Bluetooth module like change password for connection, baud rate, Bluetooth device’s name etc.
* To do this, HC-05 has AT commands.
* To use HC-05 Bluetooth module in AT command mode, connect “Key” pin to High (VCC).
* Default Baud rate of HC-05 in command mode is 38400bps.
* Following are some AT command generally used to change setting of Bluetooth module.
* To send these commands, we have to connect HC-05 Bluetooth module to the PC via serial to USB converter and transmit these command through serial terminal of PC.

**Applications:**

1. Wireless communication between two microcontrollers

2. Communicate with Laptop, Desktops and mobile phones

3. Data Logging application

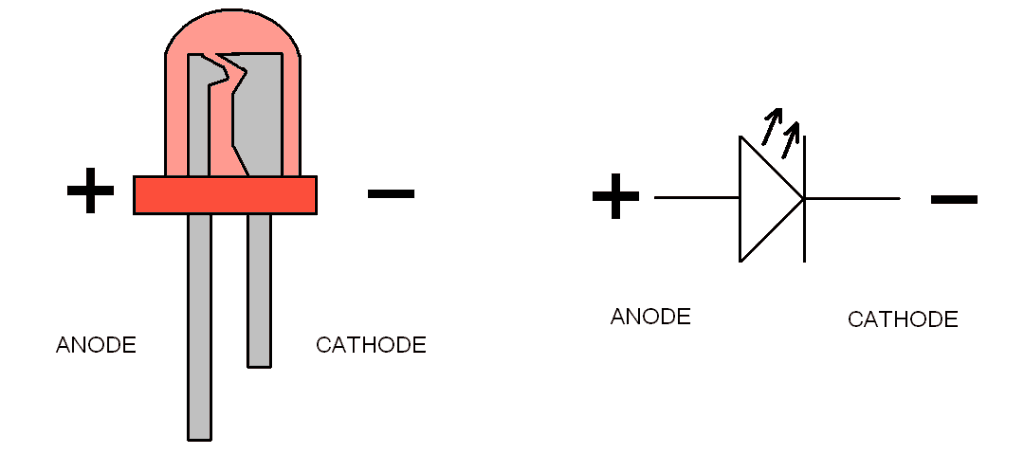
4. Consumer applications

5. Wireless Robots

6. Home Automation

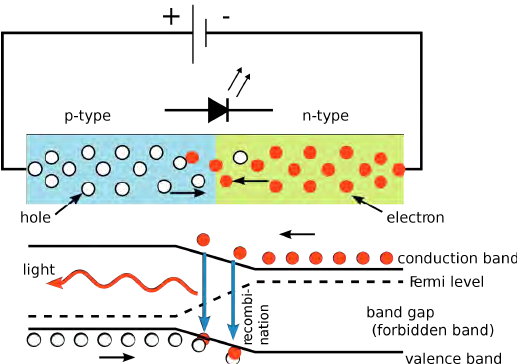
**5.5 BLUE LIGHT**

* A **light-emitting diode** (**LED**) is a two-lead semiconductor light source. It is a p–n junction diode that emits light when activated. When a suitable current is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons. This effect is called electroluminescence, and the color of the light (corresponding to the energy of the photon) is determined by the energy band gap of the semiconductor. LEDs are typically small (less than 1 mm2) and integrated optical components may be used to shape the radiation pattern.
* Appearing as practical electronic components in 1962, the earliest LEDs emitted low-intensity infrared light. Infrared LEDs are still frequently used as transmitting elements in remote-control circuits, such as those in remote controls for a wide variety of consumer electronics. The first visible-light LEDs were of low intensity and limited to red. Modern LEDs are available across the visible, ultraviolet and infrared wavelengths, with very high brightness.
* Early LEDs were often used as indicator lamps for electronic devices, replacing small incandescent bulbs. They were soon packaged into numeric readouts in the form of seven-segment displays and were commonly seen in digital clocks. Recent developments have produced LEDs suitable for environmental and task lighting. LEDs have led to new displays and sensors, while their high switching rates are useful in advanced communications technology.
* LEDs have many advantages over incandescent light sources, including lower energy consumption, longer lifetime, improved physical robustness, smaller size, and faster switching. Light-emitting diodes are used in applications as diverse as aviation lighting, automotive headlamps, advertising, general lighting, traffic signals, camera flashes, lighted wallpaper and medical devices .They are also significantly more energy efficient and, arguably, have fewer environmental concerns linked to their disposal.



**WORKING PRINCIPLE:**

* A P-N junction can convert absorbed light energy into a proportional electric current. The same process is reversed here (i.e. the P-N junction emits light when electrical energy is applied to it). This phenomenon is generally called electroluminescence, which can be defined as the emission of light from a semiconductor under the influence of an electric field The charge carriers recombine in a forward-biased P-N junction as the electrons cross from the N-region and recombine with the holes existing in the P-region. Free electrons are in the conduction band of energy levels, while holes are in the valence energy band. Thus the energy level of the holes is less than the energy levels of the electrons. Some portion of the energy must be dissipated to recombine the electrons and the holes. This energy is emitted in the form of heat and light.
* The electrons dissipate energy in the form of heat for silicon and germanium diodes but in gallium arsenide phosphide (GaAsP) and gallium phosphide (GaP) semiconductors, the electrons dissipate energy by emitting photons. If the semiconductor is translucent, the junction becomes the source of light as it is emitted, thus becoming a light-emitting diode. However, when the junction is reverse biased, the LED produces no light and—if the potential is great enough, the device is damaged.
* A 4x zoom photo of an ultra-violet LED has been shown in picture.
* Unlike a laser, the color of light emitted from an LED is neither coherent nor monochromatic, but the spectrum is narrow with respect to human vision, and for most purposes the light from a simple diode element can be regarded as functionally monochromatic.



**5.6 BUZZER**

### Buzzer Features and Specifications

* Rated Voltage: 6V DC
* Operating Voltage: 4-8V DC
* Rated current: <30mA
* Sound Type: Continuous Beep
* Resonant Frequency: ~2300 Hz
* Small and neat sealed package
* Breadboard and Perf board friendly

### Equivalents for Passive Buzzer

Piezo Electric buzzer, [Speaker](https://components101.com/misc/8-ohm-speaker), Active Passive Buzzer with Module

### How to use a Buzzer

A **buzzer**is a small yet efficient component to add sound features to our project/system. It is very small and compact 2-pin structure hence can be easily used on [breadboard](https://components101.com/misc/breadboard-connections-uses-guide), Perf Board and even on PCBs which makes this a widely used component in most electronic applications.

There are two types are buzzers that are commonly available. The one shown here is a simple buzzer which when powered will make a Continuous Beeeeeeppp.... sound, the other type is called a readymade buzzer which will look bulkier than this and will produce a Beep. Beep. Beep. Sound due to the internal oscillating circuit present inside it. But, the one shown here is most widely used because it can be customised with help of other circuits to fit easily in our application.

This buzzer can be used by simply powering it using a DC power supply ranging from 4V to 9V. A simple 9V battery can also be used, but it is recommended to use a regulated +5V or +6V DC supply. The buzzer is normally associated with a switching circuit to turn ON or turn OFF the buzzer at required time and require interval.

### Applications of Buzzer

* Alarming Circuits, where the user has to be alarmed about something
* Communication equipments
* Automobile electronics
* Portable equipments, due to its compact size

**CHAPTER 6**

**SYSTEM SPECIFICATION**

**6.1. Software Tools Used:**

Spyder software

**6.2. Hardware Tools**

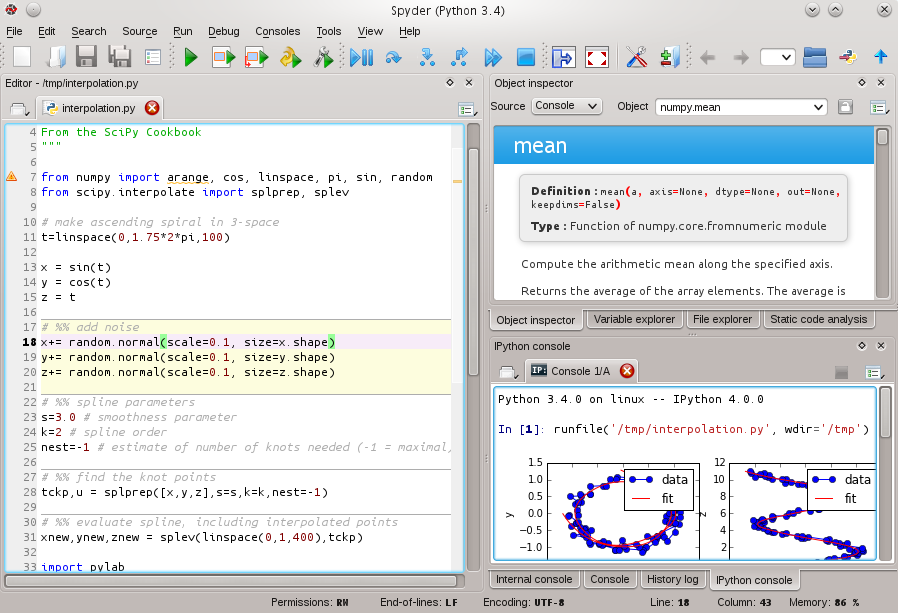
1. Atmega 8A microprocessor
2. Power supply
3. Mq2 sensor
4. Ultrasound patch
5. Bluetooth
6. Buzzer
7. light

**6.1 Spyder software:**

spyder (Scientific PYthon Development EnviRonment, previously known as Pydee) is a simple and lightweight, yet powerful interactive development environment for scientific programming in the Python language. This software is open source and cross-platform. It can be installed on Windows using Python(x,y), WinPython or Anaconda, on Mac OS via Anaconda or MacPorts. Spyder also can be integrated to widely used Linux distributions (Ubuntu, Debian, Fedora, OpenSuse, Gentoo, ArchLinux).



Spyder is useful and reliable Python software that features advanced editing, tools for data inspection, interactive testing and debugging. Also it embeds Python-specific code quality assurance and introspection instruments, such as Pyflakes, [Pylint](http://quintagroup.com/cms/python/pylint) and rope. This IDE integrates such tools as NumPy, SciPy, Matplotlib and [IPython](http://quintagroup.com/cms/python/ipython), as well as other open source software.  
Spyder is a part of spyderlib, a Python module based on PyQt4, pyflakes, rope and [sphinx](http://quintagroup.com/cms/python/sphinx) that provides robust PyQt4 or PySide widgets like source code editors, Python console or GUI-based dictionary, lists/tuples and NumPy array editors. Spyder offers MATLAB-like PYTHONPATH management dialog box (works with all consoles).

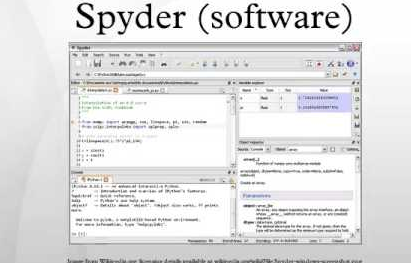


Since IDE is mainly directed on code writing and testing, a powerful editor is a key element of Spyder. The main editor’s features are:

* syntax coloring for Python, C/C++, Fortran;
* breakpoints and conditional breakpoints (debugger: pdb);
* powerful dynamic code introspection features (powered by rope), including code completion and calltips;
* integrated pylint code analysis;
* class and function browser;
* code outline explorer: functions, classes, if/else/try/... statements;
* occurrence highlighting;
* to-do lists (TODO, FIXME, XXX);
* errors and warnings with real-time code analysis (provided by pyflakes);
* opportunity to run a whole script or any portion of it from the editor;
* code completion and automatic link to documentation through the Object Inspector.

**Basic Console features:**

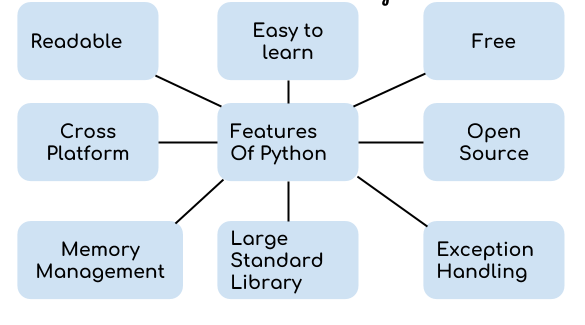
* all consoles are executed in a separate process;
* open Python interpreters or basic terminal command windows;
* many Python and Ipython consoles work simultaniously;
* Python interpreter;
* running Python script;
* User Module Deleter ( forces the Python interpreter to reload modules completely when executing a Python script);
* variable explorer: GUI-based editors for a lot of data types (numbers, strings, lists, arrays, dictionaries, etc.); import/export data from/to a lot of file types (text files, !NumPy files, MATLAB files); multiple array/list/dict editor instances at once, thus allowing to compare variable contents; data visualization.



# Introduction to Python Programming language

Python is developed by **Guido van Rossum**. Guido van Rossum started implementing Python in 1989. Python is a very simple programming language so even if you are new to programming, you can learn python without facing any issues.

## Features of Python programming language



1. **Readable:** Python is a very readable language.

2. **Easy to Learn:** Learning python is easy as this is a expressive and high level programming language, which means it is easy to understand the language and thus easy to learn.

3. **Cross platform:** Python is available and can run on various operating systems such as Mac, Windows, Linux, Unix etc. This makes it a cross platform and portable language.

4. **Open Source:** Python is a open source programming language.

5. **Large standard library:** Python comes with a large standard library that has some handy codes and functions which we can use while writing code in Python.

6. **Free:** Python is free to download and use. This means you can download it for free and use it in your application. See: Open Source Python License. Python is an example of a FLOSS (Free/Libre Open Source Software), which means you can freely distribute copies of this software, read its source code and modify it.

7. **Supports exception handling:** If you are new, you may wonder what is an exception? An exception is an event that can occur during program exception and can disrupt the normal flow of program. Python supports exception handling which means we can write less error prone code and can test various scenarios that can cause an exception later on.

8. **Advanced features:** Supports generators and list comprehensions. We will cover these features later.

9. **Automatic memory management:** Python supports automatic memory management which means the memory is cleared and freed automatically. You do not have to bother clearing the memory.

**6.2 IMAGE ENHANCEMENT**

Image Enhancement is the process of adjusting digital image so, that the result are more suitable for display or further image analysis. For example, you can remove noise, sharpen or brighten of digital images, making easier to identify key feature. Methods:

* + Filtering with morphological operation.
  + Histogram equalization.
  + Noise removal.
  + Linear contrast adjustment
  + Median filtering. Etc.
  1. **SEGMENTATION PROCESS**
* **Segmentation** partitions an image into distinct regions containing each pixels with similar attributes. To be meaningful and useful for image analysis and interpretation, the regions should strongly relate to depicted objects or features of interest. Meaningful segmentation is the first step from low-level image processing transforming a grayscale or color image into one or more other images to high-level image description in terms of features, objects, and scenes. The success of image analysis depends on reliability of segmentation, but an accurate partitioning of an image is generally a very challenging problem.
* Segmentation techniques are either contextual or non-contextual. The latter take no account of spatial relationships between features in an image and group pixels together on the basis of some global attribute, e.g. grey level or color. Contextual techniques additionally exploit these relationships, e.g. group together pixels with similar grey levels and close spatial locations.
* Image segmentation is a technique to determine the shape and size of the border. It separates the object from its background based on different features extracted from the image. After removing the noise and hair from the lesion area, the lesion needs to be separated from the skin, and therefore the analysis for diagnosis is conducted purely using the necessary area. There are a lot of segmentation methods possible for this study

**Thresholding**

* This method determines the threshold and then the pixels are divided into groups based on that criterion. It includes bi-level and multi thresholding. Thresholding method includes Histogram and Adaptive thresholding

**Color-based segmentation**

* algorithms Segmentation based on color discrimination. Include principle component transform/ spherical coordinate transform.

**Discontinuity-based segmentation**

* Detection of lesion edges using active contours / radial search techniques / zero crossing of Laplacian of Gaussian (LoG). It covers Active contours, Radial search & LoG 3.2.4 Region-based segmentation It is a method of splitting the image into smaller components then merging sub images which are adjacent and similar in some sense. It includes Statistical region merging, multi scale region growing, and morphological flooding. It is based on the techniques such as Split and merge Statistical Region Merging Multi-Scale Morphological flooding

**6.3**  **Edge Detection Technique**

* Edge detection is an image processing technique for finding the boundaries of objects within images.
* It works by detecting discontinuities in brightness.
* Filter is used to blur image and remove noise.
* It is used for image segmentation and data extraction in area such as image processing, computer vision.
* Ringworm is detected best using edge detection.

**6.4 Gray Scale**

* Grayscale is a range of monochromatic shades from black to white. It contains only shades of gray and no color.
* Grayscale values are represented as binary values as 0’s and 1’s.
* Grayscale images are composed of pixels represented by multiple bits of information, typically range from 2 to 8 bits or more.
* Grayscale measures the intensity of the light reflected from an area(dot) of plane surface and defines each pixel(picture element) as a byte.

****

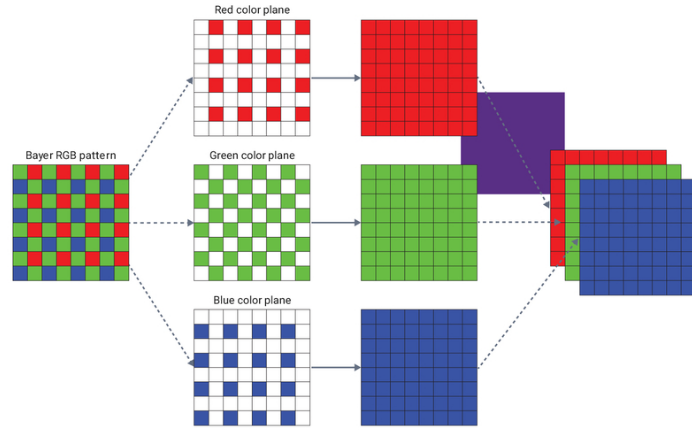
**6.5 Canny edge detector**

* Canny edge detector is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in images.
* 5 Different process in Canny edge detector are;
* Apply filter to remove noise.
* Finds intensity gradients of image.
* Apply non-maximum suppression to get rid of spurious response.
* Apply double threshold to determine potential edges.
* Track edge by hysteresis(combining all edges without break).

**Disease detection**

****

**6.6 RGB matrix classification**

****

Hence, in the case of a colored image, there are three Matrices (or channels) – Red, Green, and Blue. Each matrix has values between 0-255 representing the intensity of the color for that pixel.We have a colored image on the left (as we humans would see it). On the right, we have three matrices for the three color channels – Red, Green, and Blue. The three channels are superimposed to form a colored image.Note that these are not the original pixel values for the given image as the original matrix would be very large and difficult to visualize. Also, there are various other formats in which the images are stored. **Reading Image Data in Python**

Let’s put our theoretical knowledge into practice. We’ll fire up Python and load an image to see what the matrix looks like:

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |
| --- |
| import pandas as pd |
|  | import numpy as np |
|  | import matplotlib.pyplot as plt |
|  | %matplotlib inline |
|  | from skimage.io import imread, imshow |
|  |  |
|  | image = imread('image\_8\_original.png', as\_gray=True) |
|  | imshow(image) |

**CHAPTER 8**

**RESULTS**

**CHAPTER 9**

**CONCLUSION**

A Computer based skin disease detection system is proposed. The diagnosing methodology uses Digital Image Processing Techniques for the classification of infected skin. The unique features of the enhance images were extracted using HSV-histogram and SURF. Based on the features, the images were classified as infected skin and normal skin. This methodology has got good accuracy also. By varying the Image processing techniques and Classifiers, the precision can be improved for this system. Despite having some difficulty, these techniques are very helpful in medical science. The data we have collected will be helpful in medical field to see the clear image of the infected part in the skin .

**CHAPTER 10**

**REFERENCES**

[1] Z. Hu and C. S. Yu, “Functional research and development of skin barrier,” Chinese Journal of Clinicians, vol. 7, no. 7, pp. 3101–3103, 2017.

[2] V. Vezhnevets, V. Sazonov, and A. Andreeva, “A survey on pixel-based skin color detection techniques,” in Proceedings of Graphicon 2003, pp. 85–92, Moscow, Russia, September 2016.

[3] J. Oyola and V. Arroyo, “Detection of chickenpox vesicles in digital images of skin lesions,” in Proceedings of 17th Iberoamerican Congress on Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, pp. 583–590, Buenos Aires, Argentina, September 2018.

[4] D. H. Chung and G. Sapiro, “Segmenting skin lesions with partial differential-equations-based image processing algorithms,” IEEE Transactions on Medicine Image, vol. 19, no. 7, pp. 763–767, 2015.

[5] S. R. Yu, X. H. Zhao, and X. M. Pu, “Image characteristics of dermopathic herpesvirus disease under reflectance confocal microscope,” China Journal of Leprosy and Skin Diseases, vol. 31, no. 2, pp. 85–88, 2015.

[6] L. S. Zhong, X. Jin, and C. Quan, “Diagnostic applicability of confocal laser scanning microscopy in psoriasis vulgaris,” Chinese Journal of Dermatovenereology, vol. 25, no. 8, pp. 607-608, 2017.

[7] R. Sumithra, M. Suhilb, and D. S. Guruc, “Segmentation and classification of skin lesions for disease diagnosis,” Procedia Computer Science, vol. 45, pp. 76–85, 2018.

[8] R. Yasir, M. S. I. Nibir, and N. Ahmed, “A skin disease detection system for financially unstable people in developing countries,” Global Science and Technology Journal, vol. 3, no. 1, pp. 77–93, 2015.

[9] S. Arivazhagan, R. N. Shebiah, K. Divya, and M. P. Subadevi, “Skin disease classification by extracting independent components,” Journal of Emerging Trends in Computing and Information Sciences, vol. 3, no. 10, pp. 1379–1382, 2016.

[10] H. J. Niu, K. K. Shang, and Y. Liu, “Study of segmenting skin erythema images by reducing dimensions of color space,” Computer Engineering and Applications, vol. 13, no. 3, pp. 219–221, 2017.

[11] F. Liu and H. X. Guo, “Research progress of CT in skin diseases,” Chinese Journal of Dermatovenerol Integrated Traditional Western Medicine, vol. 313, no. 3, pp. 189–191, 2018.

[12] W. Luo, R. S. Meng, and R. K. Cai, “Application of CDIA system in measurement and research of vitiligo,” Chinese Medical Journal, vol. 22, no. 12, pp. 1059-1060, 2017.

[13] J. Lu, E. Kazmierczak, and H. Jonathan, “Automatic segmentation of scaling in 2-D psoriasis skin images,” IEEE Transaction on Medical Imaging, vol. 32, no. 4, pp. 719–730, 2018.

[14] J. A. Jaleel, S. Salim, and R. B. Aswin, “Artificial neural network based detection of skin cancer,” International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 1, no. 3, pp. 200–205, 2016.

[15] S. Salimi, M. S. Nobarian, and S. Rajebi, “Skin disease images recognition based on classification methods,” International Journal on Technical and Physical Problems of Engineering, vol. 22, no. 7, pp. 78–85, 2018.