SHRI VILEPARLE KELAVANI MANDAL'S
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Affiliated to the University of Mumbai)

Mini Project II

Gesture Controlled Robotic Arm

Project by:
VRUSHABH DESAI (60001140012)
KARAN DESAI (60001140010)
GEET GANDHI (60001140020)

PLOT NO U-15, JUHU SCHEME, BHAKTIVEDANTA SWAMI MARG, VILE PARLE (WEST), MUMBAI-56
TEL.: 26107010 / 26107461 FAX: 2619 4988 E-mail: info@djsce.ac.in Website: www.djsce.ac.in

INDEX

OBJECTIVE:

To design a minimum system using Atmega32A and interface gesture controlled robotic arm using inertia sensor (MPU 6050) using I2C communication Protocol.

APPROACH AND EMBEDDED SYSTEM DESIGN

In this project, we have used a sensor MPU-6050 and a controller Atmega-32A as our embedded platform. We used I2C communication between sensor and controller.

MPU6050:

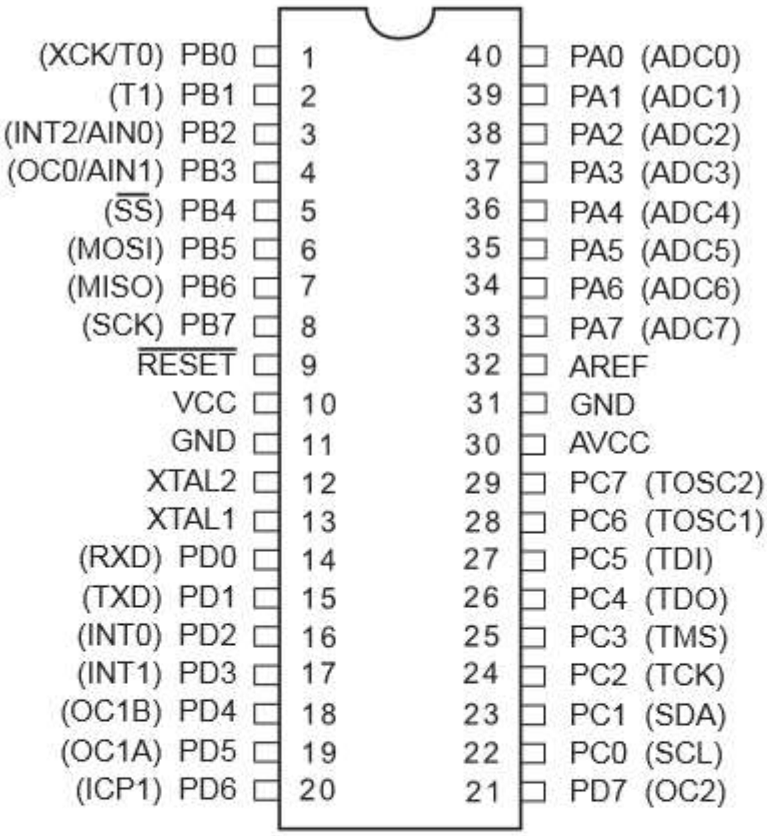
The MPU-6050 sensor contains a MEMS accelerometer and a MEMS gyroscope in a single chip (Micro Electro-Mechanical Systems technology). It is very accurate, as it contains 16-bits analog to digital conversion hardware for each channel. Therefore, it captures the X, Y and Z channel (axis) at the same time. It is used in movement detection. The output of the sensor is analog in nature. A prototype chair is implemented with a small chair and rpm motors are used to move the chair. This project is very much useful for the disabled.

The MPU-6050 is not expensive, especially given the fact that it combines both an accelerometer and a gyroscope. The MPU-6050 always acts as a slave to the Controller with the SDA and SCL pins connected to the I2C-bus. The accelerometer and gyro values are called the "raw" values.

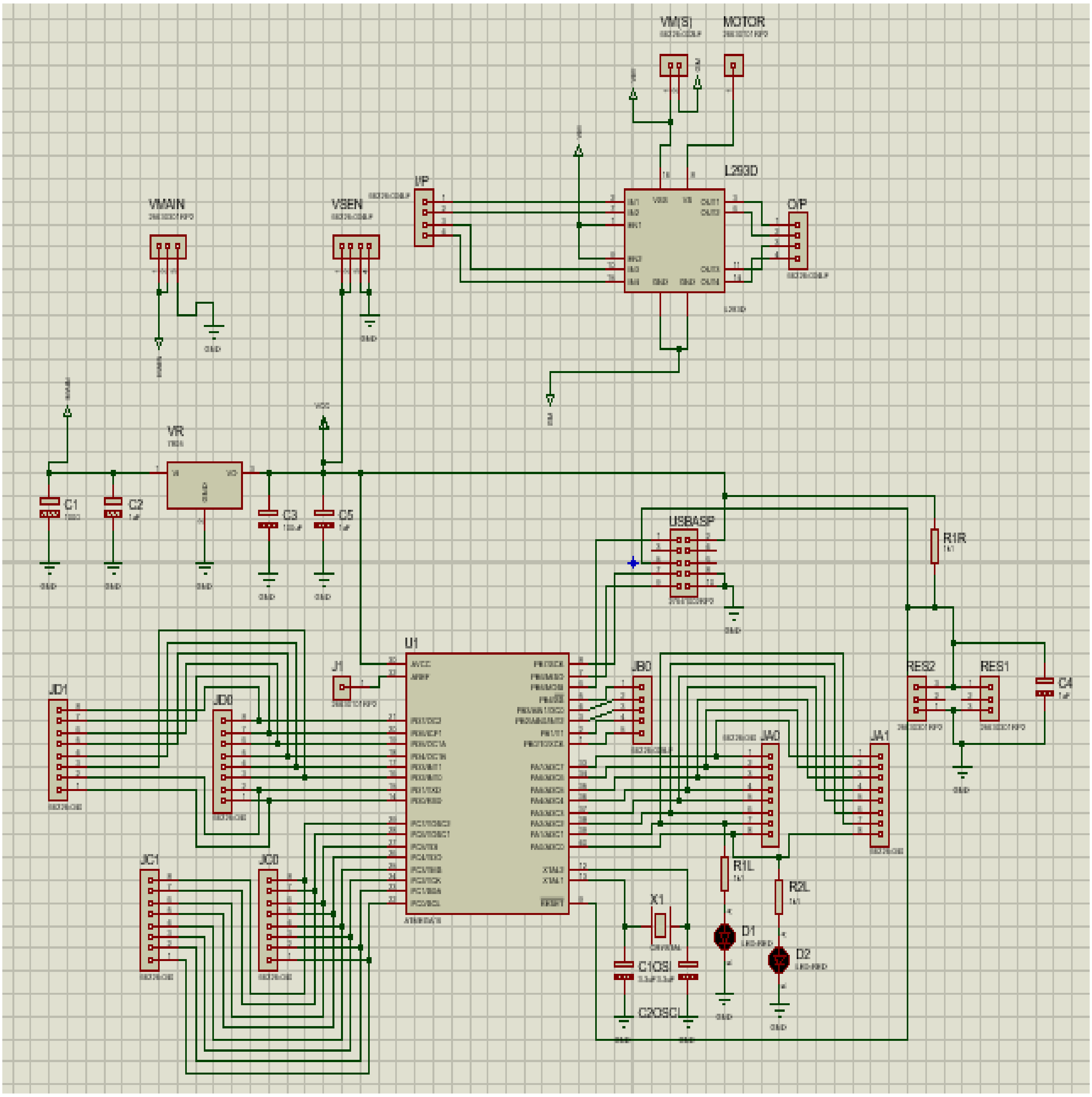
Microcontroller Atmega-32A:

The controller used in this project is Atmega-32A. Atmega-32A is a 8-bit microcontroller with 16K bytes in-system programmable flash. The ATmega32A is a low-power CMOS 8-bit microcontroller based on the Atmel AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega32A achieves throughputs approaching 1MIPS per MHz allowing the system designer to optimize power consumption versus processing speed. The ATmega32A is a powerful microcontroller that provides a highly flexible and cost-effective solution to many embedded control applications. The ATmega32A is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

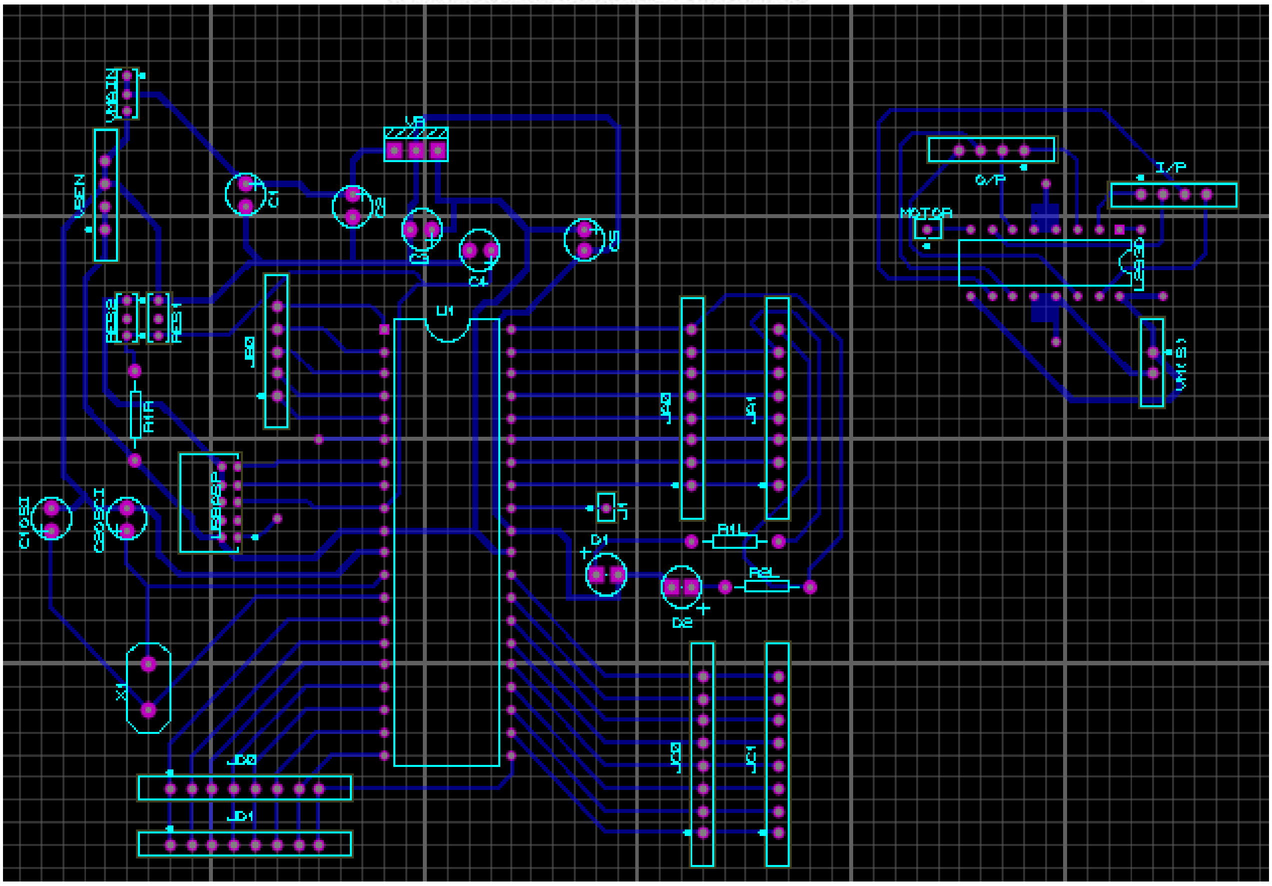
PDIP

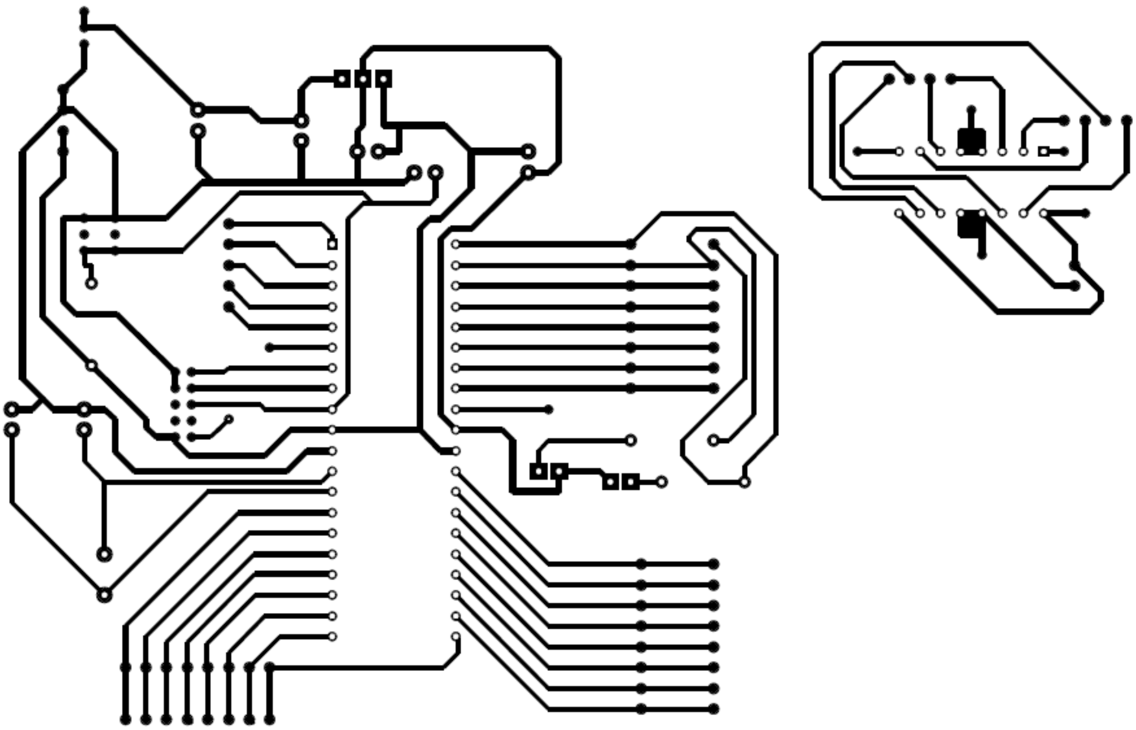


PCB Design using Proteus PCB Designing Tool:



Schematic

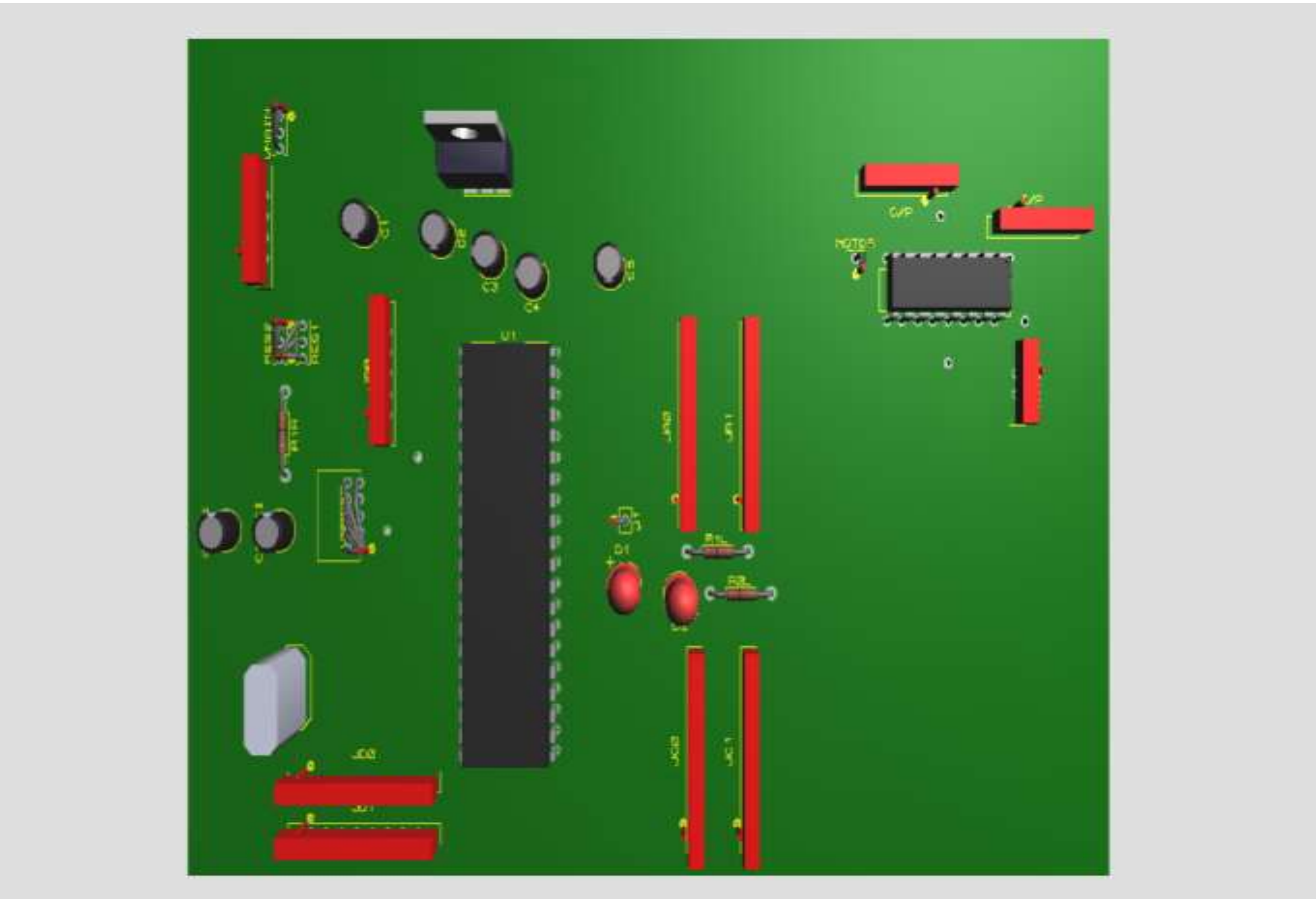

VMAIN
VGEN
C1
C2
C3
C4
C5
U1
JB0
RES2
RES1
R1A
USBASP
C1OSI
C2OSCI
X1
JA0
JA1
J1
D1
D2
R1L
R2L
JC0
JC1
JD0
JD1
O/P
I/P
MOTOR
L293D
VM(5)



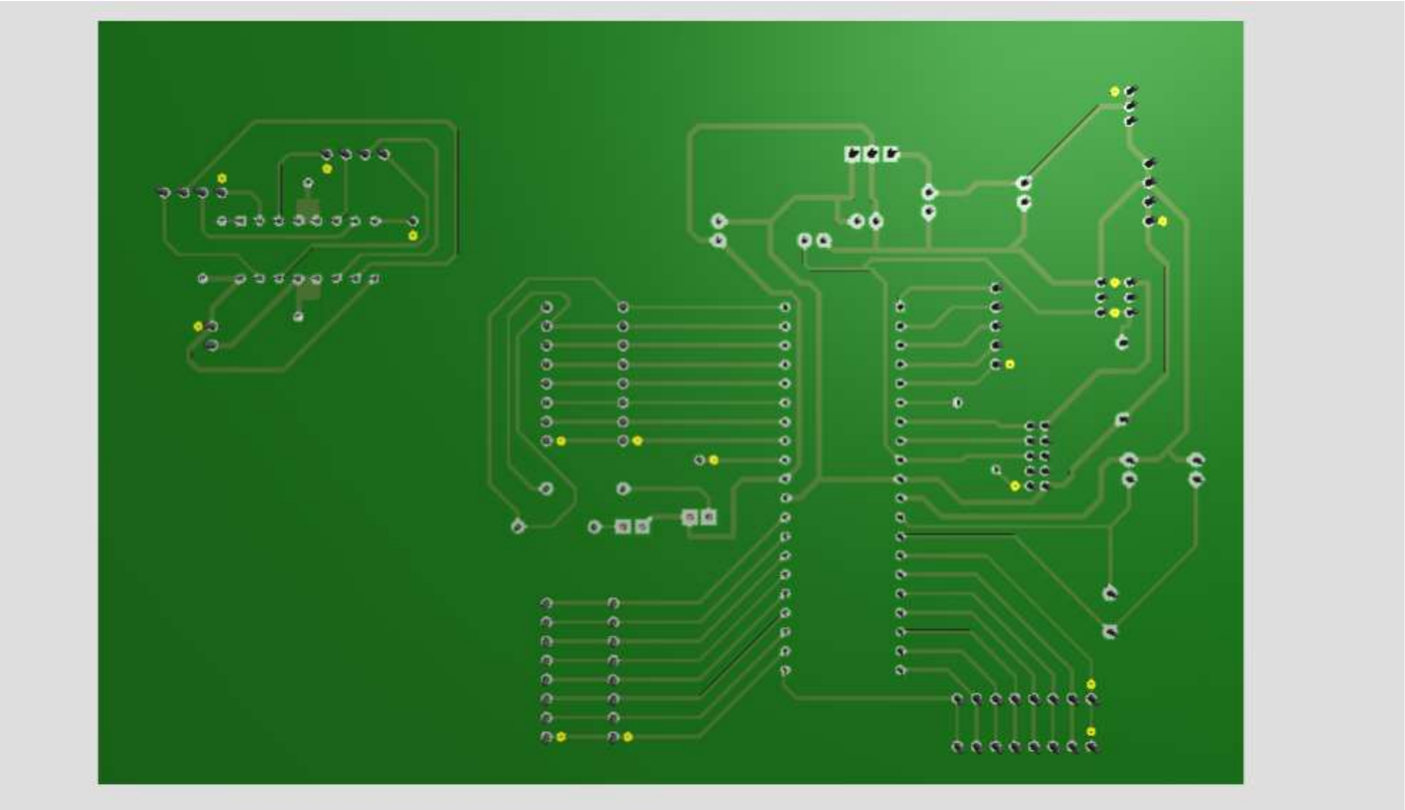
PCB Layout's

<u>Components used on Minimum System:</u>

1. AT Mega 32A
2. L293D (Motor Driver IC)
3. IC 7805 (Voltage Regulator)
4. Capacitor (1uF,100uF,22nF)
5. Crystal Oscillator
6. Reset Switch
7. Resistor (1K)
8. USBASP Connectors
9. Bug strip
10. Diode



Final PCB with Components (Front View)



Final PCB with Components (Back View)

Applications:

- Such types of robotic arms can be use in various types of applications as follows:
- Control of various functions of robots in arduous and dusty atmospheres industrial jobs as in painting shops, shot blasting chambers etc. The operator controls the robotic functions from outside the hazardous chambers looking through a glass door.
- Automatic picking of small objects (bottles, bags, tumblers etc.) moving on a conveyor and placing at other desired location in industries manufacturing various types of cosmetics, food products, medicines etc.
- Automatic metal cutting machines desire profiles, which are in high temperature zones.
- Advanced robotic toys operated with state-of-art hand operated control systems.
- Robots controlled cranes, lifting forks etc. operated from a distance with fingers/hands controlled remote system

APPROACH, LOOPHOLES AND POSSIBLE SOLUTIONS

In the very first step, while making a PCB, it is highly possible that some of the tracks get wiped off at some points. In addition, weak soldering can lead to terrible problems at a later stage.

We faced such situations multiple times while we were working in the project.

Solution: Always check the connectivity for each track using a digital multimeter and see to it that the soldering is done very efficiently as it is the foundation of your project.

Second, while designing the schematic we did not map the pins of USBASP in the correct order, we had to use female to male wires to connect the board with the USBASP, which is not desirable as it increases the possibility of errors because of connectivity issues and also it looks very untidy.

Solution: Design the schematic very precisely, such that all the pins are mapped properly, and no jumpers need to be soldered separately.

We had to make a whole new PCB board because of some faulty components. We could not figure out what was wrong with the board.

Solution: Always check all the components before you solder it on the board.

CONCLUSION

We have successfully implemented gesture control robotic arm which helped us understand I2C Communication and various factors considered while designing an embedded system. Also, we learned how to calculate degree of freedom of Robotic Arm.

REFERENCES/BIBLIOGRAPHY

WEBSITES:

[1] www.alldatasheets.com

[2]http://www.engineersgarage.com/contribution/accelerometer-based-hand-gesture-controlled-robot

[3] www.atmel.com

[4] www.allaboutcircuits.com

[5] www.howstuffworks.com