

# Motion Imitation Robotic Arm, (MIRA)

Akshay P. Shende<sup>1</sup>, Ravikiran S. Anande<sup>2</sup>

<sup>1</sup>Student of B.E. Electronics engineering, P.V.P.I.T, Budhgaon (Sangli), Maharashtra, India

<sup>2</sup>Assistant Professor, P.V.P.I.T, Budhgaon (Sangli), Maharashtra, India

\*\*\*

**Abstract** - In the recent technology advancements, employing robots for handling precise operations were seen in abundance like industrial, medical and rescue operations etc. Introducing human interaction with the robotic arm in the real-time scenario opens new doors in innovations and applications of the robotic arms. This paper reports the making and testing of a miniature prototype robotic controller arm using joystick which can be used to manually control an actual robotic arm that has 3 degrees of freedom. The main objective is to design the robotic arm, which can mimic the motion, robotic arm imitate as it does when the arm is operated by a human operator. This can be achieved with the help of simple potentiometers. The data is exchanged by the robotic arm and the main robotic arm with the help of joystick.

**Key Words:** servomotor, ATmega 328P, Driver PCA9685, robotic gripper, joystick

## 1. INTRODUCTION

Recent years brought an increasing interest into human-robot interaction, in particular in the context of service robotics. The immediate-future service robots will have to work among people, execute tasks that are assigned to them and exhibit similar motions to their human counterparts for a friendly and predictable interaction. In this context, one of the most important aspects is motion planning. Programming a robot to execute a prescribed movement is a very complex and tedious task.

The programmer has to establish the motor control sequences for every joint of the robot, as well as their interactions, while dealing with a large number of degrees-of-freedom simultaneously. This can be an extremely time-consuming process. Part of this burden can be eliminated by using intelligent approaches that allow a robot to imitate the movement.

We can plan the motion of robotic arm in three modes. In first mode, we can store the particular motion through programming in IDE arduino software and that program is upload in arduino board that gives motion which we stored in program. Second mode is manual mode, we can control the motion of robotic arm using joystick and third mode is the motion imitation in that we can do the motion for particular amount of time and that motion is uploaded on robotic arm.

## 2. COMPONENT SELECTION

**Table -1:** component

No.	Name of component	Number
1	ATmega 328P	1
2	Servomotor	5
3	Motor driver	1
4	Joystick	2
5	Gripper	1

### 2.1 System controller

The ATmega 328P is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega 328P achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed. The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single

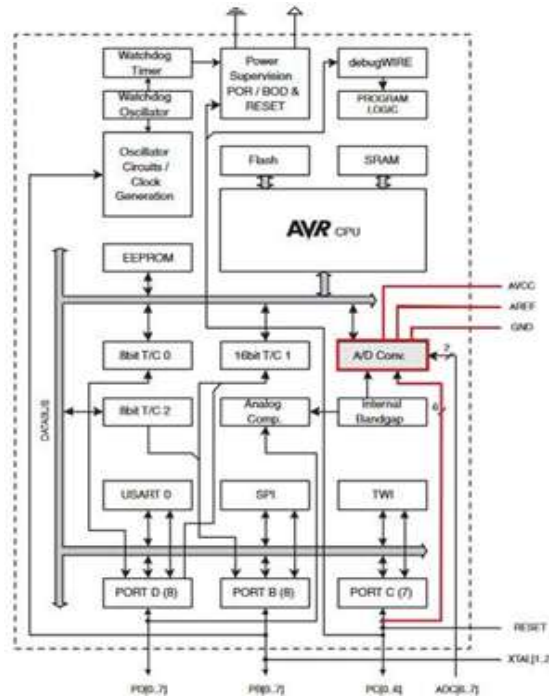


Fig -1: Microcontroller architecture

instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

## 2.2 Servomotor

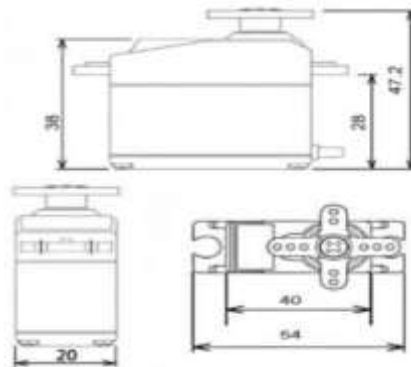


Fig-2: dimension of servomotor

This high-speed standard servo can rotate approximately 120 degrees (60 degree in each direction). We can use any servo code, hardware or library to control these servos, so it's great use for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. The MG995 Metal Gear Servo also comes with a selection of arms and hardware to get set up nice and fast. Position of servomotor is depending upon duty cycle.

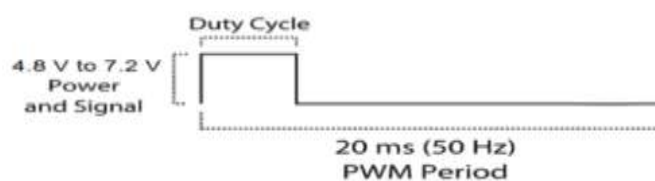


Fig -3: Duty cycle

## 2.4 Motor control module

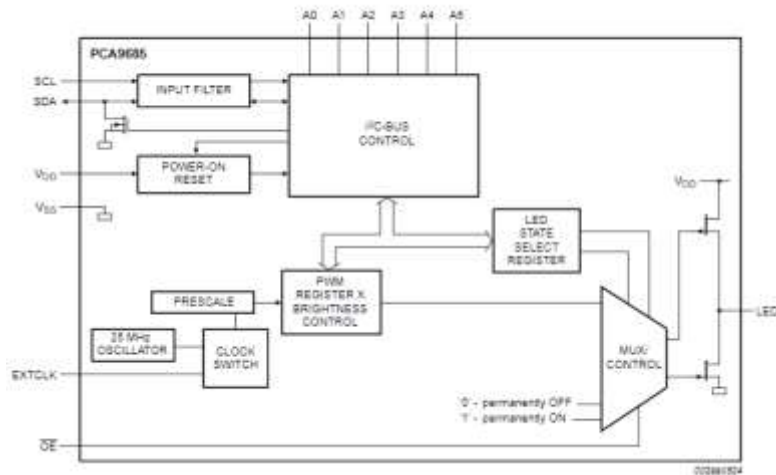


Fig -4: 3D design of arm

The PCA9685 is an I2C-bus controlled 16-channel servomotor controller, optimized for servomotor application. Each servomotor has its own fixed frequency individual PWM controller that operates at a programmable frequency from a typical of 24 Hz to 1526 Hz with a duty cycle that is adjustable from 0 to 100% to allow the servomotor to be set at specific value. All output can be set to the same PWM frequency. Each servomotor output can be set at its individual PWM value. The servomotor output driver is programmed to be either open-drain with a 25mA current sink capability at 5V or totem pole with a 25mA sink, 10mA source capability at 5V. The PCA9685 operates with a supply voltage range of 2.3 V to 5.5 V and the inputs and outputs are 5.5 V tolerant. The PCA9685 is in the new Fast-mode Plus (Fm+) family. Fm+ devices offer higher frequency (up to 1 MHz) and more densely populated bus operation (up to 4000 pF).

The PCA9685 has 4096 steps (12-bit PWM) of individual servomotor brightness control. The PCA9685 has an external clock input pin that will accept user-supplied clock (50 MHz max) in place of the internal 25 MHz oscillator. This feature allows synchronization of multiple devices. The PCA9635 does not have external clock input feature. The OE (output enable) can also be used to externally “pulse width modulates” the outputs, which is useful when multiple devices need to be controls together using software control. Software programmable servomotor all call and defined groups of PCA9685 devices to respond to a common I2C-bus address, allowing for example, all servomotor to be turned on or off at the same time or marquee chasing effect, thus minimizing I2C-bus commands. Six hardware address pins allow up to 62 devices on the same bus. The Software Reset (SWRST) General Call allows the master to perform a reset of the PCA9685 through the I2C-bus, identical to the Power-On Reset (POR) that initializes the registers to their default state causing the outputs to be set LOW. This allows an easy and quick way to reconfigure all device registers to the same condition via software.

## 2.4 Joystick

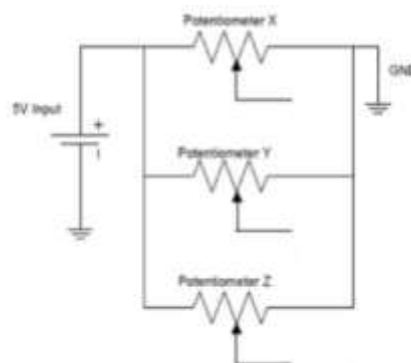


Fig -5: Joystick construction

The joystick module is similar to analog joysticks found in gamepads. It is made by mounting two potentiometers at a 90 degrees angle. The potentiometers are connected to a short stick centered by springs. This module produces an output of around 2.5V from X and Y when it is in resting position. Moving the joystick will cause the output to vary from 0v to 5V depending on its direction. If we connect this module to a microcontroller, you can expect to read a value of around 512 in its resting position (expect small variations due to tiny imprecisions of the spring and mechanism). When you move the joystick, you should see the values change from 0 to 1023 depending on its position. There are directional movements are simply two

potentiometers- one for each axis. It is compatible with Arduino interface. Dimensions are 1.57 in x 1.02 in x 1.26 in (4.0 cm x 2.6 cm x 3.2 cm). There are five pin for this module.

Pin configuration:

1. GND: ground
2. +5V:5V DC
3. VRx: voltage proportional to x position
4. VRy: voltage proportional to y position
5. SW: switch push button

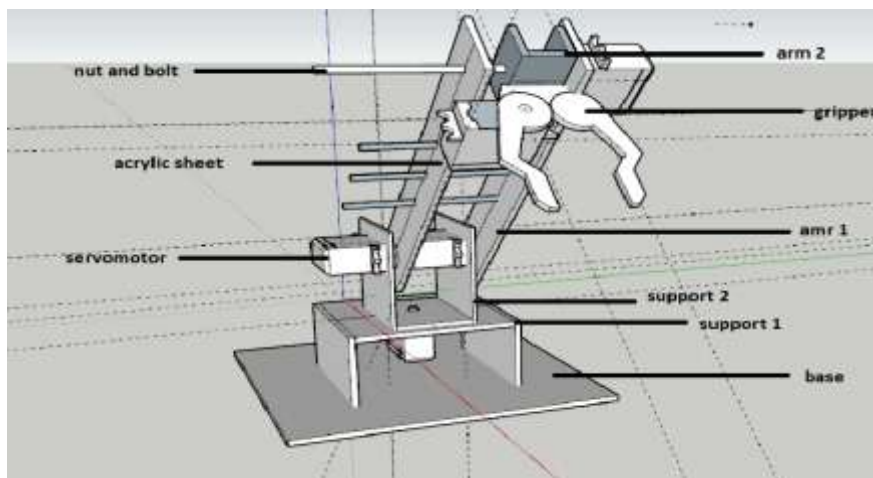
### 2.5 Gripper

A gripper is a motion device that mimics the movements of people, in the case of the gripper it is the fingers. A gripper is a device that holds an object so it can be manipulated. It has the ability to hold and release an object while some action is being performed. The fingers are not part of the gripper, they are specialized custom tooling used to grip the object and are referred to as “jaws”.

### 3. DESIGN METHODOLOGY

Design is most crucial part in this project because it contain a lot moving part and it just has to resize for motion what happening, first model is made up with metal sheet and it just unbalance the model and not work properly. Motors are not able to carry out its weight, so in second time it made with acrylic sheet to reduce weight dimensions also deduce up to section level.

First, we made a prototype on sketch up. Sketch is tool which we use to construct 3D model of robotic arm and we design arm according to our requirement in this tool. We use sketch tool in constructing arm because it is simple and easy to use. Following table shows dimensions of arm and they are show in table 2: Dimensions of arm.



**Fig -6:** 3D design of arm

**Table -2:** Dimensions of arm

Dimensions					
No.	Name	Length	Breath	height	unit
1	Base	20	20	0.1	cm
2	Support 1	10	10	7	cm
3	Support 2	7	4	7	cm
4	Arm 1 (Acrylic sheets)	23	4	0.4	cm
5	Arm 2	4	2	2	cm
6	Gripper	6	4.8	4.4	cm
7	servomotor	4	2	4	cm
8	Nut and bolt	0.4	0.4		cm

#### 4. Working Methodology

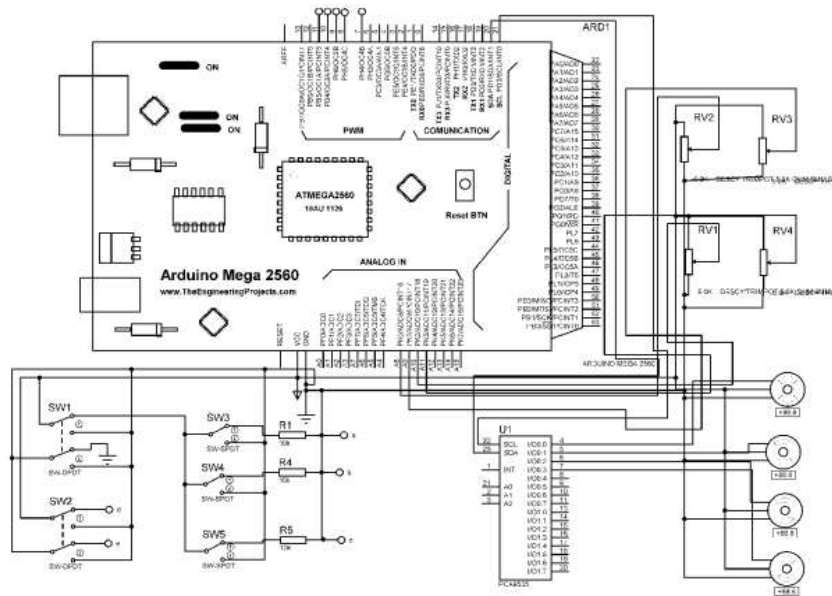


Fig -7: circuit diagram

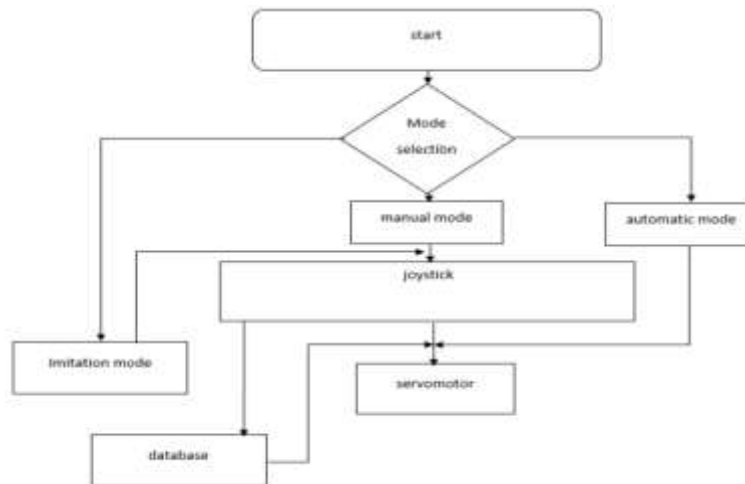


Fig -8: Flowchart

Circuit diagram shows the connection of all components. The circuit diagram is made by using proteus software. Proteus is software which is used to construct any circuit layout and it is very easy to use. Flowchart is use to understand the flow of code. Flowchart gives clear idea of how actual code is work and it is very useful at time of making code. The basic function of Motion Imitation Robotic Arm (MIRA) is done by its joints. Joints are analogous to human joints and are used to join the two consecutive rigid bodies in the robot. They have rotary joint. To add a joint to any link of a robot, we need to know about the degree of freedom and degree of movement for that body part. Degree of freedom implement the linear and rotational movement of body and degree of movement imply the number of axis the body can move. In our project, we built the robotic arm which has three degree of movement. The robotic arm consists of two rigid bodies on a fixed base, connected together with servo motor. The first rigid body is fixed and supports the second rigid body to which gripper is connected. The second rigid body is provided with movement in all three axes and has a three degree of freedom. It is connected to the first body with a servomotor. The gripper should accommodate three degree of freedom in order to take the component.

On a whole, the motion imitation robotic arm works as follows: For three degree of motion, we joint three servo-motor (one for each joint). The servomotor works on PWM signal which is provided by servo-motor driver PCA9685. The communication between the drive PCA9685 and Arduino is based on I2C protocol.



Working of project is carried out by three modes and they are as follow:

- 1) In automatic mode, fix program is store in arduino which do action according to the program. We have done programming in IDE software.
- 2) In manual mode, we control our arm motion with the help of joystick provides controlling signal to arduino. As per the controlling signal, arduino provide control signal to servomotor driver PCA9685 and driver provide PWM signal to servomotor. Our project consists of two joysticks. One joystick which is used for vertical and horizontal movement of robotic arm and another joystick used for controlling the motion of robotic gripper. Here robotic gripper is use to pick (hold) the object.
- 3) In motion imitation mode, we first do action for particular time constraint and then we imitate that action on arm. Here potentiometers are located in joystick are used to provide feedback loop. We keep the record of reading provided by potentiometer in arduino database. Here we provide dual pole dual throw switch for recording and imitating the motion, in first position, we do movement for particular amount of time and store motion. For storing purpose, we use array which is part of database then we switch to another position all recorded data feed to motor driver for copy motion.

## 5. RESULTS

After selection of mode, arm will configure in manual mode. Then it will follow the controlling signal of joystick and move to that particular position. In motion imitation first, we have to do action for particular time and after that it will continuously repeat that action again and again. For automatic mode, we will have following result (code is fix and we assign position to it). First it goes to position 1 and then move to position 2 and grabs an object, after that it will moves to position 3 and at last position 4 where it will put an object.



Fig-9: Position 1



Fig-10: Position 2



Fig-11: Position 3



Fig-12: Position 4

## CONCLUSION

In recent technology area, robot will use for so many applications instead of human. For precise and accurate result of work robot handle operations are better than human handling. From observation, it clearly shows that its movement is precise, accurate, easy to control and user friendly to use. The robotic arm has been developed successfully as the movement of robot can be controlled precisely. This robotic arm control method is expected to overcome this problem such as placing or picking object that away from the user, pick and place hazardous object in a very fast, safe and easy manner.

## ACKNOWLEDGEMENT

With profound sense of regard and gratitude, we thank Prof. RAVIKIRAN S. ANANDE for his invaluable guidance continuous interest and constructive suggestions during the course of project. This project wouldn't have been possible without his zeal and interest throughout task. We thank for his valuable and immense knowledge and timely help, which made this a reality. We feel gratified to record cordial thanks to our Head Prof. R.D.PATIL and their staff members of Electronics Department for their support, help and assistance which they extended as and when required.

We are deeply indebted to our Principal Dr.D.V.GHEWADE and the college for providing us with a platform to excel in my life. Last but by no means the least, we extremely grateful to our batch mates for their timely and constructive criticism.

## REFERENCES

- [1] Saugat Bhattacharyya, "Motor Imagery and Error Related Potential Induced position control of a Robotic Arm" IEEE/CAA JOURNAL OF AUTOMATICA SINICA, VOL. 4, NO. 4, OCTOBER 2017
- [2] Shih-An Li, Chung-Wei Weng, Yi-Hong Chen, "Servo motor controller design for robotic manipulator" 2012 IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2012) November 4-7, 2012
- [3] Kadirimangalam Jahnavi and Sivraj P, "Teaching and Learning Robotic Arm Model" 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)
- Sulabh Kumra, Rajat Saxena, Shilpa Mehta, "Design and Development of 6-DOF Robotic Arm Controlled by man machine interface" 2012 IEEE International Conference on Computational Intelligence and Computing Research
- [4] Neo Ee Sian, Kazuhito Yokoi, Shuuji Kajita, Fumio Kanehiro, Kazuo Tanie, "Whole Body Teleoperation of a Humanoid Robot Development of a simple Master Device using Joysticks" , Proceedings of the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems EPFL, Lausanne, Switzerland- October 2002
- [5] Muhammad Yaqoob, Sardar Rushan Qaisrani, Muhammad Waqas, Yasar Ayaz, Sohail Iqbal, Sajid Nisar, "Control of Robotic Arm Manipulator with Haptic feedback using programmable system on Chip", 2014 International Conference on Robotics and Emerging Allied Technologies in Engineering (iCREATE) Islamabad, Pakistan, April 22-24, 2014
- [6] Jimin Liang, Gong Zhang, "Development of two degrees of freedom deterministic parallel robotic arm unit", Proceedings of the 6th International Conference on Automation, Robotics and Applications, Feb 17-19, 2015, Queenstown, New Zealand
- [7] Pronadeep Bora, Vishwajit Nandi, "Low Cost Shadow Function based Articulated robotic arm", 978-1-4678-6503-1/15/\$31.00 © 2015 IEEE
- [8] Hairong Jiang, Juan P. Wachs, "3D Joystick for Robotic Arm Control by Individuals with high level spinal Cord Injuries" 2013 IEEE International Conference on Rehabilitation Robotics June 24-26, 2013 Seattle, Washington USA
- [9] G. Sen Gupta, S.C. Mukhopadhyay, C. H. Messom and S. Demidenko, "Master-Slave Control of a Teleoperated Anthropomorphic Robotic Arm with Gripping Force Sensing" IMTC 2005 – Instrumentation and Measurement Technology conference Ottawa, Canada, 17-19 may 2005.

## BIOGRAPHIES



**Akshay Prabhakar Shende**  
B.E. (ELECTRONICS)



**Ravikiran S. Anande**  
ASSISTANT PROFESSOR  
(B.E. ELECTRONICS)