Solar Based Robotic Arm Using Microcontroller

Vaibhav Kirnapure[1], Sukeshni Waghmare[2], Vaishnavi Khapre[3], Prajakta Kuhikar[4], Rina Nikalje[5], Prof. Nutan Moghe[6]

[1-5]Dept. Of Electrical Engineering, GWCET Nagpur, Maharashtra, India
[6]Professor, Dept. Of Electrical Engineering, GWCET Nagpur, Maharashtra, India

---***---

Abstract - *Integration of robotic arms into working tasks has currently increased magnificently in performing the very repetitive, dangerous or difficult tasks. Typically, a robotic arm is a mechanical arm that is programmable to mimic the behaviour of a human arm in terms of how it functions. Computers and microcontrollers have widely been used in the control of robotic arms with the help of sensors, levers, buttons, wireless devices, just to mention but a few. More advanced technology has lately revolutionized their control, ranging from the haptic technology using accelerometers to human-brain control through noninvasive technology. This research focused on the design, implementation and control of a robotic arm with five degree of freedom (DOF) using servo motors. It was designed to entirely operate by itself in a repetitive routine. A control circuit based on a PIC18F4550 microcontroller interfaced with a servo motor was built and a suitable software for the control of the rotation of motor developed.*

***Key Words*: Robotic Arm1, Micocontroller2, Sensors3, Accelerometers4, Servo Motor5, etc.**

1 INTRODUCTION

Robots are increasingly being integrated into performing repetitive, dangerous or difficult tasks. Robots can be used in many fields of operations including but not limited to; constructions of roads and buildings, performing repetitive tasks in industries, loading and offloading in transport, drilling, picking tools and minerals in mining, move cameras that capture videos from elevated, crowded and dangerous environment, install power cables and picking and placing medical equipment for easier surgery operations in hospitals. Fundamentally, robots should mimic human motion, and to enable them do so they have features which include foot and arms.

A robot possibly will take a sense of aptitude or imagination by itself through mimicking a natural look or systematizing actions. Robotic arms can be autonomous or manually controlled. Control of robotic arms has been done through computer terminals, joysticks, graphical user interface, accelerometers, sensors and tentatively interfacing them with internet so that they can be remotely controlled from any part of the world (ISO 8373-1994). Recently there has been more advanced ways of controlling robotic arm among them being haptic technology electromyography and human mind. According to Wikipedia, haptic technology provides a tactile feedback which creates the sense of touch by applying forces, vibrations or motion to the user. It gives the feel of touch, sense and force. Its detected through an accelerometer that measures the acceleration of a moving or vibrating body. In the application of electromyography, muscle tension generates a signal which can be transmitted to a computer using a wire or via Bluetooth to control a robotic arm.

1.1 Statement of Research

Problem Human beings are often tasked with activities that are very difficult and tedious to perform. Others demands for unwavering concentration to produce the same results over and over again. Some of these activities include and not limited to; landscaping and site clean-up in building and road constructions, loading and off-loading construction materials onto the transport trucks, lifting heavy construction materials like metal chases and poles, digging holes and heavy demolitions. Robotic arms are therefore needed to work in such areas to make it easy, fast and efficient. Bulldozers, excavators and backhoe loaders are some of the robots available that have arms that help in these activities. The arms in these robots need personnel throughout the activity to give it instructions using gears, levers and sometimes buttons. There is therefore a need to compliment the movement of the robotic arm involved in such operations so that it can automatically perform its functions without the influence or interference of a human being. This arm will be programmed to pick and place objects from a fixed position repetitively without the influence of an operator.

2 OBJECTIVES

- To design a control circuit using a PIC microcontroller and interface it with a servo motor.
- To develop and implement software using C programming language for controlling the rotation of a servo motor.
- To use the microcontroller circuit to send the appropriate signals to different motors to produce the required rotation.
- To assemble and join the materials in construction of the robotic arm.
- To test and implement the robotic arm

3 HARDWARE DESIGN

The hardware designing took four major aspects that included the system design, circuit designs and implementation, and construction of the robotic arm. All the circuits were designed using the Proteus-ISIS7 professional software for schematic circuit design and software simulation. Actual interface was done to come up with the circuit for the control of the servo motors.

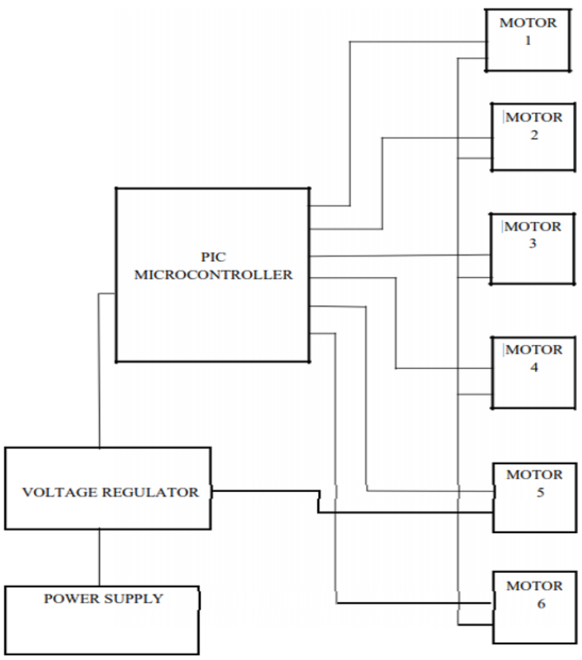

Fig-1 Block diagram of the hardware

3.1 Power Supply

The servo motor needed 6.5V voltage and 3.0A minimum DC individually while the microcontroller needed 5.0V voltage and 2.0A minimum DC. The project's power supply was designed using a 240V to 12V step down transformer, bridge rectifier and two LM2596 adjustable voltage regulators.
The step down transformer was connected to the main power supply through -pin plug socket and its 12V output to the 12V AC to DC bridge rectifier. The 12V DC from the output of the bridge rectifier was fed to two separate LM2596 adjustable voltage regulators, one to supply 6.5V to the servo motors and the other to supply 5.0V to the microcontroller. The two GND from the LM2596 regulators were connected together.

In order to minimize interference from the motors while the microcontroller initializes, a relay input was interfaced to RA2 pin of the microcontroller. Its output was connected to the motor's power supply in order to switch on the supply through the software by the microcontroller after the initialization process was complete.

In order to minimize interference from the motors while the microcontroller initializes, a relay input was interfaced to RA2 pin of the microcontroller. Its output was connected to the motor's power supply in order to switch on the supply through the software by the microcontroller after the initialization process was complete.

3.2 Control Circuit

In the control of one servo motor, a circuit was designed using PIC18F4550 microcontroller, on white breadboard to allow correction, jumpers, wires, 8MHz crystal, two 10kΩ resistor, two 22pF capacitors and a digital push button.
The breadboard was used as the working space. The MCLR pin of the microcontroller was connected to +5.0V through a 10kΩ resistor. The two VDD pins were connected to +5.0V and the GND pins to 0V. To provide a more stable and accurate clock signal, the 8MHz crystal was connected to OSC1 and OSC2 pins and grounded through two parallel 22pF capacitors. The PIC programmer was connected to the PC through a USB and to the PIC microcontroller with its pins, The input digital push button was connected to pin RD0 of the microcontroller and to +5.0V through a 10kΩ pull-up resistor. The control wire of the servo motor was connected to pin RA0 of the microcontroller for PWM signal, power wire to +6.5V and the GND wire to GND. The circuit to control different motors was developed from the circuit to control one motor.

The clock signal circuit and MCLR reset pin remained the same. Three pins; RD0, RD1 and RD2 were connected to control lines of servomotors 1, 2 and 3 respectively. The servo power lines were connected to +6.5V power supply and their ground lines to GND. To select which motor to rotate, push buttons were connected to pins RA0, RA1 and RA2 through 10 kΩ resistors for motors 1, 2 and 3, respectively.

4 SOFTWARE DESIGN

In order to control the rotation of the servo motors and hence the robotic arm, the microcontroller was programmed to give the appropriate instructions to the control wire of the motors through output pins. Software codes were developed at each stage during the design and construction of the robotic arm.

In order to generate the various codes required in the research, the following software were installed in the PC;

- MPLAB_IDEversion6.6.1 from Elektronika Software Company, an Integrated Development Environment (IDE) and compiler.

- PIC programmer software version 3.10 used to import the Hex files from the PC and write them to the PIC microcontroller through the PIC programming kit and a USB cable.
- Timer calculator software for generating the counts the timers make before generating an interrupt in the production of PWM signal.

The software to produce the PWM signal needed was designed using interrupts with its flow chart drawn as shown in figure 4.8 to help in the programming. Timer1 (T1) was used to set period at 20 ms and Timer0 (T0) to set the PWM signal. The microcontroller was set to operate at an external clock frequency of 8 MHz. T1 was started to count 20 ms by pressing the push button after which an interrupt occurred due to overflow (overflow occurs when counts reaches 0xFFFF or 65535). The interrupt of T1 was set to start T0 and set control pin RA0 HIGH. T0 was set to develop interrupt after making its full count for the time the pin RA0 was supposed to be HIGH, for instance 1.0 ms, 1.5 ms and 2.0 ms. The interrupt of T0 was set to put the pin RA0 LOW and to disable itself. The time the pin is set HIGH gives the desired PWM signal that control the position of the servo motor shaft. This method of producing PWM signal was found to work only with three servo motors since there are only four Timers available in PIC18F4550, i.e. T0 for setting the period and T1, T2 and T3 for three control lines.

Due to this limitation of the number of Timers in the PIC18F4550, another method of producing the PWM signal for five servo motors was used. Since a servomotor receives a signal every 20 ms (20000 μs), this time was divided into ON and OFF time so that the control pin to the motor control line is set ON (HIGH) for the duration the signal is47 required and with its completion the pin is set OFF (LOW) for the remaining time of the 20000 μs.

4.2 Debugging methods

Different ways of debugging were used in the course of doing the project which included UART reading and writing, LED display, LCD display and current flow indication by the digital multi meter. FTDI adaptor (USB to serial converter module) was used to enable communication between the microcontroller and PC. It was connected to the TX and RX pins of the PIC18F4550 and GND and a code to write to the PC through the MPLAB_IDE virtual instrument was designed. This was used in different stages of the software development to find out where the codes were working by writing „OK" to the MPLAB_IDE Virtual instrument or ERROR to indicate that the code had failed at a specific point and needed to be checked. Whenever such indications were required, this function was called. The LCD was used the same way by displaying text and later for communication without using the PC and whenever the serial pins were engaged in other uses. Its function was called immediately after an execution that needed to be debugged. For instance when a motor was selected and given the angle to rotate, the LCD displayed the current motor in use. The LED pin was toggled so as to be ON and OFF depending on the duration of delays in every function called and in case it failed at any point it was used as an indication that a certain part of the software was not working and needed the attention of the programmer. The LED was connected to pin RA3. The pin was toggled as follows;

```
LATA.B3=~LATA.B3;            //pin RA3 toggling
```

The hardware part of the project was also debugged at various stages using the digital multimeter by checking whether the connections were okay. Connecting a microcontroller pin to the GND through the meter terminals could indicate whether that pin was outputting the required voltage. The buzzer part of the meter was used to show wires that were faulty from inside and not allowing current to flow.

5. CONCLUSIONS

A PIC microcontroller-based five degree of freedom robotic arm using servo motors was designed, constructed and implemented. The control circuit for the robotic arm was designed in three phases starting with the one to control one servo motor, then to control different servo motors and lastly to control the arm. In the control of one servo motor, a circuit was designed using PIC18F4550 microcontroller, white breadboard, jumpers, wires, 8 MHz crystal, two 10 kΩ resistor, two 22pF capacitors and a digital push button. The power supply to the microcontroller and servo motors was designed using a step down transformer, bridge rectifier and two LM2596 adjustable voltage regulators. The software to produce the PWM signal needed was designed by use of C programming language on MPLAB_IDE (Integrated Development Environment) and compiler,

REFERENCES

[1] Abdul, K. N. L., Ahmad, S. A., Che, A. S. and Ishak, A. T. (2012). Development of Adjustable Gripper for Robotic Picking and Placing Operation: International Journal on Smart Sensing and Intelligent Systems, 4: 1019-1043.

[2] Aishwarya, D., Mayuresh, M. and Virushali, D. (2016). Haptic Arm Robot: International Journal of Advanced Research in Electronics and Communication Engineering, 5: 1438-1440.

[3] Ajayi, A., Awodele, O. and Jegede, O. (2007). Development of a Microcontroller Based Robotic Arm, Journal of Proceedings Of the 2007 Computer Science and IT Education Conference, 1: 549-557.

[4] Alan, L., Ryan, M. and Vuk, Z. (2006). Report on Project, Monitor and Control of an Excavator Robot: Department of Electrical and Computer Engineering, QUEEN"S University, Kingston, Canada.
[5] Amarashid, P., Arriffuddin, J., Muhammad, S. S. and Vijaya, K. R. V. (2009). Development of a Single DOF Robot Arm Using PIC Microcontroller: Journal of Malaysian Technical Universities Conference on Engineering and Technology 1: 36-38.
[6] Amey, P. R., Bhagwat, S. D., Prateek, R. N. and Vipul, J. G. (2013). Robotics Arm Control Using Haptic Technology: International Journal of Latest Research in Science and Technology, 2: 98-102.
[7] Angeliki, B., Bin, H., Bryan, B., Jaron, O., Jianjun, M. and Shuying, Z. (2016). Scientific Report on Noninvasive Electroencephalogram Based Control of a Robotic Arm For Reach and Grasp Tasks: University of Minnesota, College of Science and Engineering and The Medical School.
[8] Arian, F. (2014). Thesis on Design, Implementation and Control of a Robotic Arm Using PIC16F877A Microcontroller: Eastern Mediterranean University, Gazimagusa, North Cyprus.
[9] Arid, K., Mahesh, V. and Puran, S. (2013). Design of a Robotic Arm with Gripper and End Effector for Spot Welding: Universal Journal of Amity school of Engineering and Technology 3: 92-97.
[10] Ashutosh, P. and Rajiv, R. (2013). Thesis on Robotic Arm Control Through Human Arm Movement Using Accelerometers: Electronics and Instrumentation Engineering Department; National Institute Of Technology, Rourkela.