International Journal of Recent Trends in Engineering & Research
ISSN (ONLINE) : 2455 – 1457
IMPACT FACTOR : 4.101

4 Axis Pick and Place Robot using Arduino UNO
Design and Experimental investigation

Venkatramanan C.[1], Barathraj D.[2], Abhishek Sagar T. P.[3], Nandhakumar R.[4]

[1,2,3,4] *Mechatronics Engineering, Agni College of Technology*

Abstract— This paper presents design and experimental investigation on performance of four axis Pick And Place Robot. The objective of this project is to pick the filled water bottles from the conveyors and to place and arrange them in the respective boxes for packing. This achieved by using Arduino UNO as a main controller and servos for performing tasks. Also the Arduino is programmed in such a way that it can sync with the conveyor belt run timing to pick up the bottles at the right time and also it is equipped with proximity sensors to detect bottles if required.

Keywords— Arduino UNO, Pick and Place Robot, IR, Proximity, and Packing Automation

I. INTRODUCTION

Mechanical is the branch of engineering science & Technology related to machinery, and their design, manufacture, application, and structural disposition. Robotics is related to electronics, mechanics and software. Robotics research today is focused on developing systems that exhibit modularity, flexibility, redundancy, fault tolerance, general software environment and seamless connectivity to other machines, some researchers focus on completely automating a manufacturing process or a task, by providing the sensor based intelligence to the mechanical arm, while others try to solidify the analytical foundations on which many of the basic concepts in robotics are built. In this highly developing society time and man power are critical society time and man power are critical constrains for completion of task in large scales. The automation is playing important role to save human efforts in most of the regular and frequently carried works. One of the major and most commonly performed a work is picking and placing the jobs from source to destination.

Present day industry is increasingly turning towards computer based automation mainly due to the need for increased productivity and delivery of end products with uniform qualities. The inflexibility and generally high cost of the hard-automation systems, which have been used for automated manufacturing tasks in the past, have led to a broad based interest in the use of mechanical arm capable of performing a variety of manufacturing functions in a flexible environment and at lower costs. The use of Industrial mechanical arm characterizes some of contemporary trends in automation of the manufacturing process. However, present day industrial mechanical arm also exhibit a monolithic mechanical structure and closed-system software architecture. They are concentrated on simple repetitive tasks, which tend not to require high precision. The pick and place mechanical arm is a human controlled based system that detects the object, picks that object from source location and places at desired location.

II. DESIGN METHODOLOGY

In this paper purpose, The robotic arm can be designed to perform any desired task such as welding, gripping, spinning, etc., depending on the application. For example, robot arms in automotive assembly line perform a variety of tasks such as welding and parts rotation and placement during assembly. This robot can be autonomous or controlled manually and can be used for perform variety of task with great accuracy. The designed robotic arm uses the microcontroller

i.e Arduino UNO and the controller gets programmed using Arduino programming. This process works on the principle of interfacing servos and potentiometers. Interfacing is done by using Arduino board. The remote is fitted with potentiometers and the servos are attached to the body of the robotic arm. This converts the mechanical motion into electrical motion. This servo will respond with the regards to the pulses which result in the moment of the arm. DC motor equipped with the servo is used for precise control of angular position. The robotic arm which is made by them can be used remotely and task can be accomplished. This kind of the robotics arm has various applications in different domain.

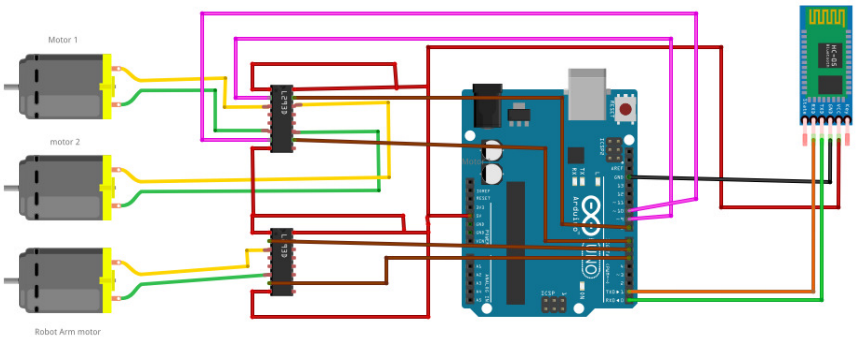

Figure 1. Circuit Layout

2.1 Hardware requirement

The hard ware components used in this project are Arduino UNO, Servo motors, PC9685.

2.1.1. Arduino UNO controller

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

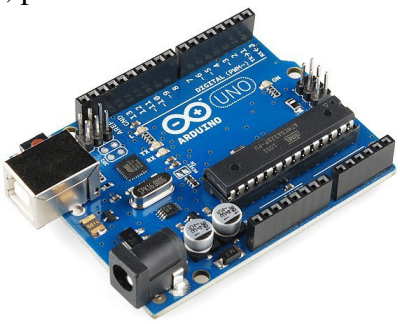

Figure 2. Arduino UNO

2.1.2. Motor

The High-Torque MG996R Digital Servo features metal gearing resulting in extra high 5Kg and 1 Kg stalling Torque in a tiny package. It is the upgraded version of the famous MG995 servo. The gearing and the motor have also been upgraded to improve dead bandwidth and centering. It can rotate up to 120 degrees (60 in each direction). It also comes with a selection of arms and hardware to get setup nice and fast. The specifications are as follows

1. Dimensions - 40.7x19.7x42.9 mm approx.
2. Weight - 55gm
3. Stall Torque - 9.4 kgf-cm (4.8V),11 kgf-cm (6 V)
4. Operating speed - 0.17s/60º (4.8V),0.14s/60º (6V)
5. Operating voltage - 4.8V a 7.2 V
6. Dead baud width - 5µs
7. Temperature range - 0ºC-55ºC

Figure 3. DC Motor

2.1.3. PCA9685 servo motor control board

The PCA9685 is a 16-channel I2C-bus controlled LED controller optimized for Red/Green/Blue/Amber (RGBA) color backlighting applications. Each LED output has individual 12-bit resolution (4096 steps) PWM controller with a fixed frequency. The controller operates at a programmable frequency from a typical 24 Hz to 1526 Hz. All outputs are set to the same PWM frequency. With the PCA9685 as the master chip, the 16- channel 12-bit PWM Servo Driver only needs 2 pins to control 16 servos, thus greatly reducing the occupant I/Os. Moreover, it can be connected to 62 driver boards at most in a cascade way, which means it will be able to control 992servos.

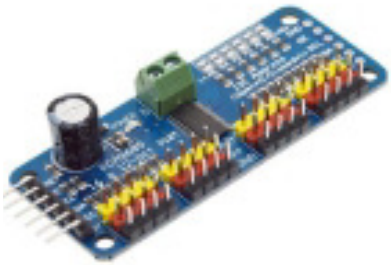

Figure 4. PCA9685 Motordriver

2.2 Software

The software part consisted of a programming language and Integrated Development Environment (IDE). The codes were installed to Arduino board to be compiled and executed.

2.2.1. Programming language

The Arduino Programming Language is basically a framework built on top of C++. We can argue that it's not a real programming language in the traditional term, but we think this helps avoiding confusion for beginners. A program written in the Arduino Programming Language is called sketch. A sketch is normally saved with the .ino extension (from Arduino). The main difference from ―normal‖ C or C++ is that you wrap all your code into 2 main functions. You can have more than 2, of course, but any Arduino program must provide at least those 2. One is called setup(), the other is called loop(). The first is called once, when the program starts, the second is repeatedly called while your program is running. We don't have a main () function like you are used to in C/C++ as the entry point for a program. Once you compile your sketch, the IDE will make sure the end result is a correct C++ program and will basically add the missing glue by preprocessing it. Everything else is normal C++ code, and as C++ is a superset of C, any valid C is also valid Arduino code. One difference that might cause you troubles is that while you can spawn your program over

multiple files, those files must all be in the same folder. Might be a deal breaking limitation if your program will grow very large, but at that point it will be easy to move to a native C++ setup, which is possible.

2.2.2. IDE

When we work with Arduino we commonly use the Arduino IDE (Integrated Development Environment), a software available for all the major desktop platforms (macOS, Linux, Windows), which gives us 2 things: a programming editor with integrated libraries support, and a way to easily compile and load our Arduino programs to a board connected to the computer.

III. IMPLEMENTATION

The figure 5, represents the pick and place robotic arm mounted on chassis. The pick and place robot so implemented is controlled using Arduino UNO. The chassis is supported for the displacement of robotic arm by four servo controlled DC motors. The robotic arm implemented has 4 axis and 2 DOF. The circuit powered up by 12V 1A battery fitted within the body of the chassis. The robot so programmed for pick and place operation can be made versatile and more efficient by providing the feedback and making it to work on own than any human interventions. It can be made possible by tool interfaced with this Arduino. The features that can be added on to improve its efficiency, make it operate on its own thought without any human intervention are line follower, wall hugger, obstacle avoider, metal detector, bomb diffuser etc.

Figure. 5 Pick and Place Robot

IV. RESULTS AND DISCUSSIONS

This pick and place robotic arm is mainly done for picking and placing of object in process of manufacturing or packing which can be fruitful under the suitable working condition but it is not directly developed for the industry . This work had to gone through the picking and placing of measured area. Thus, the adjustment made for program and working area several times for making this bot useful for the wide range of working area. Currently the bot is controlled based on Arduino IDE, this made the range of working area to be limited. It is possible to made change in working area by reprogramming on Arduino. If repetition needed daily then you have to plug your power source and switched on daily.

V. CONCLUSION

Our project mainly targets the small scale industry to increase the productivity by decrease the time for the packing process. In future, our project will hold the better position for small scale industrial revolution. Our project creates a good impact over workers to take over the process with our innovation. If it continues for 05-10 years for all industrial process, our state will self-sufficient in manufacturing sector. In the marketing perspective the above said areas have a great scope with further upgrades this can be implemented in the variety of small scale industries. The technology which is involved in our model is the picking and placing the product over the movement area. Consider a packaging unit of a perfect squared area and the mobile bot is set up at an initial position.

When the bot is switched on the driver circuit starts the motor with the required torque. The weight to torque ratio is satisfied here and the motor can enough torque to rotate the arm in programmed manner. The speed is not needed much in this scenario.

If speed is high then the picking and placing of the object is not so evenly spread so the motors having high torque and low rpm is used. The upgrade includes automatic refill and solar powered charging unit, in case if it stuck in anbgular movement over a long time.

REFERENCES

I. Ramya, B. Palaiyappan, T. Akilan, Embedded System for Robotic Arm Movement Control Using Web Server and Zigbee Communication, Internatiaonal Conference on Research Trends in Computer Technologies, 2013, pp. 30-34.
II. Pradeep Kumar ,et al, design and implementation is based on pneumatic principles, journal of scientific and technology research volume 2, issue 8, august 2013.
III. Mohhammad Hafizuddin. B.A.S et al., use of programmable logic controller to control all the robot movement, April 2008.
IV. Mohammed Naufal Bin omer, "pick and place robotic arm controlled by computer ," International conference on WESAS, April 2007.
V. Kensuke Harada "An object placement planner for robotic pick-and place tasks", National Institute of Advanced Science and technology ASIT, volume 4, issued May 2014.
VI. B.O.Omijeh, "design analysis of remote controlled pick and place robotic vehicle", International Journal of Engineering Research and Development, volume 10, issue 4.
VII. Bhattacharya S.K., and R.R. Tummala, "Integral passives for next generation of electronic packaging: application of epoxy/ceramic nanocomposites as integral capacitors", Microelectronics Journal 32 (2001) 11-19, print.
VIII. https://en.wikipedia.org/wiki/History_of_robots
IX. Hunt K. and Primose E., (1993), Assembly configurations of some in parallel-actuated platforms, Journal of Robotic ystems, 6(6):703-720.
X. Electronics Data Book (1998), http://www.circuitidears.com

