Sudan University of Science and Technology

College of Graduate Studies

BASIC ROBOT TWO AXIS WITH PLC SEQUENCER CONTROL

أساس الروبوت ذي المحورين مع حاكمة المنطقة المتسلسلة القابلة للبرمجة

A Thesis Submitted in Partial Fulfillment of the Requirements for Degree of M.Sc in Electrical Engineering (Control and Microprocessor)

Presented By:

Samah Osman Mohammed Salih

Superviser By:

Ust.Abdallah Salih Ali

December /2016/

Sudan University of Science & Technology

College of Graduate Studies

بسم الله الرحمن الرحيم

جامعة السودان للعلوم والتكنولوجيا

كلية الدراسات العليا

SUST

كلية الدراسات العليا

Ref: SUST/ CGS/A11

Approval Page

(To be completed after the college council approval)

Name of Candidate: عبده | محمد | عثمان | حسن

Thesis title: Basic Robot Two Axis with PLC Sequence control
الروبوت الأساسي ذي المحورين مع التحكم بالمنطق المبرمج المتسلسل.

Degree Examined for: M.Sc. in Control & Microprocessor

Approved by:

1. External Examiner

Name: د. أسامة حسن شيبه

Signature: .... Date: 18/4/2017

2. Internal Examiner

Name: د. كامل حسن

Signature: .... Date: 18/4/2017

3. Supervisor

Name: ABDALLA SALIH ALI

Signature: .... Date: 18/4/2017

الآيـــــة

قَالَ تَعَالَىٰ:

﴿ وَيَسْـَٔلُونَكَ عَنِ ٱلرُّوحِۖ قُلِ ٱلرُّوحُ مِنۡ أَمۡرِ رَبِّي وَمَآ
أُوتِيتُم مِّنَ ٱلۡعِلۡمِ إِلَّا قَلِيلٗا ٨٥ ﴾

صدق الله العظيم

سورة الإسراء: ٨٥

الشكر والعرفان

الحمد لله القائل (لئن شكرتم لازيدنكم) والشكر اولا واخيرا لله عز وجل
اتقدم بخالص الشكر الجزيل والعرفان بالجميل والاحترام والتقدير لمن غمرني بالفضل واختصني بالنصح وتفضل علي بقبول الاشراف علي رسالتي استاذي ومعلمي الفاضل دكتور /**عبدالله صالح**

واخص بالشكر من كلله الله بالهيبه والوقار، من علمني العطاء دون انتظار، من احمل اسمه بكل افتخار

أبي العزيز

كما اختص بالشكر ايضا من هم اقرب الي من روحي وبهم استمد عزتي واصراري

إخوتي

واتقدم ايضا بالشكر والعرفان الي الاستاذ **محمد عثمان محمد** لما قدمه لي من مساعده طيلة فتره المشروع

إهداء

إلى من نذرت عمرها في أداء رسالة

صنعتها من أوراق الصبر *** وطرزتها في ظلام الدهر

على سراج الأمل *** بلا فتور أو كلل

رسالة تعلم العطاء كيف يكون العطاء *** وتعلم الوفاء كيف يكون الوفاء

إليك ...

أمي

أستاذة/ سعاد محمد الحسن محمد

أهدي هذه الرسالة وشتان بين رسالة ورسالة

جزاك الله خيراً وأمد في عمرك بالصالحات

فأنت زهرة الحياة ونورها

Abstract

A two axis robot is to pick and place with different discrete position using PLC. Using only PLC may make the operation slow. the slowness of operation may make some items missed there pick or place position. To avoid miss of pick and place the system speed most be increased. When the system speed is increased the miss position is avoided. The operation of pick and place using PLC controller is shown in eleven steps.

المستخلص

الربوت ذي المحورين استخدم في الالتقاط والوضع لاصناف مختلفه من السلع باستخدام حاكمة المنطق القابله للبرمجه . PLC وعند استخدام حاكمة المنطق القابله للبرمجه نجد ان سرعة الربوت تكون بطيئه وهذه قد تقود لفقدان حالتي الالتقاط والوضع . لتفادي عدم الالتقاط ووضع الاصناف فان سرعة النظام يجب ان تزيد . عندما تمت زيادة السرعه فان حالة ضياع او تبديد الالتقاط والوضع قد تمت ازالتها تماما . التشغيل تم في خلال احد عشره حاله .

TABLE OF CONTENTS

CHAPTERONE

Introduction

1.1 General

Robotics can be described as the current pinnacle of technical development. Robotics is a confluence science using the continuing advancements of mechanical engineering, material science, sensor fabrication, manufacturing techniques, and advanced algorithms. The study and practice of robotics will expose a dabbler or professional to hundreds of different avenues of study. For some, the romanticism of robotics brings forth an almost magical curiosity of the world leading to creation of amazing machines. A journey of a lifetime awaits in robotics.

Robotics can be defined as the science or study of the technology primarily associated with the design, fabrication, theory, and application of robots. While other fields contribute the mathematics, the techniques, and the components robotics creates the magical end product. The practical applications of robots drive development of robotics and drive advancements in other sciences in turn. Crafters and researchers in robotics study more than just robotics.

The promise of robotics is easy to describe but hard for the mind to grasp. Robots hold the promise of moving and transforming materials with the same elan and ease as a computer program transforms data. Today, robots mine minerals, assemble semi-processed materials into automobile components, and assemble those components into automobiles. On the immediate horizon are self-driving cars, robotics to handle household chores, and assemble specialized machines on demand. It is not unreasonable to imagine robots that are given some task, such as reclaim desert into photovoltaic cells and arable land, and left to make their own way. Then the promise of robotics exceeds the minds grasp.

In summary, robotics is the field related to science and technology primarily related to robotics. It stands tall by standing the accomplishments of many other fields of study.

A Mechatronic Device is a degenerate robot with these components

- Sensors, which detect the state of the environment
- Actuators, which modify the state of the environment
- A Control System, which controls the actuators based on the environment as depicted by the sensors

Basic structure of robots is very much similar to humans. How do humans sense? For example a human sees something and sends neural signals to the brain via neurons and reacts accordingly. The development of all these senses artificially is achieved through 'Sensors'. Sensors are the transducers which receive the physical changes of the environment and convert them into electrical or electronic signals. These analog signals are converted into digital by using analog-to-digital convertors.Control system functions as a brain in robotic systems.

PID controller is a popular method for controlling robots. Point of the control system is to get robot actuators do what one wishes to. An encoder on actuator, basically a sensor, determines what all is changing. Program that one writes, defines final results and actuators make changes. Other sensor senses environment, providing robot a better sense of working. Robots are made from an assortment of materials and are driven in many ways too. These can be constructed from sturdy, heavy steel or light weighted plastics. Surgical robots and Robotic arms have uncomplicated rotational joints. These are driven by hydraulics and electrical motors along with longitudinal joints that are moved with rotating screws. Some mobile robots have various wheels that can operate on various planes while others can walk on different terrains with multiple legs. You can find robots with sensing mechanisms like cameras operating as eyes and touch sensors for feeling the environment.

1.2 Problem Statement

Design the robots used for pick and place using PLC alone makes the movement slow as we discover , The movement can be improve by using PID controller.

1.3 Objective

Basictwo axis robots with PLC sequence

A PLC is used to operate robots. Robots to be used for illustrate for pick and place robots which have various discrete positions for their gripper assembly,the position are determined by discrete signal, on , cause the robots axis and manipulated to move to one extreme position ,off, move then to other extreme position

1.4 Methodology

Study the PLCand how to be programming it.

UsingPLCto design basic two axis robots

Analysis the simulation results

1.5 Thesislayout

Chapter two present back ground of programmable logic controllerincluding introduction ,structure and programming. In chapter three system implementation including basic two axis robots, operation to move apart from position A to B and programs for controllingrobot .chapter four present the simulation result and discussion. Finally chapter five provides the conclusion and recommendation.

CHAPTER TWO

Programmable Logic Controller (PLC)

2.1 Introduction

A Programmable Logic Controller (PLC) is a device that is capable of being programmed to perform control functions. The first PLC was introduced in the late 1960s to replace relay logic controls in the automotive industry. Compared to relay logic controls, the PLC's advantages include easy programming and installation, high control speed, hardware and software security, network compatibility, troubleshooting and testing convenience, and high reliability. PLCs are currently used widely in industrial and commercial environments. They can be found in almost any manufacturing facility. There are several manufacturers of PLCs. While the instruction formats may not be the same for different brands, the hardware structures and programming concepts are very similar.

- **Advantages and disadvantages of Programmable logic controller (PLC):**

Advantages	**Disadvantages**
1. Rugged and designed to withstand vibrations,temperature, humidity, and noise.	1. There's too much work required in connecting wires
2. Have interfacing for inputs and outputs already inside the controller	2. There's difficulty with changes or replacements.

3. PLCs are easily programmed and have an easily understood programming language.	3. It's always difficult to find errors; And require skillful work force.
	4. When a problem occurs, hold-up time is indefinite, usually long.

2.2 PLC controller components

PLC is actually an industrial microcontroller system (in more recent times we meet processors instead of microcontrollers) where you have hardware and software specifically adapted to industrial environment. Block schema with typical components which PLC consists of is found in the following Figure 2-1. Special attention needs to be given to input and output, because in these blocks you find protection needed in isolating a CPU blocks from damaging influences that industrial environment can bring to a CPU via input lines. Program unit is usually a computer used for writing a program (often in ladder diagram).

2.2.1 Central Processing Unit – CPU

Central Processing Unit (CPU) is the brain of a PLC controller. CPU itself is usually one of the microcontrollers. Aforetime these were 8-bit microcontrollers such as 8051, and now these are 16- and 32-bit microcontrollers. Unspoken rule is that you will find mostly Hitachi and Fujicu microcontrollers in PLC controllers by Japanese makers, Siemens in European controllers, and Motorola microcontrollers in American ones. CPU also takes care of communication, interconnectedness among other parts of PLC controller, program execution, memory operation, overseeing input and setting up of an output. PLC controllers have complex routines for memory checkup in order to ensure that PLC memory was not damaged (memory checkup is done for safety reasons). Generally speaking, CPU unit makes a great number of check-ups of the PLC controller itself so eventual errors would be discovered early. You can simply look at any PLC controller and see that there are several indicators in the form of light diodes for error signalization.

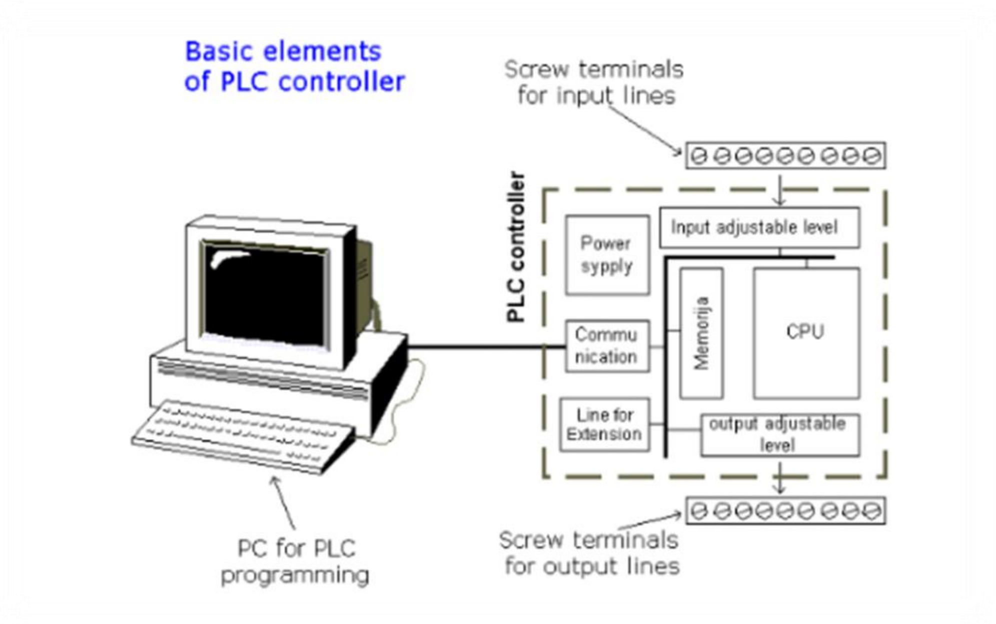


Figure 2-1: Basic elements of PLC controller

2.2.2 System Memory

Memory (today mostly implemented in FLASH technology) is used by a PLC for aprocess control system. Aside from this operating system it also contains a user program translated from a ladder diagram to a binary form. FLASH memory contents can be changed only in case where user program is being changed. PLC controllers were used earlier instead of FLASH memory and have had EPROM memory instead of FLASH memory which had to be erased with ultraviolet light and programmed on programmers. With the use of FLASH technology this process was greatly shortened. Reprogramming a program memory is done through a serial cable in a program for application development. User memory is divided into blocks having special functions. Some parts of a memory are used for storing input and output status. The real status of an input is stored either as "1" or as "0" in a specific memory bit. Each input or output has one corresponding bit in memory. Other parts of memory are used to store variable contents for variables used in user program. For example, timer value, or counter value would be stored in this part of the memory.

2.2.3 Programming a PLC controller PLC

Controller can be reprogrammed through a computer (usual way), but also through manual programmers (consoles). This practically means that each PLC controller can programmed through a computer if you have the software needed for programming. Today's transmission computers are ideal for reprogramming a PLC controller in factory itself. This is of great importance to industry. Once the system is corrected, it is also important to read the right program into a PLC again. It is also good to check from time to time whether program in a PLC has not changed. This helps to avoid hazardous situations in factory rooms (some automakers have established communication networks which regularly check programs in PLC controllers to ensure execution only of good programs). Almost every program for programming a PLC controller possesses various useful options such as: forced switching on and off of the system inputs/out puts (I/O lines), program follow up in real time as well as documenting a diagram. This documenting is necessary to understand and define failures and malfunctions. Programmer can add remarks, names of input or output devices, and comments that can be useful when finding errors, or with system maintenance. Adding comments and remarks enables any technician (and not just a person who developed the system) to understand a ladder diagram right away. Comments and remarks can even quote precisely part numbers if replacements would be needed. This would speed up a repair of any problems that come up due to bad parts. The old way was such that a person who developed a system had protection on the program, so nobody aside from this person could understand how it was done. Correctly documented ladder diagram allows any technician to understand thoroughly how system functions.

2.2.4 Power supply

Supply is used in bringing electrical energy to central processing unit. Most PLC controllers work either at 24 VDC or 220 VAC. On some PLC controllers you'll find electrical supply as a separate module. Those are usually bigger PLC controllers, while small and medium series already contain the supply module. User has to determine how much current to take from I/O module to ensure that electrical supply provides appropriate amount of current. Different types of modules use different amounts of electrical current. This electrical supply is usually not used to start external inputs or outputs. User has to provide separate supplies in starting PLC controller inputs or outputs because then you can ensure so called "pure" supply for the PLC controller. With pure supply we mean supply where industrial environment can not affect it damagingly. Some of the smaller PLC controllers supply their inputs with voltage from a small supply source already incorporated into a PLC.

2.2.5 PLC controller inputs

Intelligence Of an automated system depends largely on the ability of a PLC controller to read signals from different types of sensors and input devices. Keys, keyboards and by functional switches are a basis for man versus machine relationship. On the other hand, in order to detect a working piece, view a mechanism in motion, check pressure or fluid level you need specific automatic devices such as proximity sensors, marginal switches, photoelectric sensors, level sensors, etc. Thus, input signals can be logical (on/off) or analogue. Smaller PLC controllers usually have only digital input lines while larger also accept analogue inputs through special units attached to PLC controller. One of the most frequent analogue signals are a current signal of 4 to 20 MA and mili volt voltage signal generated by various sensors. Sensors are usually used as inputs for PLCs. You can obtain sensors for different purposes. They can sense presence of some parts, measure temperature, pressure, or some other physical dimension, etc. (ex. inductive sensors can register metal objects). Other devices also can serve as inputs to PLC controller. Intelligent devices such as robots, video systems, etc. often are capable of sending signals to PLC controller input modules (robot, for instance, can send a signal to PLC controller input as information when it has finished moving an object from one place to the other.)

2.2.6 Input adjustment interface

Adjustment interface also called an interface is placed between input lines and a CPU unit. The purpose of adjustment interface to protect a CPU from disproportionate signals from an outside world. Input adjustment module turns a level of real logic to a level that suits CPU unit (ex. input from a sensor which works on 24 VDC must be converted to a signal of 5 VDC in order for a CPU to be able to process it). This is typically done through opto-isolation, and this function you can view in the figure 2-2. Opto-isolation means that there is no electrical connection between external world and CPU unit. They are "optically" separated, or in other words, signal is transmitted through light. The way this works is simple. External device brings a signal which turns LED on, whose light in turn incites photo transistor which in turn starts conducting, and a CPU sees this as logic zero (supply between collector and transmitter falls under 1V). When input signal stops LED diode turns off, transistor stops conducting, collector voltage increases, and CPU receives logic 1 as information.

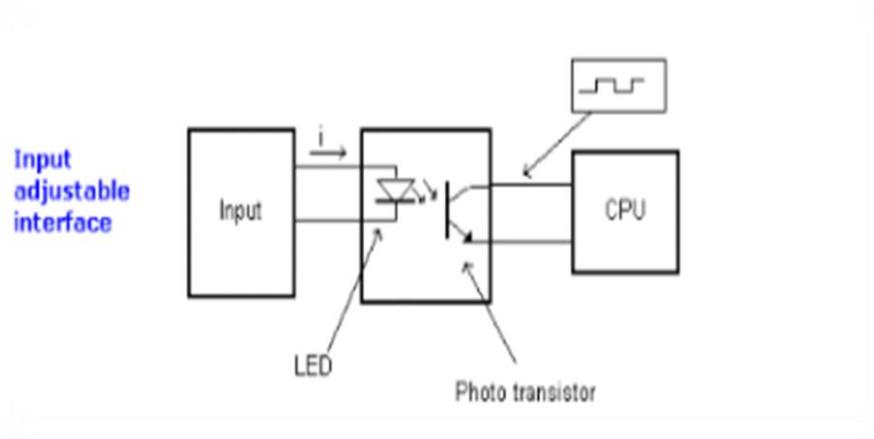


Figure 2-2: Input adjustable interface

2.2.7 PLC controller output

Automated system is incomplete if it is not connected with some output devices. Some of the most frequently used devices are motors, solenoids, relays, indicators, sound signalization and similar. By starting a motor, or a relay, PLC can manage or control a simple system such as system for sorting products all the way up to complex systems such as service system for positioning head of CNC machine. Output can be of analogue or digital type. Digital output signal works as a switch; it connects and disconnects line. Analogue output is used to generate the analogue signal (ex. motor whose speed is controlled by a voltage that corresponds to a desired speed).

2.2.8 Output adjustment interface

Output interface is similar to input interface. CPU brings a signal to LED diode and turns it on. Light incites a photo transistor which begins to conduct electricity, and thus the voltage between collector and emmiter falls to 0.7V , and a device attached to this output sees this as a logic zero. Inversely it means that a signal at the output exists and is interpreted as logic one. Photo transistor is not directly connected to a PLC controller output. Between photo transistor and an output usually there is a relay or a stronger transistor capable of interrupting stronger signals. Show in figure 2-3

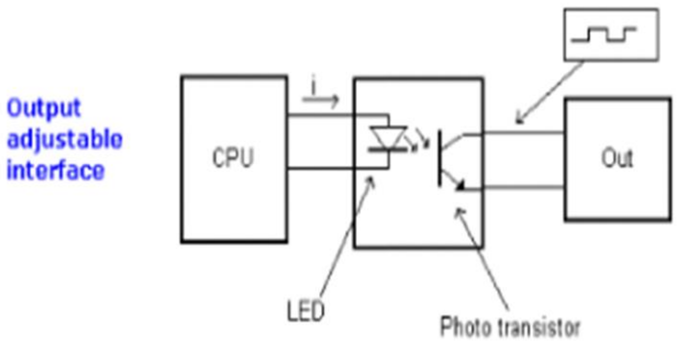


Figure 2-3: Output adjustable interface

2.3 Programmable controllers

Programmable controllers are generally programmed in ladder diagram (or "relay diagram") which is nothing but a symbolic representation of electric circuits. Symbols were selected that actually looked similar to schematic symbols of electric devices, and this has made it much easier for electricians to switch to programming PLC controllers. Electrician who has never seen a PLC can understand a ladder diagram.

2.3.1 Ladder diagram

There are several languages designed for user communication with a PLC, among which ladder diagram is the most popular. Ladder diagram consists of one vertical line found on the left hand side, and lines which branch off to the right. Line on the left is called a "bus bar", and lines that branch off to the right a instruction lines. Conditions which lead to instructions positioned at the right edge of a diagram are stored along instruction lines. Logical combination of these conditions determines when and in what way instruction on the right will execute. Basic elements of a relay diagram can be seen in the figure2-4

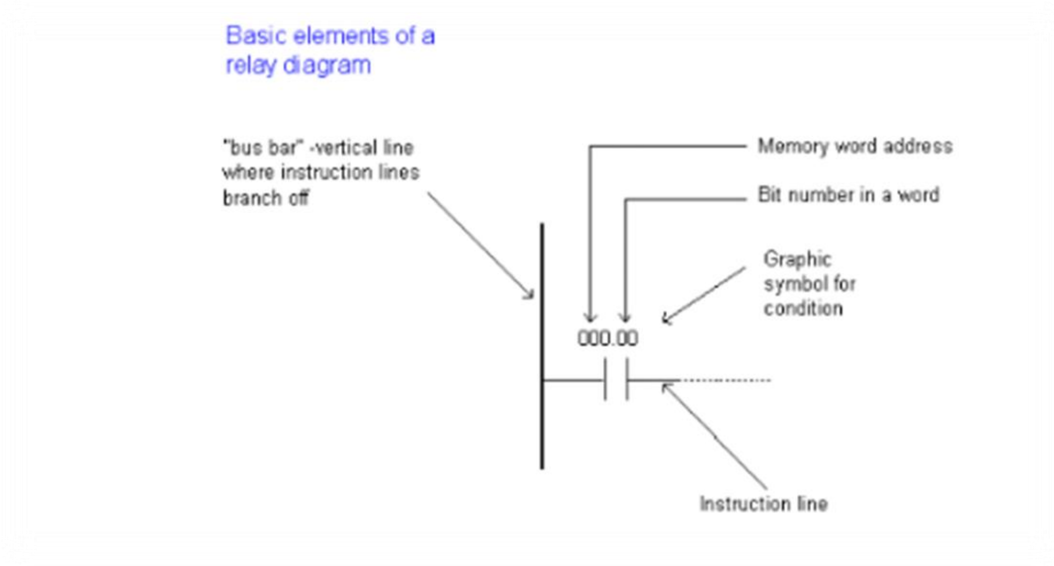


Figure 2-4: Basic element of a relay diagram

Most instructions require at least one operand, and often more than one. Operand can be some memory location, one memory location bit, or some numeric value -number. In the example above, operand is bit 0 of memory location IR000. In a case when we wish to proclaim a constant as an operand, designation # is used beneath the numeric writing (for a compiler to know it is a constant and not an address.)

Based on the figure 2-4 above, one should note that a ladder diagram consists of two basic parts: left section also called conditional, and a right section which has instructions. When a condition is fulfilled, instruction is executed, and that's all.

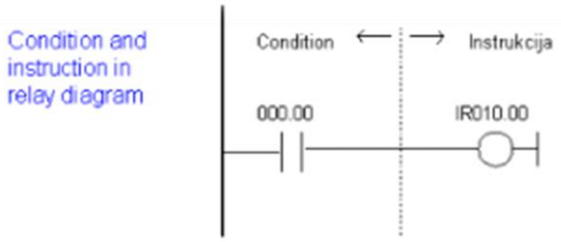


Figure 2-5: Condition and Instruction in relay diagram

Figure 2-5 above represents a example of a ladder diagram where relay is activated in PLC controller when signal appears at input line 00. Vertical line pairs are called conditions. Each condition in a ladder diagram has a value ON or OFF, depending on a bit status assigned to it. In this case, this bit is also physically present as an input line (screw terminal) to a PLC controller. If a key is attached to a corresponding screw terminal, you can change bit status from a logic one status to a logic zero status, and vice versa. Status of logic one is usually designated as "ON", and status of logic zero as "OFF". Right section of a ladder diagram is an instruction which is executed if left condition is fulfilled. There are several types of instructions that could easily be divided into simple and complex. Example of a simple instruction is activation of some bit in memory location. In the example above, this bit has physical connotation because it is connected with a relay inside a PLC controller. When a CPU activates one of the leading four bits in a word IR010, relay contacts move and connect lines attached to it. In this case, these are the lines connected to a screw terminal marked as 00 and to one of COM screw terminals.

2.3.2 Normally open and normally closed contacts

Since we frequently meet with concepts "normally open" and "normally closed" in industrial environment, it's important to know them. Both terms apply to words such as contacts, input, output, etc. (all combinations have the same meaning whether we are talking about input, output, contact or something else). Principle is quite simple, normally open switch won't conduct electricity until it is pressed down, and normally closed switch will conduct electricity until it is pressed. Good examples for both situations are the doorbell and a house alarm. If a normally closed switch is selected, bell will work continually until someone pushes the switch. By pushing a switch, contacts are opened and the flow of electricity towards the bell is interrupted. Of course, system so designed would not in any case suit the owner of the house. A better choice would certainly be a normally open switch. This way bell wouldn't work until someone pushed the switch button and thus informed of his or her presence at the entrance. Home alarm system is an example of an application of a normally closed switch. Let's suppose that alarm system is intended for surveillance of the front door to the house. One of the ways to "wire" the house would be to install a normally open switch from each door to the alarm itself (precisely as with a bell switch). Then, if the door was opened, this would close the switch, and an alarm would be activated. This system could work, but there would be some problems with this, too. Let's suppose that switch is not working, that a wire is somehow disconnected, or a switch is broken, etc. (there are many ways in which this system could become dysfunctional). The real trouble is that a homeowner would not know that a system was out of order. A burglar could open the door, a switch would not work, and the alarm would not be activated. Obviously, this isn't a good way to set up this system. System should be set up in such a way so the alarm is activated by a burglar, but also by its own dysfunction, or if any of the components stopped working. (A homeowner would certainly want to know if a system was dysfunctional).

Having these things in mind, it is far better to use a switch with normally closed contacts which will detect an unauthorized entrance (opened door interrupts the flow of electricity, and this signal is used to activate a sound signal), or a failure on the system such as a disconnected wire. These considerations are even more important in industrial environment where a failure could cause injury at work. One such example where outputs with normally closed contacts are used is a safety wall with trimming machines. If the wall doors open, switch affects the output with normally closed contacts and interrupts a supply circuit. This stops the machine and prevents an injury.

Concepts normally open and normally closed can apply to sensors as well. Sensors are used to sense the presence of physical objects, measure some dimension or some amount. For instance, one type of sensors can be used to detect presence of a box on an industry transfer belt. Other types can be used to measure physical dimensions such as heat, etc. Still, most sensors are of a switch type. Their output is in status ON or OFF depending on what the sensor "feels". Let's take for instance a sensor made to feel metal when a metal object passes by the sensor. For this purpose, a sensor with a normally open or a normally closed contact at the output could be used. If it were necessary to inform a PLC each time an object passed by the sensor, a sensor with a normally open output should be selected. Sensor output would set off only if a metal object were placed right before the sensor. A sensor would turn off after the object has passed. PLC could then calculate how many times a normally open contact was set off at the sensor output, and would thus know how many metal objects passed by the sensor.

Concepts normally open and normally closed contact ought to be clarified and explained in detail in the example of a PLC controller input and output. The easiest way to explain them is in the example of a relay. Show in figure 2-6

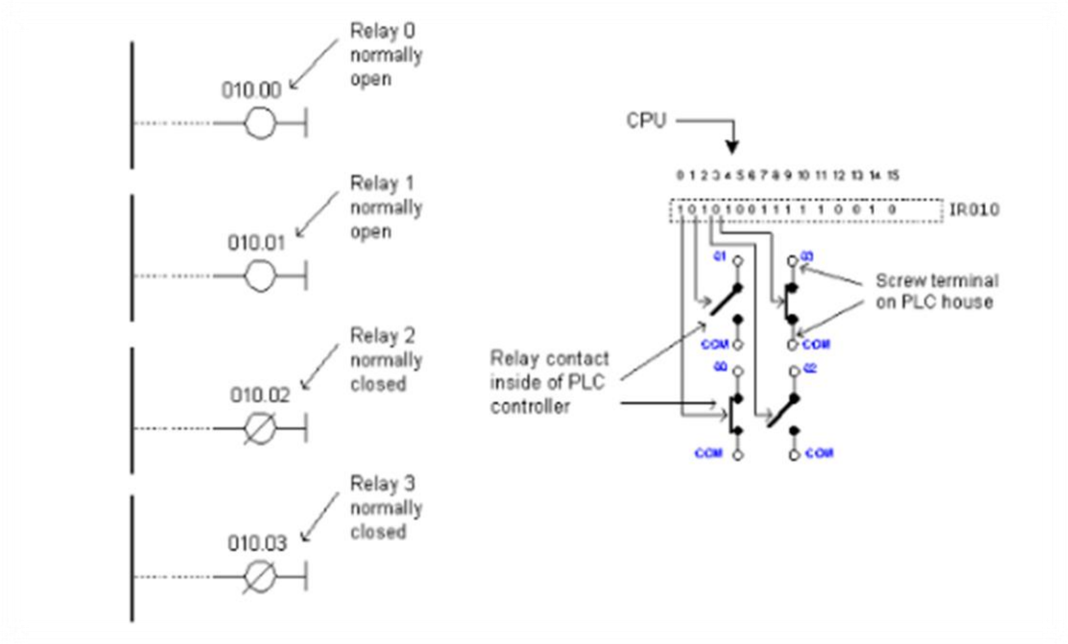


Figure 2-6: Normally open and Normally close contact

Normally open contacts would represent relay contacts that would perform a connection upon receipt of a signal. Unlike open contacts, with normally closed contacts signal will interrupt a contact, or turn a relay off. Previous figure 2-7 shows what this looks like in practice. First two relays are defined as normally open , and the other two as normally closed. All relays react to a signal! First relay (00) has a signal and closes its contacts. Second relay (01) does not have a signal and remains opened. Third relay (02) has a signal and opens its contacts considering it is defined as a closed contact. Fourth relay (03) does not have a signal and remains closed because it is so defined.

Concepts "normally open" and "normally closed" can also refer to inputs of a PLC controller. Let's use a key as an example of an input to a PLC controller. Input where a key is connected can be defined as an input with open or closed contacts. If it is defined as an input with normally open contact, pushing a key will set off an instruction found after the condition. In this case it will be an activation of a relay 00.

If input is defined as an input with normally closed contact, pushing the key will interrupt instruction found after the condition. In this case, this will cause deactivation of relay 00 (relay is active until the key is pressed). You can see in figure 2-7 below how keys are connected, and view the relay diagrams in both cases.

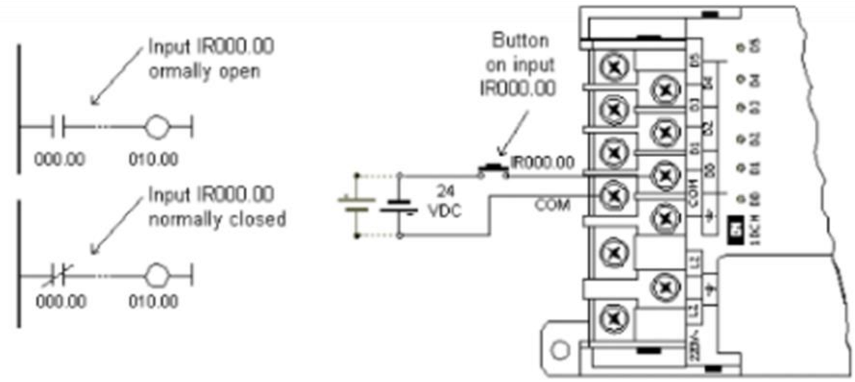


Figure 2-7 : Keys are connected and relay diagrams in both cases

Normally open/closed conditions differ in a ladder diagram by a diagonal line across a symbol. What determines an execution condition for instruction is a bit status marked beneath each condition on instruction line. Normally open condition is ON if its operand bit has ON status, or its status is OFF if that is the status of its operand bit. Normally closed condition is ON when its operand bit is OFF, or it has OFF status when the status of its operand bit is ON. When programming with a ladder diagram, logical combination of ON and OFF conditions set before the instruction determines the eventual condition under which the instruction will be, or will not be executed. This condition, which can have only ON or OFF values is called instruction execution condition. Operand assigned to any instruction in a relay diagram can be any bit from IR, SR, HR, AR, LR or TC sector. This means that conditions in a relay diagram can be determined by a status of I/O bits, or of flags, operational bits, timers/counters, etc.

CHAPTER THREE

SYSTEM IMPLEMENTATION

3.1: INTRODUCTION

A PLC may be used to run a robot .the robots to be used for illustration are pick . and .place robots , which have various discrete position for their for their gripper assembly . the position are determined by discrete singles "on" causes the robots axes and manipulators to move to one extreme position . and "of" mouse them to the other extreme position . PLC control system are used for industrial type pick and place robots .

3.2: BASIC TWO AXIS ROBOT WITH PLC SEQUENCER CONTROL

Two schemes of controlling the basic pick and place robot shown in figure 3-1 we use switches first , and then a drum controller sequencer .

The robot use for illustration starts operating from the position shown which is the "at-rest" lower .left .initialized position . the step – by – step sequence .

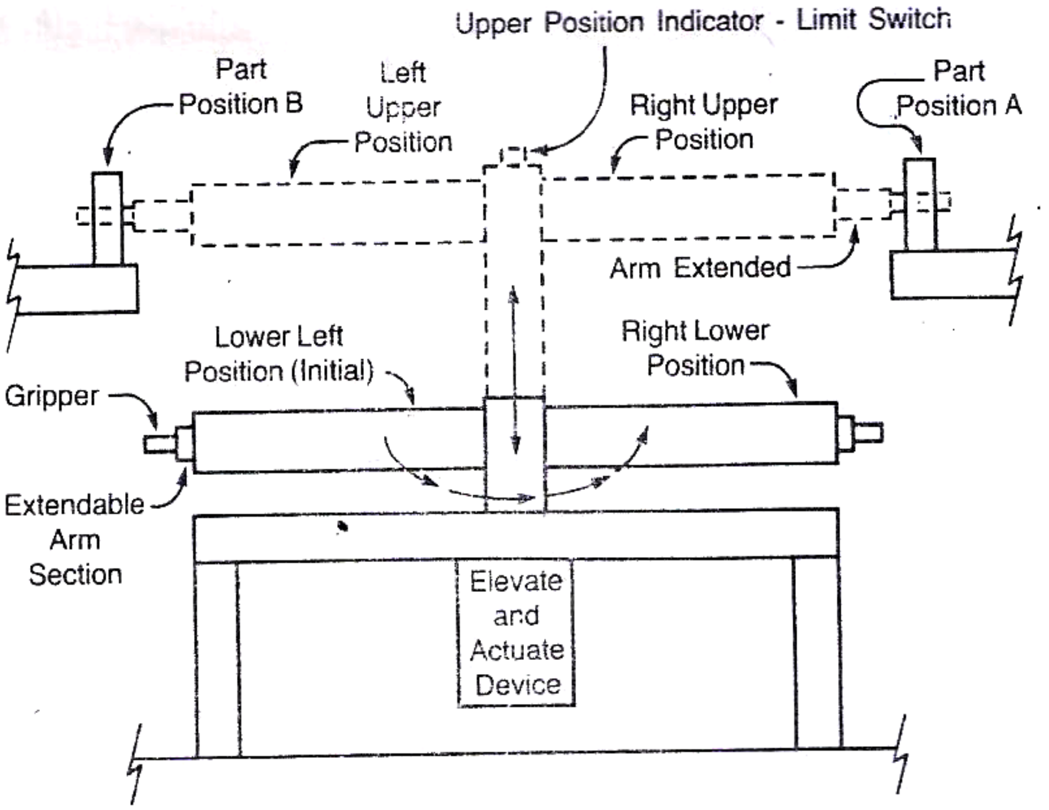


Figure 3-1**Basic pick – and – place robot**

3.3: OPERATION TO MOVE APART FROM POSITION A TO B IS AS FOLLOWS

1. Arm is initially in the down – position as shown .gripper is open and not extended .
2. Arm move to upper position .
3. Arm rotates to right .
4. Hand extends to position A .
5. Gripper closes .gripping part .
6. Arm swings back to the left to position B .
7. Gripper opens .releasing part.
8. Hand retracts .
9. Arm lowers to the initial position .

The robots has lour power pneumatic solenoids if all solenoids are off . no air is applied to the robots actuators . in this initial position . the robots is in the lowered . left position with the hand retracted and the gripper open . energizing each of the four solenoids causes the following action to occur :

1. Rotate – arm rotates full right .
2. raise – arm rises to the upper position .
3. extend – hand extends form the arm .
4. Grip – the gripper closes .

An operational matrix for the robot to move a part from position a to position b shown in table 3-1 .

An o indicates the opposite position , down , left , in , or open

Table 3-1: Part movement robot operational matrix .

Step	**Up**	**Rotate right**	**Hand out**	**Grip close**
Initialized	O	O	O	O
1	X	O	O	O
2	X	X	O	O
3	X	X	X	O
4	X	X	X	X
5	X	O	X	X
6	X	O	X	O
7	X	O	O	O
8	O	O	O	O

As imply control system for the robot shown in Table3-1 could consist of four switches , one for each motion .

- **DISADVANTAGE OF THE FOUR SWITCH CONTROL :**

1. The four switch control is that someone would have to do controlling continuously .
2. Turning off the switch would not immediately stop the are it would spring . return to its initial position which would be hazardous to any one expecting it to stop immediately .
3. Problem in mechanical interface during operation in the upper position with the arm extended , moving the are down could break off the arm on the convey or below it also . if the gripper opened up while the arm was making a swing . the part would be dropped or thrown outward .

3.4: PROGRAMS FOR CONTROLLING THE ROBOT :

Two basic programs for controlling the robot of figure 3-1

1. Program is a plc version of the switch relay system shown in figure 3-2 .

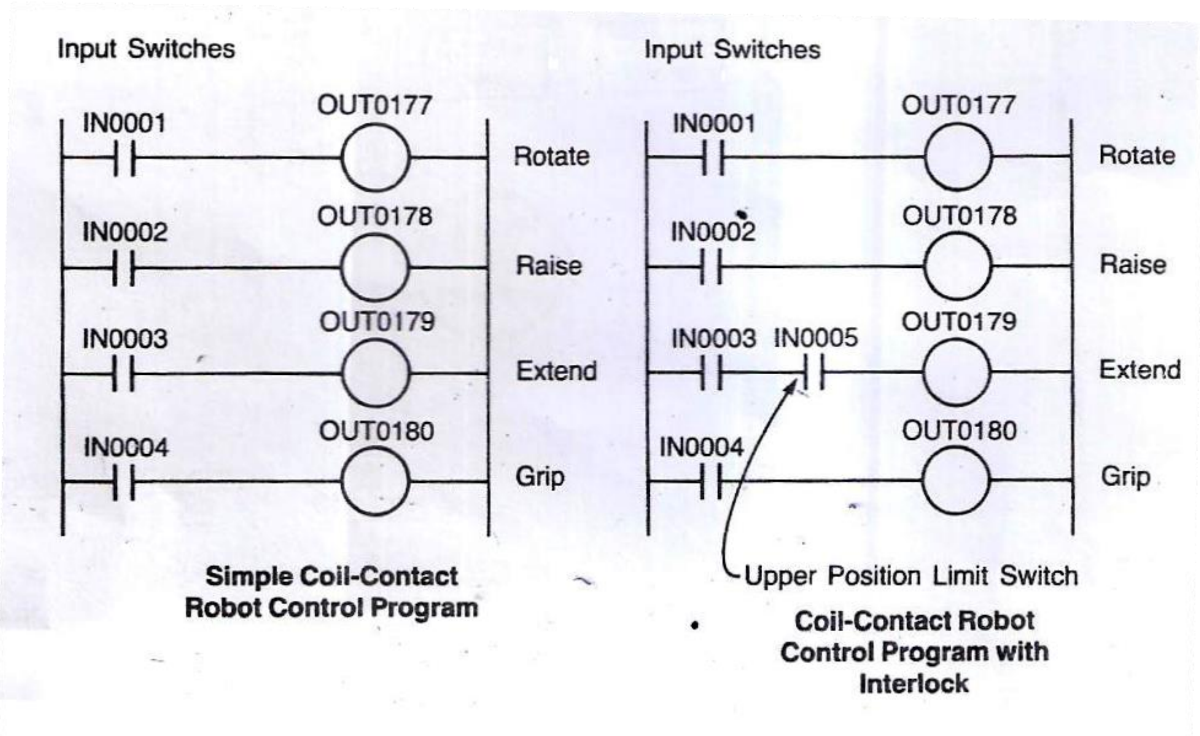


figure 3-2 plc program for robot control

2. ADR function and register .

For this program there are some possible operational problems if a pneumatic cylinder failed or some thingbe camejammed . The plc program would continue unabated . There could be equipment damage or even personal injury . Additional programming would be necessary to inclue inter locks .Sensor , Positive emergency stop .And the like .

CHAPTER FOUR

SIMULATION AND RESULT

4.1 INTRODUCTION

A pick and place robot is a robot that can be programmed to literally pick an object up and place it somewhere. These robots are popular among business owners who require speedy and precise automation applications and material handling systems. They are especially practical in places such as assembly lines, where repetitive and difficult tasks need to be performed with accuracy.

A main advantage of using a pick and place robot is that it can get a job done faster than humans. This can lead to increase production rates, especially in industries where production is time-sensitive. Using these robots also can pad profits, because they can output more products in less time than human can.

A pick and place robot can be programmed and reprogrammed to complete a variety of functions. It can be programmed to handle multiple sizes of items or even different types of items. In addition, its thin arms, which can reach far, can be flexible enough to accommodate almost any type of product. Accuracy is another benefit of using these types of robots. They can be specifically programmed, so their results will always be the same. The robots' owners know that they will be as precise as they want them to be.

4.2 Pick and Place Robots work

High-speed pick and place robots can be mounted on a stand to allow the robots to access their entire working envelope. Product will enter the robotic work envelope after its orientation has been identified by an upstream vision system. Using a custom end of arm tool, the product will be picked and placed by the robot at the desired location. Product can pass/fail inspection based on customer defined specifications for length, straightness, shape, etc .

4.2.1 Benefits of Pick and Place Robots

- Automates to processes that involve fast-moving conveyor
- Provides extremely high-speed output
- Eliminates fixtures therefore reducing cost.
- Two dimential vision is flexible and easy to adapt.
- Improved productivity and product quality.

4.2.2 Applications for Pick and Place Robots

Bastian Solutions utilizes a creative concept and design process to ensure we implement the most appropriate tool for the application. Common applications include:

- Fast Assembly – robots can pick parts from incoming conveyor and assemble them onto work pieces carried by an outgoing conveyor with high precision.
- Inspection and Quality Control – robot visually inspects and picks out defective products moving on a fast conveyor.
- Fast Packaging – vision spots products spread out on moving conveyor and then the robot transfers them into packaging containers at high speed.
- Fast Sortation - robot instantaneously picks and separates different types of parts passing through its vision domain.

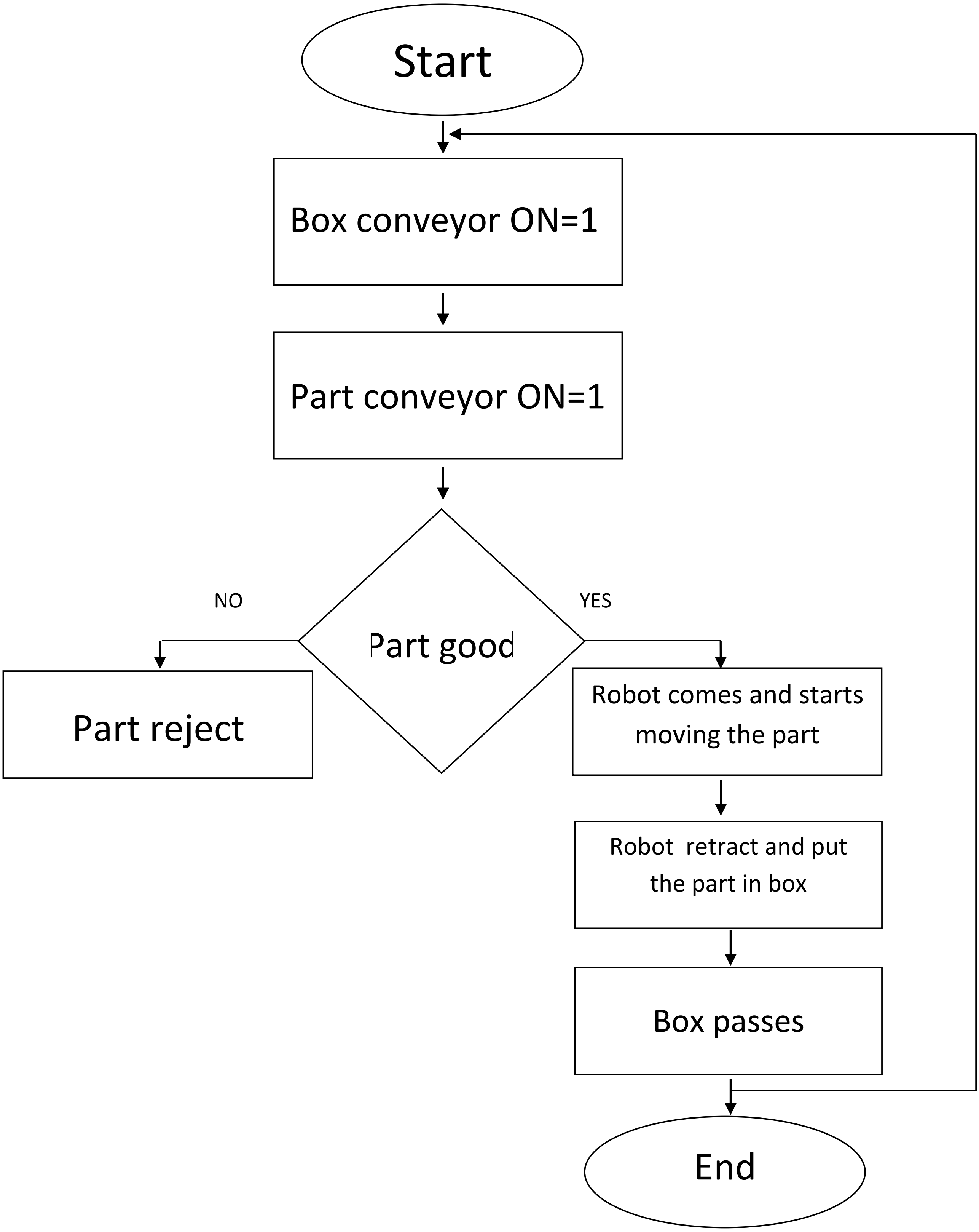
4.2.3 Pick and Place Robot Specifications

- Throughput can reach up to 200 products per minute
- Vision can identify 100 or more products on the moving conveyor per second.
- High picking accuracy and tool compensation can bring placement error down to 0 mm.
- Vision sensor and accessories are stationary-mounted
- Robotic system interfaces with PC or PLC

.

4.3 flow chart

The flow chart shows the sequence of operation. the sequence of operation is show in network of 11 steps.



4.4 SIMULATION

Case 1

System start when start push bottom is pressed and green lamp will lit shown in figure 4-1

Figure4-1: starting of Robot

Case 2:

When the system start the X_ axis will move two steps in the minus direction every step is activated for 1 second in order to get

the full movement shown in figure 4-2

Figure **4-2: starting with X-axis two step**

Case 3:

Then the boxes conveyor will start moving to feed empty boxes shown in figure 4-3

Figure 4-3: **Moving of conveyor to feed empty boxes**

Case 4:

The parts conveyor will start moving when an empty box is present and the boxes conveyor will stop also when an empty box is present on the loading position (box present sensor =1,ON)shown in figure 4-4

Figure 4-4: conveyor movement when empty boxes are present

Case 5:

By moving the parts conveyor the parts will come in three different sizes ; two of them are good and one of them is wrong ; shown in figure 4-5a,b and c ; so when the wrong detection sensor is ON then this part will not loaded to the boxes while the other two part types will loaded to the boxes

Figure 4-5a: loading good part

Figure**4-5 b: loading another good part**

Figure**4-5c: unloading wrong part**

Case 6:

The picking sequence will take place if a good part is detected and reach the picking point (box_ present _sensor=1) then the part conveyor will stop shown in figure 4-6 a and b

Figure 4-6a: stopping of the part conveyor incase agood part

Figure 4-6b: passing of the part conveyor incase wrong part

Case 7:

The X_ axis moves to the minus direction 1 step until the X_ minus _ limit _ ON; the picking vacuum is activated and the Z _ axis will retract shown in figure 4-7

Figure 4-7: move of X-axis one step and Z-axis retract

Case 8:

The X _axis will move one step in the plus direction and when it reached it is limit the Z_ axis extend will activated shown in figure 4-8a and b

Figure 4-8A: X- axis in for word direction

Figure 4-8b: Z- axis will activated

Case 9:

When the Z_ axis is fully extended (Z_ extent _ limit=1) the picking vacuum is deactivated in order to drop the part in the empty slot out the box then the Z_ axis will retract shown in figure 4-9a and b

Figure 4-9a: Z- axis is fully extended

Figure **4-9b: Z-axis retract**

Case 10:

Anew sycle will start by resting all memories ; and the boxes conveyor will start moving again to feed anew empty box on net work 3 shown in figure 4-10

Figure 4-10: memory reset and new sycle

Case 11:

At any time if the stop _ push bottom is pressed the boxes conveyor will stop and the green lamp will goes off which deactivate all the program shown in figure 4-11

Figure **4-11: stop push bottom is pressed and deactive all program**

4.5 RESULT

By programming the memory the robot start operating , the starting is show by green LED . Then the x-axis is started in two step ,empty boxes and feeded . conveyor start moving when empty boxes are . present ,when items come in three different size which two are good and bad items will be reject . the sequence of operation will take and place when the good items are detected and reach the location and the conveyor will stop.

CHAPTER FIVE

CONCULSION AND RECOMMENDED

5.1 Conculsion

Mechanical interface could be a problem during the operation .In the upper position will arm extended moving arm down could break the arm .To overcome this problem ladder diagram will interlock .

Various control method are used to operate robot or to program the robot to represent automatically, then may include manual control . Mechanical control can use EPROM and computer control .Computer control may be for the divided. One complete cycle

5.2 Recommended

- This project need to be implemented with hard components ; even aprototype in order to test it in machine handling
- Using fuzzy logic and PID controller for pick and place of items
- Using microcontroller

Reference

1\ Jhon W.Weeb And Robardo, “Programmable Logic Controller” Printa Itall, 1993.

2\ Bryons, “Programmable Controller”, Industrial Text , 1998.

3\ Janes C.T. And Bryons,“Programmable Controller And Concept Application”, IPC/ASTEC, 1983.

4\ Kisslland James, “Under Study And Use Programmable Contoroller”, Printa Itall 1986.

5\ Englewood Clift,” Programmable Contoroller”, New York Magraw Itill 1989.

APPENDICES

OB1 - <offline>

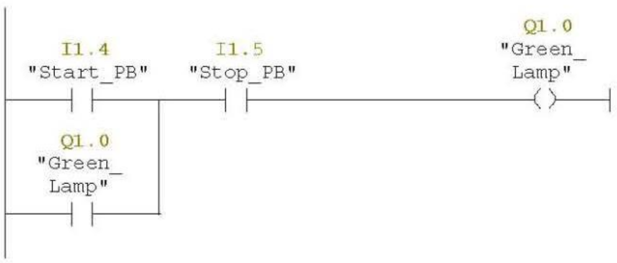
"Cycle Execution"

Name: **Family:**
Author: **Version:** 0.1
Block version: 2
Time stamp Code: 03/30/2016 11:18:11 PM
Interface: 02/15/1996 04:51:12 PM
Lengths (block/logic/data): 00504 00370 00022

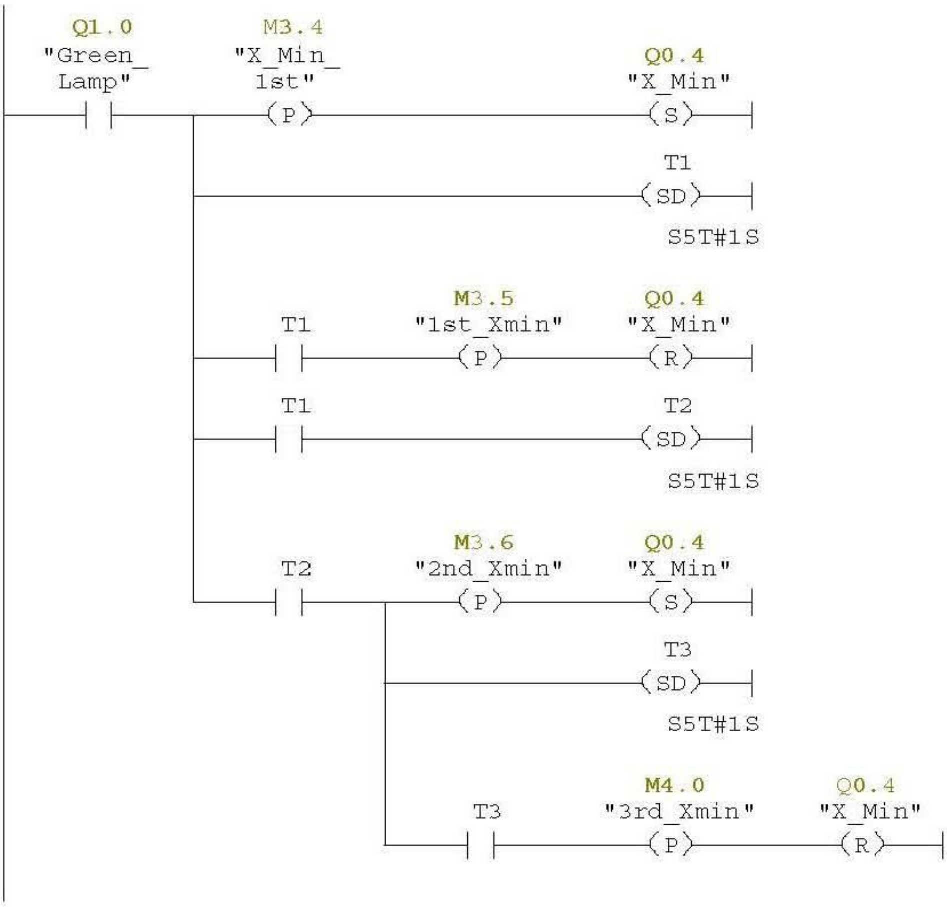
Name	Data Type	Address	Comment
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Block: OB1 "Main Program Sweep (Cycle)"

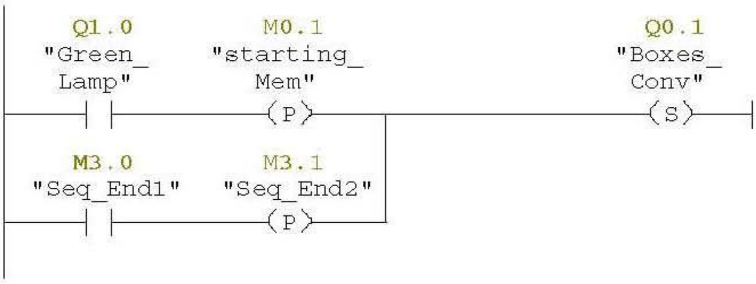
Network: 1 System Start when Start push bottun is pressed and green lamp wi



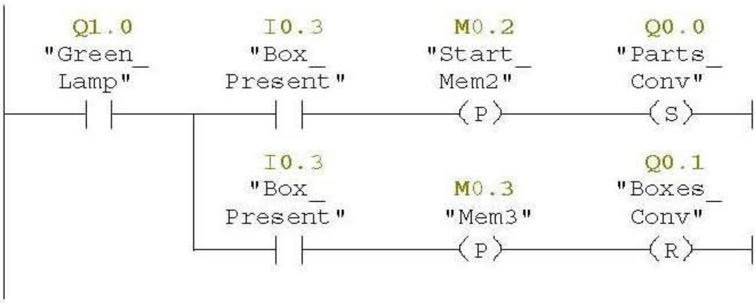
Network: 2



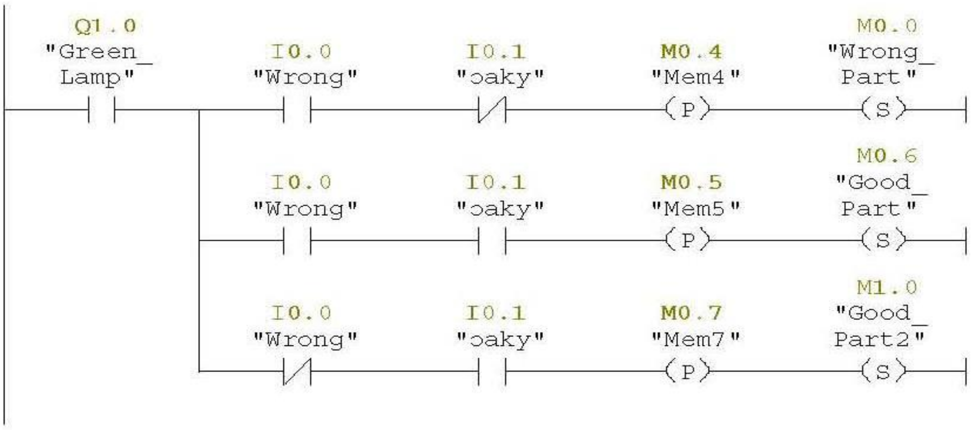
Network: 3



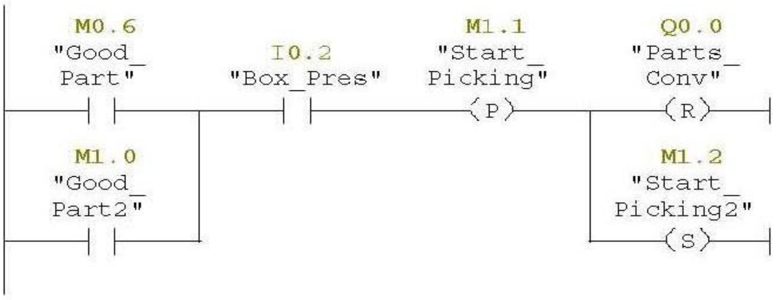
Network: 4



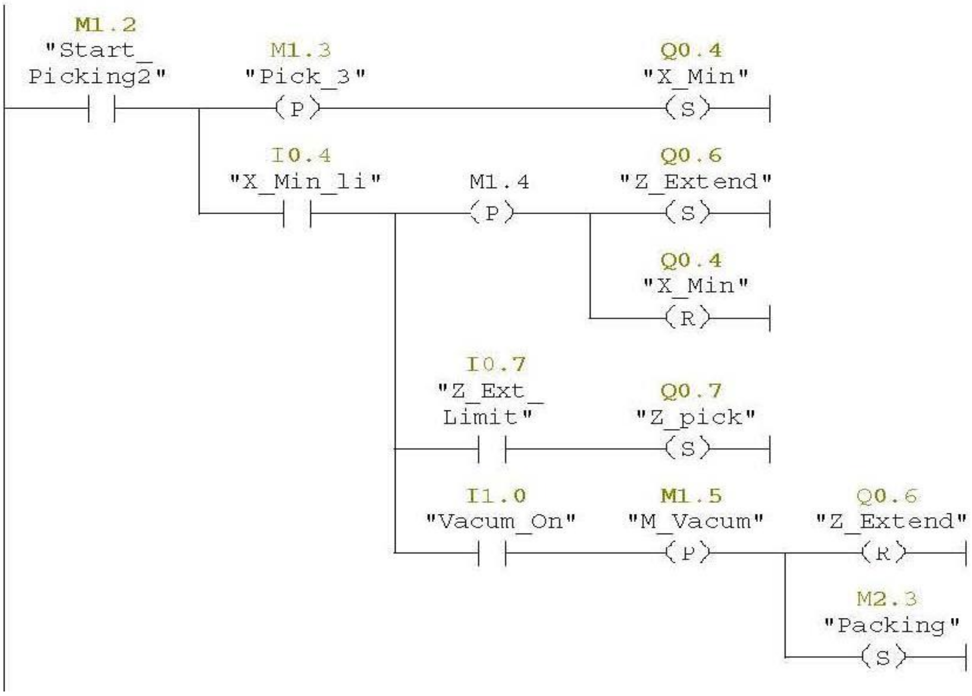
Network: 5



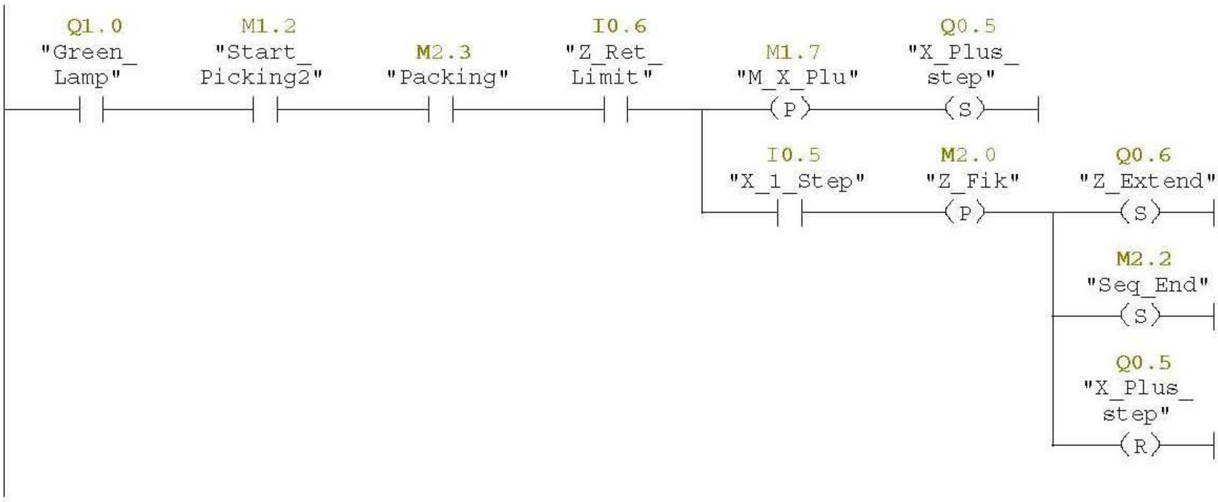
Network: 6



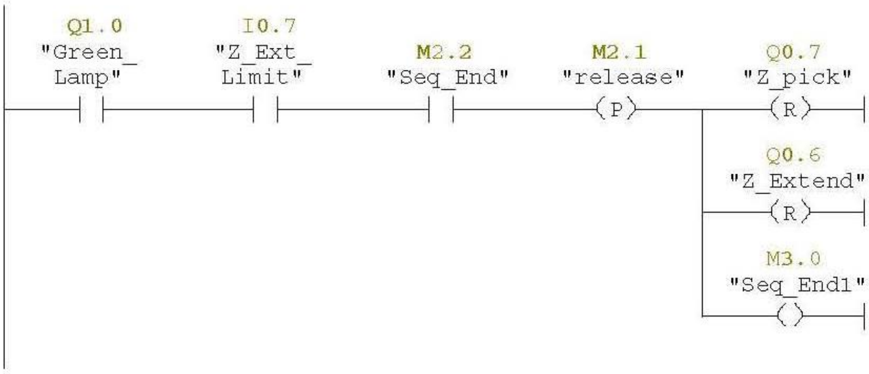
Network: 7



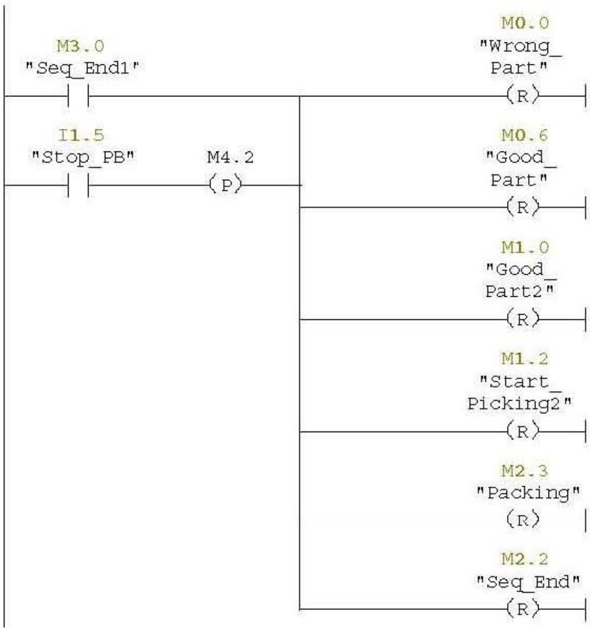
Network: 8



Network: 9



Network: 10



Network: 11