

Intelligent Sorting System

by

Kevin Soh Ping Han

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical and Electronic Engineering)

MAY 2004

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

E

TS

225

PK

S682

2004

- 1) Programmable controllers
- 2) Microcomputers

CERTIFICATION OF APPROVAL

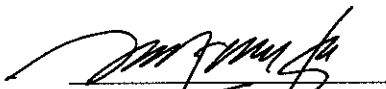
Intelligent Sorting System

by

Kevin Soh Ping Han

A project dissertation submitted to the
Electrical and Electronic Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(ELECTRICAL AND ELECTRONIC ENGINEERING)

Approved by,



(Rosdiazli bin Ibrahim)

ASR. Rosdiazli Ibrahim
Lecturer
Electrical & Electronic Engineering
Universiti Teknologi PETRONAS

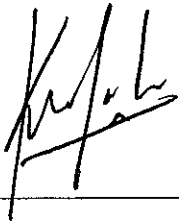
UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

May 2004

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



KEVIN SOH PING HAN

ABSTRACT

This project presents the design of an intelligent sorting system that integrates the pneumatic system and X-Y Robot as actuator, the Vision Sensor and the Programmable Logic Controller (PLC). This automated sorting system is designed with the main aim of reducing production time in manufacturing line that involves product sorting. In the industry, most sorting procedures are still done by humans which are error prone and slow. With the industry moving towards application of automation, this project is thus relevant.

The main objective in constructing this project is to integrate system components into a sorting system using PLC, actuators which are the X-Y Robot and pneumatic system, and the Vision Sensor. A Human Machine Interface (HMI) is created for the PLC to be controlled via personal computer using Visual Basic.

The scope of this project will be within PLC programming for system controller, pneumatic system design, setup and programming of the X-Y Robot, Vision Sensor control, Visual Basic HMI creation, and system design and integration of the whole system. Scope of literature review is within sorting system technology and approach.

This project starts off with preliminary research and literature review followed by tools identification. Following steps include PLC parameter set-up and programming, actuator design and setup, vision controller setup and system integration. Each component of the system is tested individually and then integrated into a complete system. The system is then tested to operate in production mode to cater for industrial objective, which in this case is to sort target objects.

The project achieves all the main objectives underlined. Each major component is designed, developed and tested to be functioning well. The system integration comprises all the relevant components, all connected and communicating with each other. System integration is complete and successful. The project is completed within the time frame and serves the problem statement.

ACKNOWLEDGEMENT

First and foremost, the author would like to thank God, for all His blessings and guidance which made all things possible for the author while undertaking the tasks in this project.

The author would like to take this opportunity to thank project supervisor, Mr. Rosdiazli bin Ibrahim. Throughout the period of the project, Mr. Rosdiazli had given the author guidance and technical knowledge any undergraduate would have hope for. Mr. Rosdiazli's ever-willingness to help is a true inspiration and it paths the way for the author to complete the project.

Thanks also to control laboratory technician, Mr. Azhar for providing the facilities and technical help in guiding the author during the project physical implementation. Mr. Azhar's experience as a technician is vital in helping the author to troubleshoot project components.

To fellow undergraduate mates and friends in Universiti Teknologi PETRONAS, special thanks for all the help, cooperation and teamwork that make the author's project execution and completion a successful and unforgettable experience.

Last but not least, the author would like to thank his parents and family for all their unconditional love, understanding and support throughout the period of undertaking the project.

TABLE OF CONTENTS

CERTIFICATION OF APPROVAL	i
CERTIFICATION OF ORIGINALITY	ii
ABSTRACT	iii
ACKNOWLEDGEMENT	iv
CHAPTER 1: INTRODUCTION	1
1.1 Background of Study	1
1.2 Problem Statement	1
1.2.1 Problem Identification	1
1.2.2 Significance of Project	2
1.3 Objectives and Scope of Study	3
CHAPTER 2: LITERATURE REVIEW	4
2.1 PC Control on Robotics	4
2.2 Programmable Logic Controller in Robotic Application	4
2.3 Modular Assembly Integration in Manufacturing	5
2.4 Fuzzy Logic Algorithm for a Robotic Sorting System with Colour Vision	6
2.5 Guidelines for the Design of Robotic Gripping System.	6
2.5.1 Reduce Cost	7
2.5.2 Increase Throughput	8
2.5.3 Increase Reliability	9
2.6 FieldBus Technology for Safety in Automation	10

CHAPTER 3:	METHODOLOGY	11
3.1	Procedure Identification	11
3.1.1	Preliminary Research	11
3.1.2	Project Planning and Research	12
3.1.3	PLC Setup and Assembly	12
3.1.4	X-Y Robot Setup and Assembly	13
3.1.5	Operation and Testing of X-Y Robot in Manual Mode	15
3.1.6	Pneumatic System Design and Setup	15
3.1.7	Vision Sensor Parameter Setup	16
3.1.8	Vision Sensor Measurement Setup	19
3.1.9	Vision Sensor Testing	21
3.1.10	System Visual Basic Interface Design	22
3.1.11	Serial Connection Setup between Interface and PLC	23
3.1.12	System Design and Implementation	24
3.1.13	System Process Identification and Ladder Diagram Construction	25
3.1.14	System Testing, Running and Monitoring	26
3.2	Tools Identification	27
CHAPTER 4:	RESULTS AND DISCUSSION	29
4.1	Findings	29
4.1.1	PLC Setup and Assembly	29
4.1.2	X-Y Robot Setup and Assembly	29
4.1.3	Operation and Testing of X-Y Robot in Manual Mode	32
4.1.4	Pneumatic System Design and Setup	33
4.1.5	Vision Sensor Parameter Setup	34
4.1.6	Vision Sensor Measurement Setup	36
4.1.7	Vision Sensor Testing	38
4.1.8	System Visual Basic Interface Design	45

4.1.9	Serial Connection Setup between Interface and PLC	46
4.1.10	System Design and Implementation	47
4.1.11	System Process Identification and Ladder Diagram Construction	48
4.1.12	System Testing, Running and Monitoring	49
4.2	Discussion	50
CHAPTER 5:	CONCLUSION AND RECOMMENDATION	60
REFERENCES	62
APPENDICES	64

LIST OF FIGURES

Figure 4.1:	Basic Architecture of the X-Y Robot	30
Figure 4.2:	Pin to pin connection of CV500-CIF01 (MCU) to RS-232C (PC)	31
Figure 4.3:	Servo driver to servo motor connection diagram	32
Figure 4.4:	Registered circle for experiment 1	39
Figure 4.5:	Position Compensation Grey Search value using perfect circle	39
Figure 4.6:	Grey Search Measurement of perfect circle	40
Figure 4.7:	The amount of displacement due to Position Compensation by shifting object to the right	40
Figure 4.8:	Grey Search Correlation OK for Position Compensation by shifting object to the right	41
Figure 4.9:	Grey Search measurement of circle with minor defect	41
Figure 4.10:	Grey Search measurement of circle with medium defect	42
Figure 4.11:	Grey Search measurement of circle with major defect	42
Figure 4.12:	Flow Chart of multiple scenes measurement for shape classification	55

LIST OF TABLES

Table 4.1:	The inputs combination to switch scene in the vision system	36
Table 4.2:	Grey Search Measurement Results for Circle, Square and Triangle based on Different Registered Perfect Model	43
Table 4.3:	Average Density Readings for Square and Triangle in multiple orientation	44
Table 4.4:	Compolet Series Table	46

LIST OF APPENDICES

Appendix I:	Project Flow-Chart	64
Appendix II:	Gantt Chart	65
	- Gantt Chart for First Semester	65
	- Gantt Chart for Second Semester	66
Appendix III:	Ladder Diagram	67
	- Ladder Diagram for X-Y Robot Manual Mode Operation	67
	- Ladder Diagram for Sorting System	68
Appendix IV:	Visual Basic Interface Layout	79
	- Visual Basic Interface Layout for X-Y Robot Manual Mode Operation	79
	- Visual Basic Interface Layout for Sorting System	80
Appendix V:	Visual Basic Codes	81
	- Visual Basic Codes for the X-Y Robot Manual Mode Operation Interface Layout	81
	- Visual Basic Codes for the Sorting System Interface Layout	84
Appendix VI:	Pneumatic System Design Layout	87
Appendix VII:	Connection Diagram of the Vision Controller to the PLC	88
Appendix VIII:	Vision Sensor Testing Results	89
	- Grey Search with Perfect Model: Circle	89
	- Grey Search with Perfect Model: Square	90
	- Grey Search with Perfect Model: Triangle	91
	- Average Density of Circle	92
	- Average Density of Square	93
	- Average Density of Triangle	94
Appendix IX:	Sorting System Process Flowchart	95
Appendix X:	System Design Layout of the Intelligent Sorting System	96

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND OF STUDY

The project creates an automatic sorting system consisting of the Vision Sensor, the Programmable Logic Controller (PLC), and an actuator system. There are two options for the actuator system. The options are the mechanical form using the X-Y Robot or the pneumatic form. The X-Y Robot can function manually or automatically by programming the PLC. The pneumatic system uses pressured air to move solenoid valves and also programmable through the PLC. The actuators and Vision Sensor are connected to the PLC. As the main controller, the PLC serves as the “brain” of the system. The PLC is interfaced to the personal computer using Visual Basic to create a user-friendly control environment from the personal computer for end user.

1.2 PROBLEM STATEMENT

1.2.1 Problem Identification

The project is done in response to the industry need of an automated sorting system in a typical manufacturing line. Most of the sorting systems in the industry are handled by operators manually. This process is prone to human error and reduces cycle time which will affect the production output amount. Uncertainty in the production rate and amount in a manually operated line will cause problems in projecting output and meeting demand.

In a manufacturing line, the normal process cycle consists of many stages. The actuator system and the vision sensing system might be independent systems. Separate systems take up more space plus adding extra processes which will increase cycle time. Time delay between stages is unpredictable and might cause a very high loss time. Therefore, there clearly is a need to reduce manufacturing stages and integrate more processes together.

Conventional actuator in manufacturing line normally functions in manual mode, requiring a reasonably high amount of attention from operator. This is a rather direct approach to controlling the actuator. Some functions required the sorting system to perform complicated movement. This requires the actuator to perform functions automatically. In the industry nowadays, the key to a good system is not on functionality, but rather more on versatility and flexibility. This is due to an increasing need to produce a variety of products on the same manufacturing line. In essence, the sorting system has to be programmable to cater various functions.

1.2.2 Significance of Project

This project is an integration of the Vision Sensor, PLC controller and the actuator system. The PLC serving as the main controller will operate the system in manual and automatic mode as required. More importantly, the usage of PLC ensures the system is programmable and can be tailored to a variety of functions. This project uses the PLC as controller, plus a Human Machine Interface (HMI) constructed with Visual Basic to create a user friendly interface for operators to manage in the industry. The Vision Sensor sets the visual conditions for the sorting system to distinguish and sort target objects. This creates an intelligent function to ensure a fully automated system.

The gist of this project is to create a flexible sorting system to cater for a variety of target objects. The system has to be programmable rather than fixed to sort only a single target object. This feature is desirable in today industrial manufacturing demand. Most company cannot afford to purchase a single machine for a single product. Moreover in a modern production such as semiconductor, the product life cycle is very short and virtually new devices are being introduced monthly. An excellent sorting system must be able to cater for a range of target products within its own category.

This system will increase production and cycle time in industry that utilize sorting mechanism comparatively to manual sorting. The system also reduces process stoppage or slowdown due to sorting errors. The system user friendly features coupled with automatic functions ease the work of manufacturing line operators. These are desirable attributes to a competitive industrial automation environment.

1.3 Objectives and Scope of Study

The project is relevant as it is an approach of integrating numerous independent systems into one process. It is a system design project and it emphasizes many layers of operation consisting of system components.

The objectives of this project:

1. To set up and develop the actuators: the X-Y Robot and pneumatic system to function in automatic and manual mode.
2. To set up and develop the Vision Sensor to determine the conditions for sorting target object.
3. To set up, develop and program the PLC controller for the automated sorting system.
4. To create and program a Human Machine Interface (HMI) for the system controller using Visual Basic.
5. To integrate the vision sensor, actuators, and PLC into a single sorting system.

The scope of study of this project is the integration of the actuator and the vision sensor to create an intelligent sorting system using PLC as controller. The project deals with designing the function of the sorting system to serve industrial purpose. The procedures will involve PLC programming for system controller, X-Y Robot setup and programming, pneumatic system design, vision sensor control, Visual Basic HMI creation, and integration of the whole system. This project does not recreate or redesign the components of the sorting system. It is not essentially a machine design. It is a system design.

This project is feasible within two semesters with proper planning and organization of work. This project will be divided into 4 phases:

- 1 Phase I: Actuator setup and development, with work on X-Y Robot parameter setup and programming, pneumatic system design, and controller using PLC.
- 2 Phase II: Vision Sensor set up with work on vision controller, plus integration to the PLC.
- 3 Phase III: Construction of HMI and its interfacing to the PLC.
- 4 Phase IV: System integration, testing and troubleshooting.

CHAPTER 2

LITERATURE REVIEW

2.1 PC CONTROL ON ROBOTICS

With the growing acceptance of PCs based on Intel Pentium processors and Microsoft Windows NT operating systems, open solutions are spreading into most areas of factory automation. Key benefits for users include communications, connectivity, component standardization and availability, and cost reduction. Studies of robot programs reveal actual robot control code is a small percentage of most programs' total code. Most program lines deal with initialization, logic testing, and branching. Standard PC languages can handle tasks like these, so it was only natural to build a standard control platform on one-Visual Basic with special robot control extensions added .[1]

Therefore, the concept of using a PC Control robotic sorting system is of great importance to the industry. As commercial demands become more competitive, it is only natural to create a system that reduces cycle time in the manufacturing that will yield higher output. This can be easily and conveniently achieved by implementing information technology into conventional mechanical machine, serving as controller or sub-controllers.

2.2 PROGRAMMABLE LOGIC CONTROLLER (PLC) IN ROBOTIC APPLICATION

PLCs are used in automated workcells. In fact, a PLC is usually the main controller in a workcell. Even in a workcell with a robot, the robot is usually a slave to the far cheaper PLC. Why not use the expensive robot to control the PLC? PLCs are optimized for supervisory control, whereas robots controllers are optimized to control a single actuator. The robot controller is designed to perform the actions sequentially. A robot program might take from a few minutes to several hours to execute just once. A PLC, on the other hand, executes a full scan cycle every milliseconds, executing all its control actions almost simultaneously. [2]

This explains the reason of using the PLC as the controller of the system. Since it is possible to directly control the robot with PLC, it will not be implemented as the X-Y Robot needs to be commanded by its native language which is the G-code. However there are systems which are naturally suited for direct PLC connection. The pneumatic system is one such example.

It is much easier to program sequential operations in the robotic programming language, which inherently finishes one instruction before executing the next. To program PLC to move the robot, latch bits are needed to keep track of the sequential step that is supposed to be active. The robot controller is also optimized for control as it contains programs in its ROM memory that coordinates the movement of all motors. PLC program, although possible will be more complex, less flexible and less optimum to control robot directly. [2]

2.3 MODULAR ASSEMBLY INTEGRATION IN MANUFACTURING

The modular assembly system is the concept of designing automation to cater for many functions that the machinery could be used for. The essence of modular system is to spend expensive design time working on parts of the machine that are in contact with the product, while the other related components should be generic. The machinery designed in a modular system could be modified to cater many different products because it is designed this way in the beginning. A perfect example of modular system is automation with integrated vision system, which is the feature of this project. The objective of modular approach is optimum accuracy, repeatability, and performance. The project is to be targeted along this concept. It does not have to be designed perfectly for one specific function. Rather what the industry is looking for today is a machine that could cater new products and support incoming technology.

2.4 FUZZY LOGIC ALGORITHM FOR A ROBOTIC SORTING SYSTEM WITH COLOUR VISION

The algorithm implements images understanding based on line segment coded algorithm to approximate shape of target object. Implement real-time object tracking and manipulation, allowing the robot to hold the objects successfully with two line segments of object boundaries with following conditions:

- a. Two line segments should be parallel or near parallel
- b. The object body must exist between the two line segments
- c. There is the projection of two line segments on each other
- d. The handling area, or pick-up section, formed by the projections of the two line segments on each other and their distances sides, must satisfy size constraints of the robot end-effectors.

Fuzziness is used to determine best two lines if there are more than 2 that satisfy all the conditions. The vision system also worked well under varying lighting conditions indicated by some experiments done on the algorithm.

2.5 GUIDELINES FOR THE DESIGN OF ROBOTIC GRIPPING SYSTEMS

Gripper design is an important and often overlooked aspect of the design of a complete assembly system. The shift of automation to a more reconfigurable and reprogrammable system places an even more important leverage on end-of-arm tooling design. The design of a gripping system is not a trivial task, and most often overlooked until the system is almost complete at which point too little time or resources left to perform a good job. Such complacency should be avoided and ideally the design of the gripper should occur concurrently with the design of the rest of the system. The reason literature review was done on the design of the gripper is that many texts and references discuss types of grippers and the quality of grasp without placing much importance in the area of design parameters and guidelines for end effectors. In this case, the organized listing of the gripper design guidelines are arranged into three major categories: guidelines to save cost, guidelines to increase throughput, and guidelines to increase reliability.

2.5.1 Reduce Cost

The objective of saving cost in gripper design must extend beyond the immediate gripper cost. An extra investment in gripper design might be beneficial in the long run as the system may need not continual operator attention due to grippers dropping or mishandling parts.

Use parallel or rotary motion pneumatic actuators

Generally, the use of parallel or rotary motion actuators is less expensive. An important factor in saving cost is the selection of the gripper actuator. Although at first, it seemed a more expensive actuator is required, a more clever and creative design can utilize one of the less expensive units.

Use off-the shelf components in gripper designs

It is often advantageous and desirable to design a solution that works with the best readily available gripper. Off-the shelf components usually have a large installation base of users who have already debugged the design. They are specially designed for their intended purposes, more reliable, and make repair or replacement of components a quick and easy process. This factor makes this idea a cost-saving option in the long run operation of the system.

Handle multiple parts with a single gripper

Rotary wrist mechanisms and automatic tool changers should be always avoided unless absolutely necessary. With more time spent on design and careful evaluation, most likely a single actuator with properly designed fingers can easily handle multiple parts.

2.5.2 Increase Throughput

Increasing throughput can be achieved in many ways such as increasing the robot speed, decreasing the chance of waiting for a part to arrive, or increasing the functions and versatility of the gripper to do more than just handing parts.

Minimize the interference measure

Using the smallest interference measure possible is important in flexible feeding applications where parts may be situated close to one another and at random orientations. Too large interference measure will disable the gripper from gripping the part.

Chamfer the exterior of gripper feature

This technique allows the gripper to displace neighboring parts as the target part is being approached.

Minimize the gripper weight

Reduced weight allows increase in acceleration and reduces overshoot with regard to a fixed payload capacity of the gripper in the system.

Grasp parts securely

This objective is easily achieved by designing the geometry of the gripping surfaces of the gripper fingers to compliment the shape of the part being handled. Advantages include higher speed thus reducing cycle time.

Avoid tool changes

Automatic tool changes are time consuming and this extra time consumed decreases the throughput of the system.

Grip multiple parts with a single gripper

This avoids too changes. It is also a good alternative to design multiple gripping surfaces actuated by a single actuator.

Install multiple grippers on a single wrist

This also avoid too changes and allows the robot to have more than one gripper ready for use and may decrease cycle time and reduces robot motion that contributes to wear and tear.

Increase functionality in gripper fingers

This can speed the system by allowing the gripper to perform a task that would usually be done by an additional piece of hardware.

2.5.3 Increase reliability

System reliability is generally enhanced by designing grippers to work in the presence of errors and maintain a secure and accurate grasp of components.

Grasp parts securely

This decreases the likelihood that the part will be dropped or will shift in the gripper during motion and misaligned when placed.

Fully encompass the part with the gripper

This helps hold the part securely and align the part in the gripper jaw in the presence of uncertainties in the pickup location.

Do not deform the part during grasping

Deformed parts will cause problems when trying to insert the part into a fixture or adding it to an existing sub-assembly because of its change in shape.

Minimize finger length

The longer the fingers, the more they are going to deflect when grasping and this causes misalignment and the quality of the grasps suffers.

Provide an Ample Approach Clearance

It is important to ensure that there is enough clearance to approach the pickup location so that if there is some uncertainty in the location of the part, a collision will not occur.

Design for proper gripper-part interaction

Important aspects include the shape compatibility of the gripper-part and also the amount of friction. In some cases, low friction interface is desired so parts may slide relative to gripper jaw for alignment purposes. However, in cases when a flat surface is used, then a high friction is desired since the part would be aligned anyway and the higher friction increases the secureness of the grasp.

Do not rely on added parts for location

Added parts increased the possibility of the parts being misplaced or missing, which will fail the assembly operation. It is therefore desirable to use permanent features so that if there was an error in a previous step, the current operation will not be affected.

Assembly grippers should align parts

Added features which align parts before assembly operation takes place should be incorporated into grippers to reduce alignment problem during the assembly operation.

Incorporate functionality into gripper fingers

Increase in functionality of the gripper avoids dedicated assembly hardware task. There is less risk of mishandled part and alignment problem when a gripper can perform many functions with a single grip of the part.

2.6 FIELDBUS TECHNOLOGY FOR SAFETY IN AUTOMATION

The fieldbus is a technology incorporating a network to control the machinery. It introduces less wiring, reduced complexity, faster installation, distributed intelligence and self-diagnostic. In a sense, it gives more control and monitoring to the automation, and these processes can be done remotely. More importantly, it allows a rapid fault identification and rectification compared to the conventional safety system of using manual push buttons. This concept could be implemented in the project but it is well beyond the scope of study. It is a good recommendation, and perhaps could be a new additional project to the sorting system.

CHAPTER 3

METHODOLOGY

3.1 PROCEDURE IDENTIFICATION

The procedures identified are methods and approaches that have been implemented to achieve the objectives underlined for this project. Procedures underlined are not necessarily executed in the following arranged order and more often than not, they are repeated when the need arises.

3.1.1 Preliminary Research

The preliminary research involves researches and work done by others related to the project. This step also includes literature review of materials regarding robotic control, PC control of hardware, vision system and also PLC application as controller. Additional information could be obtained from published journals on related automation field such as fieldbus technology and sorting techniques. This research supports the relevancy of the project and its potential application in the industry. Following research includes identification of the components of the system. The tools identification includes software and hardware involved in the construction of the system.

Research for the components of the project involves obtaining information from the OMRON manuals regarding the system's hardware setup and operation. The manuals are separated into the Programmable Logic Controller, Motion Control Unit (MCU), servo drivers, Vision Sensor and photoelectric switch. Other references have to be obtained to provide information on setting up and configuration of the parts in the pneumatic system. All manuals provide an extensive hardware connection, setup procedure and components functionality. Research is directed and focused on methods to implement the functions of the respective components to achieve project objectives rather than to understand the components completely.

3.1.2 Project Planning and Research

The procedures will be divided into 4 phases for system construction and integration. Phase I will deal with the configuration and setup of the X-Y Robot and the pneumatic system. The phase will complete with integration of PLC controller to the actuator system. Phase II will involve the set-up of the Vision Sensor and also its integration to the PLC controller. Phase III involves creating a Human Machine Interface using Visual Basic to provide an interface to control the PLC from the personal computer. The following and final phase which is phase IV will be system integration, testing and troubleshooting. This will be done after hardware implementation of the whole system. Refer to Appendix I for project flowchart and Appendix II for Gantt Chart.

3.1.3 PLC Setup and Assembly

The PLC serves as the main controller of the system. It controls the X-Y Robot, the pneumatic system, the vision system and also the supporting components: photoelectric switch and conveyor belt. In the sorting system, the PLC essentially controls all the functions whether in manual or automatic mode. The PLC and the accompanying CX- Programmer software to create ladder diagram is initialized using the following steps:

1. Ensure PLC is connected to the computer via RS-232 cable to com port.
2. Switch on Computer and PLC.
3. Open the CX- programmer software.
4. Set the PLC type as C200HE.
5. Set the PLC CPU setting to CPU 42
6. Set the driver to COM1 or COM2

The PLC assembly is done using the PLC C200HE CPU 42 with com port connection to the personal computer. The PLC C200HE is a modular type of PLC by OMRON. This is one of the medium types of PLC which provides up to a total of 16 inputs and 16 outputs. The Motion Control Unit is attached to the PLC. The PLC is programmed through ladder diagrams constructed with the CX-Programmer Version 2.0.

The PLC controls external devices via ladder diagram programming. The ladder diagram can be constructed after this initial setup. To transfer the ladder diagram from the computer to the memory of the PLC, the following steps are required:

1. Construct and end the ladder diagram.
2. Set the CX-programmer to online mode.
3. Set the PLC to program mode.
4. Transfer the program to PLC.
5. Set the PLC to run mode.

With that the ladder diagram will be stored in the PLC's memory until being overwrite. The program can be triggered with the input switches or any external switches. The output can be indicated on the PLC lamps or any other external devices. The inputs and outputs can also be forced to trigger ON or OFF via the CX-Programmer itself.

While the PLC is used as the main control of the system, other forms of programming is needed to complement the system. Besides ladder diagram, the system also uses G-code for the X-Y Robot automatic mode and Visual Basic code for the system interface. The PLC only fully controls the pneumatic system, vision sensor and the manual operation of the X-Y Robot. The PLC alone is sufficient to operate the robot in manual mode. The steps required to operate the robot in manual mode are servo lock, origin search and jogging in sequential order. This is possible using the PLC only as each function is assigned a PLC address identified by the Motion Control Unit.

3.1.4 X-Y Robot Setup and Assembly

The hardware setup of the X-Y Robot includes physical connection among the MCU, servo drivers, servo motors, PLC, and personal computer. The PLC and MCU are connected to the PC via com port, so it is important to have two com ports on the PC. The connection from the MCU to the PC via RS-232C has to be handled with care as there is a specific pin to pin assignment between the 25 pin MCU side and the 9 pin PC side. The MCU is connected to the servo drivers on dedicated cables. The servo drivers are connected to the servo motors on the robot through dedicated cables. The

limit sensors and the origin sensors are connected to the input output module, which is connected to the MCU. This module sends the signals from the sensors to the MCU for processing and interpretation. 240 AC power supply is needed for the PLC, PC, and servo drivers. The servo drivers and MCU will need an additional 24V DC supply. This can be tapped from the 24V output of the PLC.

Compliment to the hardware setup, the related software that is used to control the hardware is installed on the personal computer. Testing and simulation are carried out to ensure response from the hardware through commands from the software. The CX-Motion is a software to construct and transfer machine parameters and G-code programs to the MCU on the PLC. The following parameter need to be setup on the CX-Motion:

Number of axis: 2
Number of tasks: 1
Task 1 axis: XY
Task 2 axis: *

Further settings are done involving the machine, coordinate, feed rate, and servo parameter. The machine parameters include minimum setting unit, display unit, pulse rate, and positive and negative software limits. The coordinate parameters include reference origin offset value and workplace origin offset value. The feed rate parameters include maximum high-speed feed rate, maximum interpolation feed rate, origin search high speed, origin search low speed and maximum jog feed rate. The servo parameter is set for the in position. All the parameters are saved in a file name.

The test and run procedure is done after all hardware and software have been setup and initialize. A program of the ladder diagram operation in manual mode is used to testing purposes. The robot can be tested to run in manual mode or automatic mode. In the manual mode, only the PLC and MCU are involved while in the automatic mode, the MCU needs a G-code program. Testing procedures include all error detections and troubleshooting.

3.1.5 Operation and Testing of X-Y Robot in Manual Mode

Each task of the MCU can be executed either in manual or automatic mode. In the automatic mode, MC programs created using CX-Motion software in G-code form are downloaded to the MCU and executed. In the manual mode, manual commands from the PLC are executed. These commands are programmed directly by constructing a ladder diagram using the CX-Programmer. Testing in the manual mode requires a ladder diagram that provides such step-by-step function as indicated below:

1. Set ON Servo Lock X (105.09)
2. Set ON Servo Lock Y (107.09)
3. Set ON Origin Search X (105.01)
4. Set ON Origin Search Y (107.01)
5. Set ON Jog X (105.03) and X-direction (105.13)
6. Set ON Jog Y (107.03) and Y-direction (107.13)

Each function is shown with the address in the PLC in parentheses. To trigger for example servo free X, a switch is used to trigger the output bit 105.90. The ladder diagram for this manual mode operation is shown in Appendix III. The first step in manual mode operation is servo lock X and Y axis. Servo locking clamps the two axes and ensure only the servo drivers can move the robot. External force applied will not move any of the axes.

3.1.6 Pneumatic System Design and Setup

The pneumatic system is used as an actuator system in the project. Just like the X-Y Robot, it is an alternative for the system actuator system. The pneumatic system uses the pressured air supply in the region of 30 psi as a main moving force. It does not involve electrical supply or any mechanical force by motors like the X-Y Robot. It operates by receiving electrical signal from the PLC to perform its functions. Therefore it is a raw system with no intrinsic controller and thus it is fully and directly controlled by the PLC. Choices between using the X-Y Robot or the pneumatic system are selected after setting up and evaluating the pros and cons of each system. These are reported and discussed in the system design section.

The first step in designing the pneumatic system is to define the requirement and come up with the standard design sketch of the system. Standard sketch shows the air flow direction and the electrical connection. This is followed by identification of the components. In the system, it is defined that objects are sorted into three categories. Therefore, three solenoid valves and three valves are required for the system. Each valve is used to supply air to push or pull the one solenoid valve. The next step involves electrical and pneumatic connection. While electrical connection uses normal wires, the pneumatic system requires hollow air tube. Connections are done referring to the standard design sketch. After all connections are done, it can be tested by connecting the electrical inputs of the valves to the output of the PLC.

3.1.7 Vision Sensor Parameter Setup

Before the Vision Sensor is to be used, some parameters setup needed to be done. The setup of the Vision Sensor was done together with some initial testing to get an idea on the capabilities of the system. The first step is to identify the components of the Vision Sensor. The basic components of the Vision Sensor are:

1. F150-C10E-3 Vision Mate Controller
2. F150-S1A Camera
3. F150-VS Camera Cable
4. F150-KP Operating Console

Other accessories of the Vision Sensor such as the external monitor and the tripod to hold the camera have to be identified and obtained too. The next step is to determine the operating environment of the Vision Sensor. The F150 Vision Sensor is highly reliable and resistant to most environmental factors. However it is recommended that the system is not to be operated at temperature outside 0-50°C, places with rapid temperature fluctuations, relative humidity outside 35% to 85%, contains corrosive or flammable gasses, and direct exposure to sunlight.

Background suppression has to be done to an image before a measurement takes place. The back ground suppression (BGS) is done to exclude the background of the measurement object from the measurement process. BGS changes image areas with densities below the lower limit to 0 and images areas with densities above the upper limit to 255. Images areas with densities between the lower and upper limits are graded from 0 to 255 so that only images with densities between the lower and upper limit are measured. The following steps are done for BGS setting:

1. On the Menu of the vision system, go to ADJUST
2. Go to BGS Level
3. Set the Upper limit
4. Set the Lower limit
5. Click OK

The origin has to be set to determine the physical coordinates of the image being measured. This origin will be used to output results of the object deviation from the position of the origin. These values can indicate the consistency of the placement of object under the camera. The origin is useful for the Position Compensation feature of the Camera. The origin can set as followed:

1. On the Menu of the vision system, go to ADJUST
2. Go to Calibration
3. Go to Parameter Input
4. Set the coordinate system
5. Set the magnification
6. Set the unit
7. Click END

The next step in setting up the Vision Sensor is adjusting the light level. There are 15 lighting patterns registered with the F150. The lighting can be set simply by going through the different patterns in order or selecting the one that gives the clearest image. The light level for each section can be set separately to a value between 0 and 7. The 0 value means unlit state and the higher the setting value, the higher the light level. The following procedures apply for the lighting setup:

1. On the Menu of the vision system, go to ADJUST
2. Go to Light Control
3. Switch a lighting sample pattern using the SHIFT + Left/Right keys
4. For fine adjustment of light levels, move to the desired digit and adjust
5. Press ENT to store setting.

To use the terminal blocks as output, there is some setting to be done. However the default setting is to transfer output through the terminal blocks. The following communication settings are done:

1. Go to SYS
2. Go to Communication
3. Go to Terminal Blocks
4. Set the On state
5. Set the Handshaking feature
6. Set the output period
7. Set Gate ON delay

3.1.8 Vision Sensor Measurement Setup

The Vision Sensor measurement setup is the function that determines the requirement of the target object. It determines which target object is acceptable and which is not. There are two functions to the measurement procedure, the Position Displacement Compensation and Measurement.

The position displacement compensation function is used when the position and orientation of measurement objects are not consistent. By using this function, the displacement between the reference position and the current position is obtained, and this displacement is compensated for in measurements. The following procedures ensue for the position displacement function:

1. On the Menu of the vision system, go to Position Compensation
2. Go to Adjust
3. Adjust the suitable Filtering and BGS
4. Go to Region
5. Draw the region of target object
6. Register to region, which is known as the model
7. Set the range in which a search is to be made for the model
8. Set the evaluation criteria
9. Go to Direction
10. Select the reference values for X, Y and θ
11. Select END

The measurement is used to evaluate the target object and determines whether it is acceptable or not. There are 9 different functions of measurement which the vision system is capable of; however for the interest of this project, using two functions are sufficient. The functions to be used are the Density Averaging and Grey Search.

The Density Averaging is obtained for each pixel and measurements are made based on the average density. This feature could not precisely process an image accurately; rather it is more likely to be used as measure of the existence of the object. The steps of Density Averaging are simple and involve only drawing the measurement region and setting the evaluation criteria.

The Grey Search registers an image pattern (the model) in advance. The part of an input image that is most similar to the model is found, and the degree of similarity is expressed with a correlation value. With Grey Search, the position is obtained in pixel units. The following steps are done for the Grey Search setup:

1. On the Menu of the vision system, go to Measurement Region
2. Go to Grey Search
3. Go to Model Registration
4. Draw the region showing the image of perfect target object
5. Register to region, which is known as the model
6. Set the range in which a search is to be made for the model
7. Set the evaluation criteria
8. Select END

When all the setup has been done, the measurement is done by two methods. The first is to press the TRIG button on the console. The other alternative would be to input a signal to the STEP on the vision controller terminal block.

Setting the evaluation criteria indicates setting the range of correlation value. The extent of conformity is expressed as a correlation value, between 0 to 100, where 100 is perfect similarity. A lower limit and upper limit threshold can be set on this correlation value and the system will indicate OK for image above threshold and NG for image below threshold. This function enables the system to detect shapes or even patterns or characters based on the settings of the correlation value. The range will be set depends on the stringent ness of the measurement required. However care has to taken to ensure the vision controller does not reject acceptable images due to too small of a correlation region.

3.1.9 Vision Sensor Testing

To evaluate the capabilities of the vision system, experimentations and simulations are carried out on three different target objects: circle, triangle, and square. The target objects and the simulation results will be integrated into the complete sorting system if proven successful. Underlined below are 7 experiments carried out:

1. Approximation to a circle using Grey Search
2. Differentiate circle from square and triangle using Grey Search
3. Differentiate square from triangle and circle using Grey Search
4. Differentiate triangle from square and circle using Grey Search
5. Average Density testing for Circle
6. Average Density Testing for Square
7. Average Density Testing for Triangle

For each experiment, the perfect model is being registered first such as circle for the first and second one, square for the second one, and triangle for the third one. The vision system is triggered for each criterion using the three mentioned shapes. For the first one, some deforms circle objects are used as measurement subjects. The second, third and fourth experiment test for all three shapes. The fifth, sixth, and seventh experiment test for circle, square, and triangle respectively placed in different orientation.

Since the first four experiments use the Grey Search measurement, the important aspect is the setting of the lower and upper limit of the correlation value. The range of the correlation must be chosen to be suitable to reject the undesirable shape. However, the correlation range cannot be too stringent so that the position displacement can be done even though the object is not acceptable.

Average Density depends on the area of the object under the vision with respect to a registered search region. For the next three experiments that use the Average Density measurement, the acceptable figure has to be evaluated a few times to obtain repetitive average density values of the object. These values are analyzed to obtain a suitable density value range to pass the object and reject other shapes. The range is then set to pass the object.

3.1.10 System Visual Basic Interface Design

The Visual Basic interface is a simple controller interface constructed with the objective of creating a creative, user friendly and pleasant looking controller for the end user. Without the interface, the system could be directly controlled by the switches in the PLC. This requires direct access to the PLC, while controlling the system from the interface in the personal computer enables user to control the system from a remote place as long as the connection between the PLC and the personal computer is established. Moreover, Visual Basic offers good aesthetic tools such as text labeling, color coding, click button definition and also good processing tools such as database access.

The interface design is done using drag and drop functions of the Visual Basic. The command, label and text caption can be done using the simple functions listed on the software toolbars. The timer function is also on the toolbars and could be programmed on the delay duration and source of trigger. Any objects on the interfaced can be coded by double-clicking on them. Coding is based on C programming. It uses instruction similar to C and is less stringent comparatively to C. Proper Visual Basic code was typed to each object in the interface to communicate with the PLC.

The interfaces are done on the manual mode operation of the X- Y Robot and on the operation of the overall sorting system. The first step in designing the interface is to identify all the components or processes that need to be controlled. Each entity will then be assigned to one click button on the interface. The click button is then coded as required with reference to its function.

For the manual mode operation of the X-Y Robot, it covers the basic functions of servo lock, origin search and jogging in both directions for both axes. For the sorting system, the interface provides control for the pneumatic system, vision controller and the conveyor belt. The interface provides operation in manual and automatic mode. In the manual mode, each component can be controlled independently. This is suitable for troubleshooting single component. In automatic mode, the system automates itself and integrates the functions of all components, responding in an orderly and required fashion as programmed by the PLC.

3.1.11 Serial Connection Setup between Interface and PLC

The ladder diagram of the manual mode operation of the X-Y Robot or the sorting system is downloaded to the PLC using the CX-Programmer. After the ladder diagram is stored in the PLC, the following steps ensue:

1. Set the PLC in Monitor Mode and Work Offline
2. Open the FinsGateway Service Manager
3. Set the CPU Unit and Serial Unit to RUNNING
4. Set the Local Network and Units to be compatible with PLC.
5. Open Visual Basic Interface
6. Click Project→ Components→ OMRON SYSMAC C Control
7. Drag the Control Object to the Interface
8. Set the Properties of the Control Object as shown:
 - Network Address: 1
 - Node Address: 240
 - Unit Address: 0
9. Code for the Visual Basic with reference to addresses in the ladder diagram

These will ensure the Visual Basic set up a connection to the PLC through the node and unit address. The testing procedure of the Visual Basic communication to the PLC is done by running the interface while observing the PLC response. If the communication is not setup properly, the Visual Basic program will trigger an addressing error upon running. While the CX-Programmer is used to command the functions of the PLC, it also serves as a monitoring tool. By clicking the monitor icon, the CX-Programmer will indicate the flow of current path on the ladder diagram. By matching the monitor mode observation of the ladder diagram to the codes of the Visual Basic, the effectiveness of the communication can be observed.

3.1.12 System Design and Implementation

The system design consists of component identification and placement, system layout design and integration of all components in the sorting system. Its physical implementation will ensue upon completion and verification of system design. This is the last stage of the project, followed by testing and troubleshooting procedures. It is important to get the system design planned properly to ensure all the components are optimally arranged to serve its respective purpose. A good design will path the way for easier physical implementation with less time and effort spent on troubleshooting and verifying errors.

The actuator design involves pneumatic design and the X-Y Robot. The overall design of the system involves the integration process of the vision sensor, the actuator system, and the PLC. The system design involved two important aspects. The first is component placements and the second is the wiring connections. The wiring connections include both the electrical wiring and the air tube wiring.

The component placement process involves:

1. Identify all components
2. Identify the placement of the vision camera on the conveyor
3. Identify the height of the vision camera to the target object
4. Identify the placement of the solenoid valves on the conveyor
5. Identify the placement of X-Y Robot pick and place arm on the conveyor
6. Identify the sorting bin, its classification and placement
7. Identify the placement of PLC
8. Sketch the overall design

The wiring connections process involves:

1. Identify all electrical inputs and outputs
2. Identify the electrical connection from the PLC to the personal computer
3. Identify the electrical connection from the vision system to the PLC
4. Identify the electrical connection from the PLC to the pneumatic system
5. Identify the air tubing connection from the within the pneumatic system
6. Identify the electrical connection within the X-Y Robot components
7. Identify the electrical connection from the X-Y Robot to the PLC.
8. Sketch the circuit diagram
9. Sketch the pneumatic diagram
10. Implement the diagram

3.1.13 System Process Identification and Ladder Diagram Construction

The system process indicates the working flow of the system. It identifies the key stages of the system, determines the order of the stages and underlined dependencies between stages. The ultimate product from this procedure is a comprehensive system flow chart. From the flow chart, programming of the ladder diagram and Visual Basic can proceed. This is a vital procedure especially to operate the system in automatic mode. The step by step procedures are:

1. Identify the objectives of the system
2. Identify the functions of system's components
3. Identify stages of the process flow
4. Identify sequential order of the stages
5. Identify decision making processes
6. Construct system process flow chart based on stages identified
7. Based on flow chart, construct ladder diagram
8. Download ladder diagram to PLC
9. Test and troubleshoot the system

3.1.14 System Testing, Running and Monitoring

The test and run procedure is done after all hardware and software have been setup and initialized. The system ladder diagram program is constructed and used to test the functionality of the system and its underlying components. Testing procedures include all error detections and troubleshooting. Testing is done on individual components at initial stages, and then on the overall system after integration of all the components and physical implementation.

While the software is used to command the functions of the hardware, it also serves as a monitoring tool. The PLC can be monitored through the CX-programmer by triggering the monitor icon. This will indicate the flow of current path on the ladder diagram. The MCU condition can be monitored through the CX-Motion software with the MC monitoring function. It indicates the coordinates and pulses of the X and Y axis. It also shows the code of error if there is any from the MCU. Errors are coded systematically and are available in the MCU manual. For example, an error code of 62 indicates Clockwise Over Travel Error.

Monitoring of the pneumatic system is done physically by observing the in and out movement of the solenoid valves. The vision system is monitored by observing the video display on the external monitor. The display indicates the acceptance of images by showing OK for good images and NG for rejected images. The results of the vision system images can also be observed from the PLC when the terminal blocks are connected to the PLC.

3.2 TOOLS IDENTIFICATION

The tools that are required for the execution and completion of this project are categorized into software and hardware. The software is basically the controller, and the hardware is the controlled component.

The software consists of:

1. CX-Motion from OMRON

A Window based G-code programming software to set X-Y robot parameters and provide G-code instruction to the MCU to operate the X-Y robot in automatic mode

2. CX-Programmer for PLC C200HE

A ladder diagram programming software to create ladder diagram to operate the X-Y robot in manual mode or trigger the G-code in automatic mode. Ladder diagram is used to program the functionality of the sorting system in manual and automatic mode.

3. Visual Basic 6.0

A Human Machine Interface (HMI) software to control the PLC via the personal computer through a graphical mean from the monitor.

4. FinsGateway

A communication interface for all OMRON Factory Automation networks that allows shared data including data link.

5. SYSMAC Compolet Version 2

A software component that provides application programs with functions so that SYSMAC C series PLC which is connected to computer, can be easily controlled from the computer.

The hardware is divided into:

1. X-Y Robot

The X-Y Robot is basically divided into a Motion Control Unit C200H-MC221 (MCU), two servo drivers, two servo motors, and the robot itself. The MCU is the device that controls the robot and holds the G-code program. It is mounted on the PLC and interfaces the robot to the personal computer. The servo drivers drive the servo motors on the X-Y robot to move the robot in X axis and Y axis.

2. Pneumatic System

The Pneumatic system uses solenoids valves that are connected to air supply and to the PLC. Using air as a medium for mechanical movement, it is an efficient and power saving alternative. This is the actuator for the sorting system, an alternative for the X-Y Robot. Component include 3 double acting cylinders (solenoid valves) and 3 5/2(14) Valves.

3. PLC C200HE

The PLC is the main controller of the system. It consists of switches to trigger the operation of the robot in manual or automatic mode. It operates according to the ladder diagram programmed and transferred using the C-X Programmer.

4. Vision Camera F150 series OMRON F15-C10E-3

The vision camera is the requirement setting of the sorting system. It is to be integrated to the actuators to create a sorting system to sort target objects based on requirements that are set and sensed by this vision sensor. Components involve controller, camera and operating console.

5. Sony External Monitor

This external monitor is connected to the vision controller. It serves as the display of real time images captured by the vision system camera. Settings of the vision controller and the results of simulation are done and shown on this external monitor.

6. Photoelectric Switch E3S-AT21-L (transmitter) and E3S-AR21 (receiver)

Manufactured by OMRON, the photoelectric switch consists of a transmitter and a receiver. It sends an output signal when the light transmitted failed to reach the receiver. This sensor switch is used in the first stage of the process to detect the presence of object.

7. Conveyor Belt

The conveyor belt is used to move the target objects to the various stages of the system. The target object will pass through the photoelectric switch, vision camera, and the actuator system on this conveyor. The starting point and uploading of object is on the conveyor.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 FINDINGS

4.1.1 PLC Setup and Assembly

The connection of the PLC to the personal computer is working perfectly via com port. The C-X programmer version 2.0 is able to transfer the programmed ladder diagram to the PLC successfully. The PLC has been tested to run successfully on run, program and monitor mode. Online editing function has also been tested successfully. The addresses corresponding to the inputs, outputs and intermediate outputs have been identified and tested.

External connections of the PLC are tested by connecting the terminal blocks of the vision controller, the conveyor belt, photoelectric switch and the pneumatic system to the PLC. All components work well upon output signal from PLC. In the case of the photoelectric switch, the PLC can read the signal generated from the switch. Therefore, PLC is able to drive the actuator, vision system and the conveyor belt at the same time able to read the vision system and photoelectric switch output. This desirable characteristic justifies the application of PLC as the main controller of the system. This narrows down any troubleshooting procedures of the system operation to the ladder diagram stored in the PLC, instead of separate control entity in each component.

4.1.2 X-Y Robot Setup and Assembly

The X-Y Robot assembly involves collecting all related hardware that is essential to the operation of the robot. The required devices are the PC, the PLC, the MCU, two servo drivers and two servo motors. An important aspect of the physical arrangement is the connection of each hardware to another. A correct connection will ensure good operation while enhancing the safety of the system.

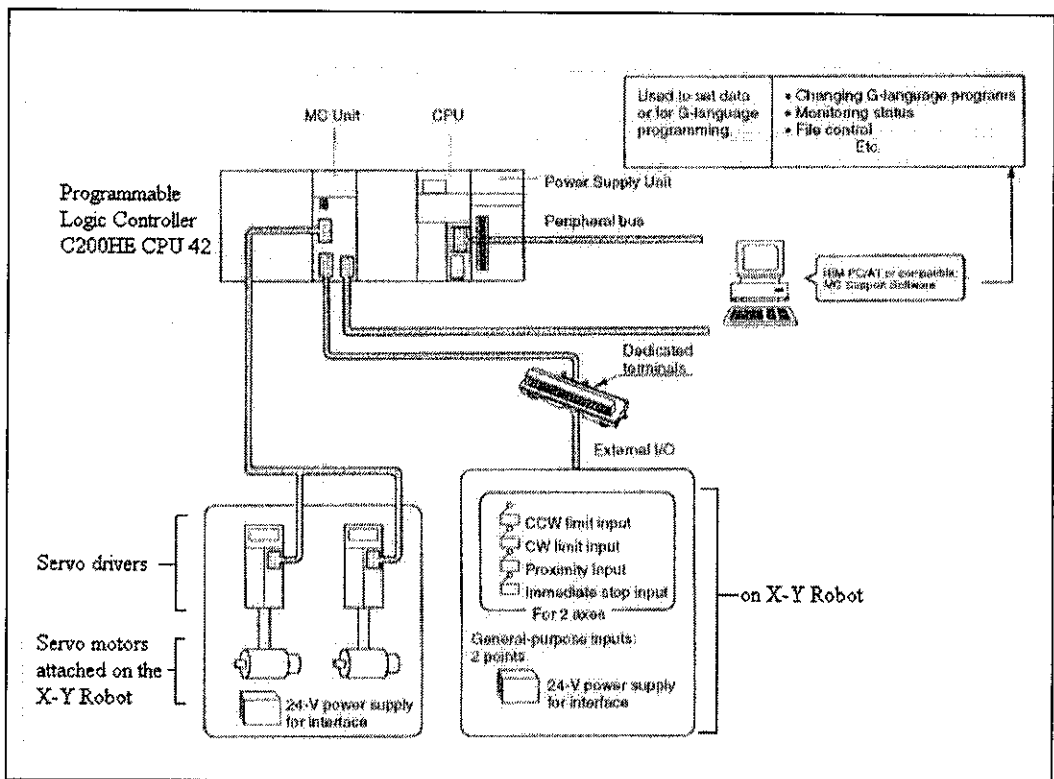


Figure 4.1: Basic Architecture of the X-Y Robot

Figure 4.1 shows the basic structure of the X-Y robot with all related devices and connections. The X-Y Robot has two axes, the X and the Y axis. Each axis has a servomotor connected to the robot. The two servomotors are connected to two servo drivers which are then connected to the Motion Control Unit (MCU). The MCU used for this project is model number C200H-MC221 by OMRON. It is designed to control positioning using linear or circular interpolation.

The MCU is also connected to the personal computer via com port. The connection is physically an interface from the 20 pin side of the MCU to the 9 pin side of the personal computer. Since there is an interface of 25 pins from the 20 pin side of MCU via CV500-CIF01 OMRON, the 25 pins are connected to the 9 pins com port of the PC via the diagram shown in Figure 4.2

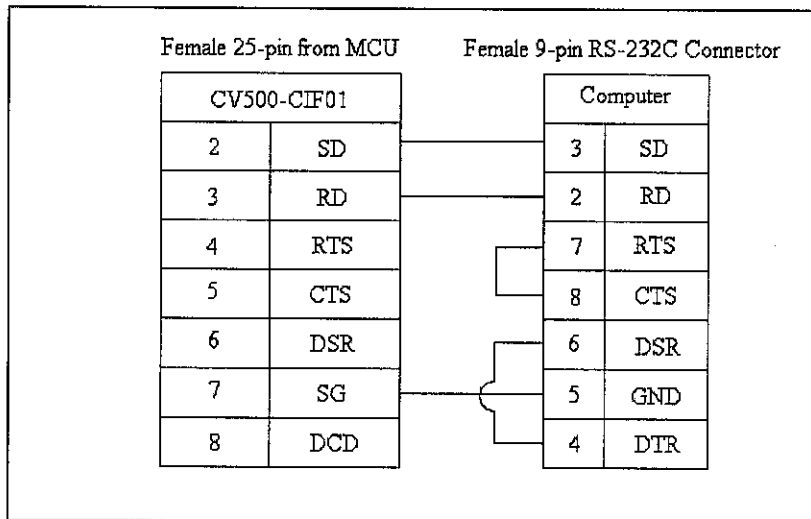


Figure 4.2: Pin to pin connection of CV500-CIF01 (MCU) to RS-232C (PC)

The transfer of parameter settings and MC programs to the MCU via the CV500-CIF01 to RS-232 had been tested to be successful. Two-way communication is possible between this link as the MCU could upload existing information into the CX-Motion software as well. The CX-Motion also provides a good monitoring tool of the status of the robot through the MC monitoring function. Error indicator is also a feature available to pin point the type of error generated plus suggesting corrective actions.

The connection from the servo drivers and servo motors is also an important issue. The connection diagram is shown in Figure 4.3. As the servo drivers are the operator of the servo motors that will ultimately move the robot, it is a necessary to get all connections and setup right. The setup of the servo drivers were successful with no error indicated.

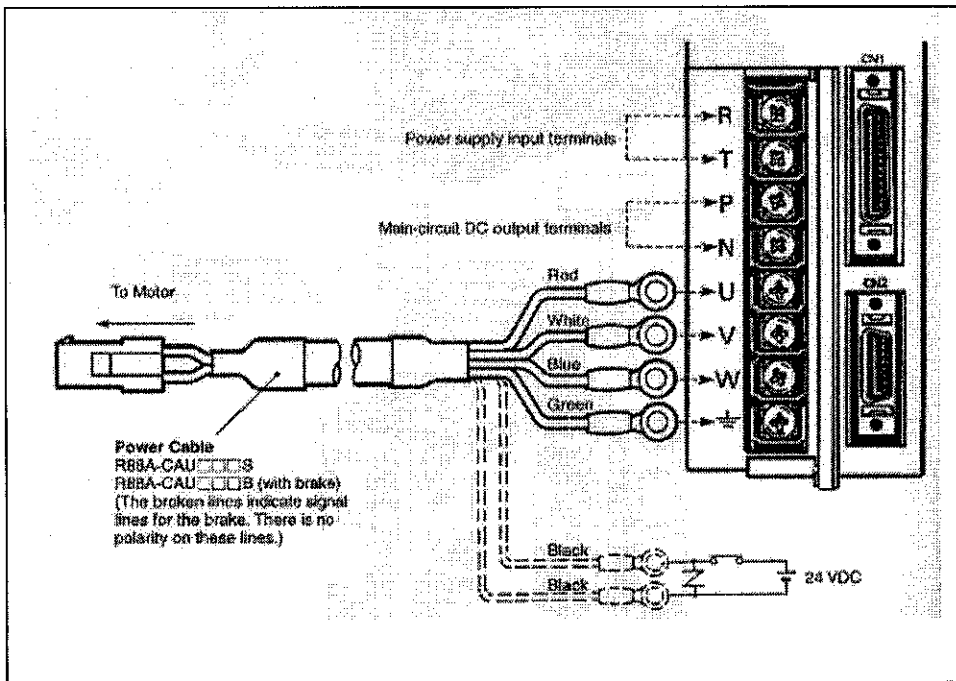


Figure 4.3: Servo driver to servo motor connection diagram

4.1.3 Operation and testing of X-Y Robot in Manual Mode

The operation of the X-Y Robot in the manual mode was tested by constructing a ladder diagram based on the triggering of servo lock, origin search, and jogging in both directions for both axes. The X-Y Robot manual mode operation ladder diagram is shown in Appendix III. The ladder diagram simply consisted of normally open switches to trigger the output with addresses associated with their respective functions. The addresses are pre-assigned and they are understood by the MCU.

Testing of the robot proved to be futile as the robot was not responding appropriately to the servo lock command. The limit sensors and the origin sensors were tested and working fine. The error that was generated when the servo lock was activated was Error 61: Error counter overflow which indicated the number of pulses in the error counter exceeded the upper limit of 65535 pulses. This error could be caused by a disconnected/broken encoder cable, electrical noise, improperly adjusted servo system or improperly adjusted/faulty machine. This error indication clearly showed a problem regarding the hardware part of the system.

Triggering the origin search bit and the jogging bit showed Error 70: Servo lock error indicating an attempt to execute commands without servo locking the axes. This response was expected as servo lock was required for all operations.

Using an operating console on the servo drivers, the X-Y Robot could servo lock and also jog in both directions on both axes. Since the problems lied neither on the servo drivers nor the PLC, the cause of the problem had to be from the MCU or the cables between the MCU and the servo drivers. These two components were sent to the supplier, OMRON for further testing. Feedback from OMRON indicates instability in the MCU operation with occasional success and failures. There was a suspected fault inside the connection of the MCU and the cables. It was advisable not to proceed with the X-Y Robot in the project as its instability would cause problem to the overall system.

Since the servo lock command could not be executed, the development and experimentation of the X-Y Robot could not proceed. This prompted the usage of solely the pneumatic system as the actuator system in the project. This result would then affect the system design and selection of target object that suited the pneumatic system.

4.1.4 Pneumatic System Design and Setup

For the pneumatic system, the components involved are three double acting cylinder (solenoid valves) and three 5/2(14) valves. These components were required to sort target objects into three categories, which was selected to be the purpose of the project. Take note that the division of target objects into three categories was chosen without any constraint, and the target objects could be sorted into more than three categories which would however require more components. Selecting three categories was good enough to indicate the functionality of the system.

The solenoid valves were required to push IN and OUT whenever required by the system. Therefore the electrical connection to the valve had to be connected to control the solenoid valves in two states: IN and OUT. It was important to take note than to push the solenoid valve OUT, air inlet to push out the solenoid valve had to be ON and the air inlet to pull the solenoid valve had to be OFF. If both inlets were ON or OFF, the equal air pressure would not budge the solenoid valves. This was to be taken into consideration when constructing the system ladder diagram to control the solenoid valves.

As the pneumatic system was fully and directly controlled by the PLC, there was no need for any controller configuration, initialization or setup. The only concern would be the connection of all the parts whether electrically or pneumatically. The electrical signal would be 24V while main air pressure would be at 30 psi. The standard pneumatic diagram connection is shown in Appendix VI.

4.1.5 Vision Sensor Parameter Setup

All the components of the vision sensor are obtained and connected properly. The operating console is the basic controller for the system. It has directional keys and ESC and ENT mode for execution. The trigger button is used to start measurement of the image. The image will display OK for acceptable or NG for unacceptable on the image measurement based on some criteria specified.

To start of with, a sample object (a paper with 5 different alphabets) was placed in front of the camera. This object is to be used as experimentation on the effects of changing the parameter on the images. The background suppression function is used to eliminate the background. It has the ability to filter image areas densities which could be set to eliminate the desired amount of background. By setting the limit at 150 and the lower limit at 100, the background is completely suppressed. In fact this setting also suppressed all the lines and dots that are on the paper that is not part of the alphabets. This range of background suppression could be altered to suit the target objects.

The coordinates are set to default using the left hand coordinate system and origin at upper left. The lighting level can be adjusted at 8 different areas in the camera for uneven lighting. In this case, all areas are set to maximum “77777777” value to obtain the brightest and even lighting. This is also configurable if the target object had different brightness densities along its body. It was assuming that the target object are evenly colored that the maximum lighting was used.

To use the terminal blocks as output, there is some setting to be done. However the default setting is to transfer output through the terminal blocks. The following communication settings are done:

1. ON state: This selects whether to turn ON the OR and DO 0 to DO 15 terminals for OK or NG. The default is ON for NG.
2. Handshaking: Set to Off, using to output the measurement result without synchronizing with the external device.
3. Output Period: Set to 10.0ms. This value has to be greater than Gate ON delay + Gate ON time, and less than the measurement interval. If this period is longer than the measurement interval, the output timing will fall behind as the measurements are repeated.
4. Gate ON Delay: Set to 1.0ms. This set the time from when the result is output to the terminal block to when the Gate signal is turned ON. This time is used to wait until the data output becomes stable. Time set is longer than the delay time for external device.

The terminal blocks can be used to input and output various signals, for example to input a measurement trigger from a photoelectric or other sensor and to output the overall judgment to the PLC or any other external devices. The STEP terminal serves as the input as one measurement will be performed when the terminal is triggered. Other form of inputs can be entered through the port DI 0 to DI 7. A combination of binary values will give the desired input. For the interest of this project, it is important to know the input for changing scenes, as there are multiple requirements to be considered to sort the target objects. Table 4.1 shows the combination to change scenes:

Switching	DI 7	DI 6	DI 5	DI 4	DI 3	DI 2	DI 1	DI 0	Function
Scenes					Scene No.				Switches
	1	0	1	0					scene
Example									Switches
	1	0	1	0	0	0	1	0	to scene 2

Table 4.1: The inputs combination to switch scene in the vision system

Whenever the measurement is performed, the judgement is sent to output. The data is only sent to external output in Run mode. There is no output to the terminals in Monitor mode. After the measurement is performed in Run mode, the data output to the OR terminal and DO terminal will be held until the next time measurement is performed in Run mode. The output status will be held even if the mode is changed.

The overall judgement is output to the OR terminal. This overall judgement will be NG if there are one or more NG judgements for the set measurement regions or output expressions. The output can be set to go ON if an NG is detected or an OK is detected. Judgement outputs 0 to 15 set in Expression/Judge will be output from terminals DO 0 to DO 15.

4.1.6 Vision Sensor Measurement Setup

After all the settings had been done, the Position Displacement Compensation was carried out. The position displacement compensation function is used when the position and orientation of measurement objects are not consistent, as always will be the case. By using, this function, the displacement between the reference position and the current position is obtained, and this displacement is compensated for in measurements.

Firstly, the search region was set. Using the Grey Search method, a model was registered and a reference position of the model was assigned. Then, a range of area was set in which a search was to be made for the model. Then the evaluation criteria were set by assigning the threshold for the correlation value. If the correlation value with the model is low, the wrong place may be found. Therefore it was important to set a reasonably high correlation value but not too high until the area cannot be detected. A good range would be above 70.

After the settings were done, the trigger button was pushed. The image show that object was at c oordinate $X=0$ a nd $Y=0$ displacement. By shifting the object for a certain distance but still within the search range, the system was triggered again. The system managed to capture the registered model and also indicated the X and Y displacement value. This function is important to scan multiple objects which position on the camera would be different due to pick and place randomness. It is important to assign a common reference point that exists on all objects to be set as the model.

There are actually nine measurement methods for this model of the vision system as stated in the manual, only two functions will serve the interest of this project: Grey Search and Density Averaging. These two methods though are sufficient to detect object's existence, and its shape. The vision system could not differentiate color as it is monochrome. However, it can measure the density and intensity of the image.

The Grey Search is used to inspect the shape of an object. To allow this function to work, a n i mage pattern for a satisfactory p roduct i s registered i n advance. This is referred to as the model. Then the vision system will check all other objects for conformity with the registered model. The extent of conformity i s expressed a s a correlation value, between 0 to 100, where 100 is perfect similarity. A threshold can be set o n this c orrelation value and t he system will i ndicate OK f or image a bove threshold and NG for image below threshold. This function enables the system to detect shapes or even patterns or characters. Initial testing with this function on alphabetical characters is successful, as it can detect the presence of a certain alphabet in object after registering the image of the alphabet.

The density averaging is used to perform measurement based on differences in the average brightness. The density averaging function obtains the density of each pixel (between 0 and 255), obtains the average, and performs measurement on this value. Density averaging is not a good detection of object shape; rather it is used to calculate object area or simply to verify the existence of an object. Initial testing is successful where the density averaging function responds well to object existence.

The vision system can basically operate in two distinct modes which are the monitor mode and the run mode. The monitor mode is used to check the measurements are being correctly performed with the set measurement conditions. The measurement results are not output via the terminals blocks and so this procedure only tests operation for the vision sensor itself. Testing in the Monitor mode is simple and successful.

The Run mode is used to perform actual measurements. When the measurement is triggered, the measurement will be executed and the measurement results will be displayed on the screen and output to external devices. Apart from outputs, it is able to receive inputs via the STEP or the DI 0 to DI 15. Testing the vision system on the Run mode is successful as outputs signals were detected when the measurements were carried out, and the system is able to receive signal from external signal as input. The signal testing was done by connecting the input and output ports to the PLC and setting the appropriate ladder diagram on the PLC.

4.1.7 Vision System Testing

Experimentations and simulations are carried out on three different objects: circle, triangle, and square. Underlined below are 7 experiments carried out:

1. Approximation to a circle using Grey Search
2. Differentiate circle from square and triangle using Grey Search
3. Differentiate square from triangle and circle using Grey Search
4. Differentiate triangle from square and circle using Grey Search
5. Average Density Testing for Circle
6. Average Density Testing for Square
7. Average Density Testing for Triangle

For the first experiment, the objective is to test the vision system to detect a defected circle from a perfect circle. How much the vision system can accept the image based on the amount of defects was set by adjusting the range of the correlation value. In this case the correlation value was set from 80 to 100. The following figures show the result of the testing:

A circle is registered:

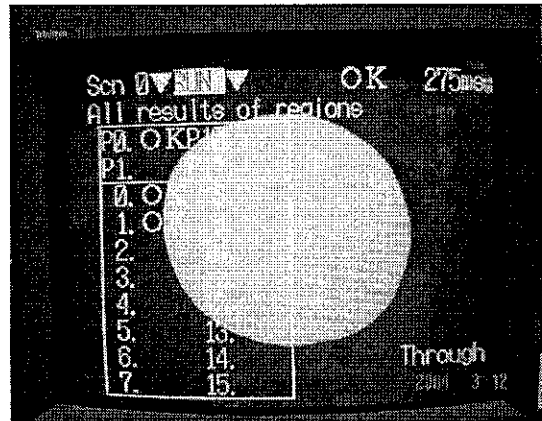


Figure 4.4: Registered circle for experiment 1

By triggering the measurement using the same picture, the correlation value for the position compensation was 100:

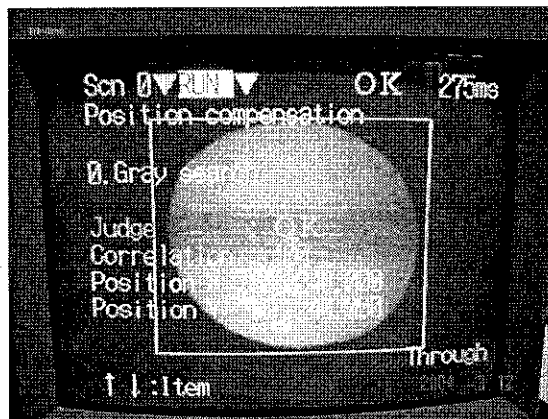


Figure 4.5: Position Compensation Grey Search value using perfect circle

The correlation value for measurement in Grey Search was good at 100 correlation value:

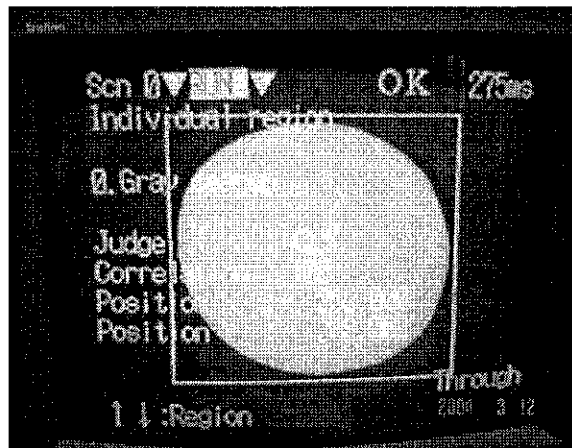


Figure 4.6: Grey Search Measurement of perfect circle

The experiment was carried on by shifting the circle to the right of the screen. This was to test the function of Position Displacement Compensation. From the following pictures, the measurement criteria were OK for all measurements that were Position Displacement Compensation and Gray Search. In the Position Displacement Compensation, the correlation value detected was 96. The amount of displacement from the original image was also displayed.

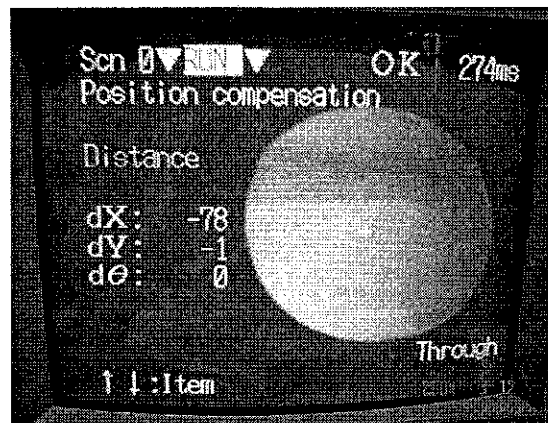


Figure 4.7: The amount of displacement due to Position Compensation by shifting object to the right

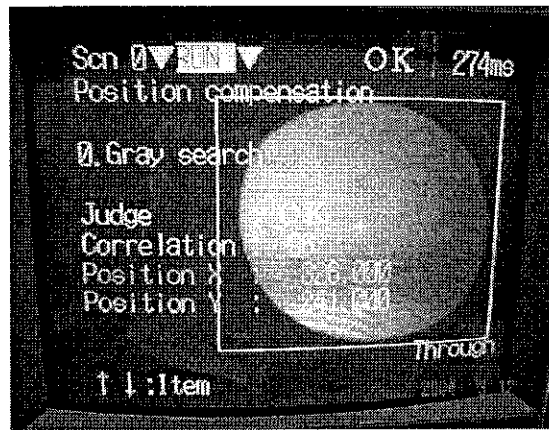


Figure 4.8: Grey Search Correlation OK for Position Compensation by shifting object to the right

The next step of the experiment was to apply a minor defect to the circle. Using the model registered and the same criteria, the measurement was carried out. It was found out that the object was deemed acceptable at correlation value of 86. The resulting images are shown below:

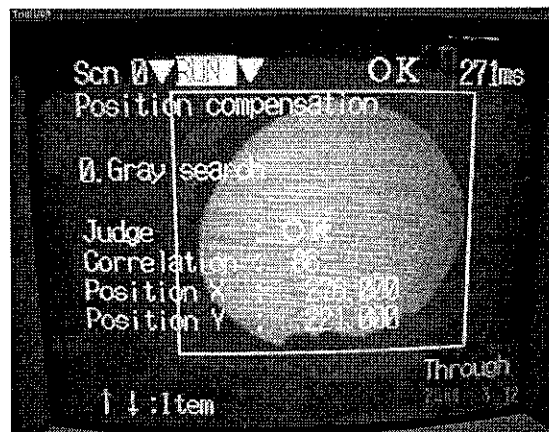


Figure 4.9: Grey Search measurement of circle with minor defect

More defects were applied to the circle and at the level of medium defect, the image was acceptable, however it was at the border line with correlation value of 81.

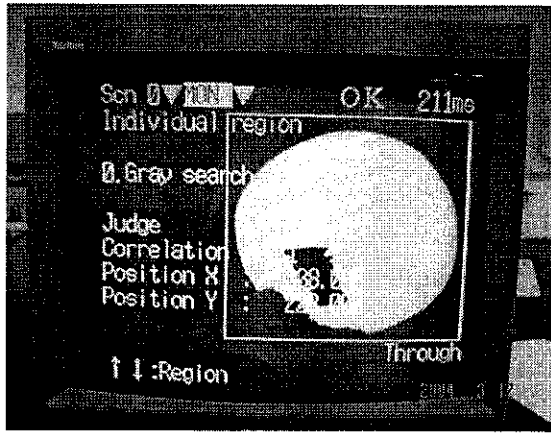


Figure 4.10: Grey Search measurement of circle with medium defect

With major defect applied to the circle, the image was no longer acceptable with correlation value 71 and the NG signal appeared on the top right of the screen. However the Position Compensation was still good since the correlation range was from 60 to 100. The Density Averaging was acceptable as the limit was not stringent. Note that Density Averaging was just used to detect the presence of object. The results are shown in the following diagram:

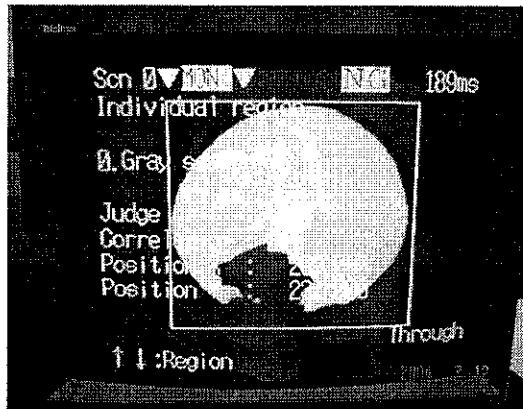


Figure 4.11: Grey Search measurement of circle with major defect

The following three experiments also used the Grey Search measurement method. Each experiment was registered a perfect model of either circle, square or triangle and then all three objects were tested to the model. The results of the three experiments registering perfect model of circle, square, and triangle respectively are shown in the table below:

Grey Search Measurement Registered Model Characteristic	Correlation Value of Target Object		
	Circle	Square	Triangle
Perfect Model Registered as Circle	96	66	62
Perfect Model Registered as Square	63	92	45
Perfect Model Registered as Triangle	56	57	97

Table 4.2: Grey Search Measurement Results for Circle, Square and Triangle based on Different Registered Perfect Model

From the results shown in the table, the Grey Search is a suitable method to differentiate the shape of an object with others. Object that matches the perfect model shows correlation value of above 90. A good range for the condition of the correlation value would be 80 to 100. As observed from the table, when the target object does not match the perfect model, the correlation value was well below 80. The visual results of the experiments are shown in Appendix VIII.

However, the Grey Search measurement method has its drawback. It only compares the target object with the registered model. If the right shaped object is titled in different orientation, the Grey Search will fail that object. This problem does not occur with the circle as its shape remains the same no matter how it is tilted. This problem is very much affecting the measurement of square and triangle.

To overcome this problem, the Average Density measurement method is used. The Density Averaging is obtained for each pixel and measurements are made based on the average density. Therefore it is the calculation based on the area of the object. No matter how the square or triangle is tilted, the area remains the same. This is the concept that will be used to pass both the objects.

The Average Density of the circle is obtained at the value of 31.428. The visual result is shown in Appendix VIII. For the Average Density of the square and triangle, three readings were obtained based on three orientation of the object. The values are shown in the table below:

Readings based on random and different orientation of target object	Average Density Value of Target Object	
	Square	Triangle
Reading 1	41.750	28.164
Reading 2	42.614	28.437
Reading 3	42.487	27.970

Table 4.3: Average Density Readings for Square and Triangle in multiple orientation

The results shown in the table indicates a close value among three readings, even though the objects are titled in different orientation. The Average Density condition could be set to 35-to-45 range to pass the square and 25-to-35 range to pass the triangle. This therefore verifies the relevance of using the Average Density measurement method in the system. The visual results are shown in Appendix VIII.

From the result of the first four experiments, how much the defect the image can tolerate can be set upon by adjusting the correlation value range. This is arbitrary and set upon requirement. The Average Density condition could also be set to pass a certain range. Therefore, this system is excellent in terms of its programmability and flexibility. From the result all the experiments, the success in differentiating all the shapes indicate that the correlation range of 80 to 100 and proper Average Density range setting are sufficient to sort objects shaped circle, square and triangle. The success on these experiments is important to pave the way for system requirement setting and implementation. It is the feature of the vision system that provides the intelligent function of the sorting system to differentiate target objects based on shapes.

4.1.8 System Visual Basic Interface Design

Visual Basic interfaces were done on two operations: the X-Y Robot manual mode operation and the sorting system operation. The interface design is simply done with the drag and drop function of the Visual Basic 6. Refer to Appendix IV for the interface of X-Y Robot manual mode operation and for the sorting system operation.

The X-Y Robot manual mode interface design was done based on the simple functions of the manual mode operation of the X-Y Robot. The change mode button is to set the robot to be working in either the manual or automatic mode. In this case, the robot will be set to manual mode. The set and reset buttons were allocated for the servo lock, servo free, and origin search function, while the Jog function has positive and negative direction movement buttons. An additional exit button was included to exit for the application.

The sorting system interface was done on two operations: the manual and automatic mode. The manual mode was designed to operate the vision sensor change of scenes, the measurement trigger button, the IN and OUT of the three solenoid valves and the conveyor belt independently. This was not the production operation of the system and this mode normally would be used for troubleshooting single components. The automatic mode was programmed with production operation in mind. Therefore it consisted of only START and STOP button. The functionality of the system automatic mode was programmed by the system ladder diagram, and not the Visual Basic coding. The automatic mode and the manual mode could not operate at the same time.

The underlying codes for the interface of the X-Y Robot are shown in Appendix V. These codes corresponded to the addresses that the ladder diagram of the X-Y Robot manual mode operation as shown in Appendix III. The codes for the sorting system are also shown in Appendix V, corresponding to the ladder diagram of the sorting system shown in Appendix III. The Visual Basic codes are typed in such a way that they matched the address assigned in the ladder diagram. This coding method is explained in the following section.

4.1.9 Serial Connection Setup between Interface and PLC

The coding method of the Visual Basic program so that it could communicate with the PLC required a standard command line with addressing using numbers based on the Compolet Series table. For simplicity purposes, a sample ladder diagram was constructed to light up lamp 0 at address 1.00 with set switch at address 200.03 and reset at address 246.15 was constructed. A simple Visual Basic interface was designed with just a SET button and a RESET button. The Visual Basic codes used were shown below:

The RESET button codes:

```
SYSMAC_C1.writeArea plcAreaCIO, 246, 1,"8000"  
SYSMAC_C1.writeArea plcAreaCIO, 200, 1,"0000"
```

The SET button codes:

```
SYSMAC_C1.writeArea plcAreaCIO, 200, 1,"0008"  
SYSMAC_C1.writeArea plcAreaCIO, 246, 1,"0000"
```

The reason the last 4 digit or the string are used was with reference to the Compolet Series Table as shown below:

0				0				0				0			
15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
2 ³	2 ²	2 ¹	2 ⁰	2 ³	2 ²	2 ¹	2 ⁰	2 ³	2 ²	2 ¹	2 ⁰	2 ³	2 ²	2 ¹	2 ⁰

Table 4.4: Compolet Series Table

The “8000” used in the RESET button code corresponds to 15 in the table, which indicates the address 246.15. The “0008” used in the SET button code corresponds to 03 in the table, which indicates the address 200.03. Using this basic knowledge of assigning Visual Basic codes to PLC ladder diagram addresses, the coding could be done on the interface of the X-Y Robot manual mode operation and the sorting system operation.

After transferring the ladder diagram to the PLC, activating the serial unit and the CPU unit in the service manager of the FinsGateway and setting the OMRON SYSMAC C Control object in the Visual Basic to corresponding nodes and network address, the interface was tested. The testing was unsuccessful for the X- Y Robot as the servo lock could not be triggered as reported earlier. For the sorting system operation, the testing was successful as the system could operate in both automatic and manual mode. The testing of the Visual Basic interface communication to the PLC was successful according to the monitoring function of the CX-Programmer and also the responsiveness of the components after system design and physical implementation.

4.1.10 System Design and Implementation

As X-Y Robot is not working properly, the actuator system will be solely on the pneumatic system alone. The identification and arrangement of all components have been done. The sorting system design layout is shown in Appendix X. The design is a rough placement of all the big components, while the specific wiring connections are being left out. The diagram shows a placement of the sensor, camera, and the solenoid valves along the conveyor belt with each serving its respective function following the flow. With the diagram, the system can therefore proceed to physical realization. Factors to be considered during physical realization include distance between components and the speed of the conveyor belt. This will affect the values of timers to be input into the PLC ladder diagram program. It is therefore only relevant to construct and experiment the system's ladder diagram after physical implementation.

The PLC is connected to the personal computer through the RS-232 cable to the COM port of the computer. This cable determines the serial connection to the Visual Basic and also to the CX-Programmer that builds the ladder diagram. For the wiring connection, the wiring diagram of the PLC to the vision controller is shown in Appendix VII. This diagram shows a connection of the inputs of the PLC to the output of the vision controller and vice versa. This set up a two way communication path between the two devices. This wiring system had been tested and responded positively. The pneumatic system construction and air supply connection is as shown in Appendix VI. The electrical signals are from the PLC outputs.

4.1.11 System Process Identification and Ladder Diagram Construction

To construct the system process flow, it is important to underline the objective of the overall system. The main purpose of constructing this sorting system is to sort objects based on shapes. And the three shapes to be sorted are determined as circle, square and triangle. Each component in the sorting system serves its particular function to achieve the objective of the system. The components are mentioned in the tools identification section and the placement of the components in the system is shown in the system design layout as shown in Appendix X.

The target object will be uploaded to the conveyor belt at the starting point. The photoelectric switch sensor will detect the presence of the object. Since the transmitter will always send a signal to the receiver, a loss of this signal indicates object is present and the sensor will send an input to the PLC. Upon receiving the input signal from the sensor, the conveyor belt stops after a certain delay. At this point the target object is directly under the vision system camera. The camera will go to Scene 1 which was to check for the circle using Grey Search. If it passes, then conveyor belt will move again. If the vision testing fails the vision controller will send a signal through the RO terminal block to the PLC indicating a failed measurement. The vision system will then trigger Scene 2 to test for square using Average Density and the same procedures apply if the testing failed and it will go to Scene 3 to test for triangle using Average Density.

After testing the target object is moved by the conveyor to the actuator stage, where the double acting cylinder solenoid valves will push the object to their respective bins. Solenoid valve A will push the circle, Solenoid B will push the square and Solenoid C will push the triangle out of the conveyor belt. The timing of the extension of the solenoids valves is an important aspect and is programmed in the ladder diagram of the sorting system operation. If the object failed all three Scenes, no actuator will triggered and thus the object would be carried by the conveyer all the way until the end to drop in the unclassified bin.

The sorting system process flow chart is shown in Appendix IX. The ladder diagram is programmed based on this flow chart. The sorting system ladder diagram is as shown in Appendix III. The ladder diagram is constructed based on two operations: the manual mode operation and the automatic mode operation. The automatic mode operation will follow the process flowchart while the manual mode operation allows individual components to be triggered such as movement of conveyor belt, vision controller scene change and measurement trigger, and the extension and retraction of solenoid valves. The manual mode is more for testing and troubleshooting purposes while the automatic mode is tailored for production interest.

4.1.12 System Testing, Running and Monitoring

System testing and running was done on both manual mode and automatic mode. Both modes are operating properly and according to objective. Testings on all individual components are carried and no defects were found. The whole system integrates well and works well as a single process unit. Although many problems were encountered in individual components testing and system integration and testing, none warrants a need to seriously halt the project. The problems were either solved or being replaced by alternative solution. The only drawback is the component failure of the X-Y Robot which was unforeseen and unavoidable.

The system could be monitored through the external monitor connected to the vision controller, through the personal computer or just by observing the components' response. The monitoring tools are important for troubleshooting to monitor system flow and pin point error. Through the testing and troubleshooting procedures, the project is thus completed and achieves all the project objectives underlined.

4.2 DISCUSSION

The hardware and software for the project are available and assembled. The hardware components that are assembled are the X-Y Robot, servo drivers, Motion Control Unit, Pneumatic System double acting cylinders and valves, Programmable Logic Controller (PLC), the Vision Sensor, photoelectric switch sensor, conveyor belt and personal computer. The Vision Sensor consists of a controller, camera, operating console, and external monitor. As for the software, the CX-Motion, C-X Programmer, Visual Basic, FinsGateway and SYSMAC Compolet are all installed on the personal computer and tested to be functioning perfectly. All the tools that are mentioned in the tools identification are available, tested and integrated into a full sorting system that serves the objective of sorting target object based on object's shape.

The PLC assembly is done successfully, as ladder program can be transferred successfully from the PLC to the PC. The PLC gives a good monitoring function of the current flow in the ladder diagram. From the ladder diagram shown in Appendix III for the X-Y Robot manual mode operation and sorting system operation, components in the diagram involve normally open switches, normally close switches and timers. All these components are connected in a fashion to serve its objective of controlling the system. Using the PLC in the system is in essence applying a centralized control system to the sorting system. The programmability, flexibility, robustness, high reliability, low space requirement and low power consumption of the PLC make it an ideal choice as main controller. Perhaps the most important aspect of the PLC is that it allows production and programming to be done in parallel. The minor drawback of using the PLC is that the programs are run in cyclical form, and thus it is slower than parallel logic system offered by other electronic means of control. Applying the PLC as the centralized controller might reduce the overall reliability of the system. The vision controller can serve as the sub controller of the system to control the vision system in the case of PLC. However, the system will lose its integration and automatic operation mode if the PLC fails.

The X-Y Robot assembly initially gave a lot of problems, as the MCU kept generating an error. The CX-Motion program hanged the PC when an attempt to transfer the parameters from the PC was done. The problem with the irresponsiveness of the MCSS software was due to the connection of the 25pin CV500-CIF01 from the MCU side to the 9 pin com port PC side. The pin to pin assignment need to be tested again and again until the MCSS manage to transfer the parameter file to the MCU. The error on the MCU was due to some connection problem. Initially the error was on because the servo drivers were not connected to the 24V DC supply. After solving the problem, the error resurfaces again. This time, it was due to the servo motor of the robot at the position of exceeding the limit switch of the axis. This was known from the MC monitoring function which indicated error number 62 meaning clockwise over travel error. This is fixed by adjusting the robot position manually and reset the MCU.

Operation of the X-Y Robot in manual mode was not successful. The robot responded to a servo lock function by moving slightly and showed Error 61: Error counter overflow which indicated the number of pulses in the error counter exceeded the upper limit of 65535 pulses. According to the manual, this error could be caused by a disconnected/broken encoder cable, electrical noise, improperly adjusted servo system or improperly adjusted/faulty machine. The MCU generated an error when the origin search is triggered due to the machine not yet been servo locked. Even though the parameter of the X-Y robot was changed and tested using the CX-Motion several times, the same negative response was encountered. The X-Y Robot components had been sent to the manufacturer for testing, and the results of the testing indicated instability in the robot's operation due to components failure. Due to the unforeseen and unavoidable problem regarding the X-Y Robot, it would not be integrated into the sorting system.

The setup of the second actuator system which was the pneumatic system posed little problem. As no software was involved in the actuator since it was directly controlled by the PLC, there was no need for parameter setup. The only important aspect is the air pressure supplied to the system which should be in the range of 30 psi. Further research could be carried on the effect of force and friction of the actuators applied to the target object. However, this is not within the scope of this project.

The setup of the vision sensor posed little problem as it is a user friendly system with a comprehensive and easy-to-navigate set of menus. The background suppression function is a feature that must be done with care. When the image of the background is suppressed, there is a tendency that the image of the target object will be suppressed at the edges. While testing the system with a paper containing alphabetical characters, it was noticed that the alphabets became thinner when the lower limit of BGS was set to a high value. This problem requires attention and needed to be solved by constant testing through adjusting the range of suppression. The coordinate system gives a good account on object displacement; however it is limited by the vision of the camera. In a sense, this could be a potential problem when dealing with large objects as the vision system coverage area is rather small. An option would be to increase the height of the vision system. However, this will reduce the sensitivity and the accuracy of the vision system. A good tradeoff needs to be achieved between image accuracy and coverage, and this depends on the nature of the target objects and the system's objective.

While fixing the wiring system of the vision sensor to the PLC, initial problems were encountered as no signal was detected from the output in Run mode. And the system did not seem to respond to external input signal to the STEP terminal. After extensive testing with many different connection combinations, the problem had been solved. The COMOUT had to be connected to the negative power supply and the COMIN had to be connected to the positive power supply. The output was then detected. For external input signal, the signal has to go to the respective input terminal blocks while the COM of the input was connected to the negative of the power supply. With these settings the vision controller can then communicate with the PLC. This is an important aspect as it provides the interaction needed to enable the PLC as the main controller. The extensive wiring connection between PLC and the vision controller is shown in Appendix VII.

The Position Displacement Compensation and the Measurement through Grey Search and Average Density functioned properly through simple experimentation. The fact that the perfect model could be registered for measurement indicates that the system is capable of handling all types of objects. This contributes to the programmability function of the sorting system that is highly desirable by the manufacturing industry nowadays. The Position Displacement Compensation especially serves an important function as it compensates for object displacement from the assigned origin. This is vital when the object position on the camera is inconsistent, as always will be the case. However this is limited to the area of coverage of the vision system. The Measurement can then be done on the object that has been compensated. The measurement level can be considered specific enough as it measures defects in pixels. The Grey Search functions only compared the image with the registered acceptable model. This posed a problem as previously mentioned when dealing with square and triangle target objects. This problem was addressed by using the Average Density function. Using both the characteristics of Grey Search and Average Density, the system could cater for all target objects. Multiple measuring methods are sometimes needed to achieve objective as shown in this project. When using the Average Density, the Position Compensation would have to be disabled as it would interfere the measuring procedure. This is due to the fact that Position Compensation uses the Grey Search technique to compensate locations, and it might shift the search region if it detects a pass target objects.

There are limitations to the capability of the vision system. Being a monochrome camera, it could not differentiate color with high accuracy. It can however differentiate color through means of brightness densities. This is applicable if a dark color needed to be separated from a brighter one. The absence of zooming function requires the height of the camera to be adjusted manually by adjusting the tripod to cater objects of different size.

For the experimentations on the shapes of circle, square and triangle, it clearly showed that a correlation value of 80 to 100 was sufficient to differentiate the objects. The correlation range could be adjusted to increase and reduce the stringency of acceptance, which is another programmability function of the system. Not only the system can differentiate shapes, it can also detect the amount of defects on the target object as shown in the first experiment. The extent that a defect is acceptable is also determined by the correlation value. While a correlation value of 80 to 100 is applicable to measurement, a correlation value of 60 to 100 is normally sufficient for the Position Displacement Compensation. This is due to the interest of the function to display the deviation from origin rather than to pass or fail the object. A failed object could therefore still be compensated in the position. This is particularly important if a research is to be carried out on the consistency of positioning in front of the camera for different shape target objects.

From the experiment, it was clear that only one model was assigned as the perfect model. To sort the three objects, there is a need to assign three conditions and give different signal for each model. This is where the multiple scenes function comes into play. The vision controller can store up to 16 scenes from SC0 to SC15. By assigning the proper condition to each scene, the system can test for conformity in every scene. Therefore the shape of the objects could be sorted according to the output signal of each scene measurement. To change scene, a combination of input is required as shown specifically in Table 4.1. With the application of multiple scenes and assigning model for Scene 1 as circle, Scene 2 as triangle and Scene 3 as square, the simple flow of the system is as shown in Figure 4.12.

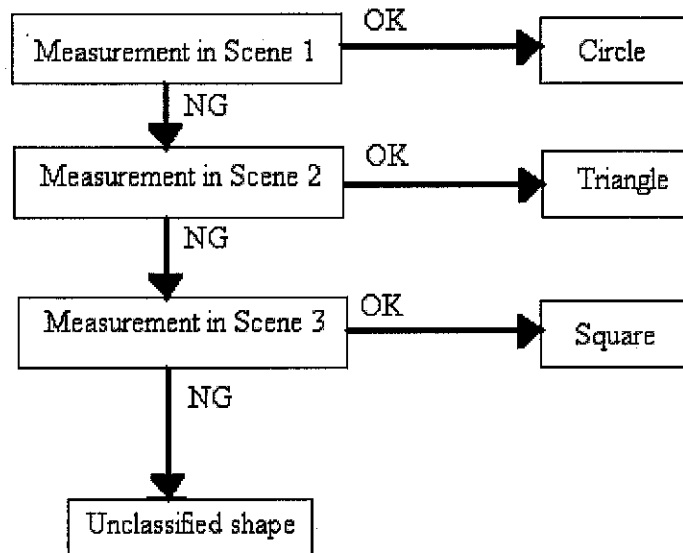


Figure 4.12: Flow Chart of multiple scenes measurement for shape classification

The Visual Basic Interface design was successfully done and configured. This was due to a user-friendly environment that the layout of the Visual basic program offered. The buttons in the interface corresponds to the functions of the manual mode operation of the X-Y Robot and the sorting system. The underlying codes in the interface were relatively easy as the keywords were similar to C programming. So virtually no problem was encountered during the construction of the interface.

While testing the communication between the Visual Basic interface and the PLC, a few problems were encountered in the initial stages. There was some confusion regarding the nodes and the unit addresses of the SYSMAC C OMRON Control Object in the Visual Basic Interface. Further research indicated the value of nodes and unit address are standard as underlined by OMRON and thus solved the confusion. After setting up the FinsGateway, the CX-Programmer could not link to the PLC. After a few testing, it was discovered that by disabling the Serial Unit in the FinsGateway only could the CX-Programmer interact with the PLC. It was therefore important to transfer the ladder diagram to the PLC first before enabling the Serial Unit. The Serial Unit was to set up a communication with the serial connection to the PLC, so was the connection from the CX-Programmer. Therefore it was not possible to enable two serial connections using one com port.

In the initial stages there was also error encountered in the Visual Basic Interface due to communication clash with the PLC. It was then discovered that to allow the interface to work, the PLC had to be in monitor mode and the CX-Programmer had to be in work offline mode. This is to avoid two programs trying control the PLC at once. After changing the mode of the PLC and the CX-Programmer, the communication between the Visual Basic program and the PLC was tested and successful.

The systems design constitutes the pneumatic system as the sorting system actuator. It is regretful that the X-Y Robot could not be integrated to the system due to unavoidable technical problems. Instead of pick and place function, the system now moves the target objects based on solenoid valves movements. One advantage is that there is no need to do research and testing on the gripper of the X-Y Robot. However, as the size and capability of the gripper is limited, only object with certain shape and size limit could be picked by the gripper. This is a limitation to the system and it reduces the flexibility and versatility of the overall system. Moreover, to do testing on the gripper, a lot of factors need to be considered such as friction and force that the target object experiences during gripping. A very specific coordinate position need to be determined for the pick and place mechanism to be successful. This will pose a problem when objects were placed randomly along the sort area. A different orientation of the object will also pose a significant problem. To counter this, a complex signal processing method is needed to ensure the optimum response of the gripper to pick the object. This will require a great deal amount of programming, experimentation and testing. On the plus side, using the X-Y Robot ensures a fully automated system as it can pick and place to a specific coordinate based on the programs in it. This feature is desirable since it reduces personnel attention to the sorting system and increases productivity. Margin of error is also small provided the system works properly with a limited predefined target objects.

Using the pneumatic system requires less research on the mechanical side of the system, as it only pushes the objects to their respective bins. It is less stringent when it comes to the object shape and size limitation. By increasing the area of the contact between the object and valves, the system can cater for bigger objects regardless of shape. Although the setup and testing of the pneumatic system is much simpler than the X- Y Robot, it is in essence a more flexible actuator system and requires less time in starting up. Its simple construction and mechanism also makes it easier to troubleshoot any arising problem when the system is functioning. The fact that this system is based on physical connection only eliminates the need to do software parameter setup and programming. This system can achieve the controllability that the X-Y Robot offers by programming the PLC; however it does not possess the capability of precise coordinate placement.

Based on the system design in Appendix X, all the components are well arranged in a flow along the conveyor belt to serve its respective function. The photoelectric switch sensor in the first stage serves a simple function of detecting the presence of the object to give a cue to trigger the vision sensor. The PLC will be the main controller and will be placed in between the vision sensor and the pneumatic system. The sorting bins will have to be placed near the conveyor belt and almost on the same level of the belt to reduce impact on target object when pushed into their respective bins. Sponges will be applied around the inside of the bins to further reduce the impact. The distances between the components are trivial as it can be easily programmed by setting the timer value in the PLC. However it is always desirable to reduce the distance creating a less space consuming system. Care must be taken to ensure the components are arranged not too cramped up until it affects the wiring system. A good amount of space between components also ensures an easier troubleshooting procedure in cases of arising problems. The height of the vision sensor is an important factor to be considered. The height very much depends on the objects to be measured as it does not have auto zoom and auto focus capability. The height of the camera can be adjusted manually by adjusting the tripod.

The process flow of the sorting system was obtained after system implementation with reference to the system objective. The process flows is the foundation of constructing the system ladder diagram. However, there are flaws and weaknesses in the process that are not easily detected in the process flow. One important aspect of the system is that the distance between an object in measurement and the next incoming object cannot be less than a certain value. This value is the distance from the vision station to the actuator. If the next object is too close to the object in measurement, the sensor will measure the incoming object and overwrite the previous measured object. Therefore, the first object would not be acted upon by the actuators. To solve this problem, whenever the photoelectric switch sensor detects an object it will trigger the system ladder diagram to indicate the production cycle is ON. The production is OFF only either if one of the solenoid valves acted upon the object or after a certain delay if the object is unclassified. Whenever the photoelectric switch sensor detects an object during production cycle ON, the system will stop indicating a System Fault as shown and indicated in the system ladder diagram in Appendix III. All the objects have to be cleared from the conveyor belt before the system starts again.

In a typical manufacturing sorting line, the target objects can come in many orientation and arrangement. This system will encounter problems if the vision system scans:

1. One object stacked on top of another
2. Two acceptable objects fall into the measurement area.

For the first case, the system might pass the object stacked on each other as it only evaluates the top view of the system. To solve this problem, an additional camera can be implemented at the side to get the side view of the objects. Multiple vision cameras are normally applied in manufacturing line to get many different views. This problem could also be solved by designing an in loader that only loads one object at a time after a certain time period.

The second case might categorize the image as unclassified even though both objects are acceptable. The best way to solve this problem is to apply the Average Density measurement method. Multiple objects in front of the measurement area will yield an unusually high value of average density. If a threshold value is exceeded, the vision system can send a signal to the PLC to indicate a possibility of multiple objects. And thus the system is stopped and required personnel attention to verify the multiple objects presence.

Testing and running the system constitutes much time and effort and is a vital part of the project. In this project, testing and troubleshooting are done in distributed form. Each component was tested before integrating into the system. Therefore any defects or error could be detected in the early stages and not affect the whole system. This is what happened to the X-Y Robot and thus it was not included in the system. System testing was done after all the components were integrated with respect to underlined process and stages. Continuous testing and proper planning ensure faults are detected and solved early. These systematic troubleshooting procedures enabled the project to be completed successfully within the time frame. With that, the project was able to achieve all the mentioned objectives and serve the problem statement.

CHAPTER 4

CONCLUSION AND RECOMMENDATION

The Intelligent Sorting System project is a system integration project that consists of relevant hardware and software component to serve the purpose of sorting target objects in a typical manufacturing line. The project is found feasible over two semesters covering procedures such as project planning, problem statement and objectives identification, literature review, tools identification, experimentation and simulation, system integration, testing and troubleshooting. The major components include the Programmable Logic Controller, the Vision Sensor and the actuator system. All major components are tested to be working and integrated into the system. The only exception is the problem surrounding the X-Y Robot, and thus the actuator system involves only the pneumatic system.

The literature review is a compilation of all researches done throughout the two semesters. It gives additional information on the studies and researches that other individual or institution has done on the sorting system. The technology and approach used by relevant authors indicate the current trend, approaches, and researches done on the industrial sorting system. With the current demand, important feature includes flexibility and relevancy over a period of time are important factors in determining a successful and self-sustaining sorting system. The project is thus deemed relevant to the industry with analysis from the literature review.

With reference to the project objectives, the project achieves all the underlined main objectives. The actuator system of pneumatic components was designed, setup and functioning properly. The Vision Sensor served the intelligent function of the system by setting the conditions of sorting the target object. The PLC functioned as the main controller to operate the whole system in manual mode. And the PLC could be controller from the personal computer using an interface created using Visual Basic. The components were integrated into a single system and operation of system in the manual and automatic mode was successful.

The project itself is only a small entity in a network of manufacturing processes. Recommendations for the future work on the project can be divided into internal and external development. Internal development is concerned on the functionality of the system itself, while the external development covers expansion of the project to other system.

Internal development recommendations could be focused on creating a system that is more flexible, versatile and offers faster production cycle. The cycle time can be reduced by a faster vision scan time. If the scanning time period is fast enough, the target object would not even have to stop under the camera. This of course needs a better capability vision controller and camera. An optimum measurement method has to be devised coupled with complex image processing to separate target objects in different orientation. Fuzzy logic technique can be implemented in the system. Image processing and fuzzy logic implementation is a large scope and constitutes a project by itself. A better actuator system can be used such a multi purpose gripper or vacuum suction tube. These actuators offer better accuracy on object shifting and placement, but however require a more complex control method.

Recommendations on external development revolve around implementing external connections to the system. This allows the system to blend into the production line while using the latest technology other system has to offer. A good recommendation is to integrate the complete system to a LAN network, which has a gateway to the outside Internet network. This is an interesting approach as it allows user to control the robot from a remote place. Another recommendation would be to integrate the system into a FieldBus, as it is the technology of the future. FieldBus is like a local area network which provides efficient control from the user to the system with less wiring and wider applicability. However, this qualifies to be a whole project by itself, and it is out of the author's project scope. It is an interesting aspect to note, and it gives a good direction of future expansion of the project.

REFERENCES

- [1] Control Engineering Europe, 1 October 1999,
<<http://www.manufacturing.net/ctl/index.asp?layout=article&articleid=CA188925>>
- [2] S. Brian Morriss, 2000, *Programmable Logic Controllers*, New Jersey, Prentice Hall
- [3] H.T. Nguyen, Zhenwei Tao, 2000, "Fuzzy Logic Algorithm for a Robotic Sorting System with Colour Vision", *International Journal from IEEE*
- [4] Juhani Heilala, Paavo Voho, 2001, "Modular reconfigurable flexible final assembly systems", *Assembly Automation Volume 21 Number 1 2001*
- [5] William Bourn, 2001, "Modular assembly: more a process than an engineering technique", *Assembly Automation Volume 21 Number 1 2001*
- [6] Richard Piggin, 2001, "Fieldbus: where are we now?", *Assembly Automation Volume 21 Number 1 2001*
- [7] Richard Piggin, Ken Young, 2001, "Machine Safety with fieldbus", *Assembly Automation Volume 21 Number 1 2001*
- [8] Greg Causey, 2003, "Guidelines for the design of robotic gripping systems", *Assembly Automation Volume 23 Number 1 2003*
- [9] Gordon Lowe and Bijan Shirinzadeh, 2003, "Designing for task oriented grasps in robotic assembly", *Assembly Automation Volume 23 Number 1 2003*
- [10] OMRON, June 1996, "C200H-MC221 Motion Control Unit: Operation Manual Introduction." OMRON

[11] OMRON, June 1996, "C200H-MC221 Motion Control Unit: Operation Manual Details," OMRON

[12] OMRON, June 1994, "User's Manual OMNUC U series AC servomotors/drivers (30 to 750-W Analog Inputs),"OMRON

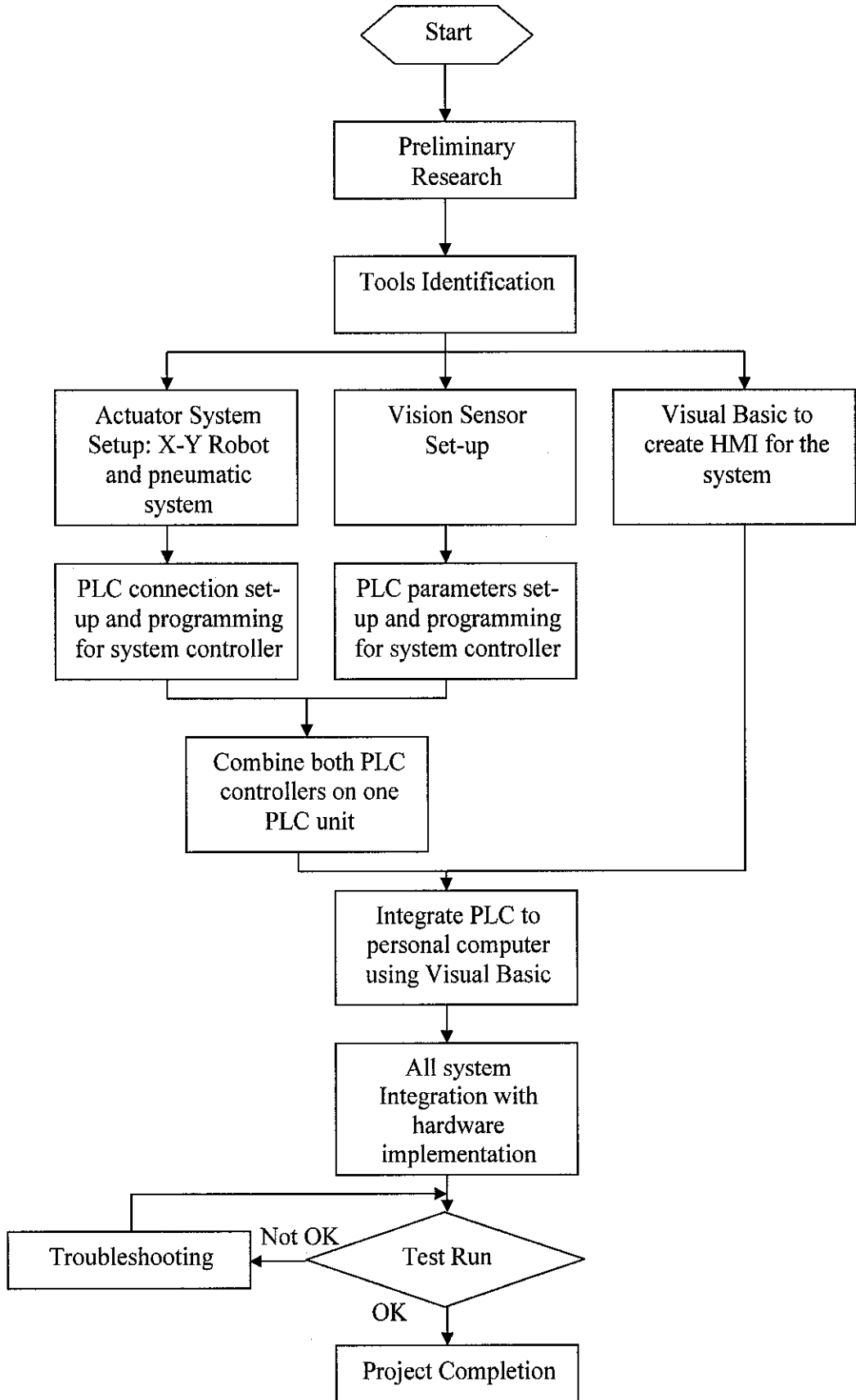
[13] OMRON, July 1999, "FinsGateway", OMRON

[14] OMRON, 2000, "SYSMAC Compolet Version2 User's Manual". OMRON

[15] OMRON, July 2003, "F150-3 Vision Sensor Manual 1: Setup Manual", OMRON

[16] OMRON, October 2000, "F150-3 Vision Sensor Manual 2: Expert Menu Operation Manual", OMRON

APPENDIX I: PROJECT FLOW-CHART

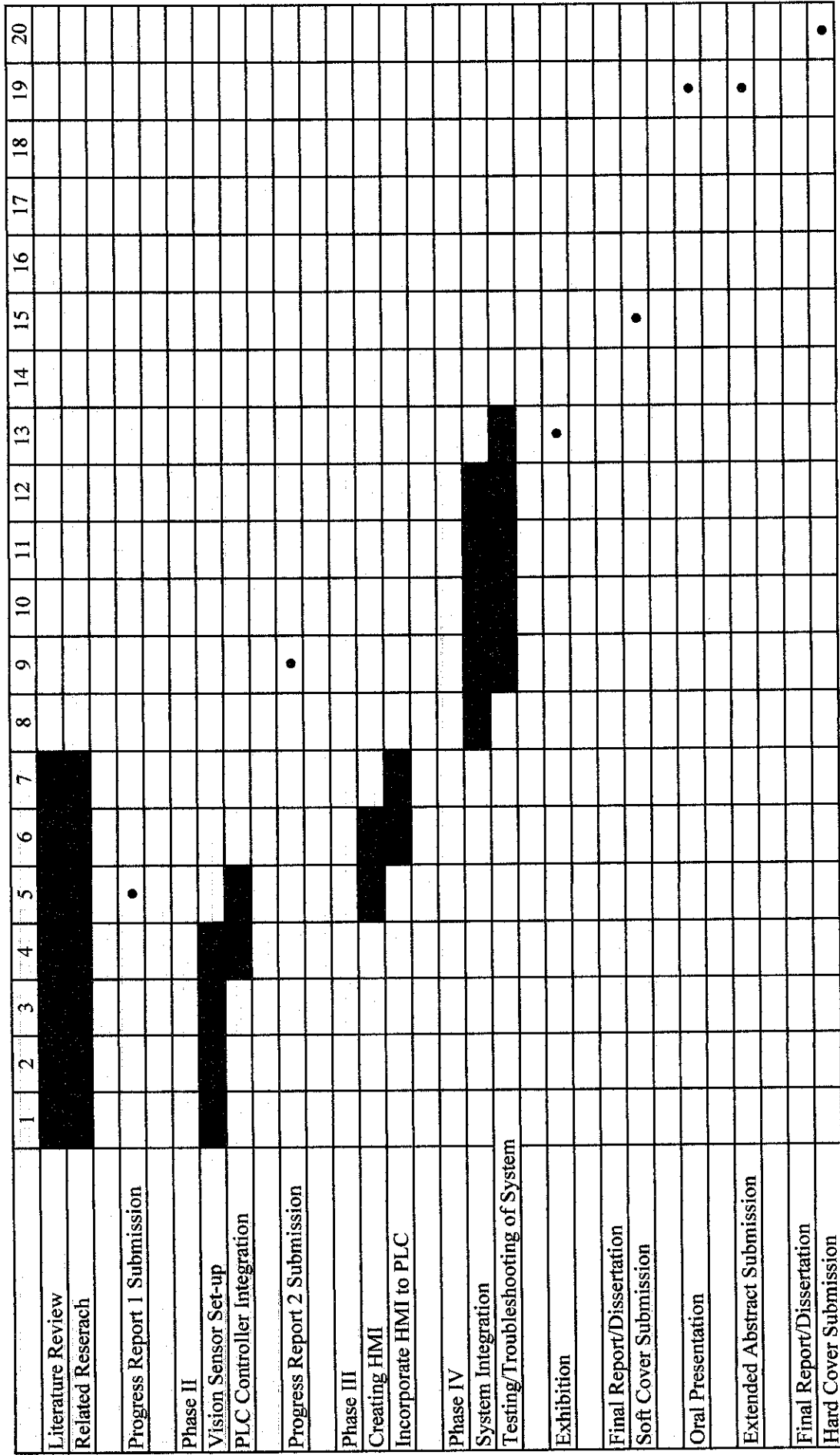


APPENDIX II: GANTT CHART

Gantt Chart for First Semester

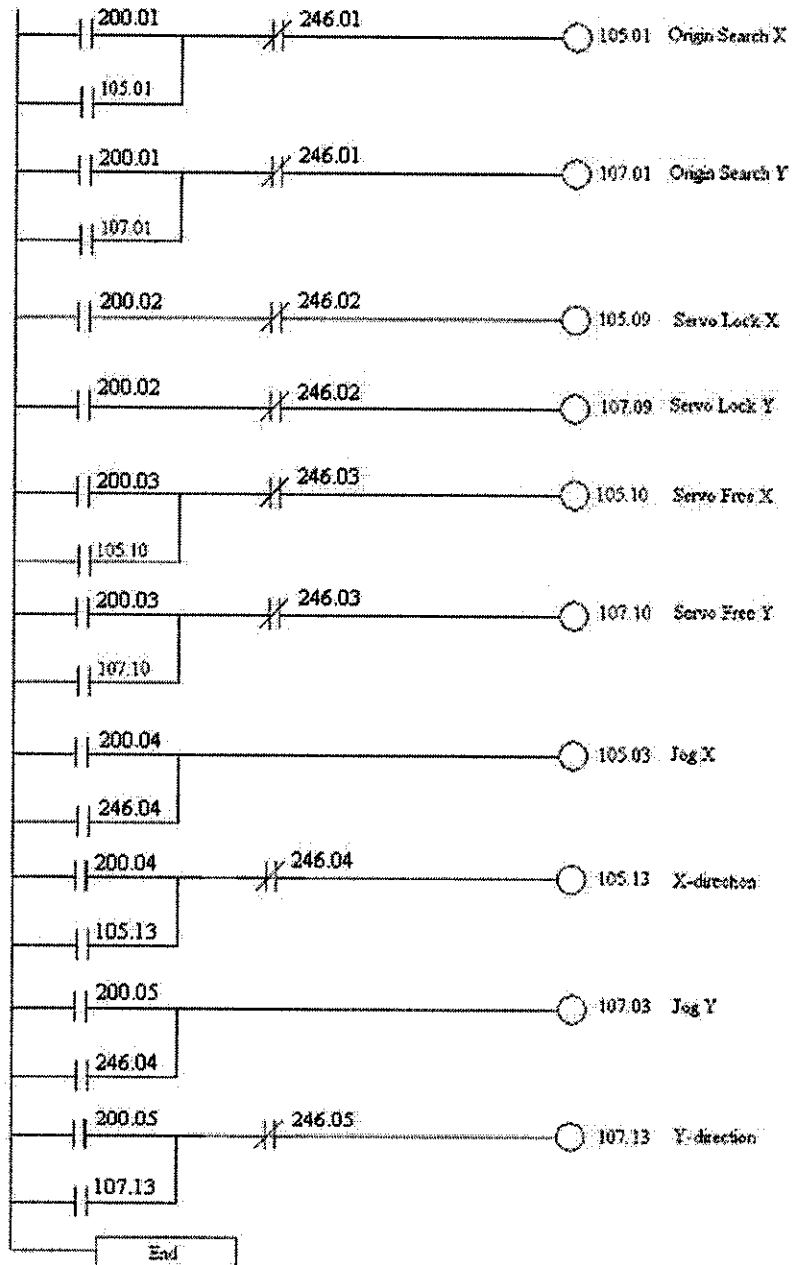
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Selection of Topic															
Topic Proposal		•													
Confirmation of Title			•												
Preliminary Research Work															
Literature Review															
Preliminary Report Submission					•										
Phase I															
X-Y Robot Set-up															
Pneumatic System Setup															
PLC Controller Integration of Actuators															
Progress Report Submission									•						
Interim Report Submission														•	
Oral Presentation															•

Gantt Chart for Second Semester

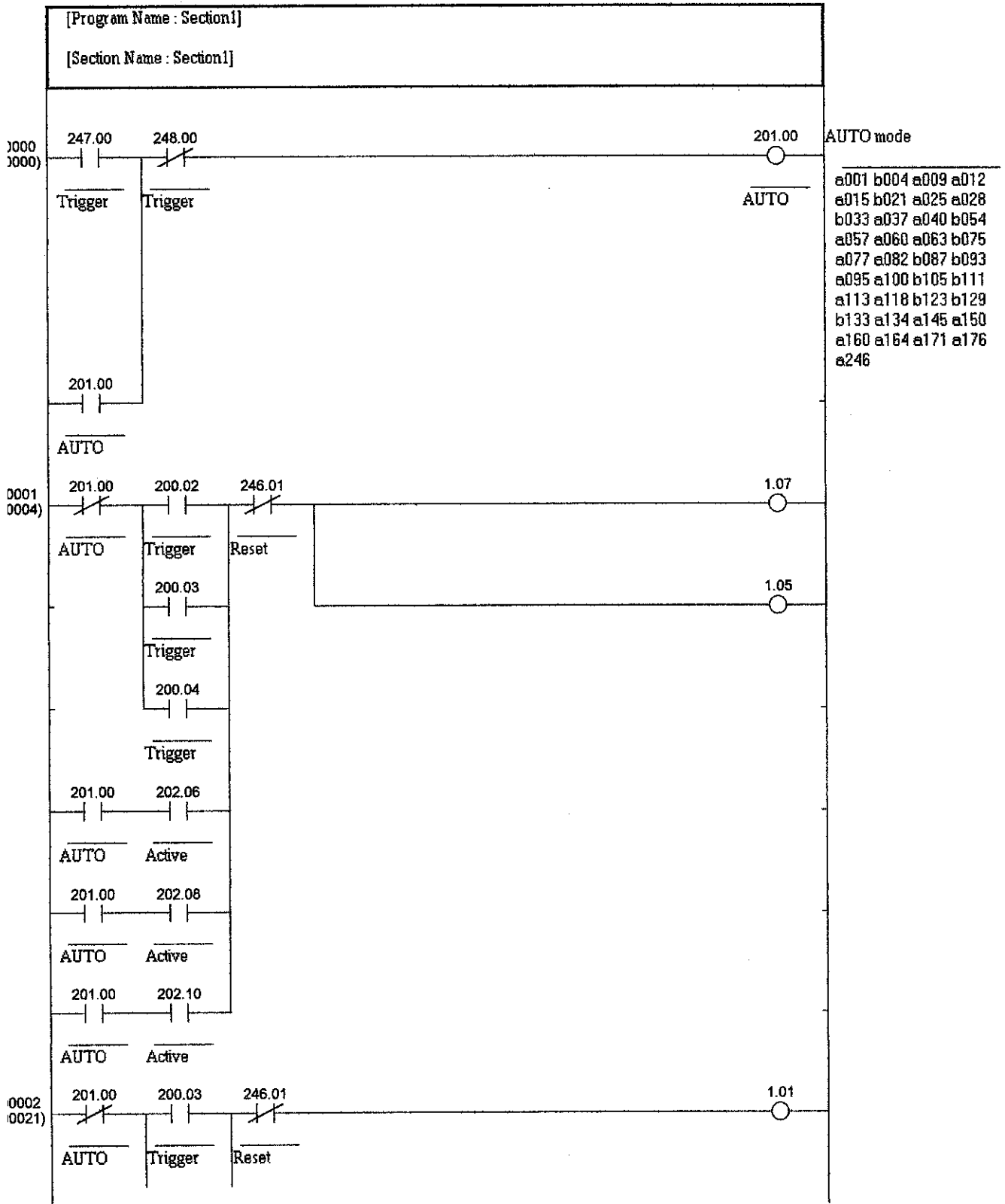


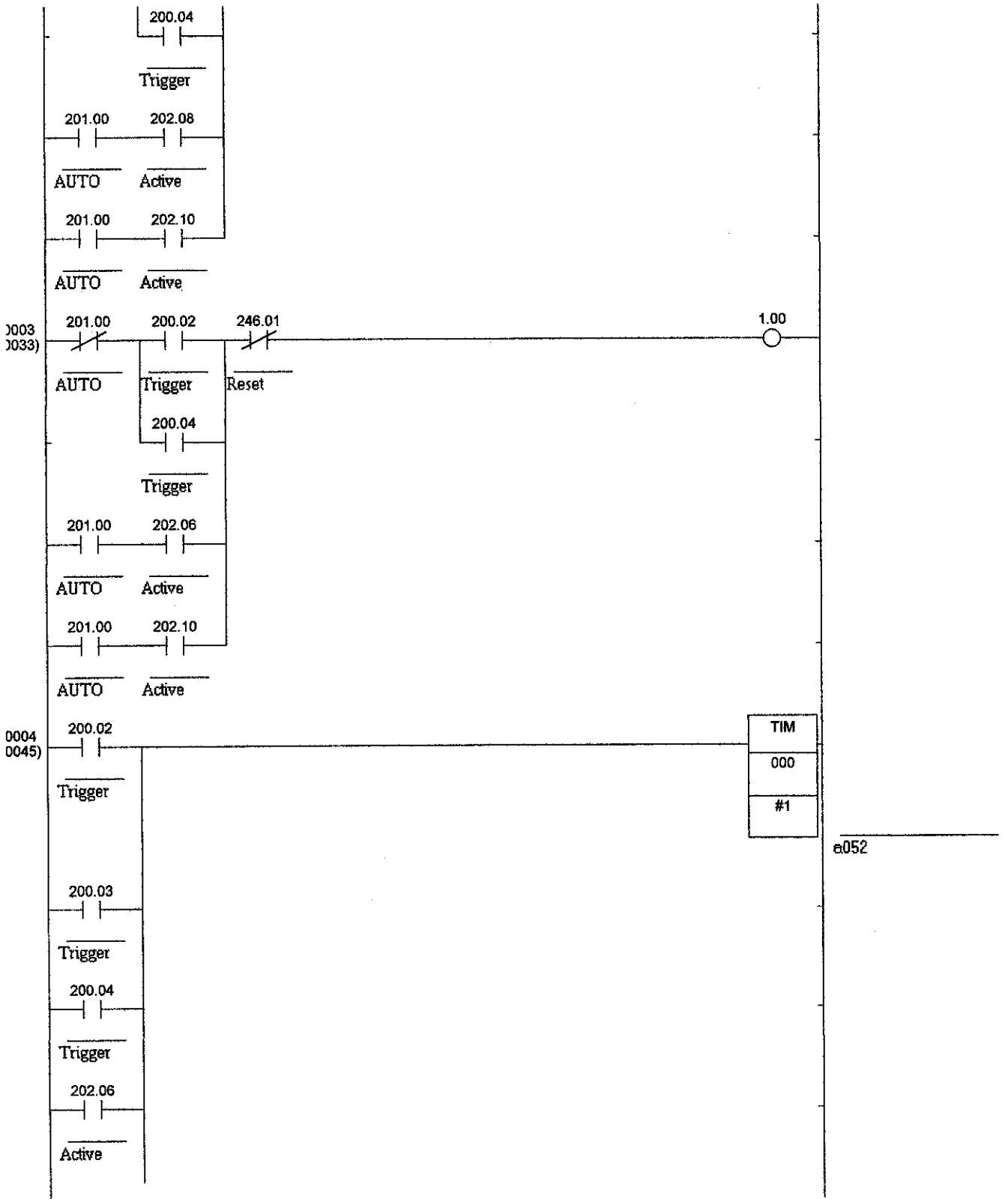
APPENDIX III: LADDER DIAGRAM

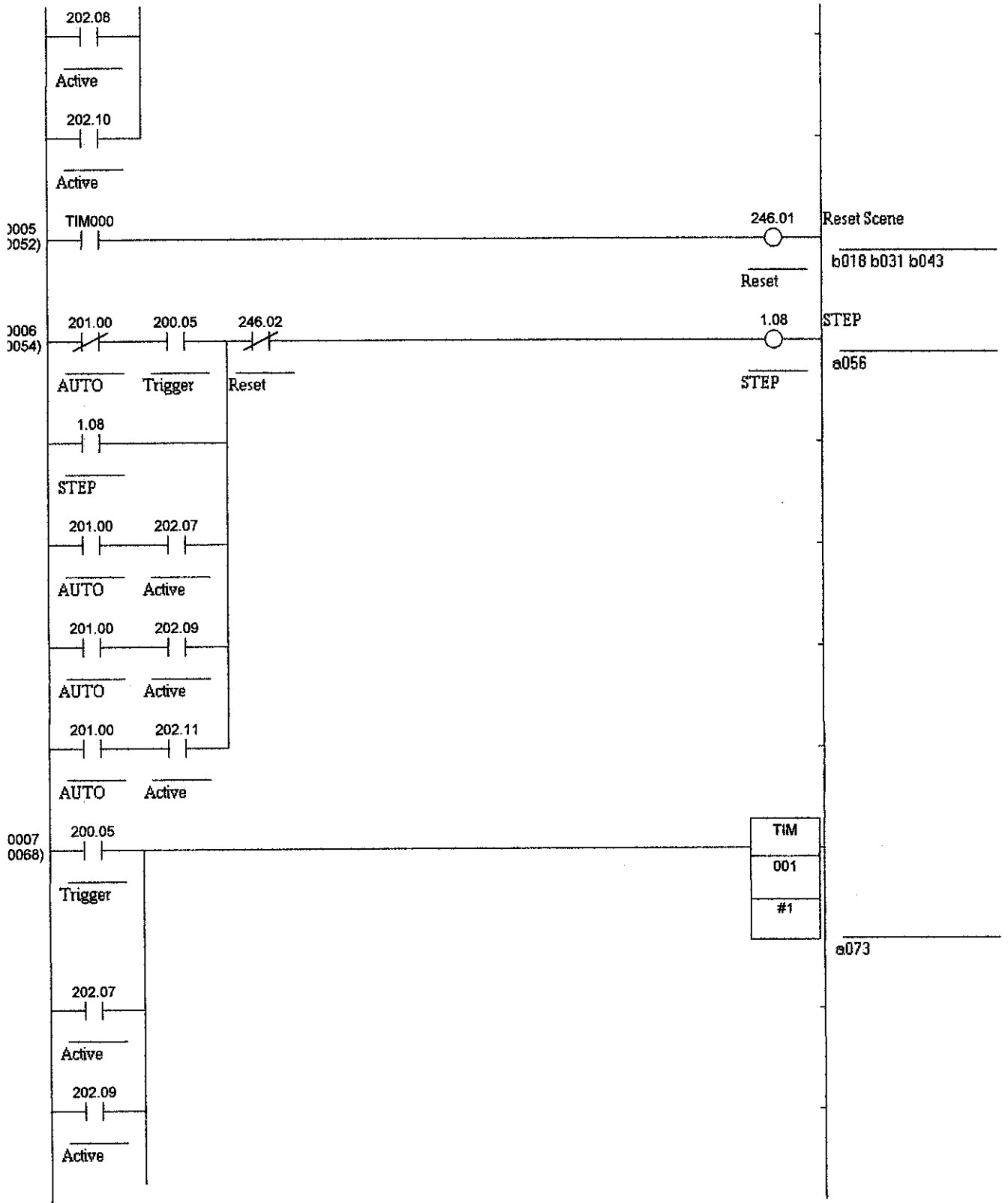
Ladder Diagram for X-Y Robot Manual Mode Operation

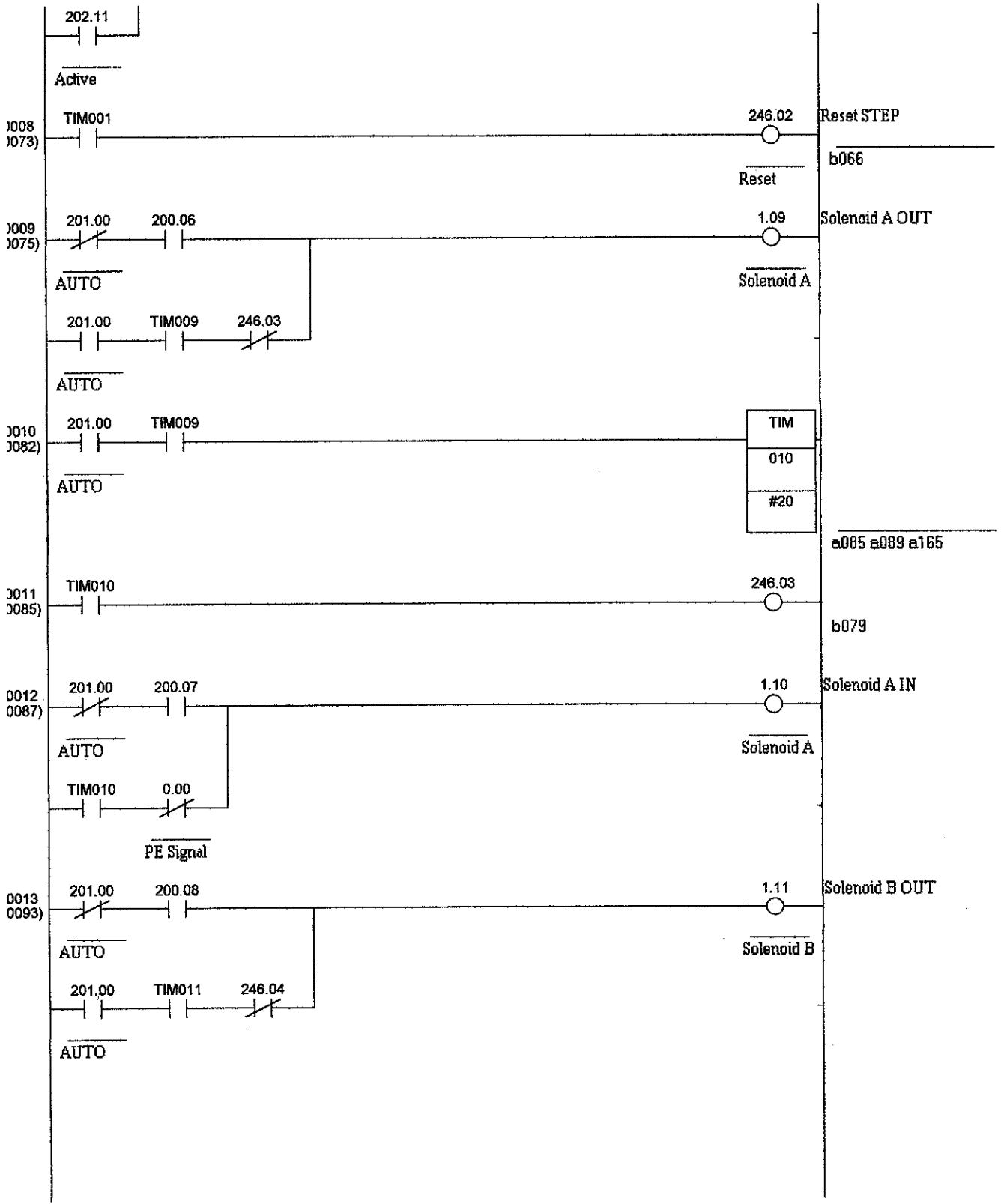


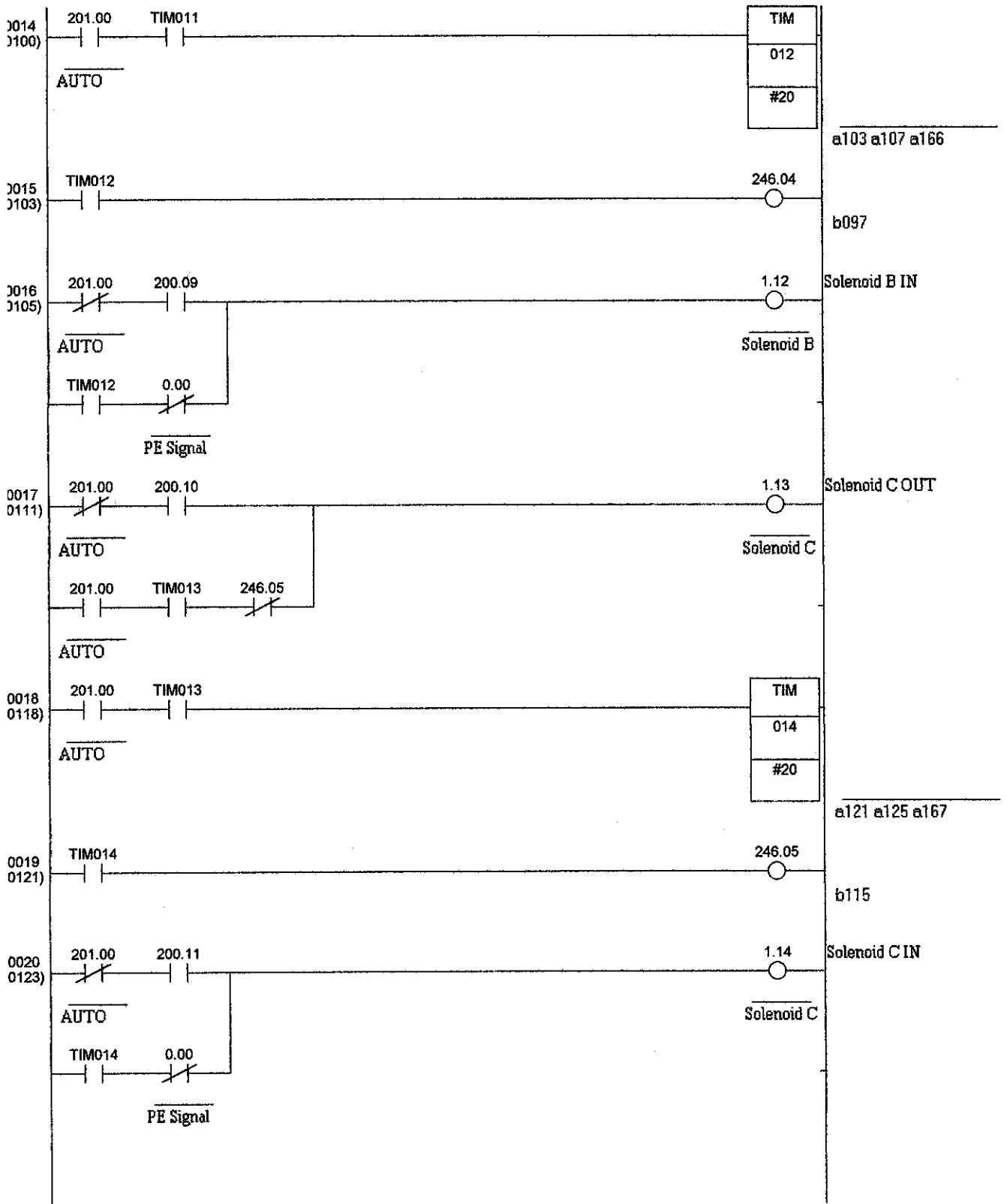
Ladder Diagram for Sorting System

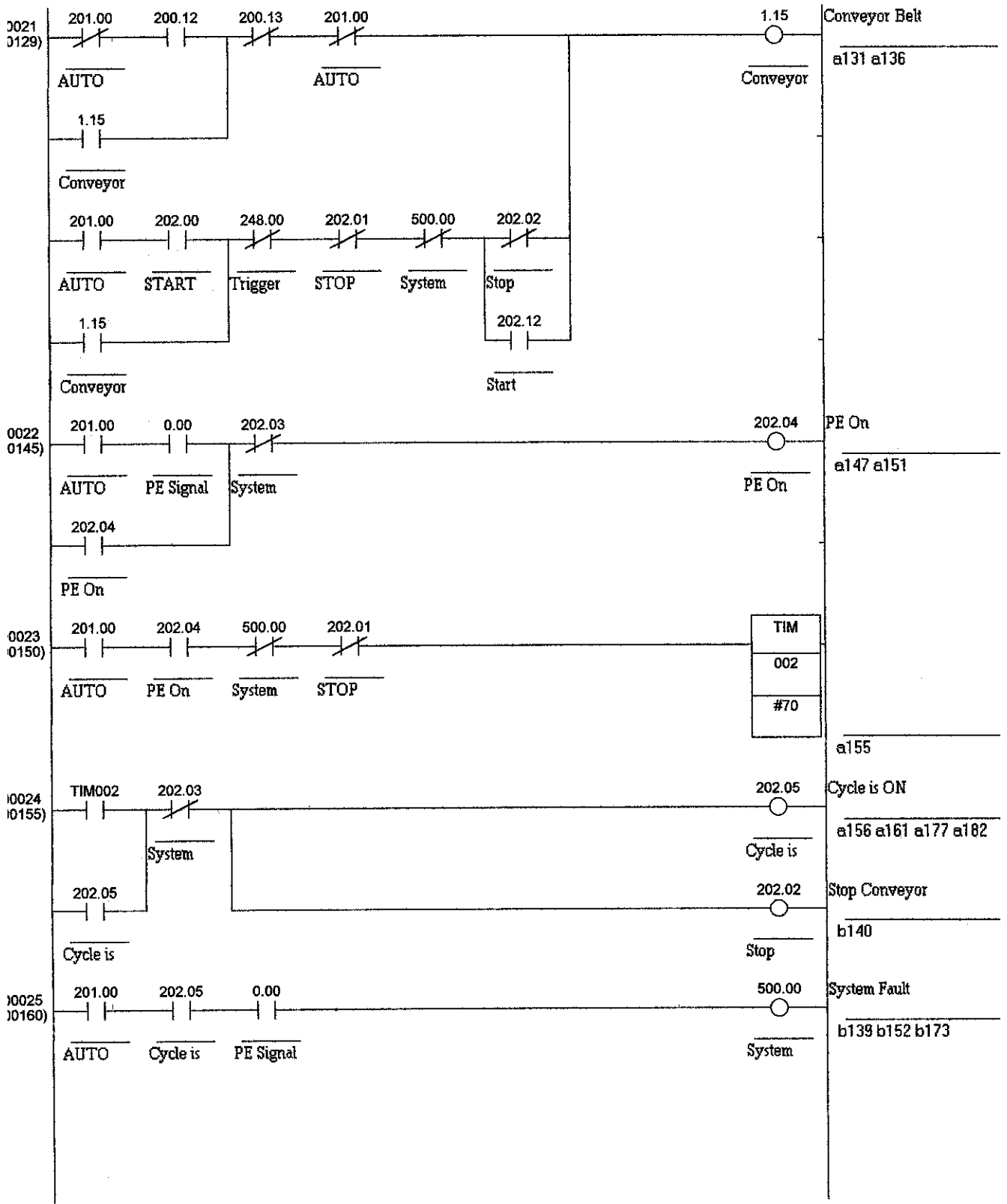


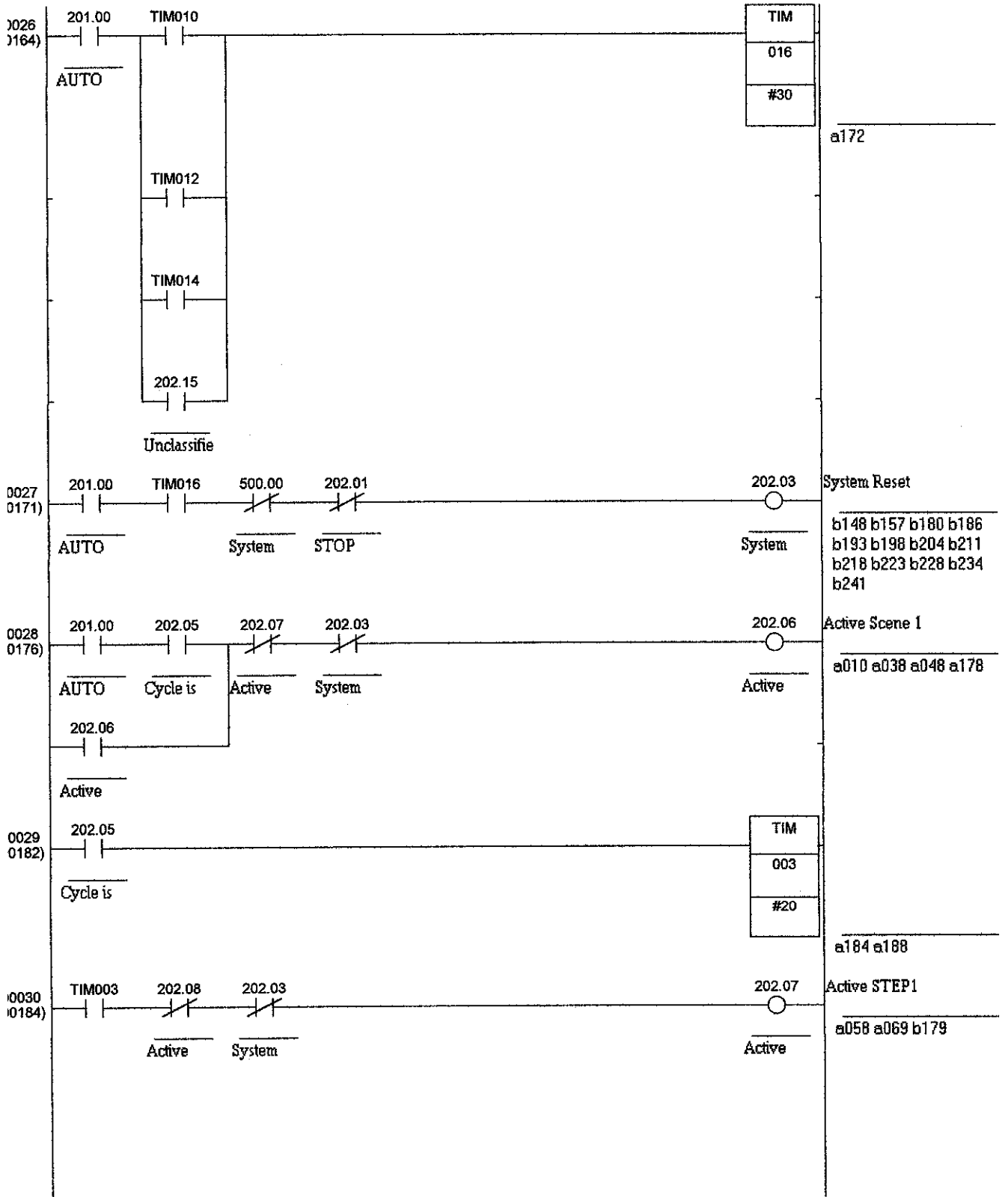


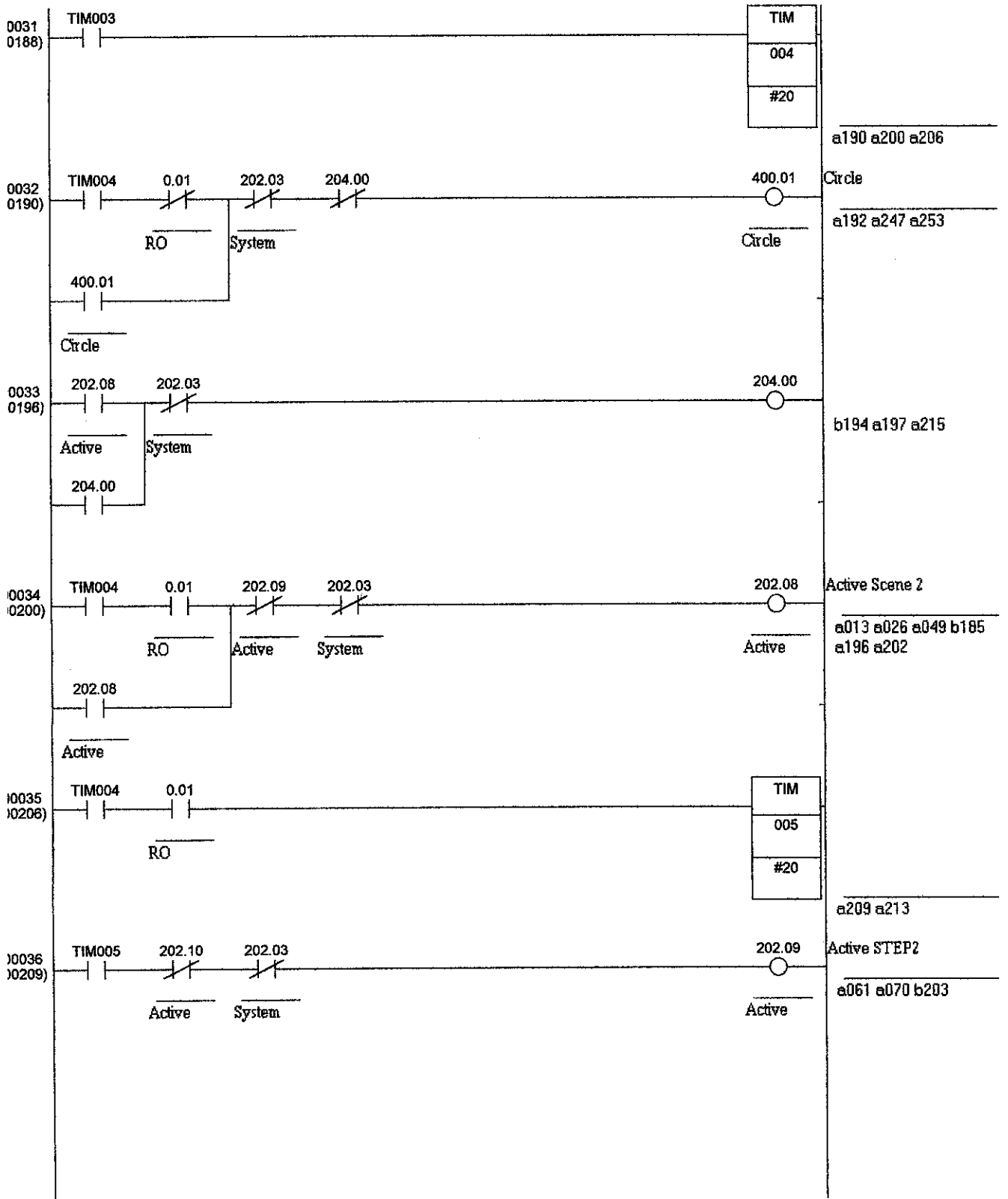


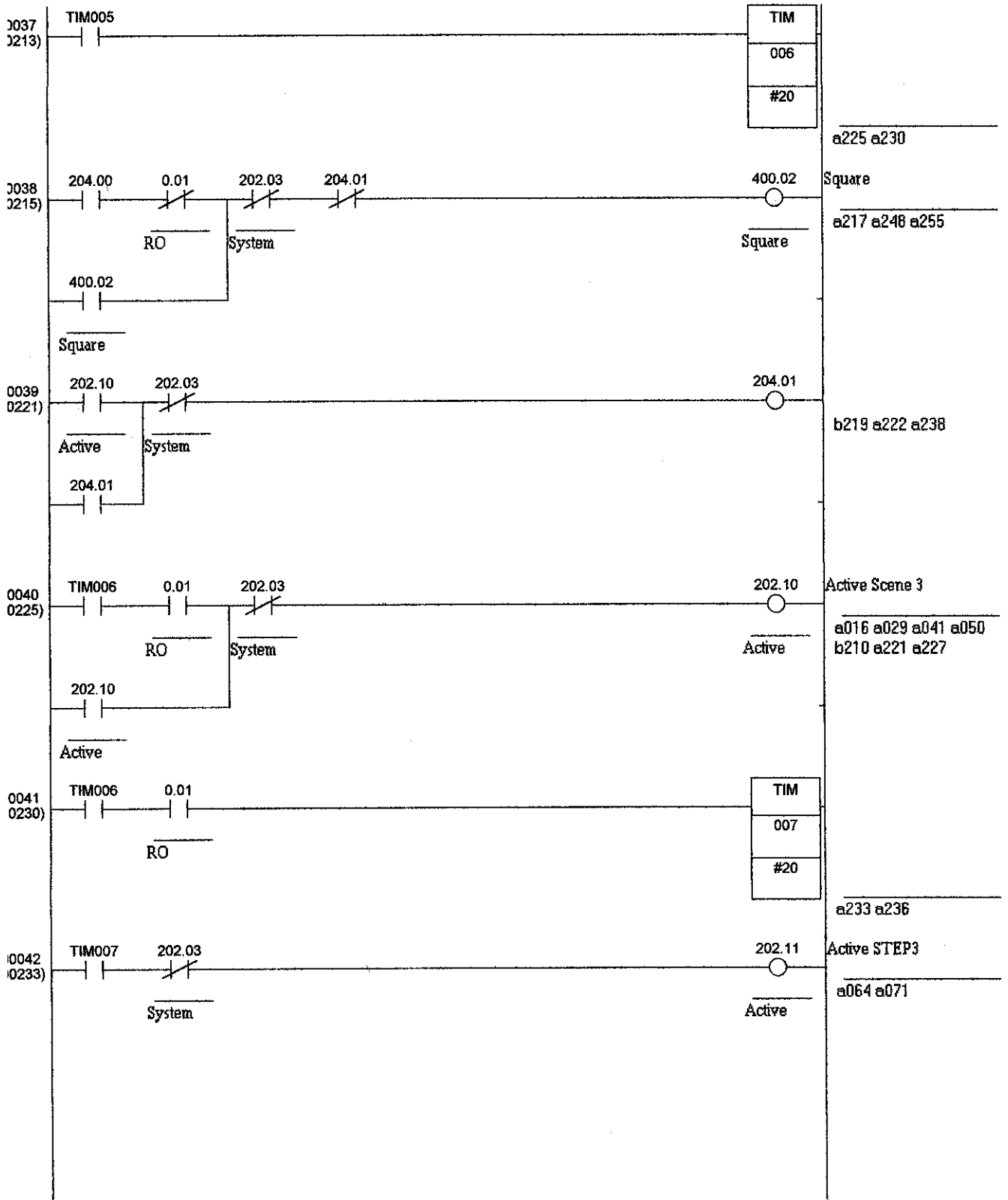


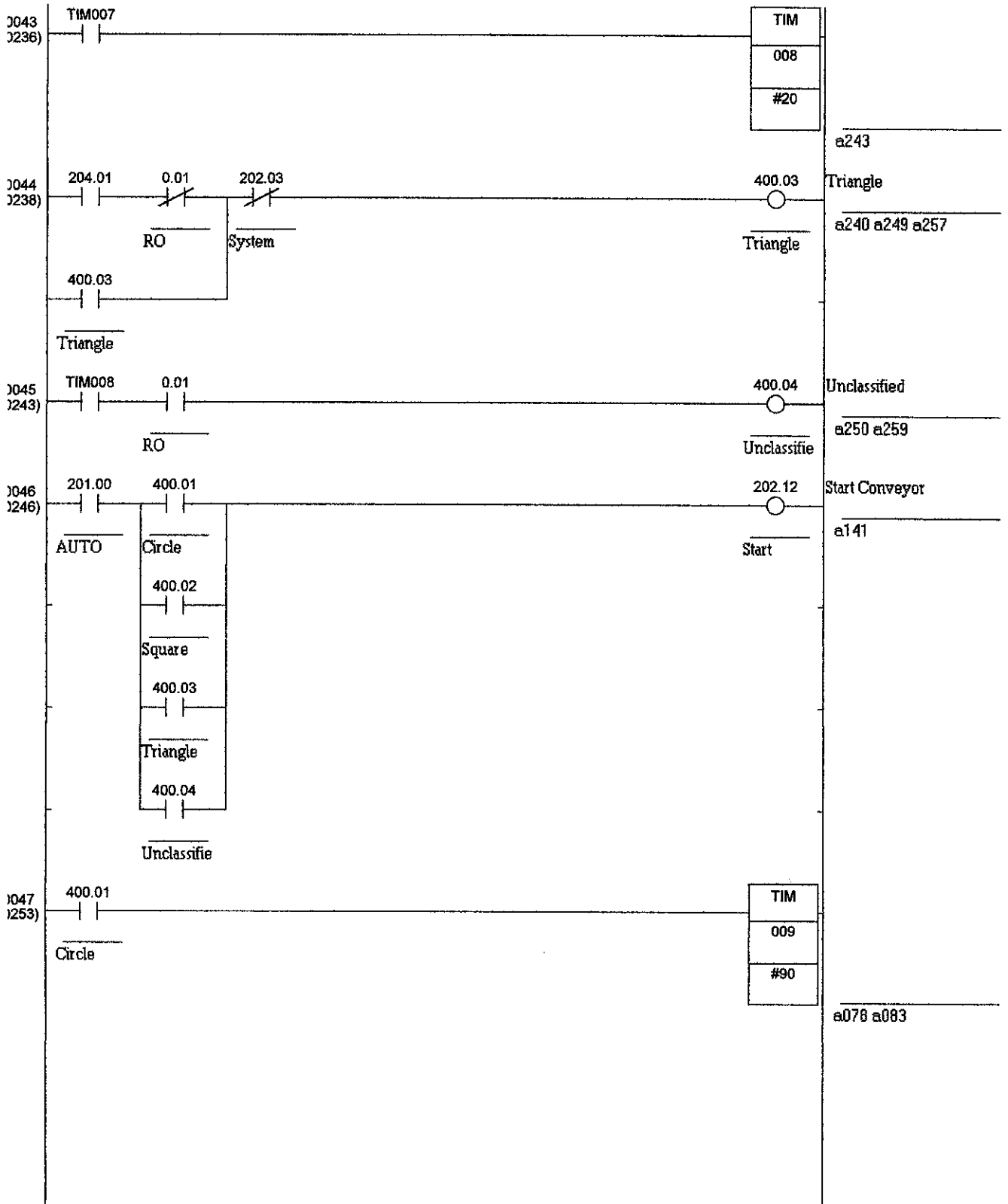


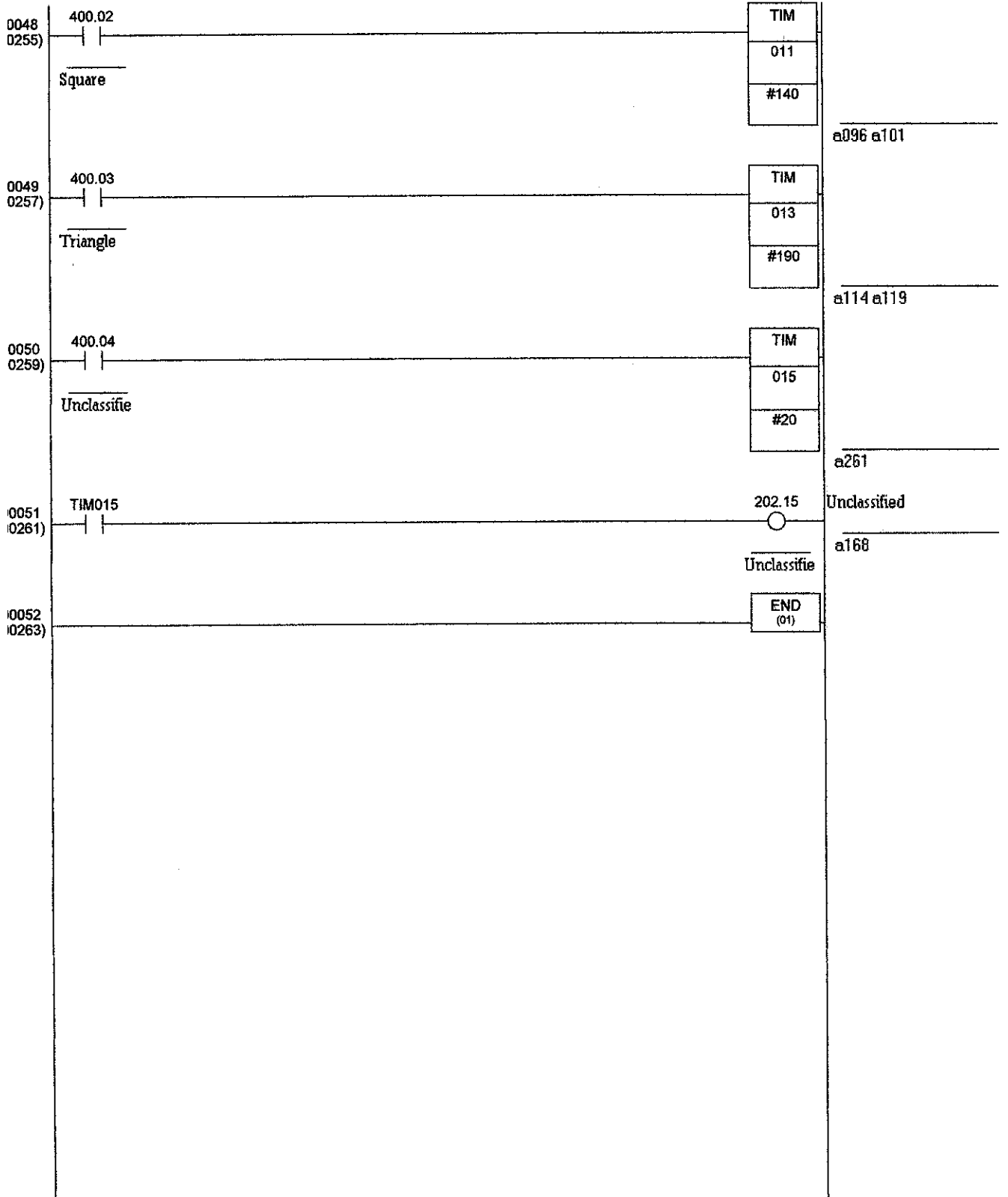






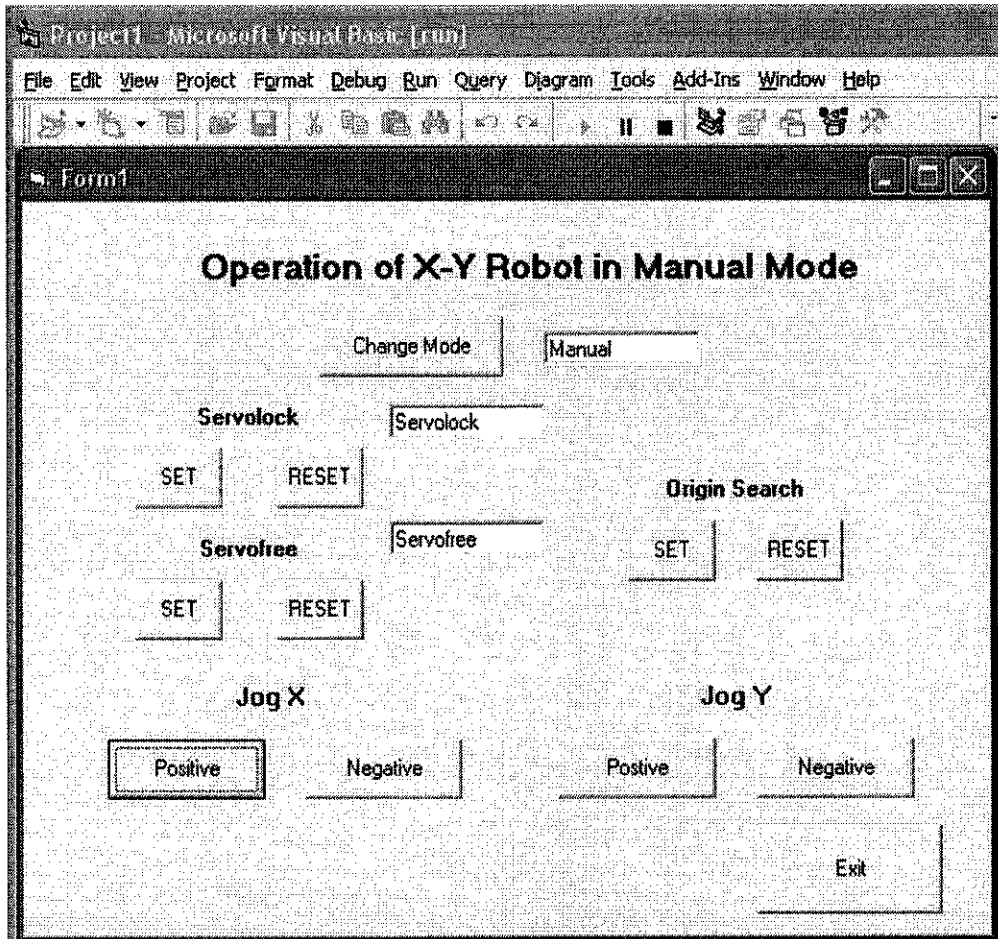




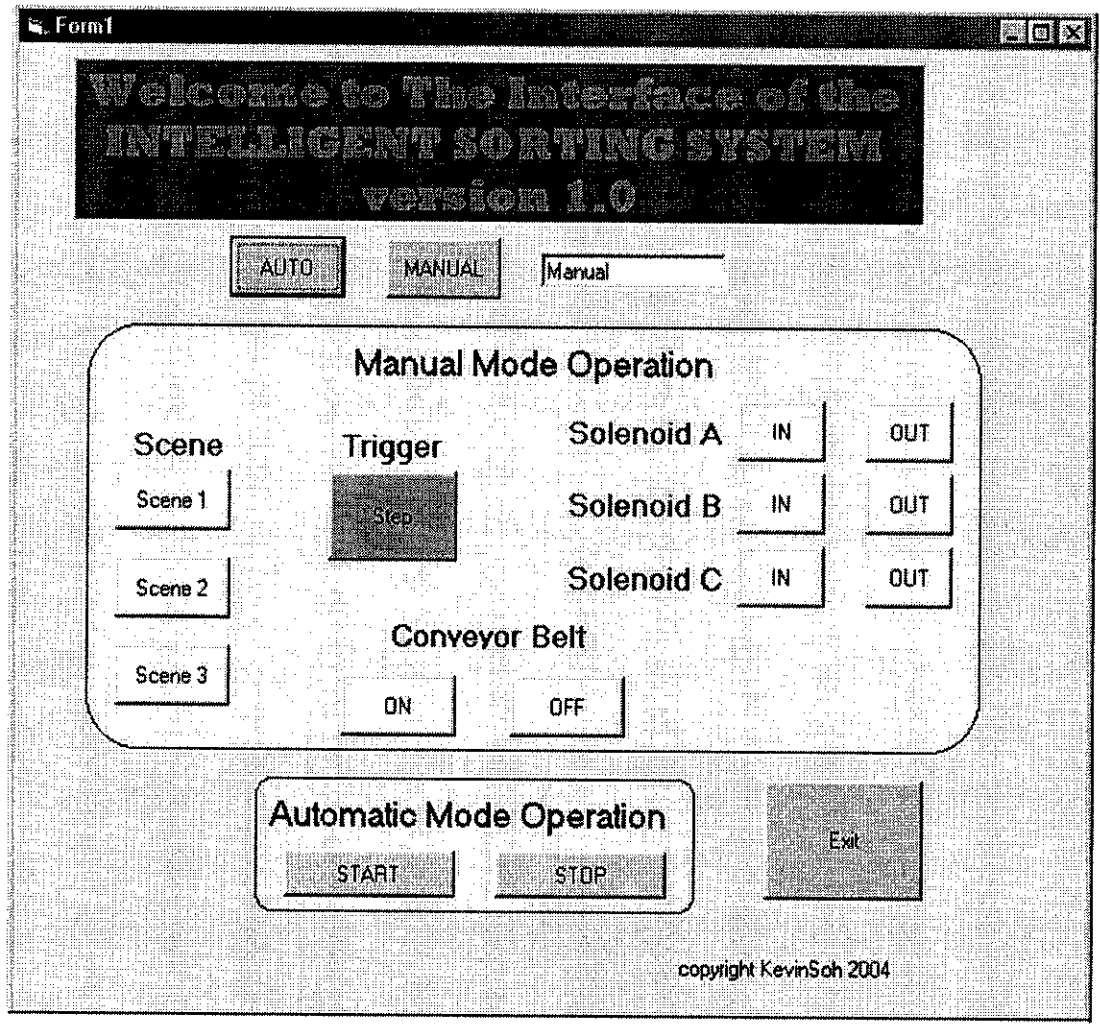


APPENDIX IV: VISUAL BASIC INTERFACE LAYOUT

Visual Basic Interface Layout for X-Y Robot Manual Mode Operation



Visual Basic Interface Layout for Sorting System



APPENDIX V: VISUAL BASIC CODES

Visual Basic Codes for the X-Y Robot Manual Mode Operation Interface Layout

```
Private Sub changemode_Click()
```

```
    If modetext.Text = "Automatic" Then modetext.Text = "Manual" Else If  
        modetext.Text = "Manual" Then modetext.Text = "Automatic"
```

```
End Sub
```

```
Private Sub exit_Click()
```

```
    End
```

```
End Sub
```

```
Private Sub originsearchset_Click()
```

```
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0002"
```

```
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0000"
```

```
End Sub
```

```
Private Sub originsearchreset_Click()
```

```
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0002"
```

```
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0000"
```

```
End Sub
```

```
Private Sub servofreereset_Click()
```

```
    servofreetext.Text = "Idle"
```

```
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0008"
```

```
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0000"
```

```
End Sub
```

```
Private Sub servofreeset_Click()
    servofree.Text = "servofree"
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0008"
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0000"
End Sub
```

```
Private Sub servolockreset_Click()
    servolocktext.Text = "Idle"
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0004"
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0000"
End Sub
```

```
Private Sub servolockset_Click()
    servolocktext.Text = "Servolock"
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0004"
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0000"
End Sub
```

```
Private Sub title_Click()
End Sub
```

```
Private Sub Xnegative_Click()
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0010"
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0000"
End Sub
```

```
Private Sub Xpositive_Click(Index As Integer)
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0010"
    SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0000"
End Sub
```

Private Sub Ynegative_Click()

 SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0020"

 SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0000"

End Sub

Private Sub Ypositive_Click()

 SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0020"

 SYSMAC_C1.writeArea plcAreaCIO, 246, 1, "0000"

End Sub

Visual Basic Codes for the Sorting System Interface Layout

```
Private Sub Ain_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0080"  
End Sub
```

```
Private Sub Aout_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0040"  
End Sub
```

```
Private Sub Auto_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 247, 1, "0001"  
    SYSMAC_C1.writeArea plcAreaCIO, 248, 1, "0000"  
    Mode.Text="Automatic"  
End Sub
```

```
Private Sub Bin_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0200"  
End Sub
```

```
Private Sub Bout_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0100"  
End Sub
```

```
Private Sub Cin_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0800"  
End Sub
```

```
Private Sub Cout_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0400"  
End Sub
```

```
Private Sub Conoff_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "2000"  
End Sub
```

```
Private Sub Conon_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "1000"  
End Sub
```

```
Private Sub Exit_Click()  
    End  
End Sub
```

```
Private Sub Manual_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 248, 1, "0001"  
    SYSMAC_C1.writeArea plcAreaCIO, 247, 1, "0000"  
    Mode.Text="Manual"  
End Sub
```

```
Private Sub Scene1_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0004"  
    Timer1=True  
End Sub
```

```
Private Sub Scene2_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0008"  
    Timer1=True  
End Sub
```

```
Private Sub Scene3_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0010"  
    Timer1=True  
End Sub
```

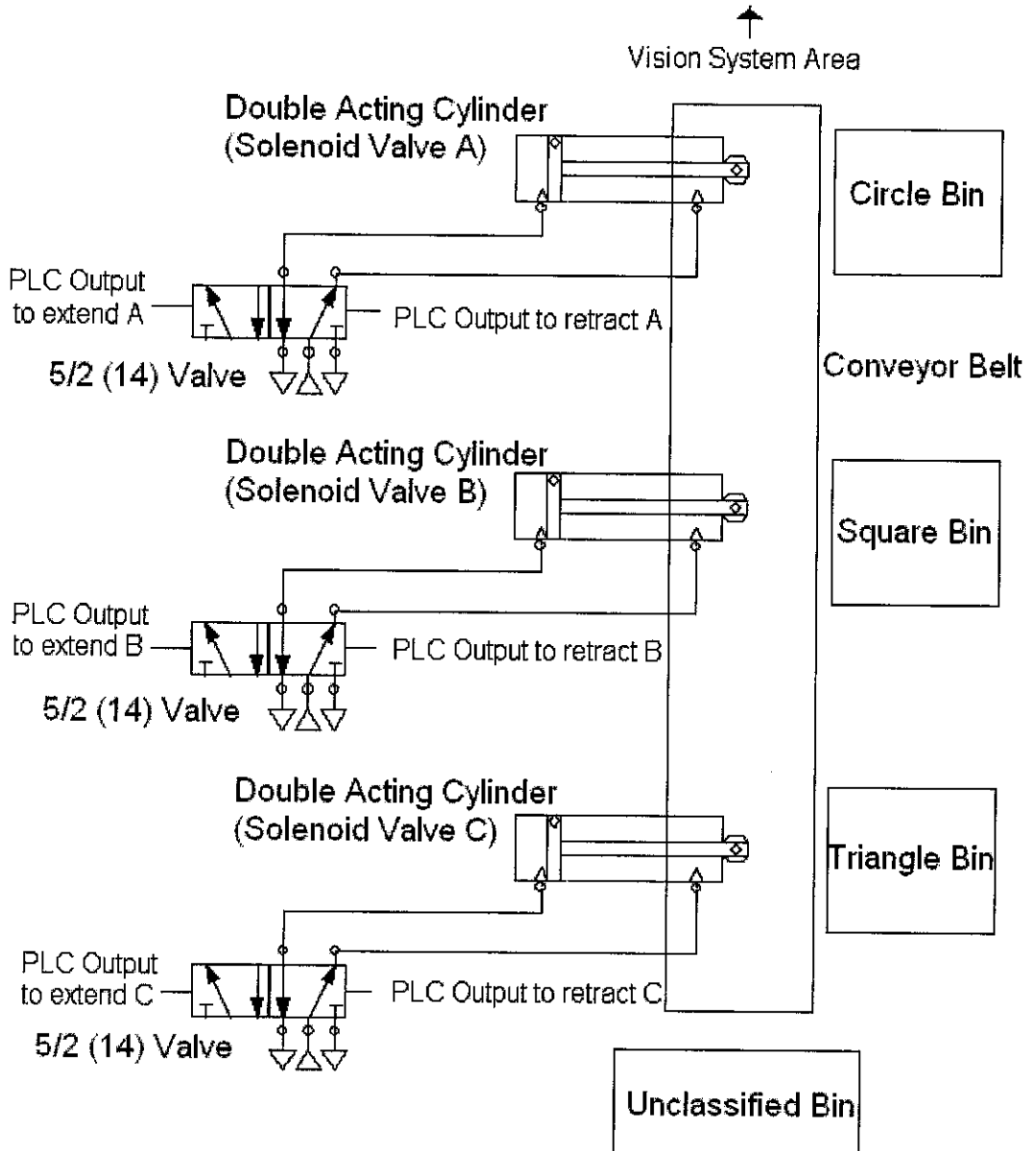
```
Private Sub startauto_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 202, 1, "0001"  
End Sub
```

```
Private Sub Step_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0020"  
End Sub
```

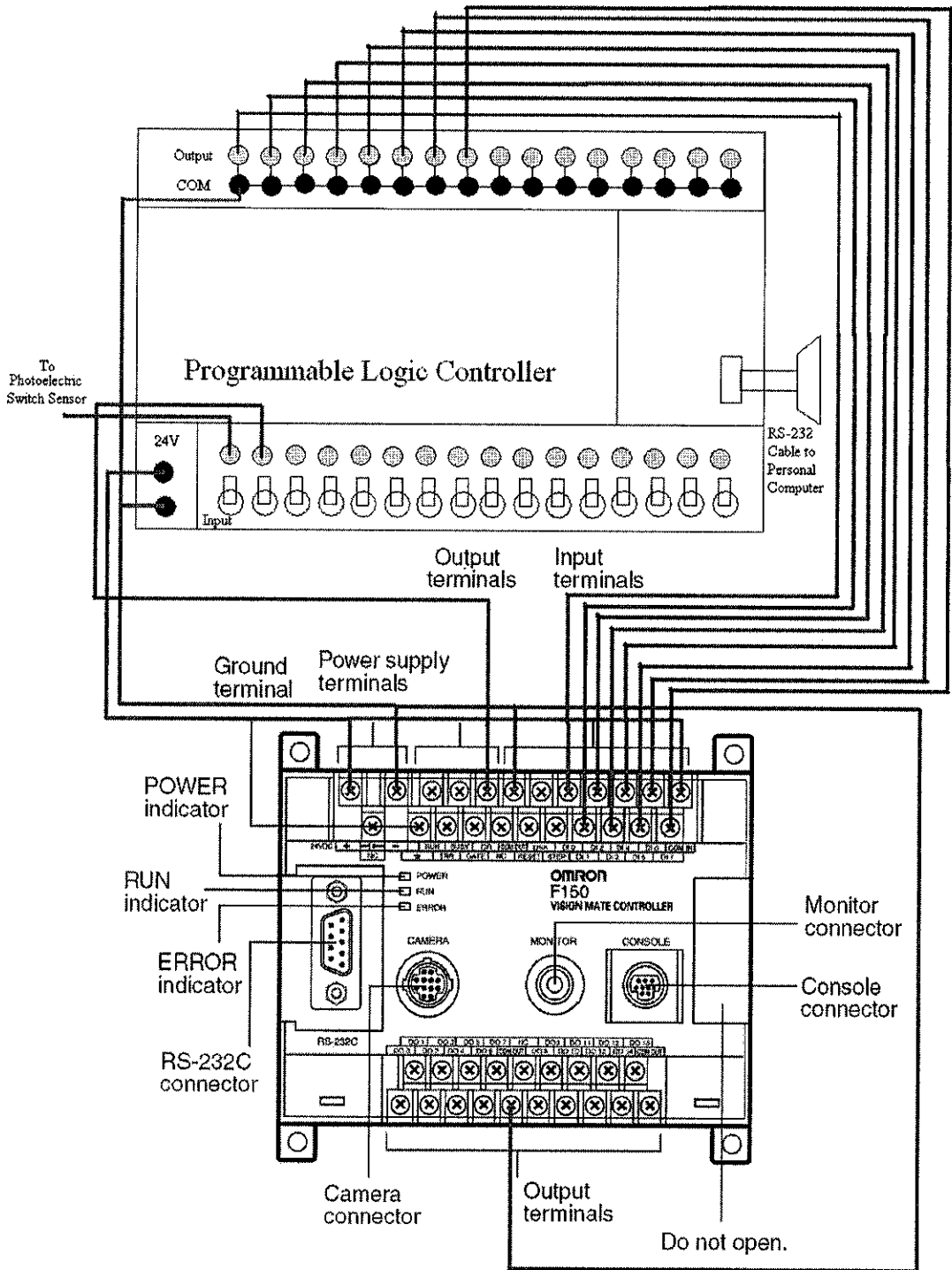
```
Private Sub stopauto_Click()  
    SYSMAC_C1.writeArea plcAreaCIO, 202, 1, "0002"  
End Sub
```

```
Private Sub Timer1_Timer()  
    SYSMAC_C1.writeArea plcAreaCIO, 200, 1, "0000"  
End Sub
```


APPENDIX VI: PNEUMATIC SYSTEM DESIGN LAYOUT

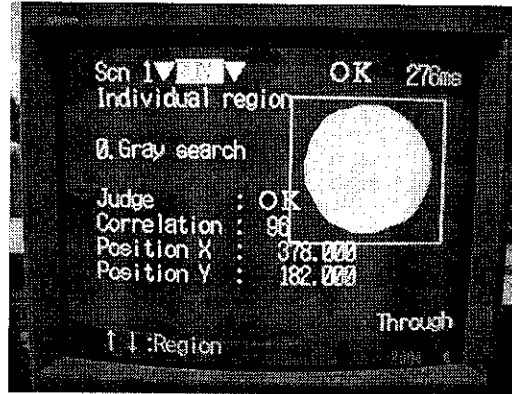


APPENDIX VII: CONNECTION DIAGRAM OF THE VISION CONTROLLER TO THE PLC

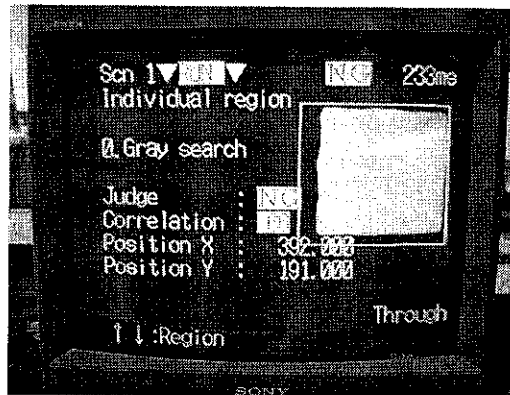


APPENDIX VIII: VISION SENSOR TESTING RESULTS

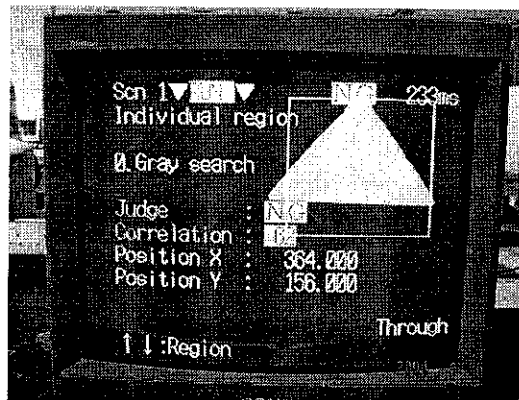
Grey Search with Perfect Model: Circle



Target Object Circle
Correlation Value: 96

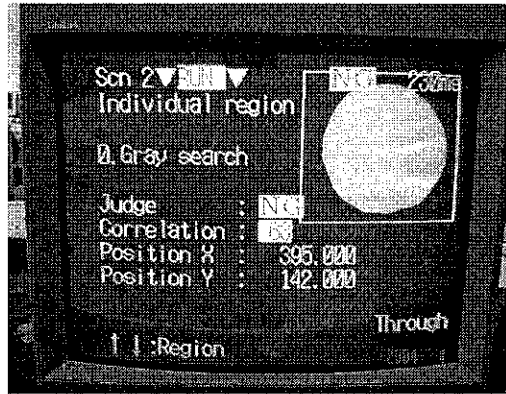


Target Object Square
Correlation Value: 66

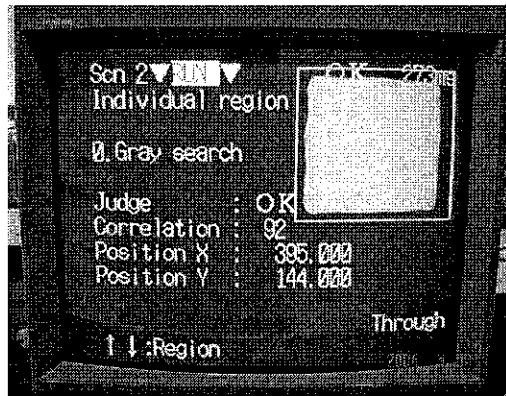


Target Object Triangle
Correlation Value: 62

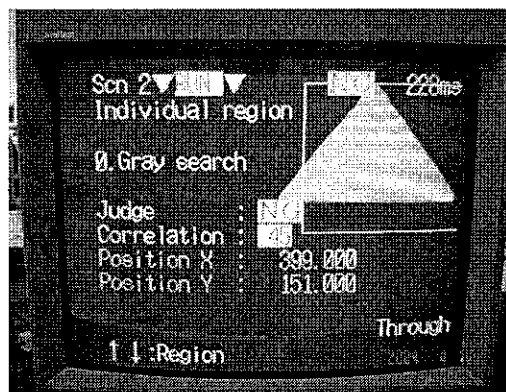
Grey Search with Perfect Model: Square



Target Object Circle
Correlation Value: 63

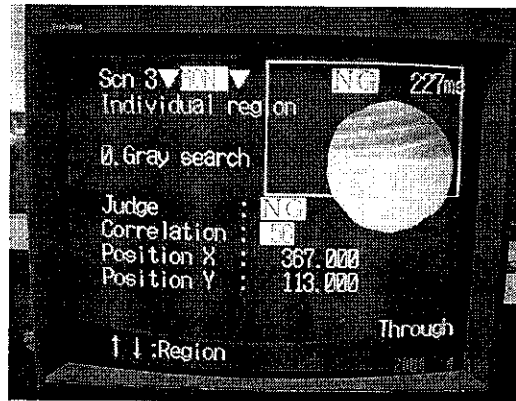


Target Object Square
Correlation Value: 92

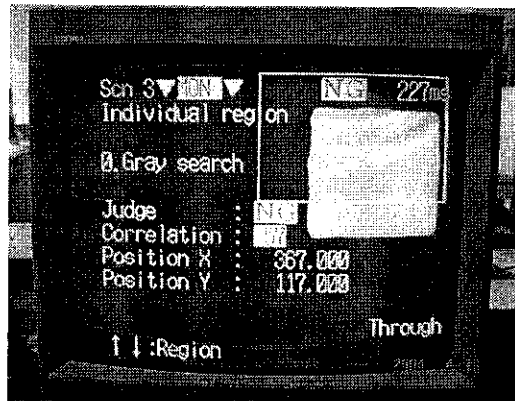


Target Object Triangle
Correlation Value: 45

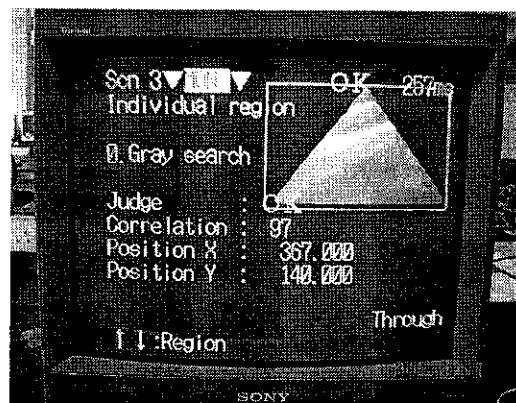
Grey Search with Perfect Model: Triangle



Target Object Circle
Correlation Value: 56

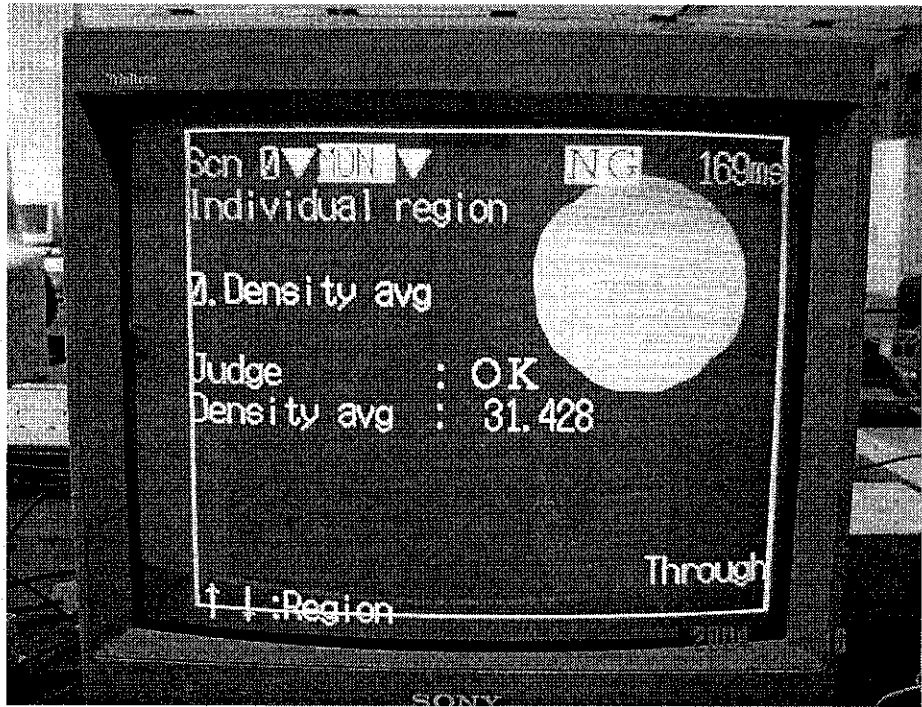


Target Object Square
Correlation Value: 57



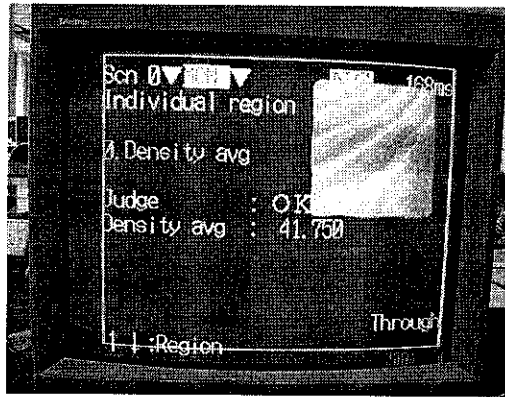
Target Object Triangle
Correlation Value: 97

Average Density of Circle

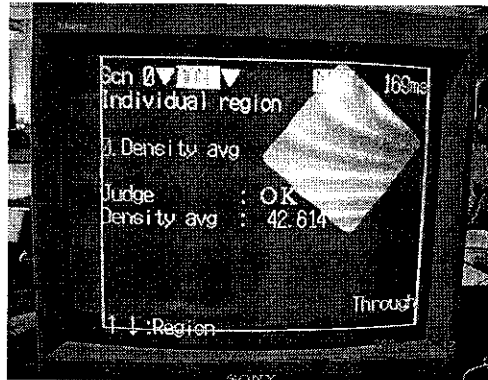


Target Object Circle
Average Density: 31.428

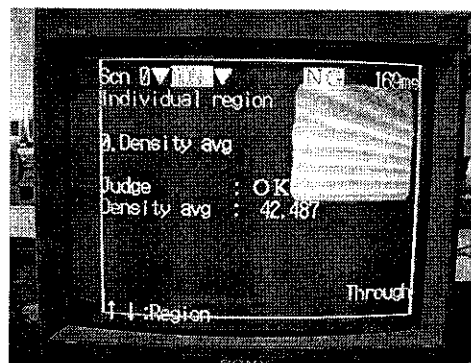
Average Density of Square



Target Object Square
Average Density: 41.750

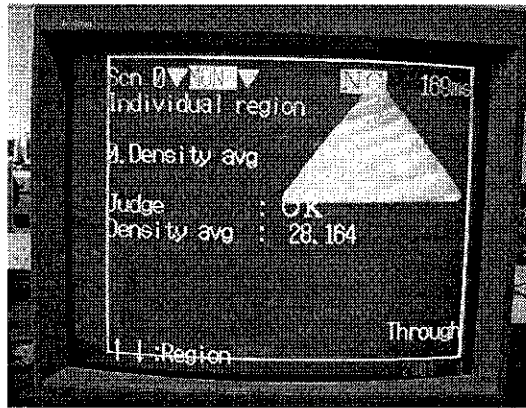


Target Object Square
Average Density: 42.614

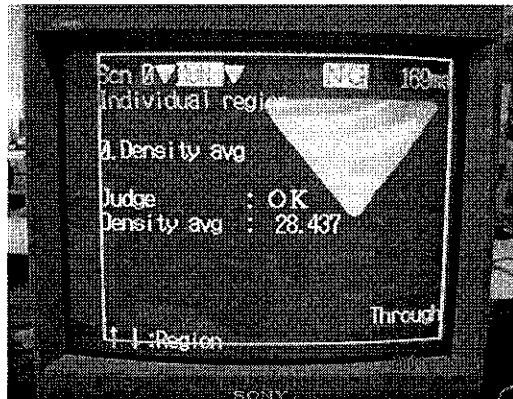


Target Object Square
Average Density: 42.487

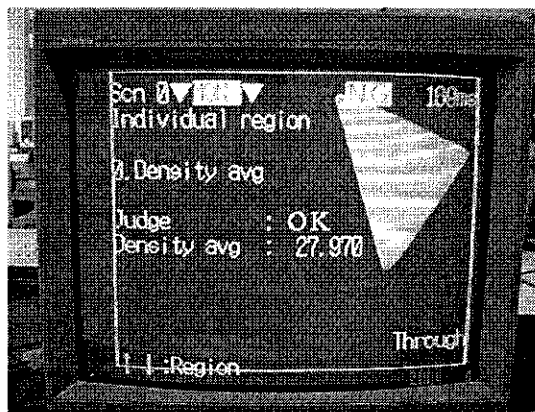
Average Density of Triangle



Target Object Triangle
Average Density: 28.164

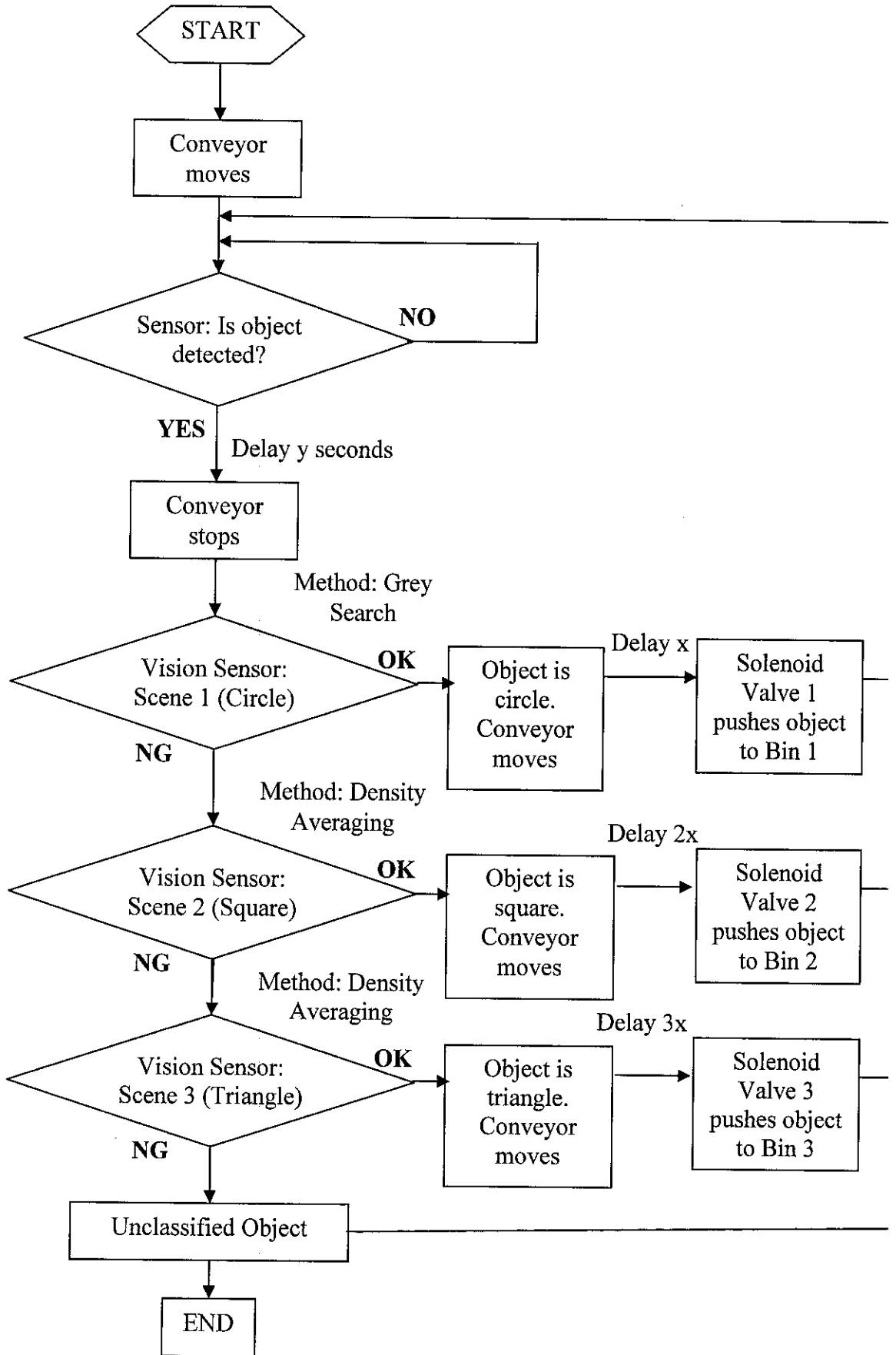


Target Object Triangle
Average Density: 28.437



Target Object Triangle
Average Density: 27.970

APPENDIX IX: SORTING SYSTEM PROCESS FLOWCHART



APPENDIX X: SYSTEM DESIGN LAYOUT OF THE INTELLIGENT SORTING SYSTEM

