CONTROL OF AN AUTOMATED STORAGE AND RETRIEVAL SYSTEM

by

Terry Lawrence

Report submitted to the Faculty of
Virginia Polytechnic Institute and State University
in the partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING

in

Industrial and Systems Engineering

Approved:

Dr. Michael P. Deisenroth

Dr. Charles F. Reinholtz

Dr. Subhash C. Sarin

December, 1996
Blacksburg, VA

Keywords: AMHS, AS/RS, Automation, Robotics

c.2
LD
5655
V851
1996
L377
c.2

CONTROL OF AN

AUTOMATED STORAGE AND RETRIEVAL SYSTEM

by

Terry Lawrence

Committee Chairman: Dr. Michael P. Deisenroth

Industrial and Systems Engineering

ABSTRACT

The Robotics and Automation Laboratory at Virginia Polytechnic Institute and State University contains a Flexible Machining and Assembly System (FMAS) and an Automated Storage and Retrieval System (AS/RS). These systems were developed for educational purposes to emulate a typical flexible manufacturing process in industry. The AS/RS hardware was designed and constructed in a previous project by Mark Eaglesham. His project consisted of hardware development only and did not include control system development.

The objective of this project is to design a set of software primitives to control the AS/RS. This set of primitive control sequences for the AS/RS could be used by future students to further develop the AS/RS control system at the device control level or to integrate the AS/RS with the FMAS at the workcell control level. The set of control primitives were developed in Relay Ladder Logic (RLL) on a Texas Instruments 565 Programmable Logic Controller (PLC). The set of primitives

include the following functions: initialize the system, store a pallet, retrieve a pallet, move to a specified location, and shutdown the system.

A communications dialog was developed for command and control interaction between the PLC and the workcell control computer. This protocol includes various functions to manage the AS/RS's operations. These functions commands from the workcell controller indicating what operation is needed and response from the AS/RS indicating its status and system error messages.

In order to test the AS/RS a program to simulate the workcell controller was developed. This project report includes a description of a test program developed using QuickC. Due to hardware difficulties, this test program was not implemented. The AS/RS primitives were tested using Texas Instrument's TISOFT RLL development and monitoring software package.

ACKNOWLEDGMENTS

I would like to thank those who have helped me throughout this project.
Dr. Deisenroth helped with his expertise and his recommendations.
Will Vest was always there to help in answering my questions and solving the many electrical problems that I couldn't.
My parents, Bill and Lesley, gave me their constant support and patience, especially through the stressful periods.

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1 Background 1

1.2 Description of the FMAS 2

1.3 History of AS/RS 4

2. PROBLEM STATEMENT AND OBJECTIVES 11

2.1 Problem Statement 11

3. LITERATURE REVIEW 13

3.1 General Automated Storage and Retrieval Systems 13

3.2 AS/RS Control Systems 14

3.3 AS/RS Control Design 15

3.4 Measurement Parameters 16

3.5 Travel Time Model 17

3.6 Trends in AS/RS 18

4. CONTROL SYSTEM DESIGN 20

4.1 Criteria 20

4.2 Constraints 21

5. CONTROL SYSTEM COMPUTER INTERFACE 25

5.1 Purpose 25

5.2 C Test Program 27

5.3 Workcell Controller and AS/RS Communication 29

6. RELAY LADDER LOGIC 32

6.1 General 32

6.2 AS/RS Ladder..... 34

7. LADDER ROUTINES..... 42
7.1 Main Program..... 42
7.2 Initialization..... 44
7.3 Store..... 45
7.4 Retrieve..... 46
7.5 Address..... 48
7.6 Travel Level..... 48
7.7 Move to Row..... 49
7.8 Move to Column..... 50
7.9 Shutdown..... 50
7.10 Global Features..... 51

8. OPERATION AND TESTING OF THE AS/RS..... 56
8.1 AS/RS System Components..... 56
8.2 Procedure..... 57
8.3 Testing..... 58

9. CONCLUSIONS..... 61
9.1 Summary..... 61
9.2 Recommendations..... 62

REFERENCES..... 64

APPENDIX A: PLC INPUT AND OUTPUT..... 66

APPENDIX B: C PROGRAM TO SIMULATE WORKCELL CONTROLLER.... 69

APPENDIX C: RELAY LADDER LOGIC..... 78

APPENDIX D: LADDER LOGIC VARIABLES 118

LIST OF FIGURES

Figure 1.1: Schematic of the Virginia Tech FMAS 3

Figure 4.1: AS/RS Storage Rack 24

Figure 5.1: Main Menu 27

Figure 5.2: C Program Output Example 31

Figure 7.1: Shuttle Mechanism Fingers and Sensors 54

Figure 7.2: Loading a Pallet onto the Shuttle Mechanism 55

Figure 7.3: Pallet Loaded on the Shuttle Mechanism 55

LIST OF TABLES

Table 1.1: Toshiba/TI565 Interconnections. 8

Table 1.2: Crane & Shuttle/TI565 Interconnections 10

Table 5.1: AS/RS Commands 25

Table 6.1: AS/RS Status Codes 36

Table 6.2: Internal Coils 41

Table 7.1: AS/RS Commands 44

Table 8.1: PLC Input/Output Configuration 58

CHAPTER ONE

INTRODUCTION

1.1 Background

Material handling contributes a significant amount to the cost and to the time of manufacturing a product. While material handling is essential to provide products to customers, it does not directly effect the product. Since material handling does not add value to the product, various methods have been designed to reduce the effect material handling has on these factors. The use of an Automated Storage and Retrieval System (AS/RS) can reduce material handling cost and production time.

The benefits of using an AS/RS are increased throughput, reduced work-in-progress, greater flexibility, and lower labor costs. As a result the manufacturing cost decreases, the inventory and the inventory cost decreases, the production lead-time decreases, and the efficiency of the process increases. In addition, an AS/RS is capable of being used by itself, such as a warehouse, and of being integrated into the manufacturing process for the storage of work in process inventories.

The Robotics and Automation Laboratory at Virginia Polytechnic Institute and State University (Virginia Tech) provides both research and teaching facilities in various areas of manufacturing automation. A Flexible Machining and Assembly System (FMAS) was designed and installed in the laboratory to emulate an industrial production process. In order to create a more realistic industry environment, an AS/RS was developed to support the FMAS. The AS/RS performs storage functions -- raw material, work-in-process (WIP), and finished product -- and delivery functions -- raw materials and WIP. The integration of the AS/RS into the

existing FMAS provides a tool for research activities in industrial automation and systems integration. Such a facility permits experimentation with the individual components of an integrated manufacturing system as well as with the control of the system itself.

1.2 Description of the FMAS

The FMAS consists of two workcells -- the kitting/assembly cell and the machining cell. The kitting portion of the kitting/assembly cell loads the required raw wax pieces onto a pallet. The raw wax pieces are machined into individual parts at the machining cell. At assembly, machined parts are assembled into a finished product.

The FMAS contains two DYNA CNC milling machines, two IBM robots, an assembly table, and a Shuttleworth Slip Torque conveyor system. Four personal computers and a Texas Instruments 565 Programmable Logic Controller (PLC) are used to control the system. The FMAS layout is depicted in Figure 1.1. A detailed account of the FMAS is presented by Eaglesham. (4)

The operation of the FMAS is captured by following the flow of parts through the system. The process begins when the AS/RS takes an empty pallet from the storage rack to the FMAS system at the AS/RS conveyor interface. The pallet is then transferred to the main conveyor which delivers it to the kitting/assembly cell. In this cell, the IBM 7547 robot places wax parts onto the pallet to build a kit. The pallet is then transported via conveyors to the machining cell where an IBM 7545

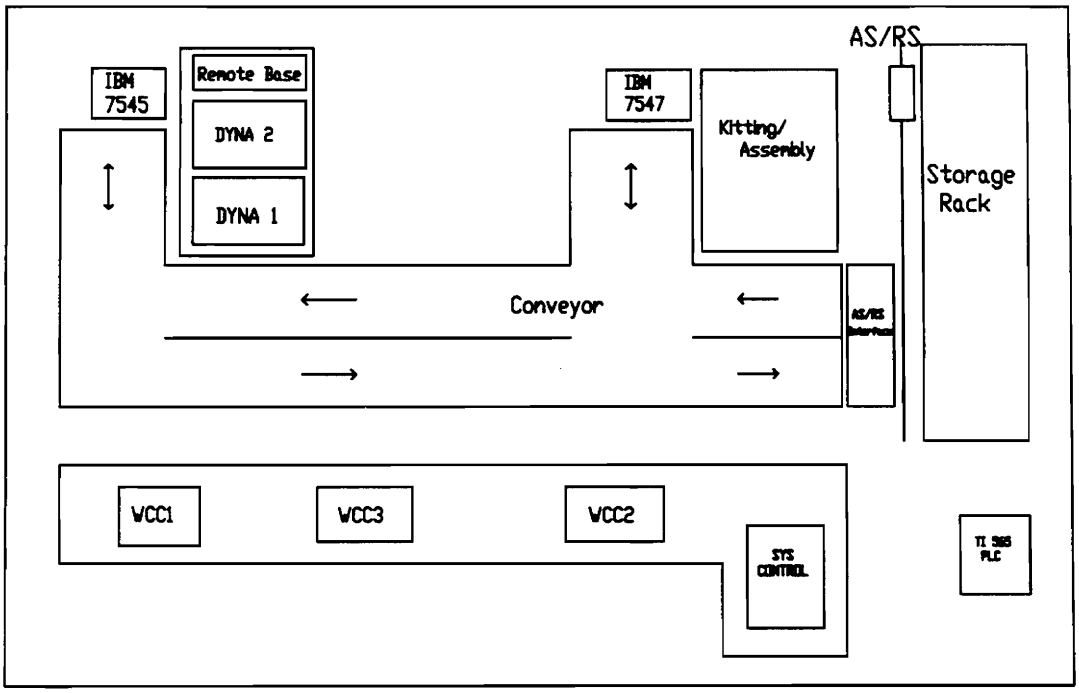


Figure 1.1: Schematic of the Virginia Tech FMAS

robot loads the wax parts onto DYNA CNC milling machines for machining. Once machined the parts are removed via the IBM 7545 robot and returned to the pallet. The pallet then travels via the conveyors back to the kitting/assembly cell where the IBM 7547 robot unloads the material from the pallet, assembles the parts, and reloads the finished product onto the pallet. The conveyor then transports the pallet to the AS/RS conveyor interface where the AS/RS crane takes the pallet to the storage rack.

The FMAS has the capability of handling raw material, WIP, and finished product. Raw material within the FMAS exists in two forms -- stock material and kits to be machined. If the kitting/assembly cell runs out of raw material, new stock material is loaded into the system either manually or automatically on a pallet. The IBM 7547 robot at the kitting/assembly cell unloads the parts from the pallet into the part feeders in the cell. Once a kit has been created, the parts are considered WIP. If a cell is down, the FMAS sends the WIP to storage to be processed later. Finished product is sent to storage for future shipping.

1.3 History of AS/RS

The design of the FMAS initially included an AS/RS; however, the AS/RS crane was not in the original design. When the FMAS was installed, it included the machining and assembly system with the main conveyor and a storage rack. The pallets designs included considerations for use by the FMAS and the AS/RS upon its completion. The storage rack, the FMAS, and the pallet design existed prior to the design of the AS/RS crane. The AS/RS system was designed to satisfy certain requirements. These requirements, among others, include the following:

- A conveyor interface from the AS/RS to the FMAS was planned to include two parallel conveyors for pickup and delivery by the AS/RS crane.

- The AS/RS crane would serve to carry pallets between the AS/RS interface and the storage rack with a maximum storage/retrieval time of 1 minute, a maximum unit load size of 20 pounds, and a throughput rate of 32 transactions per hour using single command control.

The design and fabrication of the AS/RS crane was completed in December 1995. A description of the AS/RS system follows. A more detailed description of the AS/RS hardware and its design is provided by Mark Eaglesham. (4) The proposal for the AS/RS consists of multiple steps in a continuous process. These steps include, but are not limited to, the design and construction of the crane, the development of software primitives, and the integration of the software system.

1.3.1 Storage Rack and Pallet

The storage rack has 14 columns and 8 serviceable rows. Each storage location is 10.5 inches wide and 12 inches deep. The two lowest rows have a height of 12 inches, while the other six rows are 6 inches high. The lowest row (first level) is 2 feet 1 inch from the floor. Therefore, the storage rack consists of 126 storage locations -- 98 small storage locations and 28 large storage locations.

The base of the pallet is 12 inches long, 10 inches wide and 1/4 inch thick. On each end of the pallet is a 3.5 inch high aluminum plate with an overhanging lip. The AS/RS crane uses this overhanging lip to pull the pallet onto the AS/RS. The pallets are currently being redesigned for strengthening purposes. However, the end plates on the new pallets will have the same shape as that of the current pallet.

1.3.2 Control System

A Texas Instrument 565 Programmable Logic Controller is provided to control

the Automated Storage and Retrieval System crane. The design and installation of the AS/RS crane included connecting the TI 565 input and output modules to the AS/RS. Control wiring schematics were prepared by Mark Eaglesham. (4) The PLC is programmed via an AT&T 386 personal computer.

The control of the FMAS and the AS/RS is a hierarchical system. The system controller directs the workcell controllers – WCC1, WCC2, and WCC3. The workcell controllers are personal computers which monitor the FMAS and the AS/RS's PLCs. Each PLC subsequently controls its respective system through input/output ports.

At the device level, the physical control system software consists of relay ladder logic, the language of the PLC. The control software will incorporate both the operation of the AS/RS crane itself and the interface between the AS/RS and the FMAS. The operation of the AS/RS includes storage and retrieval of pallets. The interface between the AS/RS and the FMAS consists of passing pallets to and from the AS/RS conveyor interface section.

1.3.3 AS/RS Crane

The horizontal travel of the AS/RS crane is guided by two steel rails -- one attached to floor and another located near the top of the storage rack. The main carriage is driven with a 1/2 horsepower, three phase AC gearmotor. A Toshiba inverter provides speed control of the three phase AC gearmotor. Optical sensors use markers on a bar located near the top of the storage rack as reference for horizontal positioning.

A T-section steel beam mast guides the vertical motion of the AS/RS crane. A 1/2 horsepower, three phase gearmotor is used to power travel along this vertical

mast. The Toshiba inverter, which controls the speed of the horizontal drive, also controls the speed of the vertical drive. Only one drive controller was used since the cost of the Toshiba inverter is greater than 50% of the total cost of the AS/RS. Optical sensors detect markers on a column next to the mast for vertical positioning.

The Toshiba inverter can be operate using either the control panel or discrete inputs from a remote controller. The Toshiba inverter is connected to a Texas Instruments 565 PLC to provide remote control of the horizontal and vertical drives for the AS/RS. There are four basic inputs needed to drive a motor -- interlock (ST), forward (F), reverse (R), and speed control signals (SS1, SS2, SS3). The interlock input operates as an on/off switch for the inverter and must be "on" to drive the motor. The forward and reverse inputs are use to actually drive the gearmotor. In the horizontal direction, forward motion of the crane is to the right, and reverse motion of the crane is to the left. In the vertical direction, forward is in the up direction and reverse is down. One of these two direction inputs must be on to obtain motion of the crane. Depending on the input state of these speed control signals, seven speed frequencies are possible. The eighth combination of SS1, SS2, and SS3 represents an off state. The seven speed frequencies are set manually through the Toshiba's control panel.

Additional connections on the Toshiba inverter are available. The additional inputs and outputs to the TI 565 are reset (RST), fault relay (FLA), acceleration/deceleration selection 2 (AD2). Reset is an input which allows the user to ramp the load to a stop at the set deceleration rate. Reset is also used at the control panel to reset an error on the inverter. Reset does not alter the preset values. The fault relay is an output signal which indicates an error in the Toshiba inverter. The Toshiba inverter ramps up and down using the acceleration (ACC1) and deceleration (DEC1) rates set by the user unless otherwise indicated. Acceleration/deceleration selection 2 is connected for use if another acceleration

and/or deceleration rate is desired. These values are set through the control panel. However, AD2 can be turned "on" and "off" by the TI 565 controller.

Table 1.1: Toshiba/TI 565 Interconnections

PLC I/O	DESCRIPTION	FUNCTION
X155	TOSHIBA INVERTER: FLA	fault on Toshiba inverter
Y30	TOSHIBA INVERTER: FWD	forward direction control (not when REV on)
Y31	TOSHIBA INVERTER: REV	reverse direction control (not when FWD on)
Y32	TOSHIBA INVERTER: SS1	speed selection one (to be set in Toshiba)
Y33	TOSHIBA INVERTER: SS2	speed selection two (to be set in Toshiba)
Y34	TOSHIBA INVERTER: SS3	speed selection three (to be set in Toshiba)
Y54	TOSHIBA INVERTER: RST	reset
Y55	TOSHIBA INVERTER: AD2	2nd set of acceleration and deceleration
Y56	TOSHIBA INVERTER: ST	interlock
Y61	SSR Y: SELECT VERTICAL MOTION	turn on to move crane vertically (not while horiz on)
Y62	SSR X: SELECT HORIZ MOTION	turn on to move crane horizontally (not while vert on)

The shuttle mechanism is the hardware which loads and unloads to and from the storage rack and the AS/RS conveyor interface. The shuttle mechanism moves perpendicular to the storage rack. To facilitate loading and unloading from both the storage rack and the AS/RS conveyor interface, the shuttle mechanism shifts from side to side using four pneumatic cylinders which are controlled by four solenoid valves. Actuating valves 1 and 4 causes the shuttle to shift towards the FMAS; actuating valves 2 and 3 shifts the shuttle towards the storage rack. The "home" position of the shuttle is considered to be the center. This is obtained by actuating either valves 1 and 3 or valves 2 and 4. Clearance of the storage rack and of the conveyor on either side of the shuttle occurs only when the shuttle is in the home position. To prevent vertical or horizontal travel of the crane while the shuttle is not in the center position, a microswitch is wired into the power circuit to serve as an interlock. This microswitch is not connected to the PLC.

"Push-pull fingers" on the shuttle mechanism pull a pallet onto the shuttle and push a pallet off the shuttle. These fingers are chain driven using a stepper motor. A set of short fingers and a set of long fingers travel in opposite directions on each edge of the shuttle. While the short fingers can only be used for pushing a pallet, the long fingers can push or pull a pallet. Two optical sensors mounted (at either end) on the shuttle establish the positioning of the fingers. These sensors are connected as inputs to the TI 565.

The stepper motor is driven by a stepper card and a pulse generator. Two inputs to the stepper card are required to drive the stepper motor. One input is an on/off switch. The other indicates the direction of the motor. If the direction input is active, the motor pushes a pallet into the storage location. If the direction input is not active, the motor pulls the pallet from the storage location. Rather than generate the pulses needed to operate the motor through the PLC, a pulse generator is connected to the stepper circuit and the PLC output turns the pulse stream on and off. The speed of the motor can be adjusted by altering the frequency.

Table 1.2: Crane & Shuttle/TI 565 Interconnections

PLC I/O	DESCRIPTION	FUNCTION
X118	X HOME LIMIT SWITCH	limit switch for horizontal home position
X119	Y HOME LIMIT SWITCH	limit switch for vertical home position
X120	Y VERT POSITION/OPT SWITCH	optical sensor for sensing row bars
X121	Z1 SHUTTLE POSITION/OPT SWITCH	optical sensor to sense push-pull fingers (by FMAS)
X122	Z2 SHUTTLE POSITION/OPT SWITCH	optical sensor to sense push-pull fingers (by rack)
X123	X HORIZ POSITION/OPT SWITCH	optical sensor for sensing column bars
X150	X LIMIT SWITCH - 1	extreme horizontal limit on left of rack
X151	X LIMIT SWITCH - 2	extreme horizontal limit on right of rack
X152	Y LIMIT SWITCH - 1	extreme vertical limit at bottom of rack
X153	Y LIMIT SWITCH - 2	extreme vertical limit at top of rack
Y36	CW STEPPER PULSE	stepper motor on/off
Y37	CCW STEPPER PULSE	stepper motor direction (0 -- CW, 1 -- CCW)
Y57	V1 EXTEND LEFT	extend valve one
Y58	V2 RETRACT LEFT	retract valve two
Y59	V3 EXTEND RIGHT	extend valve three
Y60	V4 RETRACT RIGHT	retract valve four

CHAPTER TWO

PROBLEM STATEMENT AND OBJECTIVES

2.1 Problem Statement

The Automated Storage and Retrieval System (AS/RS) was developed to supply the Flexible Machining and Assembly System (FMAS). The AS/RS provides storage of raw materials and finished product, delivers raw materials to the FMAS, and removes finished product from the FMAS. The first phase of the AS/RS development was the design and installation of the AS/RS crane. While consideration was given to how the crane would be controlled, the development of the control system software was not part of this phase. A crane and a shuttle mechanism were designed to interface with a conveyor section of the FMAS.

The objective of this project was to develop a working control system of motion primitives for the Automated Storage and Retrieval System. The control system was to include a Relay Ladder Logic program to control the motion of the AS/RS and a C program to simulate the functions of the storage system workcell controller (WCC3). When the AS/RS control system is fully integrated with the FMAS control system, a workcell controller (WCC3) will control the inputs and monitor the outputs of the AS/RS. Until that time, another means was needed to simulate the functions of the WCC3.

2.1.1 Develop Specifications and Relay Ladder Logic

In order to perform the storage and retrieval primitives, relay ladder logic was developed using TISOFT. The system which controls the FMAS consists of a

network of computers. Workcell controller 3, which is connected to a TI 565 PLC, controls the material handling and storage functions. The relay ladder logic developed for the AS/RS will be incorporated with the current relay ladder logic for the FMAS. While the two programs will function together, they were developed as if the systems were separate. A list of relevant inputs and outputs for the Workcell Controller and the AS/RS is in Appendix A.

2.1.2 C Program and Testing Relay Ladder Logic

In order to test the operation of the AS/RS primitives, a test program was developed using Microsoft C and tested. The Microsoft C program runs on one of the four FMAS computer and interacts with the TI 565 PLC. This program allows the user to simulate the control functions of the WCC3. The user chooses an option to perform a store or a retrieve function to a storage location specified by the user. This test program was designed but not implemented due to a lack of adequate documentation on the digital I/O board.

2.1.3 Documentation and Comparison

Documentation of the control system software for the AS/RS and its operation is provided for future users. The performance of the Virginia Tech Industrial and Systems Engineering Department AS/RS was compared with the model created by Bozer and White. This demonstrates the effect that not moving horizontally and vertically simultaneously has on the performance of the AS/RS.

CHAPTER THREE

LITERATURE REVIEW

3.1 General Automated Storage and Retrieval Systems

The *Materials Handling Handbook* defines an Automated Storage and Retrieval System as "a combination of equipment and controls which handles, stores, and retrieves materials with precision accuracy and speed under a defined degree of automation." (10) The system's control varies from manual to complete computer control. The principle components of an AS/RS are storage load, storage structures, stacker crane, pickup and delivery (P&D) stations, and controls.

Originally AS/RS systems were designed for the function of "eliminating the walking that accounted for 70 percent of manual retrieval time." (1) AS/RS systems have been developed to serve more than this original purpose. Because of the additional advantages received from using an AS/RS, such systems are widely used throughout industry both in stand-alone systems, warehouses, and in conjunction with manufacturing equipment. Among the added benefits of a current AS/RS are higher reliability, increased flexibility, faster retrieval, concentrated storage, lower inventory, and increased productivity. For example, Von Duprin Inc. "cut inventory levels 65% ..... reduced assembly cycle times and increased productivity 57%. Inventory accuracy better than 99.99% contributes to manufacturing efficiency." (6)

The College-Industry Council on Material Handling Education created "The 24 Principles of Material Handling" to aid in the design of material handing systems. The integration of the material handling system designed within the current system is considered in these principles -- "Integrated System Principle: Integrate those handling and storage activities which are economically viable into a coordinated system of manufacturing operations . . . The systems principle is especially

important when installing automated handling systems. Without complete integration of automated systems, the full benefits of an automated process may not be realized." (3)

An important portion of system integration includes the flexibility within the system. Without the flexibility to expand/reduce, to improve, and to update a system, the process becomes obsolete and useless. According to consultant Neil Glenney "flexibility means the ability to serve more than one process -- even if the additional processes have not yet been installed." (9) Flexibility ought be built into every system utilized. Adding flexibility to a system after its design and construction can be difficult and expensive. One of the dominant reasons to employ an AS/RS system is increased flexibility. With the flexibility available from an AS/RS, integration of new processes into a system is easier.

3.2 AS/RS Control Systems

When employing an AS/RS within a manufacturing process, the type of control system the AS/RS uses affects the operation of the entire process. Both mini and microcomputers as well as Programmable Logic Controllers (PLC) are currently used. The National Electrical Manufacturers Association defines a programmable logic controller as: "A digitally operating electronic apparatus which uses a programmable memory for the internal storage of instructions for implementing specific functions such as logic sequencing, timing, counting, and arithmetic to control, through digital or analog input/output modules, various types of machines or processes." (5)

PLC's are used with personal computers as control systems for AS/RS's. While personal computers are typically used to manage the AS/RS, PLC's are

employed to perform basic hard drive functions. The personal computer monitors the functions required and the placement of pallets within the storage system. The PLC drives the inputs and the outputs for motion of the AS/RS.

3.3 AS/RS Control Design

Considerations when creating the control system of an AS/RS usually include sequencing and order-picking. Three types of product are produced by the FMAS. However, the sequence of products delivered by the AS/RS was not a concern in this project. Neither is order-picking an issue. The WCC3, not the AS/RS, determines what function to perform and sends the command to the PLC. The AS/RS performs the required function without consideration of why the function is needed.

One of the major factors in controlling an AS/RS is the machine cycle command. Two types of machine cycle commands exist -- single command and dual command. These are defined by Han, McGinnis, Shieh, and White: (8)

single command (SC):

> single storage or retrieval is performed; storage cycle time is equal to the sum of times for picking up the load at the I/O station at the end of the aisle, traveling to the storage location, depositing the load in the rack, and returning to the I/O station; retrieval cycle time is similar, and

dual command (DC):

> both a storage and a retrieval are performed in a cycle; cycle time is the sum of pick-up time plus travel time to storage location plus unload time plus travel time to retrieval point plus load time plus return time to I/O station plus unload time

Single command will be used for the AS/RS in this project. The FMAS software considers only two pallets at a time in the system. This condition was employed for the ease of use by students. The need for a storage or a delivery operation will be anticipated. Using a single command system the AS/RS can move to the appropriate position for the anticipated storage or retrieval. By anticipating the operation, the cycle time can be reduced.

3.4 Measurement Parameters

In order to evaluate the AS/RS certain parameters need to be defined.

Dimensions:

L = length of rack

H = height of rack

N = total number of locations in rack

Travel Times:

t_h = time to travel horizontally from I/O station to farthest column = L/s_h

t_v = time to travel vertically from I/O station to farthest level = H/s_v

Travel Speeds:

s_h = speed in horizontal direction

s_v = speed in vertical direction

Shape Factor (b):

$b = \min (t_h/T, t_v/T) \qquad 0 \leq b \leq 1$

where $T = \max (t_h, t_v)$

3.5 Travel Time Model

The travel time for a storage or retrieval function is dependent on the vehicle velocity and on the length and the height of the rack. Bozer and White created a general mathematical model to determine the travel time of an AS/RS. (2) Several assumptions were made about the AS/RS system for this model:

- the I/O point is at the lower left-hand corner
- the S/R machine has either single or dual command operations
- the rack length and height are known
- the S/R machine travels in the horizontal and vertical direction simultaneously
- randomized storage
- the time to pickup and deposit a load are negligible

Bozer and White found the expected travel time for single command operations [E(SC)] and the expected travel time for dual command operations [E(DC)] to be:

$$E(SC) = 1/3(b^2) + 1$$

$$E(DC) = 4/3 + 1/2(b^2) - 1/30(b^3)$$

However, these equations do not apply to the AS/RS in the Robotics and Automation Laboratory at Virginia Tech. The S/R mechanism in this system does not move in a horizontal and vertical direction simultaneously. In order to determine the travel time of the AS/RS in the Robotics and Automation Laboratory, these equations will need to adapted to accurately describe the system at hand.

3.6 Trends in AS/RS

"Why?" is a question every company considers when new technology becomes available. Technology should not be used just to maintain "cutting-edge" credibility within industry. An AS/RS is not an exception to this philosophy. An AS/RS may not be applicable to the situation. However, those companies which use AS/RS's are finding more benefits and greater possibilities from the AS/RS. The latest trends among many companies are the following:

- increased system integration capability
- buffers
- modular design AS/RS

Within manufacturing processes an AS/RS is rarely used alone. AS/RS's are usually used in conjunction with other aspects of the process -- manual or automated. The AS/RS needs to be compatible with the other existing equipment. The design must account for interaction with any material handling equipment, such as a conveyor, or machines, such as numerical control milling machines, which also exists in the process. This integration is important both from a physical and a controls point of view. In order to provide a "more uniform and less costly integration of systems" open communications protocols are being developed and used. (9) Another trend within AS/RS integration is towards electronic data interchange standards.

AS/RS are no longer being used only for inventory storage. More companies are utilizing AS/RS's for "buffers" instead of storage. Throughput of the AS/RS and interconnecting equipment is of a larger concern than the storage capacity of the AS/RS. The main reason for using an AS/RS as a buffer is the increased flexibility, higher reliability, and the reduced response time of the entire process as a whole. (9)

A newer type of AS/RS is the modular AS/RS. It is considered to be a hybrid between vertical carousels and miniloads. (7) A S/R machine in a fixed position rotates to store and retrieve material with a movable insert/extract mechanism.

Although these three trends in the past five years have a large influence on the design of an AS/RS, they are irrelevant to the project at this time. The AS/RS in the Robotics and Automation Laboratory is already in existence. The hardware design is complete. As a result using a modular design is not possible. Also the function of the AS/RS is to operate as a buffer for the FMAS.

The only problem might arise from the system integration for the control systems between the AS/RS and the FMAS. However, this possibility has already been anticipated and eliminated. The workcell controller which controls the FMAS will also control the AS/RS. The AS/RS control hardware was designed to be compatible with the control hardware of the FMAS. The only system integration concern is in the software. One of the goals of this project is to develop a control system for the AS/RS which interacts with the FMAS.

CHAPTER FOUR

CONTROL SYSTEM DESIGN

4.1 Criteria

The objective of the project was to design a set of primitives for the motion of the AS/RS. In order to achieve this objective, certain requirements and criteria need to be met. Since the project is to control only the basic motions of the AS/RS, the control system must be flexible to allow adaptability for future users. Students will build on the basic motion elements developed in this project to meet additional needs of the laboratory and the FMAS.

AS/RS's are desirable within a factory since they are operable continuously for long periods of time. This AS/RS is no different; the hardware was designed to handle continuous operation for periods of time. As a result the system needs to be reliable. The AS/RS needs to execute the proper function every time a function is requested. Repeatability is a product of this. The AS/RS is required to be able to perform the same operation over and over both to the same storage location and to different storage locations.

Additional criteria include the set of functions required by an AS/RS. These basic functions were determined to be initialization, store, retrieve, move to location, and shutdown. These criteria are summarized in the following:

- Flexible for future adaptations
- Reliable
- Repeatability of operations
- Set of basic functions

4.2 Constraints

When designing any system, restrictions and constraints on the system must be identified. The constraints on the design of the control system for the AS/RS are primarily due to the physical restrictions of the AS/RS itself. These constraints fall into two major categories -- the storage rack and the motion of the crane.

4.2.1 Storage Rack

To stabilize the storage rack a brace bar is positioned horizontally along the back of the storage rack. Due to the overhanging lip on the pallet, a pallet cannot be fully positioned within a storage location in this row. The overhanging lip of the pallet hits the brace. Therefore, all storage locations within row 4 are unusable. This brace will be repositioned when the system is moved to the new Virginia Tech engineering building. Once the brace is moved, the comparison of desired row to row 4 and resulting error message should be removed from the AS/RS's relay ladder logic.

An additional column and three additional rows cannot be used because of the design of the crane and the shuttle mechanism. Column 0 on the storage rack is located next to the laboratory wall. The crane's vertical steel beam is situated on the left side of the shuttle mechanism. If one attempts to position the shuttle mechanism at column 0, the crane will collide with the laboratory wall. When the system is moved to the new building, the storage rack will be adjusted to the right to allow use of this column. In order to differentiate between column 0 and special functions using column number 0, the storage rack should be renumbered after the move to start at column 1 not column 0. This will change the last stopping position

on the horizontal from 15 to 16. Using four bits for the horizontal address restricts the highest value to be 15. Therefore, at least one of the conveyor column locations should be lined with a storage rack column to reduce the column count to a maximum of 15.

Attempting to position the shuttle mechanism at row 0 is also not possible due to the design of the system. The motor at the bottom of the crane causes a problem in placing the shuttle mechanism low enough to reach row 0. The shuttle mechanism will run into the motor before arriving at row 0. This problem will not be corrected.

Due to the counterweight on the crane, rows 8 and 9 cannot be accessed. In order to reach those two rows, the counterweight lowers to the point at which it collides with the vertical home limit switch. The problem of accessing rows 8 and 9 can be resolved. At that time the comparison of the desired row to rows 8 and 9 and the resultant error in the ladder logic should be removed. Redesign of the counterweight and/or the limit switches would give access to the top two rows. One possible solution is to lower the limit switches and attach a bar to the bottom of the shuttle mechanism. Position this bar to trigger the limit switch when the shuttle mechanism has reached the minimum level.

Stopping locations are needed to access the AS/RS interface conveyors. As a result two pseudo columns and one pseudo row are used. The two columns function as locating points for the delivery conveyor (column 12) and the pickup conveyor (column 14). Row 1 is a pseudo row used to position the shuttle at the correct elevation for the conveyor. These locations are not physical storage locations. As a result storing or retrieving a pallet from any location in column 12 and 14 or row 1 is not possible. Figure 4.1 shows a diagram of the storage rack. The columns and rows which are not usable are marked. Additionally, positive and

negative directions are identified. The numbers on this diagram are subject to change after the system has been moved to the new Virginia Tech engineering building.

4.2.2 Motion Control

Since the speed of vertical motion and the speed of the horizontal motion are controlled by the same inverter, it is not possible to move in a horizontal and a vertical motion simultaneously. Simultaneous motion requires two inverters. Unfortunately, traveling one direction at a time requires more time than simultaneous vertical and horizontal motion. In order to reduce travel time, simultaneous motion in vertical and horizontal directions may be added in a future project.

For safety reasons the motion in the horizontal direction occurs only when the shuttle mechanism is down. While the hardware is capable of horizontal travel with the shuttle up, the robustness of the hardware and of the software is unknown. Therefore, the crane is programmed to move only when the shuttle is down. For each horizontal motion desired, the shuttle mechanism moves down to a base location, travels horizontally to the appropriate column, and drives vertically to the appropriate row. This increases the travel time from one storage location above the first level to another storage location above the first level in a different column.

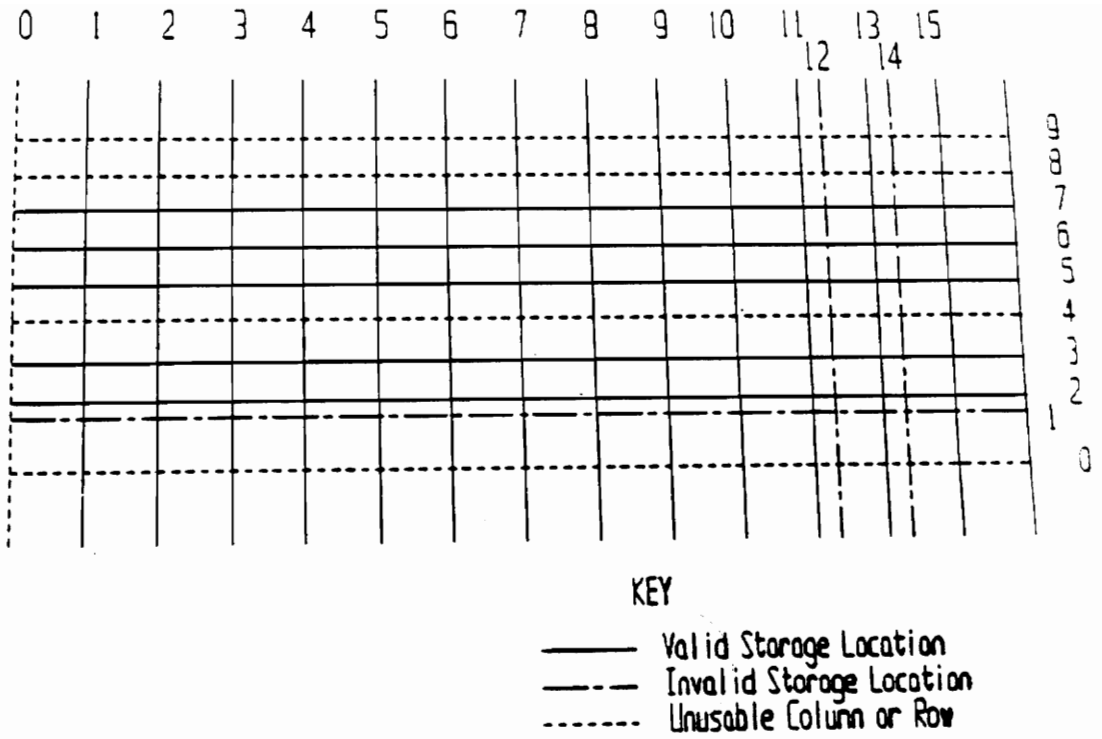


Figure 4.1: AS/RS Storage Rack

CHAPTER FIVE
CONTROL SYSTEM COMPUTER INTERFACE

5.1 Purpose

The Workcell Controller 3 (WCC3) controls the material handling aspects of the FMAS and the storage system through inputs and outputs connected to a TI 565 PLC. Based on the needs of the FMAS, the WCC3 directs the AS/RS to perform a function at a specified storage location. WCC3 uses the store/retrieve command input and the address bits, horizontal and vertical, to indicate which operation the AS/RS is to execute. Table 5.1 shows the AS/RS commands and their respective command input and address bits. The bits to indicate a location are signified by "xxxx". At this time the WCC3 does not control the AS/RS. Therefore, another means of control is required in order to test the operation of the AS/RS.

Table 5.1: AS/RS Commands

Store/Retrieve Command X103	Horizontal Address X104 - X107	Vertical Address X108 - X111	Operation
0	xxxx	xxxx	Store
1	xxxx	xxxx	Retrieve
0	0000	xxxx	Send to Column (xxxx)
1	0000	xxxx	Send to Row (xxxx)
0	0000	0000	Initialize
1	0000	0000	Shutdown
0	xxxx	0000	Invalid
1	xxxx	0000	Invalid

In a normal sequence of events the AS/RS crane is idle when it reads a command from WCC3. The AS/RS alternates between crane motion and shuttle motion while executing the operation. Once the operation is completed, the AS/RS returns to an idle state and signals WCC3 for another operation. If an error occurs during this sequence, all motion of the AS/RS is stopped. The error must be reset by the AS/RS's user. The way to fix the error depends on the error which occurs. The AS/RS then continues to execute the operation when the error is cleared. Section 5.3 further discusses this communication between the workcell controller and the AS/RS.

The services the workcell controller provides for the AS/RS include signaling the beginning of an operation, indicating the function required, and determining the storage location to be used. What is needed is a system which can emulate the workcell controller. This system must be able to perform the same functions of the workcell controller and to communicate with the AS/RS control system. Additionally, the system has to be flexible and easy to use. This system will also be utilized with the AS/RS as a teaching tools in classes at Virginia Tech. A C program -- using QuickC for Windows -- was developed for the purpose of performing these services in place of the workcell controller.

The workcell controller is a personal computer which uses a digital input/output (I/O) board to communicate with the PLC's remote base location. The digital I/O board is used to turn on the AS/RS's inputs and to monitor the AS/RS's outputs through the PLC's remote base location. However, this program does not work because sufficient documentation of the digital input/output board was not available. This documentation is expected to be located; therefore, another board was not purchased. Instead a program was developed which simulates the communication which would have occurred otherwise. These routines provide a base for further development of the WCC3's control program.

5.2 C Test Program

Upon executing the program, the "Main Menu" is presented. The Main Menu provides a means of performing each task available to the AS/RS -- Initialization, Store, Retrieve, Move to Column, Move to Row, and Shutdown -- and exiting the program. The Main Menu is shown in Figure 5.1. From each option the user is then prompted to enter any necessary data, such as storage location. This program does not receive status code bits as the actual workcell controller would since there is no means of producing an error without the physical system. The C program is available in Appendix B.

AS/RS Operation Menu

1. Initialization
2. Store Function
3. Retrieve Function
4. Move to Column
5. Move to Row
6. Shutdown
7. Quit

Enter your choice:

Figure 5.1: Main Menu

void command()

The command() subroutine displays on the screen for the user the store/retrieve command input, the horizontal bits, and the vertical bits sent to the AS/RS. It also shows the strobe bit being sent to the AS/RS. This function simulates the initial communication between the workcell controller and the AS/RS

control system -- store/retrieve command, horizontal and vertical address bits, and strobe.

void h_address()
void v_address()

The h_address() subroutine enables the user to input the storage location's horizontal address. The v_address() subroutine enables the user to input the storage location's vertical address. Both subroutines check the validity of the address inputted. If the address inputted is not within the available columns or rows, an "Invalid address" message is displayed and the user is given another opportunity to input the address.

void binary()

The binary() subroutine converts the inputted numerical horizontal and vertical addresses to four-bit binary numbers. These four-bit binary numbers are then displayed to the screen in the command() subroutine.

void move_to_col()
void move_to_row()

Columns and rows which are not available for storage locations may be possible stopping locations for the AS/RS. Therefore, these values are valid for motion of the AS/RS without performing a store or a retrieve operation. The move_to_col() subroutine reads the desired column inputted by the user and checks its validity. The move_to_row() subroutine reads the desired row inputted by the user and checks its validity.

void acknowledge()

The acknowledge() subroutine simulates the final portion of the communication between the workcell controller and the AS/RS control system. The user is requested to "Press ENTER" to simulate the AS/RS sending an acknowledge to the workcell controller. The strobe is turned off and the AS/RS is busy. The user is then prompted to "Press ENTER" to signify the end of the operation and to return to the menu.

int initialize()
int shutdown()

The initialization() subroutine checks to ensure the user wants to perform an initialization. The shutdown() subroutine checks to ensure the user wants to perform a shutdown operation. If the user does want to perform the initialization or the shutdown, the program continues. If not, the program returns to the main menu.

5.3 Workcell Controller and AS/RS Communication

In order to perform each operation, the AS/RS control system requires certain tasks and inputs. These tasks and inputs are rendered by the workcell controller, or in this case the C program and the user. The initial communication between the workcell controller and the AS/RS control system indicates the function the be performed by the AS/RS. These tasks and inputs consist of the following:

- Send "Store/Retrieve Command" input
- Transmit horizontal and vertical addresses
- Output "Command Strobe" (X102)

- Receive "Acknowledge" from AS/RS (Y10)
- Accept "Busy" signal from AS/RS (Y8)
- Receive "Done" output from AS/RS (Y9)

To signify the need of a new operation, the workcell controller transmits a "strobe" to the AS/RS. Once the AS/RS has received the strobe, it expects to receive a store/retrieve command and the horizontal and the vertical addresses of the storage location. It is recommended that the store/retrieve command input and the address bits are initiated before sending the strobe. Otherwise the AS/RS may read the input bits before WCC3 has initialized the bits. Any bits that are not read in time will be accepted either as with a status of zero or with the status of the bit from the previous operation.

After reading the horizontal and the vertical addresses from the workcell controller, the AS/RS sends an "acknowledge" signal to the workcell controller. This ensures that the AS/RS has read the command and the storage location address. The AS/RS then proceeds to perform the desired operation. Upon completion of the operation, the AS/RS sets the "done" signal to on. This indicates that the AS/RS is ready to perform another operation. The AS/RS only expects and accepts a new command when it is not busy.

Figure 5.2 is an example -- of a store function -- of the output which simulates the workcell controller. Each bit displayed on the screen is symbolic of the bit set to/read from the AS/RS/workcell controller. The store/retrieve command dependent on the option chosen by the user. The horizontal and vertical address bits are binary equivalents of the user's inputted address values.

```
Enter desired column (0-11,13,15): 8

Enter desired row (0-3,5-8): 6


 Command -- X103 = 0


   Horizontal Address
 X107  X106  X105  X104
  1     0     0     0

   Vertical Address
 X111  X110  X109  X108
  0     1     1     0


 Strobe (X102) on.

Press ENTER to acknowledge command (Y10).

 Strobe (X102) off.

 AS/RS is busy (Y8).

 Operation is done (Y9).


Press ENTER to return to menu.
 End of operation.
```

Figure 5.2: C Program Output Example

CHAPTER SIX

RELAY LADDER LOGIC

6.1 General

A relay ladder logic program looks, just as the name implies, like a ladder. The rungs on the ladder consist of contacts and coils. The condition -- active or inactive -- of the contact(s) in a rung drive the coil(s). Contacts are inputs or outputs which can be either internal or external. Coils are outputs which are either internal or external. Internal contacts and coils are used to maintain the status of the system. Internal contacts and coils do not activate any physical output. External contacts and coils represent the actual output devices which are connected to the PLC.

Most systems require the execution of mathematical operations within the control system. Relay ladder logic allows for these requirements. Functions such as addition, subtraction, counting, and bit manipulation are available within a relay ladder logic program. The program developed in this research was implemented using the Texas Instruments programming system – TISOFT. A computer specification of the programming system and language requirements can be found in reference 12.

The execution of a PLC program is vastly different from the execution of a language program, such as C++. As each rung of the ladder is read, the status of the output coils are changed. However, the ladder is scanned in its entirety before any outputs are activated. The ladder is then repeatedly scanned multiple times each second. The scan time of a ladder can be either constant or variable. Constant scan times are set by the PLC user. With variable scan time, the time

required to scan the entire ladder is dependent on several variables, such as the length of the ladder and the complexity of the ladder.

Some PLC's and their software have the capability of creating and calling subroutines within the ladder. The ladder contains the main program at the top of the ladder. The main program concludes with an END coil. All subroutines are positioned in the ladder after the END statement. When a subroutine is called within a ladder, the subroutine is performed once. While each subroutine is read each ladder scan, the outputs are activated only when the subroutine has been called. After the subroutine has been performed, the next rung on the ladder is performed.

In TISOFT, for the TI 565 PLC, subroutines are called in the following manner:

- The subroutine conditions are defined using a logical combination of contacts under which the subroutine should be performed.
- The rung ends with a GTSxxx (Go To Subroutine xxx) coil.

The subroutine itself consists of the following:

- The subroutine begins with a rung consisting of a SBRxxx (Subroutine xxx) coil.
- The following rungs contain the designed control logic.
- Each subroutine ends with a rung consisting of a RTN (Return) coil.

In order to perform the desired operations, a subroutine sometime requires more time than one scan to perform its function. For example, driving a motor to move the AS/RS crane from one point to another necessitates more than the scan's fraction of a second. Therefore, a Do-While loop needs to be simulated within the ladder logic by placing a conditional end rung immediately following the subroutine call rung. If the desired end result has not occurred, for example if the motor has

not drive the AS/RS crane to the desired location, terminate the scan before progressing further. The next scan of the ladder will include a scan of the subroutine. Once the subroutine's end result is achieved, the conditional end will not terminate the scan and the ladder can continue being processed.

6.2 AS/RS Ladder

The ladder developed for the AS/RS uses subroutines. Subroutines were chosen for several reasons. When running the AS/RS, only one function is desired at a time. Subroutines are a simple and understandable means of executing one function at a time. The subroutine for that desired operation only is called. The relay ladder logic designed for the AS/RS control system is in Appendix C. Chapter 7 discusses the ladder in detail. Certain definitions and assumptions must be defined first.

6.2.1 Definitions

"Home" position:

The AS/RS moves to its "home" position when initialized. This home position is defined to be column 1 and row 0 -- (1,0). Physically the AS/RS crane is between rows 0 and 1 when at the vertical home position. The vertical home position is defined to be 0 to facilitate counting the storage rack rows. The AS/RS is at the home position when the X HOME limit switch and the Y HOME limit switch are both active.

Travel Level:

The travel level is the vertical position at which the AS/RS is located during horizontal travel. This row is defined to be row 0 even though the crane is physically

between rows 0 and 1. The AS/RS is at the travel level when the Y HOME limit switch is active.

Status Codes:

The status codes are utilized to send information to the workcell controller on the status of the AS/RS. Four bits are available for the status codes. Hence, sixteen codes are possible. Only twelve of the codes have been used. The other four are available for future development of the AS/RS control system. These codes and their meanings are listed in numerical order in Table 6.1.

These twelve codes notify the user and the workcell controller of the status of the AS/RS. Four of these codes signify the AS/RS's current position – "AS/RS Idle", "Crane In Motion", "Shuttle In Motion", and "At Home Position". "AS/RS Idle", "Crane In Motion", and "Shuttle In Motion" are the three normal conditions of the AS/RS. The "At Home Position" code is only at active when both the X HOME and the Y HOME limit switches are closed. The other codes indicate an error with the system. These codes will be employed to determine what has caused the AS/RS downtime. Each code which signifies an error will turn on one internal error signal (C999). Therefore, it will be easy to determine if an error has occurred.

Table 6.1: AS/RS Status Codes

CODE				DESCRIPTION
Y14	Y13	Y12	Y11	
0	0	0	0	AS/RS Idle
0	0	0	1	Toshiba Fault Error
0	0	1	0	Crane In Motion
0	0	1	1	Unused
0	1	0	0	Horizontal Motor Error
0	1	0	1	Unused
0	1	1	0	Shuttle In Motion
0	1	1	1	Unused
1	0	0	0	Invalid Address
1	0	0	1	General Error
1	0	1	0	At Home Position
1	0	1	1	Vertical Extreme Limit Error
1	1	0	0	Vertical Motor Error
1	1	0	1	Unused
1	1	1	0	Stepper Motor Error
1	1	1	1	Horizontal Extreme Limit Error

6.2.2 Assumptions

Due to some physical limitations of the system, the ladder created for this project required the assumption of certain conditions. During both a store and a retrieve operation, a pallet is loaded onto the shuttle mechanism either from the conveyor or from a storage location. There is no sensor on the AS/RS to determine if the pallet to be loaded is at the proper location. Therefore it is assumed that the pallet is present at the desired pickup point if a load function is performed. Similarly,

a pallet is unloaded from the shuttle mechanism onto the conveyor or onto a storage location during both a store and a retrieve operation. There is no a sensor to determine if a pallet already exists in the location in which the pallet is to be unloaded. Hence, the assumption is made that the location is empty.

The shuttle mechanism is assumed to have a pallet present or to be empty if the "push-pull" fingers have been detected by the appropriate fingers. The order of detection of the fingers to consider the pallet loaded or unloaded will be explained later in this paper. This assumption is required since a sensor to detect a pallet's presence does not exist on the AS/RS. The default setting is that a pallet is not present on the shuttle mechanism. This default may cause a problem if a pallet is present while power to the AS/RS and PLC is lost. However, until a sensor is available to detect the presence of a pallet, an initial assumption is needed.

The final assumption made is that only one error occurs at a time. This assumption is required due to the status codes utilized to indicate errors. Four bits are available to indicate the status of the AS/RS. If more than one error occurs, the status code given will not be an accurate representation of the state of the AS/RS.

The following is a summary of the assumptions used throughout the entire ladder:

- The pallet to be loaded onto the shuttle is present at the specified location.
- The location to which a pallet is being moved is empty.
- A complete operation of the "push-pull" fingers will result in the loading or unloading of a pallet.
- A pallet not present on the shuttle is the default setting.
- Only one error occurs at a time.

6.2.3 Global Variables

A complete listing of the variables associated with this program and their values is in Appendix D. The variables for recording the values of the current location of the AS/RS and for the location of the conveyor and the desired storage location are all in variable (V) memory. The following variables are the variables required throughout the entire ladder to maintain a status of the current location of the crane and the destination location for the crane:

V120: desired storage location column
V121: desired storage location row
V122: current location column
V123: current location row
V125: conveyor row (row 1)
V126: pickup conveyor column (column 14)
V127: delivery conveyor column (column 12)

6.2.4 Internal Coils

An internal coil is defined as "A program output that is used strictly for internal purposes (does not drive a field device). It provides interlocking functions like a hardwired control relay....Internal outputs also provide a solution to ladder diagram format restrictions imposed by the programming unit." (11) Coils used in the ladder are used both for maintaining status of an output and as means to solve TISOFT restrictions. Situations in which coils are utilized for the later reason include loading values into variable memory. When loading values into variable memory, an output is not activated. However, within TISOFT every rung must end with a coil. In order

to perform the rung without activating an output, an internal coil is activated to satisfy this restriction.

A complete listing of the ladder's internal coils and their definitions is in Table 6.2. The listing is separated into seven sections. Each section is a grouping of internal coils which either have similar uses or work in conjunction with each other. The first section of coils corresponds with beginning an operation. Coil C1 maintains whether the system has been initialized upon startup. The only operation which can be performed if Coil C1 is not active is an initialization operation. Coil C2 is an internal means of declaring a busy status. Coils C3, C4, C5, and C6 are used when determining the validity of an address and special operation requests, such as "Move to Column".

Due to the difference in the operation speed of the AS/RS motors and in the PLC scan time, a time delay is employed once a column or row bar has been counted. A count can only occur when the time delay is over. This allows time for the motor to drive the crane or the shuttle past the column or row bar. Without this time delay, the same column or row bar would be counted more than once. The second set of coils indicates when this time delay has ended.

The coils in the third section work in conjunction with each other to ascertain when a pallet has been loaded onto the shuttle mechanism or unloaded from the shuttle mechanism. Coils C21 and C22 indicate when the shuttle is in the correct positioning to load or unload a pallet. The "push-pull" fingers cannot turn on until the correct positioning is achieved. A complete cycle of the "push-pull" fingers past both optical sensors is required to ensure the pallet has been loaded onto the shuttle or unloaded from the shuttle. If Coil C26 is active, then the first optical sensor has detected the "push-pull" fingers. If a pallet is loaded onto the shuttle, Coil C20 is

active. The default setting of Coil C20 is not active – no part is present on the shuttle.

The last four sections of the coil list are used to determine if the current location of the AS/RS crane or shuttle is at the desired location column or row. The current column is compared to the desired column at both the storage rack and the interface conveyor. The current row is compared to the desired row at both the storage rack and the interface conveyor.

Table 6.2: Internal Coils

COIL NUMBER	DESCRIPTION
C1	Initialization complete
C2	Processing a command
C3	Horizontal address = 0
C4	Horizontal address > 0
C5	Vertical address > 0
C6	Vertical address = 0
C9	Motion to travel level is complete
C10 & C12	Column count delay timer
C11 & C13	Row count delay timer
C20	Part present on shuttle mechanism
C21	Ready to load pallet onto shuttle
C22	Ready to unload pallet from shuttle
C26	Fingers detect by first sensor
C200	Current column = conveyor column
C101	Current column < conveyor column
C102	Current column> conveyor column
C201	Current row = conveyor row
C104	Current row < conveyor row
C105	Current row > conveyor row
C202	Current column = storage location column
C107	Current column < storage location column
C108	Current column > storage location column
C203	Current row = storage location row
C109	Current row < storage location row
C110	Current row > storage location row
C999	Error, stop motion
C4000	Horizontal limit error
C4001	Vertical limit error
C4002	Toshiba fault error
C4003	At "home" position
C4004	Vertical motor timer error
C4005	Horizontal motor timer error
C4006	Stepper motor timer error
C4007	Crane in motion
C4008	Shuttle in motion
C4009	Invalid address
C14, C15	Satisfy TISOFT ladder restrictions
C1043 - C1099	Satisfy TISOFT ladder restrictions

CHAPTER SEVEN
LADDER ROUTINES

The ladder designed to control the AS/RS contain several routines to execute the AS/RS's functions. These routines are as follows: Main Program, Initialization, Store, Retrieve, Address, Travel Level, Move to Row, Move to Column, and Shutdown. This chapter discusses the purpose, the performance steps, and the special programming features, if any, of each routine.

7.1 Main Program

The purpose of the Main Program is to manage the entire ladder. The Main Program calls the proper subroutine(s) to perform the required operation. Unlike the subroutines, the rungs in the Main Program are executed every scan.

7.1.1 Steps

The steps performed by the Main Program consist of the following:

1. Upon start-up the user must first perform an initialization to the home position. Further operations cannot be performed if an initialization has not occurred.
2. If the command strobe is active, send busy signal to workcell controller until the requested operation is performed.
3. Call "Address" subroutine one time per function to determine storage location.
4. Determine if the address is invalid; continue operation if it is not invalid.
5. Call proper subroutine(s) to perform desired operation.

7.1.2 Features

The workcell controller computer "command" consists of a single store/retrieve bit and two sets of 4 address bits – one for the row address and one for the column address. A "0" in the store/retrieve bit indicates a store operation is requested at the storage location specified by the row and column address bits. A "1" in the store/retrieve bit signifies a retrieve operation is desired at the storage location specified by the row and column address bits.

In order to perform additional operations, the store/retrieve bit is combined with various combinations of horizontal and/or vertical address bits equal to 0. The presence of "0000" in the horizontal address bits and a valid address in the vertical address bits means move to the row or to the column specified by the vertical address bits. The command bit status determines whether to move to a row or to a column. The presence of "0000" in the horizontal and the vertical address bits signifies that an initialization or a shutdown operation is requested. The status of the command bit determines which operation is desired.

Table 7.1 demonstrates the combination of command bit and address bits implemented to perform the desired operations. A valid, non-zero row or column address is represented by "xxxx".

Table 7.1: AS/RS Commands

Store/Retrieve Command X103	Horizontal Address X104 - X107	Vertical Address X108 - X111	Operation
0	xxxx	xxxx	Store
1	xxxx	xxxx	Retrieve
0	0000	0000	Initialize
1	0000	0000	Shutdown
0	0000	xxxx	Send to Column (xxxx)
1	0000	xxxx	Send to Row (xxxx)
0	xxxx	0000	Invalid

7.2 Initialization (SBR 1)

The Initialization subroutine is responsible for driving the AS/RS to the "home" position as discussed in Section 6.2.1. Global variables are initialized in the subroutine as well. The Initialization subroutine is by the user using the code specified in Section 7.1.

7.2.1 Steps

The steps performed by the Initialization subroutine are:

1. Move the shuttle mechanism vertically in the negative direction until the Y HOME limit switch is active.
2. Move the crane horizontally in the negative direction until the X HOME limit switch is active.
3. Load current column and current row global variables with (1,0).

7.3 Store (SBR 2)

A "store" operation consists of obtaining a pallet of finished product from the FMAS and storing it in a specified location in the storage rack. In the case of this project, the AS/RS gathers a pallet from the pickup conveyor and stores it in a location specified by the workcell controller.

7.3.1 Steps

The steps in the Store subroutine consists of the following:

1. Compare current column and row location to the pickup conveyor column and row.
2. If the AS/RS is not at the pickup conveyor column, call "Travel Level" subroutine to move to the travel level.
3. Move to the pickup conveyor column, add or subtract to the column count depending on the direction of the motion.
4. Move to the conveyor row, add or subtract to the row count depending on the direction of the motion.
5. Load pallet from the conveyor onto the shuttle mechanism.
6. Compare current column and row to storage location column and row.
7. If AS/RS is not at storage location column, call "Travel Level" subroutine to move to the travel level.
8. Move to the storage location column, add or subtract to the column count depending on the direction of the motion.
9. Move to storage location row, add or subtract to the row count depending on the direction of the motion.
10. Unload pallet from the shuttle mechanism to the storage location.

7.3.2 Features

The direction of the stepper motor to load and unload a pallet onto the shuttle mechanism is dependent on the pallet's location. In a store function, the pallet is loaded from the interface conveyor onto the shuttle mechanism and is unloaded from the shuttle mechanism onto the storage rack. To pull the pallet onto the shuttle mechanism, the fingers move away from the pallet's location. In this case, the stepper motor is driven in a counter-clockwise direction. To push the pallet off of the shuttle mechanism, the fingers are driven towards the unloading location. This requires a counter-clockwise motion of the stepper motor. Determining when the pallet is loaded or unloaded is fully explained in Section 7.10.2.

7.4 Retrieve (SBR 3)

The purpose of the Retrieve subroutine is to retrieve a pallet from the storage rack and place it on the delivery conveyor. The storage location in the rack is specified by the workcell controller.

7.4.1 Steps

The Retrieve subroutine performs the following steps:

1. Compare current column and row to storage location column and row.
2. If AS/RS is not at the storage location column, call "Travel Level" subroutine to move to the travel level.
3. Move to the storage location column, add or subtract to the column count depending on the direction of the motion.

4. Move to the storage location row, add or subtract to the row count depending on the direction of the motion.
5. Load pallet from storage location onto shuttle mechanism.
6. Compare current column and row to the delivery conveyor column and row.
7. If AS/RS is not at the delivery conveyor column, call "Travel Level" subroutine to move to the travel level.
8. Move to the delivery conveyor column, add or subtract to the column count depending on the direction of the motion.
9. Move to the delivery conveyor row, add or subtract to the row count depending on the direction of the motion.
10. Unload pallet from the shuttle mechanism onto the conveyor.

7.4.2 Features

In a retrieve function, the pallet is loaded from the storage rack onto the shuttle mechanism and is unloaded from the shuttle mechanism onto the FMAS delivery conveyor. To pull the pallet onto the shuttle mechanism, the fingers move away from the pallet's location. The stepper motor is driven in a clockwise direction. To push the pallet off of the shuttle mechanism, the fingers are driven towards the unloading location. This requires a clockwise motion of the stepper motor. Determining when the pallet is loaded or unloaded is fully explained in section 7.10.2.

7.5 Address (SBR 4)

The Address subroutine determines the horizontal and the vertical addresses

of the desire storage location. Both the horizontal and the vertical address inputs consist of four input bits. An active bit 0 corresponds to a value of 2^0. Bit 1 has a value of 2^1 it is active. An active bit 2 corresponds to a value of 2^2. Bit 3 has a value of 2^3 if it is active. Any inactive bit has a value of 0.

7.5.1 Steps

The Address subroutine determines the storage location by:

1. If bit 0, 1, 2, or 3 is active for the horizontal address, load into memory locations the values 2^0, 2^1, 2^2, and 2^3, respectively.
2. If bit 0, 1, 2, or 3 is not active for the horizontal address, load into memory locations the value 0.
3. If bit 0, 1, 2, or 3 is active for the vertical address, load into memory locations the values 2^0, 2^1, 2^2, and 2^3, respectively.
4. If bit 0, 1, 2, or 3 is not active for the vertical address, load into memory locations the value 0.
5. Add the horizontal address values together for the storage location column number.
6. Add the vertical address values together for the storage location row number.

7.6 Travel Level (SBR 5)

Since horizontal motion occurs for safety reasons only when the shuttle is at a base location, a subroutine has been dedicated to this function. The purpose of the Travel Level subroutine is to move the shuttle to the base travel level. This subroutine will be called throughout the ladder before the start of any horizontal motion.

7.6.1 Steps

The Travel Level subroutine consists of the following steps:

1. Move vertically in the negative direction until the Y HOME limit switch is active.
2. Load the value 0 into the current row variable.

7.7 Move to Row (SBR 6)

The Move to Row subroutine moves the shuttle to a specified row within the current column. This function was developed to allow the user to move the AS/RS to a location without performing a store or a retrieve operation.

7.7.1 Steps

The steps the Move to Row subroutine performs are as follows:

1. Compare the current row to the desired row.
2. Move to the desired row within the current column, add or subtract to the row count depending on the direction of the motion.

7.8 Move to Column (SBR 7)

The purpose of Move to Column is to move the crane to a specified column at the travel level. The shuttle remains at the travel level at the end of this operation. This function was developed to move the AS/RS to a location without

performing a store or a retrieve function. In order to move to a storage location in a different row and column without executing a store or a retrieve, this function must be done first then a Move to Row function.

7.8.1 Steps

The steps in the Move to Column subroutine consists of the following:

1. Compare the current column to the desired column.
2. If the AS/RS is not at the travel level, call "Travel Level" subroutine to move to the travel level
3. Move to the desired column, add or subtract to the column count depending on the direction of the motion.

7.9 Shutdown (SBR 8)

When initializing the AS/RS, the motors are driven at a slow speed. In order to reduce the time for the initial initialization, the AS/RS is positioned close to "home" before powering down. The Shutdown subroutine performs this operation.

7.9.1 Steps

The Shutdown subroutine performs the following steps:

1. If the AS/RS is not at the travel level, call "Travel Level" subroutine to move to the travel level
2. Move to column number 2 (close to the home position).

7.10 Global Features

Some features of the ladder are used throughout the entire program.

7.10.1 Counting of Columns and Rows

This is a feature of Store, Retrieve, Move to Column, and Move to Row subroutines. The counting of columns and rows is an important part of the AS/RS ladder. The column or row value desired changes with each operation. Within TISOFT counters do not have variable values. A counter counts to a set value. Therefore, using counters within the ladder was not feasible. Instead simple addition and subtraction is applied. Each time the AS/RS X Horizontal Position or the Y Vertical Position optical sensors detects a column or a row, respectively, a value of 1 is added to or subtracted from the current column or row count. The value 1 is added if the AS/RS is moving in a positive direction and is subtracted if the AS/RS is moving in a negative direction. A One-Shot is employed to ensure the value is added or subtracted only once. In the time required for the AS/RS to pass a column or a row marking, the ladder has been scanned multiple times. If a one-shot was not used, a value of 1 would be added each time the ladder was scanned at a column or row marking instead of each time a marking was found.

An additional consideration was needed for the counting of rows and columns. The optical sensors which detect a column or a row marking are not active at the time of detection. Upon the start of movement from the current location to the desired location, at least one of the two sensors detects a column and/or row. This column or row should not be counted. Therefore, in order to ensure that the current column or row is not counted twice, a timer is set to delay adding or subtracting from the current count.

7.10.2 Loading and Unloading a Pallet

Since the presence of a pallet on the shuttle mechanism is dependent on the performance and positioning of the "push-pull" fingers, the order of detection of the fingers by the optical sensors on the shuttle is important. A means of determining the exact location of the "push-pull" fingers is unavailable unless the fingers happen to be in sight of the optical sensors. The positioning of these sensors on the shuttle mechanism can be seen in Figure 7.1. It is assumed that if the shuttle mechanism is empty, the "push-pull" fingers are located anywhere on their path.

When loading a pallet onto the shuttle mechanism, the fingers must pass both sets of optical sensors to guarantee the pallet has been loaded. The sensors have to detect the fingers in a certain order. The sensor closest to the pallet location, either the FMAS or the storage rack, detecting the long set of fingers means that the fingers are in position to grab the pallet. Once the sensor away from the pallet's loading location has detected the set of long fingers, the pallet has been loaded. Figure 7.2 demonstrates the AS/RS loading a pallet onto the shuttle mechanism.

After the pallet has been loaded, the set of short fingers are positioned behind the pallet. Figure 7.3 illustrates the positioning of the short fingers behind the pallet. These fingers can push the pallet off of the shuttle mechanism when unloading the pallet. Detection of the set of long fingers by the sensor away from the unloading location indicates that the pallet has been unloaded.

7.10.3 Cycle

One of the main characteristics of each subroutine is the cycle of the function starts and ends with the shuttle mechanism empty. Those functions which load a

pallet onto the shuttle mechanism -- Store and Retrieve -- also unload the pallet from the shuttle mechanism. All other functions -- Initialization, Address, Travel Level, Move to Row, Move to Column, and Shutdown -- do not include any contact with a pallet.

Starting and ending every operation with an empty shuttle has a major advantage. The workcell controller (or user) cannot start another operation while the shuttle is loaded. The only two ways a pallet could be on the shuttle when an operation is begun are 1.) if a pallet is placed on the shuttle by an means outside the system and 2.) if power is lost in the middle of an operation.

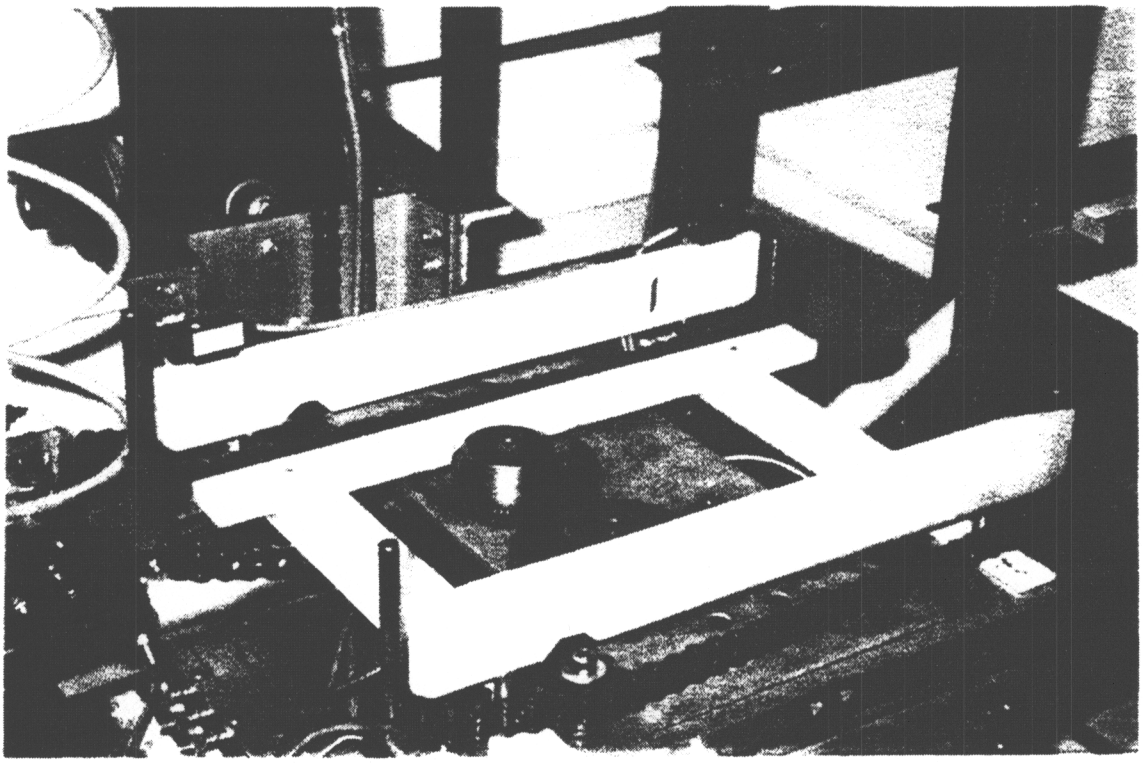


Figure 7.1: Shuttle Mechanism Fingers and Sensors

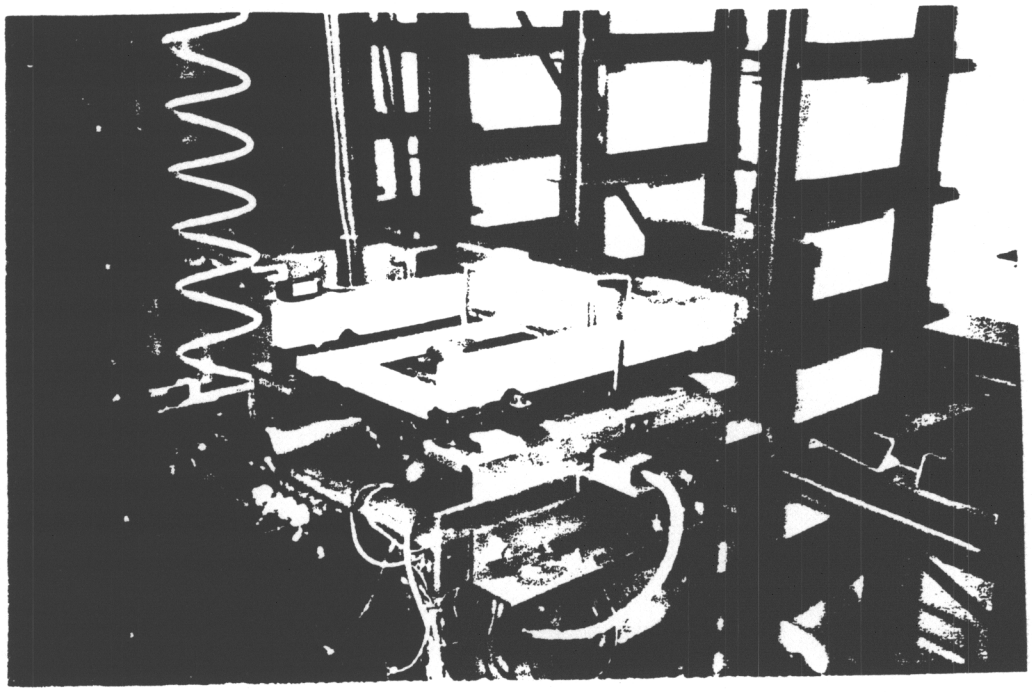


Figure 7.2: Loading a Pallet onto the Shuttle Mechanism

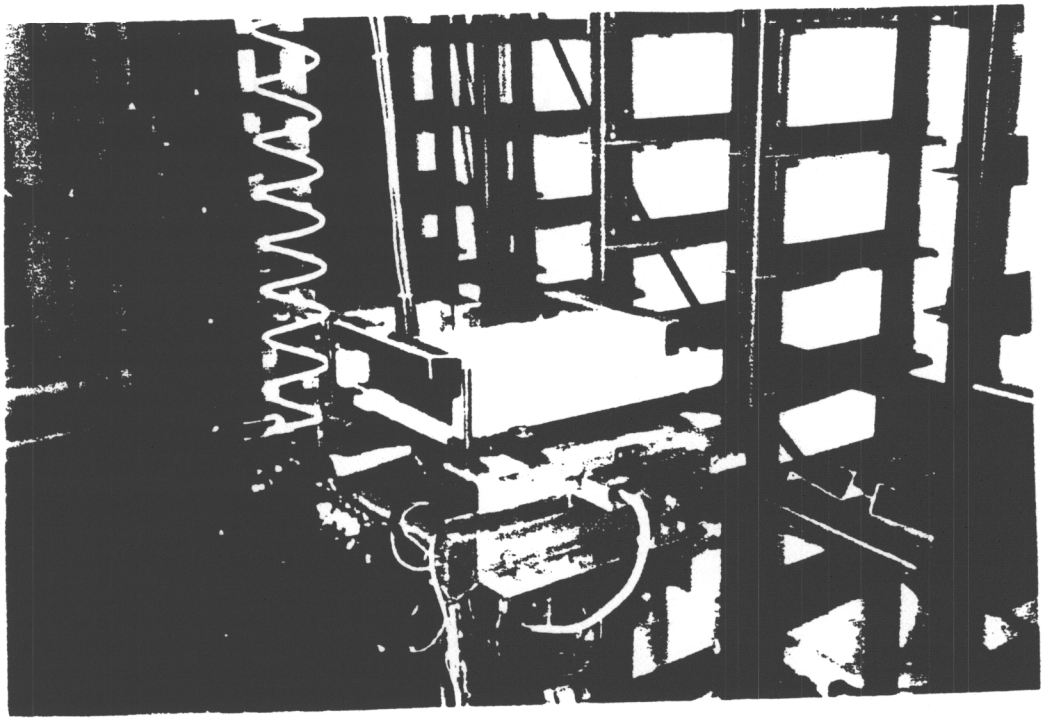


Figure 7.3: Pallet Loaded on the Shuttle Mechanism

CHAPTER EIGHT
OPERATION AND TESTING OF THE AS/RS

8.1 AS/RS System Components

As stated earlier, a Texas Instruments 565 programmable logic controller is used to control the AS/RS in this project. The TI 565 PLC works with a remote base. The remote base contains the input wires and the output wires for the FMAS and the AS/RS. The workcell controller computer system (WCC3) communicates with the PLC through two sets of discrete wires that are connected to a digital input/output board inside the computer

The TI 565 remote base includes the following elements:

- Remote Base Controller
- Power Supply
- 3 Low Voltage - DC (LVDC) 32 point Output Modules
- 4 LVDC 32 point Input Modules
- Input Simulator
- TIWAY I Network Interface

The computer system consists of the following:

- AT&T 386 personal computer
- MS-DOS version 5.0
- Windows version 3.1
- TISOFT Release 2.3 (Texas Instruments Relay Ladder Logic program)
- Quick C for Windows version 1.0
- 64 point digital input/output board

8.2 Procedure

Step 1: Power up all the required equipment. The order in which these are powered is not significant.

- Turn on workcell controller (WCC3)
- Plug in TI 565 PLC
- Plug in remote base controller
- Turn on main circuit breaker (top left switch)
- Plug in power for three phase motor
- Plug in power for 24 volts
- Turn on Toshiba inverter (gray switch on top off control cabinet)
- Turn on pulse generator
- Turn on air (recommended pressure is 60 psi)

Step 2: Load the ladder into the current TISOFT directory

Step 3: Start TISOFT program

Step 4: Choose ONLINE option for 9600 baud

Step 5: Choose CONFIO

Configure the input/output according to Table 8.1

Write the configuration to the PLC

Step 6: Load the ladder into the PLC (the ladder must be in the working directory)

Step 7: Begin running the ladder

Table 8.1: PLC Input/Output Configuration

SLOT	I/O ADDRESS	X	Y	SPECIAL FUNCTION
1	7	0	32	NO
2	33	0	32	NO
3	65	0	32	NO
4	97	32	0	NO
5	129	32	0	NO
6	161	32	0	NO
7	193	32	0	NO
8	0	0	0	NO
9	0	0	0	NO
10	201	8	0	NO
11	209	0	8	YES

8.3 Testing

The AS/RS control system was tested in two stages. First the relay ladder logic was tested by executing each function performed by the AS/RS. TISOFT has the option of "forcing" an input on or off. The PLC does not recognize the difference between a forced input and a physically active input. The difference, however, is that forcing an input does not physically activate the input. Each function was tested by forcing the appropriate store/retrieve command, the desired storage address bits, and the strobe. Secondly the C program to simulate the computer interface was tested.

For testing purposes the horizontal and vertical motors were driven at a frequency of 15 Hz during initialization and 20 Hz during all other operation. After testing was completed, the frequency during operation was increased from 20 Hz. Several frequencies were tested before settling on a frequency of 30 Hz. This frequency was chosen in order to maintain accuracy when stopping the AS/RS.

Originally the column sensor detected the column bars on the storage rack. However, due to the width of the columns a slight accuracy problem occurred when the AS/RS approached a column from different direction. To solve this problem, a bar similar to the one used to indicate rows was proposed. A bar with metal rods attached was installed at the top of the storage rack.

After testing the operations of the AS/RS, the time travel time of the AS/RS was compared to the expected travel time. The expected travel time was derived from the Bozer and White model. The following model was adapted from the Bozer and White model to account for traveling one direction at a time:

$t_h = L/s_h$

$t_v = H/s_v$

$E(SC) = t_h + t_v$

The possible travel distance in the horizontal direction is 13 ft. The vertical direction travel distance is 4 feet. At 30 Hz the horizontal motor's average speed is 0.24 ft/sec. At 30 Hz the vertical motor's average speed is 0.31 ft/sec. The expected travel time is calculated to be 67 seconds.

$t_h = L/s_h = 13 \text{ ft.}/.24 = 54.1 \text{ sec}$

$t_v = H/s_v = 4 \text{ ft.}/.31 = 12.9 \text{ sec}$

$E(SC) = t_h + t_v = 54.1 \text{ sec} + 12.9 \text{ sec} = 67 \text{ sec}$

The average time of travel along the horizontal (t_h) was 55 76/100 seconds. The average time of travel along the vertical (t_v) was 13 50/100 seconds. An average was taken of these times to reduce human error in the data collection. Based on this information the travel time of the AS/RS was found to be 69 26/100 seconds.

CHAPTER NINE

CONCLUSIONS

9.1 Summary

An Automated Storage and Retrieval System was designed and constructed to supply and support a Flexible Machining and Assembly System. Descriptions of the AS/RS and of FMAS were provided. This report specifies the need for a control system on the currently existing Automated Storage and Retrieval System. The objective of this project was to develop a system to control the motion and operation of the AS/RS.

In order to design a control system appropriate to the AS/RS, the constraints of the system and the design criteria were established. In addition to the operations required by an AS/RS -- Initialization, Store, Retrieve, and Shutdown, the operations desired by the user -- Move to Column and Move to Row -- for the Virginia Tech AS/RS were identified Using a Texas Instruments 565 Programmable Logic Controller a relay ladder logic was designed to control the motion of the AS/RS.

A program to simulate the workcell controller's management of the AS/RS was developed. This program only simulates the handshake between the workcell controller and the PLC. The actual handshake could not be implemented due to complications accessing the digital I/O board. Using this program the user inputs the desired operation to be performed and the storage location to be accessed. Based on the user's inputs the program initiates the operation request process. If the AS/RS is not busy, the strobe is turned on. The proper status for the command bit and the 8 horizontal and vertical address bits is then established.

9.2 Recommendations

9.2.1 Safety

The current design of the AS/RS and the AS/RS's control system safety system could be improved to allow for additional problems. The AS/RS's safety consists of four extreme limit switches to stop motion of the crane. If the user of the AS/RS must stop the AS/RS before reaching these limit switches, the only means available is to halt running the ladder program. This method requires being at the workcell computer and takes considerable time. Additionally, this method is not appropriate if the user is not at the computer.

The FMAS contains a set of FMAS Emergency Stop boxes throughout the system. All these Emergency Stop boxes are connected at a central control box. A connection through this location ought to be made to include the AS/RS. The AS/RS's emergency stop should function as the limit switches do -- stop motion of the crane motors.

Additionally, sensors to detect an obstacle in the path of the AS/RS ought to be added. If an object is in the path of the AS/RS, there is no means of the AS/RS to detect it. The only way to stop the AS/RS if an obstacle exists is for the user to visually observe the object and stop running the ladder program. During the operation of the AS/RS, a pallet to be loaded onto the shuttle is assumed to be present in the expected location. As well the destination location is assumed to be empty when a pallet is to be unloaded from the shuttle. Two sensors positioned on the shuttle would detect the presence of a pallet in these situations. Two sensors are required to detect pallets both in the storage rack and on the FMAS conveyors.

9.2.2 System

Several assumptions were made in the design of the control system for the AS/RS. The addition of sensors to the system would eliminate the uncertainty produced from these assumptions. The presence of a pallet on the shuttle mechanism is assumed if the "push-pull" fingers have performed the appropriate motions. A sensor on the shuttle mechanism itself could detect if the "push-pull" fingers did indeed pull a pallet onto the shuttle or push a pallet off of the shuttle.

Repositioning of the storage rack, the interface conveyor, and certain limit switches on the crane will permit access to the unusable column and rows. Shifting the storage rack to one side will alleviate the problem of hitting the wall when the crane moves to the first column in the storage rack. In order to maintain only 15 stopping positions within the AS/RS after gaining the use of the first column, the interface conveyor must be aligned with at least one column on the storage rack. Moving the storage rack brace bar up one inch will position the brace bar above the height of the pallet's lip. This will permit the use of Row 4. The position of the lower vertical limit switch poses an obstacle when reaching the top two rows of the storage rack. By relocating or redesigning this limit switch, the top two rows of the storage rack may be accessed.

REFERENCES

1. Allen, Sabrina L., "A Selection Guide to AS/R Systems", *Industrial Engineering*, March 1992, pp. 28-31.

2. Bozer, Yavuz, and John White, "Travel-Time Models for Automated Storage/Retrieval Systems," *IIE Transactions*, Volume 16, Number 16, December 1984, pp. 329-338.

3. College-Industry Council on Material Handling Education, "The 24 Principles of Material Handling," *Material Handling Engineering*, Directory Issue 1993/1994, pp. A/13-A/18.

4. Eaglesham, Mark A., "Automated Storage and Retrieval System Design Report," Department of Industrial and Systems Engineering, VPI&SU, Blacksburg, VA, 1995.

5. Eastman, Robert, *Materials Handling*, Marcel Dekker, Inc, New York, 1987.

6. Forger, Gary, "We turned an AS/RS into a materials control center," *Modern Materials Handling*, June 1995, pp. 54-55.

7 Forger, Gary, "AS/RS: changing with the times," *Modern Materials Handling*, December 1990, pp. 50-54.

8. Han, Min-Hong, et al., "On Sequencing Retrieval In an Automated Storage/Retrieval System," *IIE Transactions*, Volume 19, Number 1, March 1987, pp. 56-66.

9. Knill, Bernie, et al., "AS/RS: Full Partner in Manufacturing," *Material Handling Engineering*, May 1993, pp. 43-52.

10. Nagy, Alex, *Materials Handling Handbook*, Ed. Raymond Kulwiec, 2nd edition, John Wiley & Sons, New York, 1985, pp. 604-625.

11. Jones, C.T and L. A. Bryan, *Programmable Controllers: Concepts and Applications*, International Programmable Controls, Inc., Atlanta, GA, 1983.

12. Texas Instruments, *560 Program Design Guide: Manual No. 560-8103*, Texas Instruments Incorporated, 1987.

13. Texas Instruments, *565 Programming Instructions Manual: Manual No. 565-8102*, Texas Instruments Incorporated, 1987.

APPENDIX A

PLC INPUT AND OUTPUT

PLC	DESCRIPTION	FUNCTION	CONN #
		COMMON	
BUS	COMMON: 0 VDC		TS-26-10
BUS	TOSHIBA INVERTER: CC 0 VDC		TS-11-10
BUS	COMMON		TS-21-10
		INPUTS TO PLC FROM AS/RS	
X118	X HOME LIMIT SWITCH	limit switch for horizontal home position	TS-26-06
X119	Y HOME LIMIT SWITCH (NO)	limit switch for vertical home position	TS-26-07
X150	X LIMIT SWITCH-1 SIGNAL (NO)	extreme horizontal limit on left of rack	TS-22-06
X151	X LIMIT SWITCH-2 SIGNAL (NO)	extreme horizontal limit on right of rack	TS-22-07
X152	Y LIMIT SWITCH-1 SIGNAL (NO)	extreme vertical limit at bottom of rack	TS-22-08
X153	Y LIMIT SWITCH-2 SIGNAL (NO)	extreme vertical limit at top of rack	TS-21-01
X120	Y VERT POSITION/OPT SWITCH	optical sensor for sensing row bars	TS-26-08
X123	X HORIZ POSITION/OPT SWITCH	optical sensor for sensing column bars	TS-25-03
X121	Z1 SHUTTLE POSITION/OPT SWITCH	optical sensor to sense push-pull fingers (by FMAS)	TS-25-01
X122	Z2 SHUTTLE POSITION/OPT SWITCH	optical sensor to sense push-pull fingers (by rack)	TS-25-02
X155	TOSHIBA INVERTER: FLA	fault on Toshiba inverter	TS-21-03
		OUTPUTS FROM PLC TO AS/RS	
Y61	SSR Y: SELECT VERTICAL MOTION	turn on to move crane vertically (not while horiz on)	TS-07-05
Y62	SSR X: SELECT HORIZ MOTION	turn on to move crane horizontally (not while vert on)	TS-07-06
Y30	TOSHIBA INVERTER: FWD	forward direction control (not when REV on)	TS-11-06
Y31	TOSHIBA INVERTER: REV	reverse direction control (not when FWD on)	TS-11-07
Y32	TOSHIBA INVERTER: SS1	speed selection one (to be set in Toshiba)	TS-11-08
Y33	TOSHIBA INVERTER: SS2	speed selection two (to be set in Toshiba)	TS-10-01
Y34	TOSHIBA INVERTER: SS3	speed selection three (to be set in Toshiba)	TS-10-02
Y54	TOSHIBA INVERTER: RST	reset	TS-08-06
Y55	TOSHIBA INVERTER: AD2	2nd set of acceleration and deceleration	TS-08-07
Y56	TOSHIBA INVERTER: ST	interlock	TS-08-08
Y57	V1 EXTEND LEFT	extend valve one (keep on while in position)	TS-07-01
Y58	V2 RETRACT LEFT	retract valve two (keep on while in position)	TS-07-02
Y59	V3 EXTEND RIGHT	extend valve three (keep on while in position)	TS-07-03
Y60	V4 RETRACT RIGHT	retract valve four (keep on while in position)	TS-07-04
Y36	CW STEPPER PULSE	stepper motor on/off	TS-10-04
Y37	CCW STEPPER PULSE	stepper motor direction (0 – CW, 1 – CCW)	TS-10-05

		INPUTS TO AS/RS FROM WCC3	
X102	AS/RS COMMAND STROBE		TS-28-06
X103	AS/RS COMMAND (STORE/RETRIEVE)	store or retrieve command sent from WCC3	TS-28-07
X104	AS/RS HORIZ ADDRESS: bit 0	storage location horizontal address sent from WCC3	TS-28-08
X105	AS/RS HORIZ ADDRESS: bit 1	storage location horizontal address sent from WCC3	TS-27-01
X106	AS/RS HORIZ ADDRESS: bit 2	storage location horizontal address sent from WCC3	TS-27-02
X107	AS/RS HORIZ ADDRESS: bit 3	storage location horizontal address sent from WCC3	TS-27-03
X108	AS/RS VERT ADDRESS: bit 0	storage location vertical address sent from WCC3	TS-27-04
X109	AS/RS VERT ADDRESS: bit 1	storage location vertical address sent from WCC3	TS-27-05
X110	AS/RS VERT ADDRESS: bit 2	storage location vertical address sent from WCC3	TS-27-06

X011	AS/RS VERT ADDRESS: bit 3	storage location vertical address sent from WCC3	TS-17-07

OUTPUTS FROM AS/RS TO WCC3

Y008	AS/RS BUSY	busy signal sent to WCC3 from AS/RS	TS-14-08
Y009	AS/RS DONE	done signal sent to WCC3 from AS/RS	TS-13-01
Y010	ACK AS/RS COMMAND	AS/RS acknowledge to WCC3 for command received	TS-13-02
Y011	AS/RS STATUS CODE: bit 0	see status code list	TS-13-03
Y012	AS/RS STATUS CODE: bit 1	see status code list	TS-13-04
Y013	AS/RS STATUS CODE: bit 2	see status code list	TS-13-05
Y014	AS/RS STATUS CODE: bit 3	see status code list	TS-13-06
Y024	AS/RS TO CONV: CONV DRIVE (on/off)	turn output (from AS/RS) conveyor on and off	TS-12-08
Y025	CONV TO AS/RS: CONV DRIVE (on/off)	turn input (to AS/RS) conveyor on and off	TS-11-01

APPENDIX B

C PROGRAM TO SIMULATE WORKCELL CONTROLLER

```
#include <conio.h>
#include <float.h>
#include <io.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>


void command();          // Send command bit
void h_address();        // Choose horizontal address
void v_address();        // Choose vertical address
void binary();           // Convert addresses to 4 bit binary
void acknowledge();      // Simulate acknowledge from AS/RS to WCC3
void move_to_col();      // Choose column to move to
void move_to_row();      // Choose row to move to

int menu();          // Print out menu
int initialize();        // Make sure want to initialize AS/RS
int shutdown();      // Make sure want to shutdown AS/RS


struct horizontal            // Horizontal address structure
{
    unsigned int X107 : 1;
    unsigned int X106 : 1;
    unsigned int X105 : 1;
    unsigned int X104 : 1;
} column;


struct vertical              // Vertical address structure
{
    unsigned int X111 : 1;
    unsigned int X110 : 1;
    unsigned int X109 : 1;
    unsigned int X108 : 1;
} row;


int status;                      // Status of command bit
signed int horiz_add;            // Horizontal address inputted by user
signed int vert_add;             // Vertical address inputted by user


main(void)
{
    int choice, initial, shut;


    for(;;)
    {
        choice = menu();             //Get users input
        switch(choice)
        {
            case 1:              // Initialize AS/RS
                initial = initialize();
                if (initial == 0)
                {
                    status = 0;
```

```
        vert_add = 0;
        binary();
        command();
        acknowledge();
        break;
    }

        else
        break;

    case 2:          // Store
    status = 0;
    h_address();
    v_address();
    binary();
    command();
    acknowledge();
    break;

case 3:              // Retrieve

    status = 1;
    h_address();
    v_address();
    binary();
    command();
    acknowledge();
    break;

case 4:                  // Move to Column
    status = 0;
    horiz_add = 0;
    move_to_col();
    binary();
    command();
    acknowledge();
    break;

case 5:                  // Move to Row
    status = 1;
    horiz_add = 0;
    move_to_row();
    binary();
    command();
    acknowledge();
    break;

case 6:                  // Shutdown AS/RS
    shut =  shutdown();
    if (shut == 0)
        {
        status = 1;
        horiz_add = 0;
        vert_add = 0;
        binary();
        command();
        acknowledge();
        break;
    }
    else
        break;
```

```
            case 7:                 // Quit
                return 0;
        }
    }
}


    /* Subroutine to print AS/RS menu */

menu(void)
{
    char s[2];
    int c;


    printf("\n\n\n\n\n       AS/RS Operation Menu");
     printf("\n\n\n");
    printf("     1. Initialization\n");
    printf("     2. Store Function\n");
    printf("     3. Retrieve Function\n");
    printf("     4. Move to Column\n");
    printf("     5. Move to Row\n");
    printf("     6. Shutdown\n");
    printf("     7. Quit\n");

    do
     {
        printf("\n");
        printf("       Enter your choice:  ");
        gets(s);
        c = atoi(s);
    }while(c < 0 || c > 7);
    return c;

}


    /* Subroutine to send command to AS/RS */

void command()
{

        // Simulate WCC3 sending command bit

    printf("\n\n   Command -- X103 = %d", status);


        // Simulate WCC3 sending of address bits

    printf("\n\n\n      Horizontal Address");
    printf("\n   X107  X106  X105  X104");
    printf("\n     %d     %d     %d     %d",column.X107,column.X106,column.X105
    printf("\n\n      Vertical Address");
    printf("\n   X111  X110  X109  X108");
    printf("\n     %d     %d     %d     %d",row.X111,row.X110,row.X109,row.X108


        // Simulate WCC3 sending strobe bit

    printf("\n\n\n    Strobe (X102) on.");

}
```

```
    /* Subroutine for user to input horizontal address */

void h_address()
{
    char h_add[2];


          // Simulate WCC3 choosing storage column address

    printf("\n Enter desired column (0-11,13,15): ");
    gets(h_add);
    horiz_add = atoi(h_add);


        // Check validity of column address

    if (horiz_add < 0 || horiz_add > 15 || horiz_add == 12 || horiz_add == 14)
    {
        do
          {
            printf("\n    Invalid address.");
            printf("\n Enter desired column (0-11,13,15): ");
            gets(h_add);
            horiz_add = atoi(h_add);
         }while(horiz_add < 0 || horiz_add > 15 || horiz_add == 12 || horiz_add
     }
}


    /* Subroutine for user to input vertical address */

void v_address()
{

    char v_add[2];


        // Simulate WCC3 choosing storage row address

    printf("\n Enter desired row (0-3,5-8):  ");
     gets(v_add);
    vert_add = atoi(v_add);


        // Check validity of row address

    if (vert_add < 0 || vert_add > 8 || vert_add == 4)
    {
        do
          {
            printf("\n  Invalid address.");
            printf("\n Enter desired row (0-3,5-8): ");
            gets(v_add);
            vert_add = atoi(v_add);
         }while(vert_add < 0 || vert_add > 8 || vert_add == 4);
     }
```

```
}


    /* Subroutine converts addresses to binary and stores each bit in
       appropriate AS/RS bit location */

void binary()
{

    signed int h_step1, h_step2, h_step3, h_step4;
    signed int v_step1, v_step2, v_step3, v_step4;
    int help;

        /* Process to convert column to 4 bit binary and strore
             in column of horizontal structure */

    h_step1 = horiz_add - pow(2,3);
    if (h_step1 >= 0) column.X107 = 1;
    else
    {
        column.X107 = 0;
        h_step1 = horiz_add;
    }

    h_step2 = h_step1 - pow(2,2);
    if (h_step2 >= 0) column.X106 = 1;
    else
    {
        column.X106 = 0;
        h_step2 = h_step1;
    }

    h_step3 = h_step2 - pow(2,1);
    if (h_step3 >= 0) column.X105 = 1;
    else
    {
        column.X105 = 0;
        h_step3 = h_step2;
    }

    h_step4 = h_step3 - pow(2,0);
    if (h_step4 >= 0) column.X104 = 1;
    else column.X104 = 0;


        /* Process to convert row to 4 bit binary and strore
             in row of vertical structure */

    v_step1 = vert_add - pow(2,3);
    if (v_step1 >= 0) row.X111 = 1;
    else
    {
        row.X111 = 0;
        v_step1 = vert_add;
    }

    v_step2 = v_step1 - pow(2,2);
    if (v_step2 >= 0) row.X110 = 1;
    else
    {
```

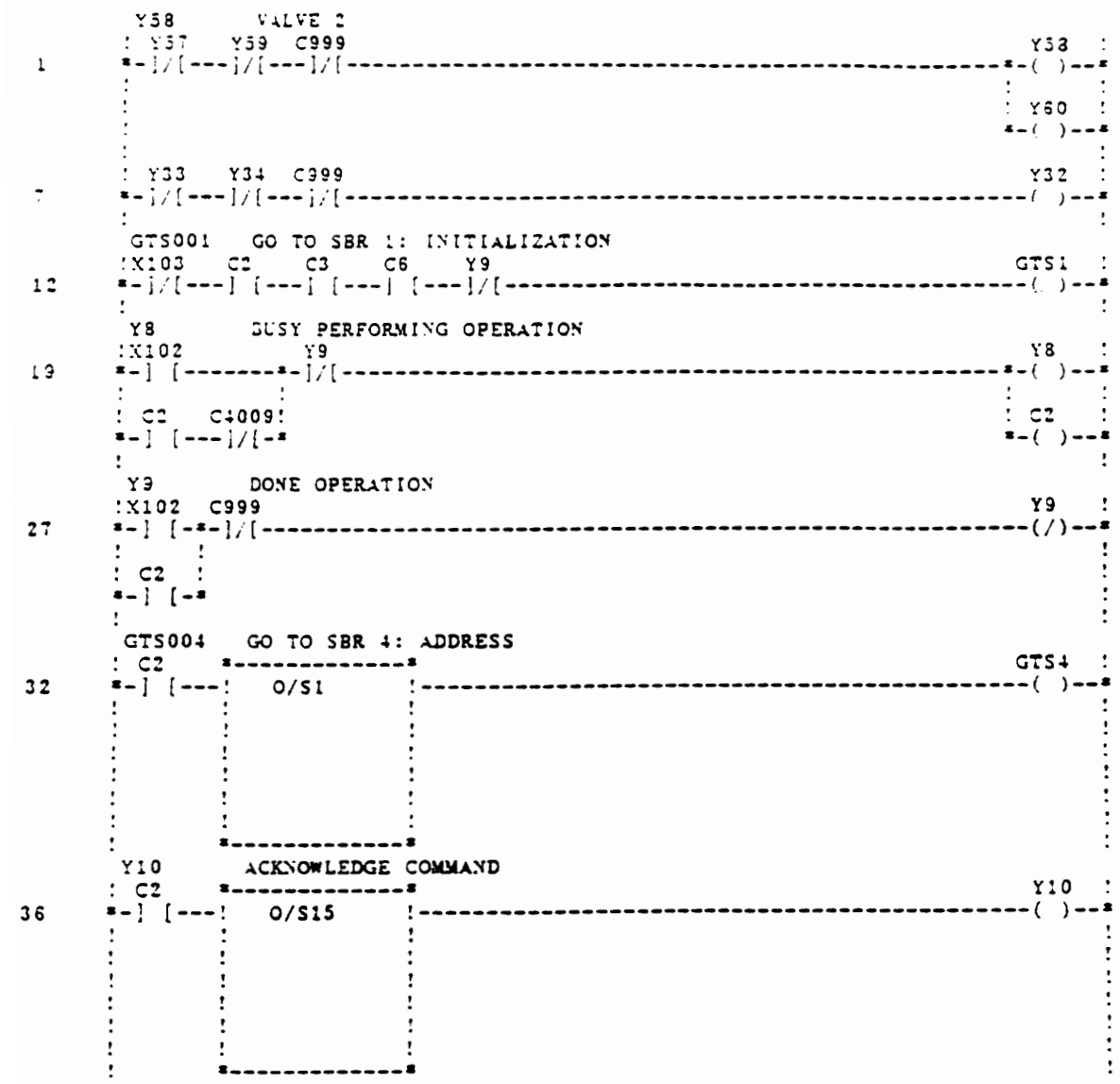
```
        row.X110 = 0;
        v_step2 = v_step1;
    }

    v_step3 = v_step2 - pow(2,1);
    if (v_step3 >= 0) row.X109 = 1;
    else
    {
        row.X109 = 0;
        v_step3 = v_step2;
    }

    v_step4 = v_step3 - pow(2,0);
    if (v_step4 >= 0) row.X108 = 1;
    else row.X108 = 0;


}


    /* Subroutine for user to input column to move to */

void move_to_col()
{
    char v_add[2];


        // Simulate WCC3 choosing column to move AS/RS to

    printf("\n Enter desired column (0-15):  ");
     gets(v_add);
    vert_add = atoi(v_add);


        // Check validity of column

    if (vert_add < 0 || vert_add > 15)
    {
        do
          {
            printf("\n  Invalid address.");
            printf("\n Enter desired column (0-15): ");
            gets(v_add);
            vert_add = atoi(v_add);
         }while(vert_add < 0 || vert_add > 15);
     }

}



    /* Subroutine for user to input row to move to */

void move_to_row()
{
    char v_add[2];


        // Simulate WCC3 choosing row to send AS/RS to
```

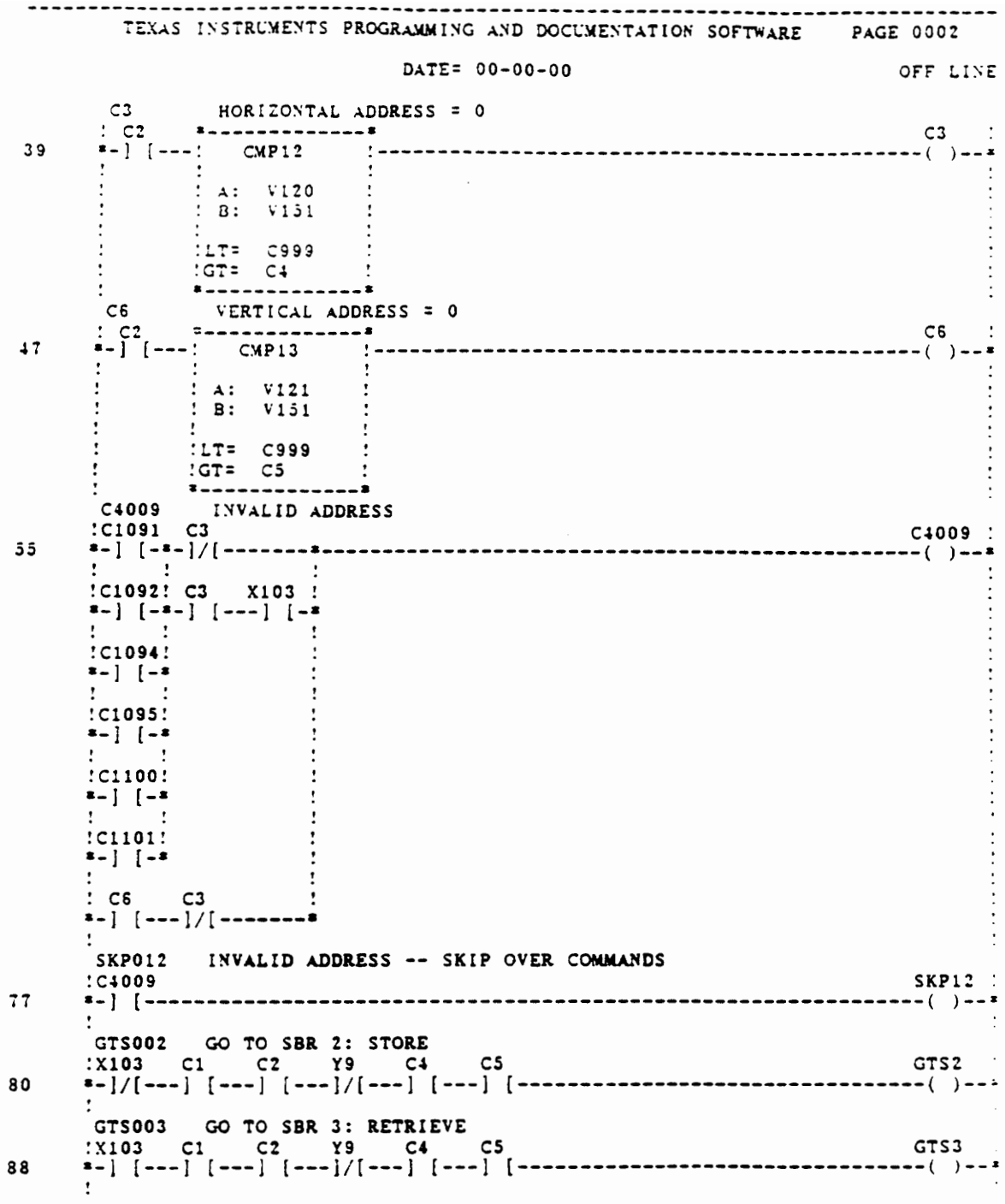
```
    printf("\n Enter desired row (0-3,5-8):  ");
     gets(v_add);
    vert_add = atoi(v_add);


        // Check validity of row

    if (vert_add < 0 || vert_add > 8 || vert_add == 4)
    {
        do
          {
            printf("\n  Invalid address.");
            printf("\n Enter desired row (0-3,5-8): ");
            gets(v_add);
            vert_add = atoi(v_add);
         }while(vert_add < 0 || vert_add > 15 || vert_add == 4);
     }

}



    /* Subroutine to simulate acknowledge from AS/RS to WCC3 */

void acknowledge()
{
    char dummy[5];

    printf("\n\n\n Press ENTER to acknowledge command (Y10).");
    gets(dummy);
    printf("\n  Strobe (X102) off.");
    printf("\n\n  AS/RS is busy (Y9).");
    printf("\n\n\n Press ENTER to return to menu.");
    printf("\n  End of operation.");
    gets(dummy);

}



    /* Subroutine to make sure user wants to initialize AS/RS */

initialize(void)
{
    char answer[1];
     int a;

    printf("\n\n Are you sure you want to initialize the AS/RS? Y or N ");
    gets(answer);
    a = strcmp(answer,"Y");
    return a;
}


    /* Subroutine to make sure user wants to shutdown AS/RS */

shutdown(void)
{
    char ans[1];
```

```
    int b;

   printf("\n\n Are you sure you want to shutdown the AS/RS? Y or N ");
   gets(ans);
   b = strcmp(ans,"Y");
    return b;
}
```

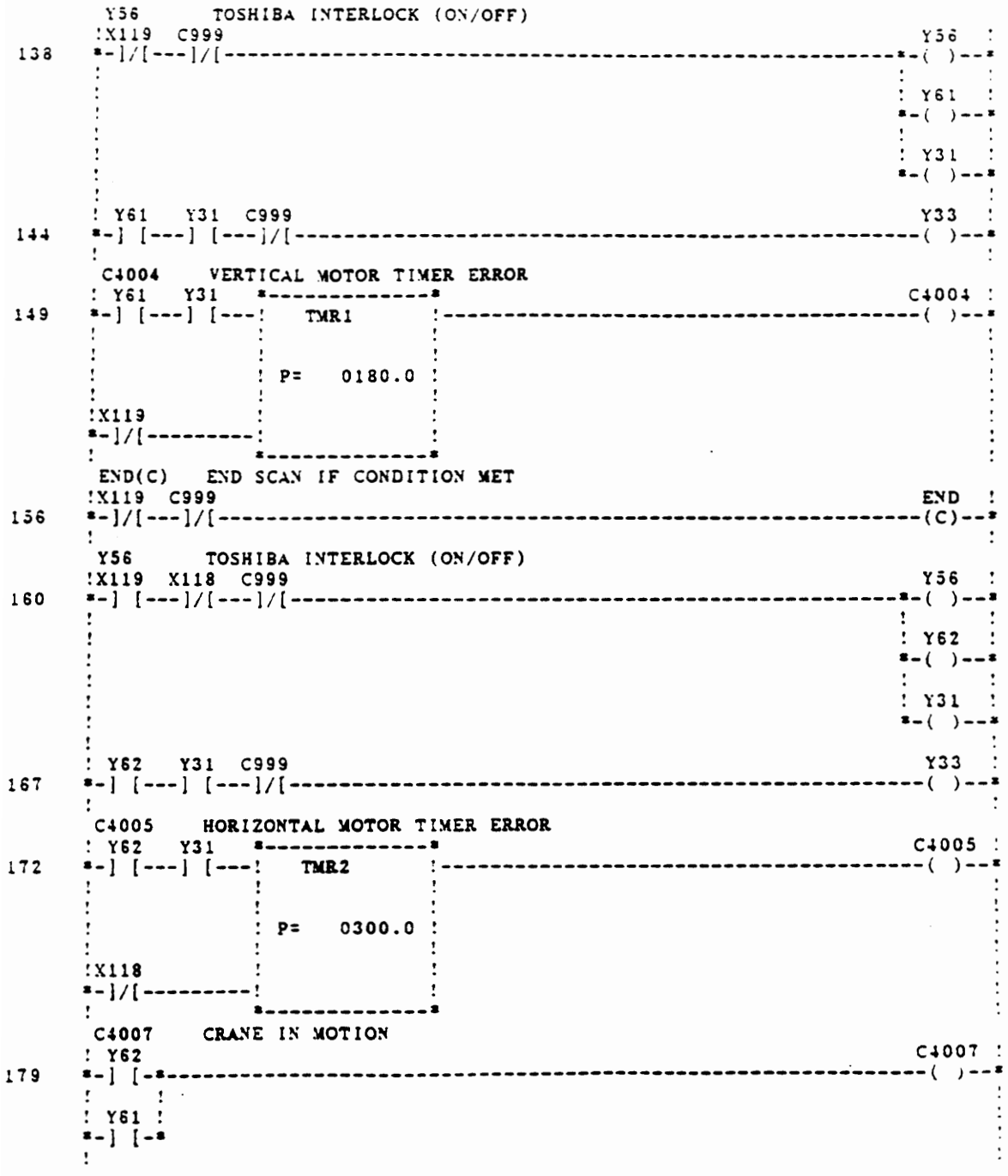
APPENDIX C

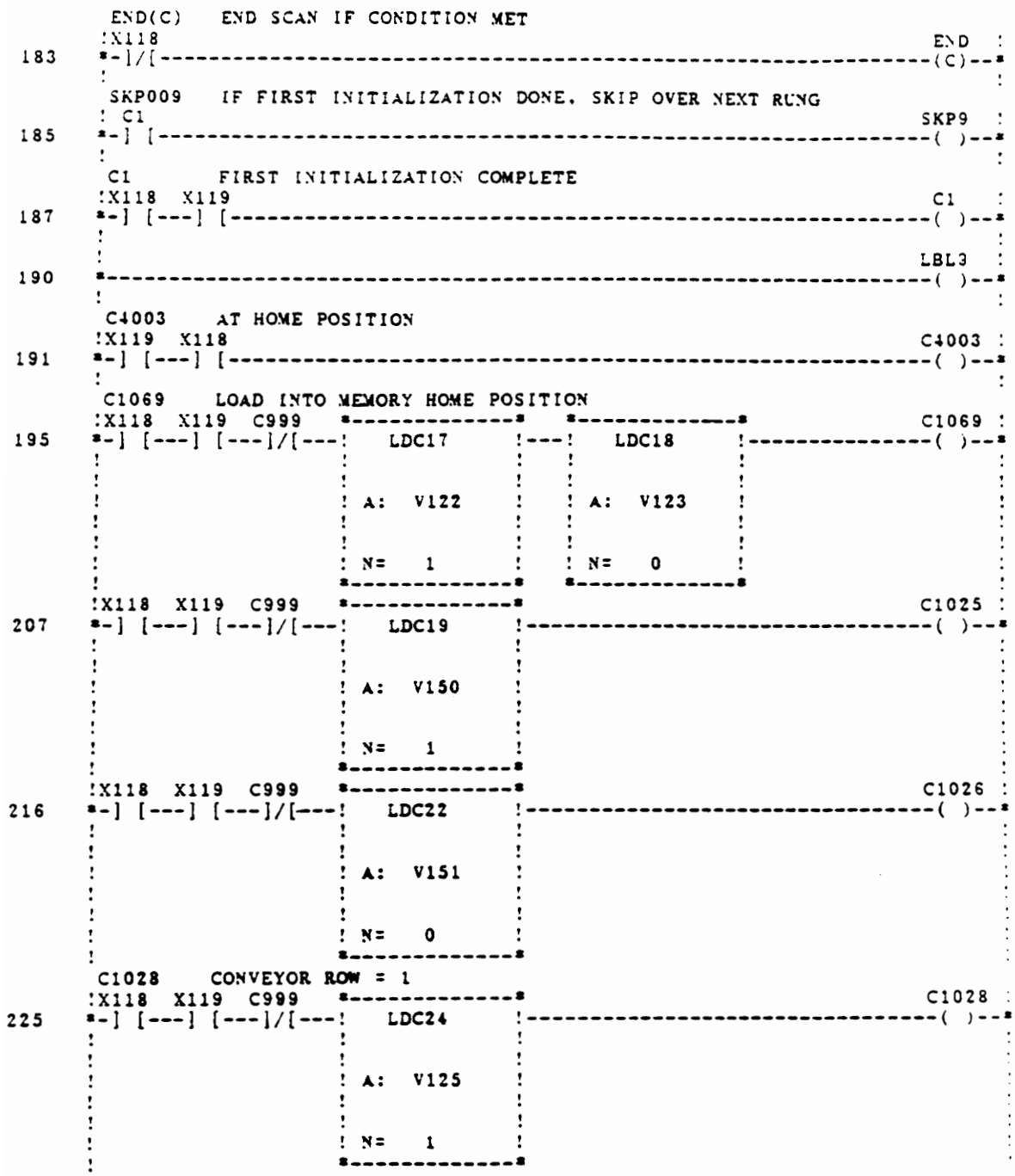
RELAY LADDER LOGIC

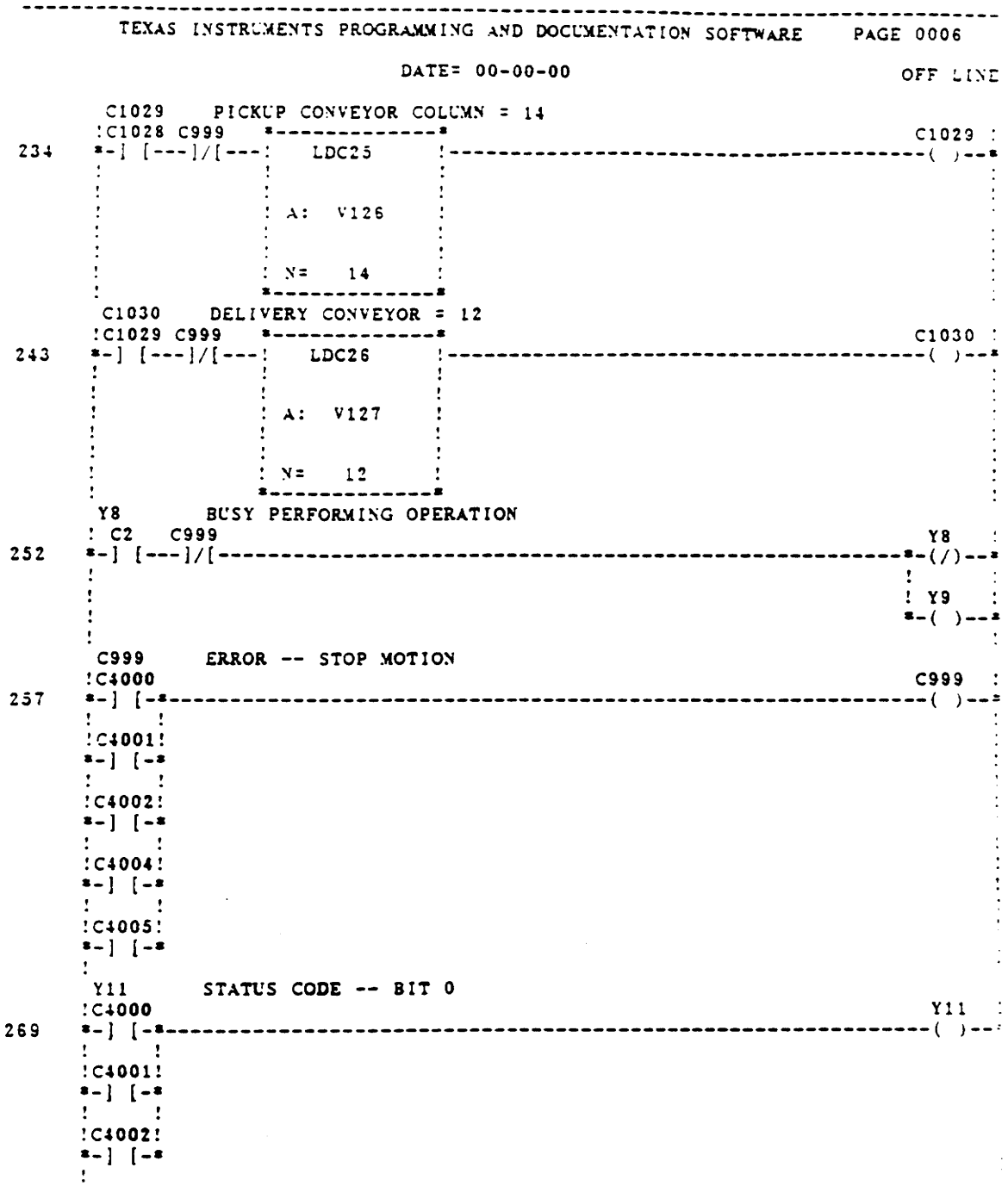

Y58 VALVE 2
Y57 Y59 C999
Y58
Y60
Y33 Y34 C999
Y32
GTS001 GO TO SBR 1: INITIALIZATION
X103 C2 C3 C6 Y9
GTS1
Y8 BUSY PERFORMING OPERATION
X102 Y9
Y8
C2 C4009
C2
Y9 DONE OPERATION
X102 C999
Y9
C2
GTS004 GO TO SBR 4: ADDRESS
C2
O/S1
GTS4
Y10 ACKNOWLEDGE COMMAND
C2
O/S15
Y10

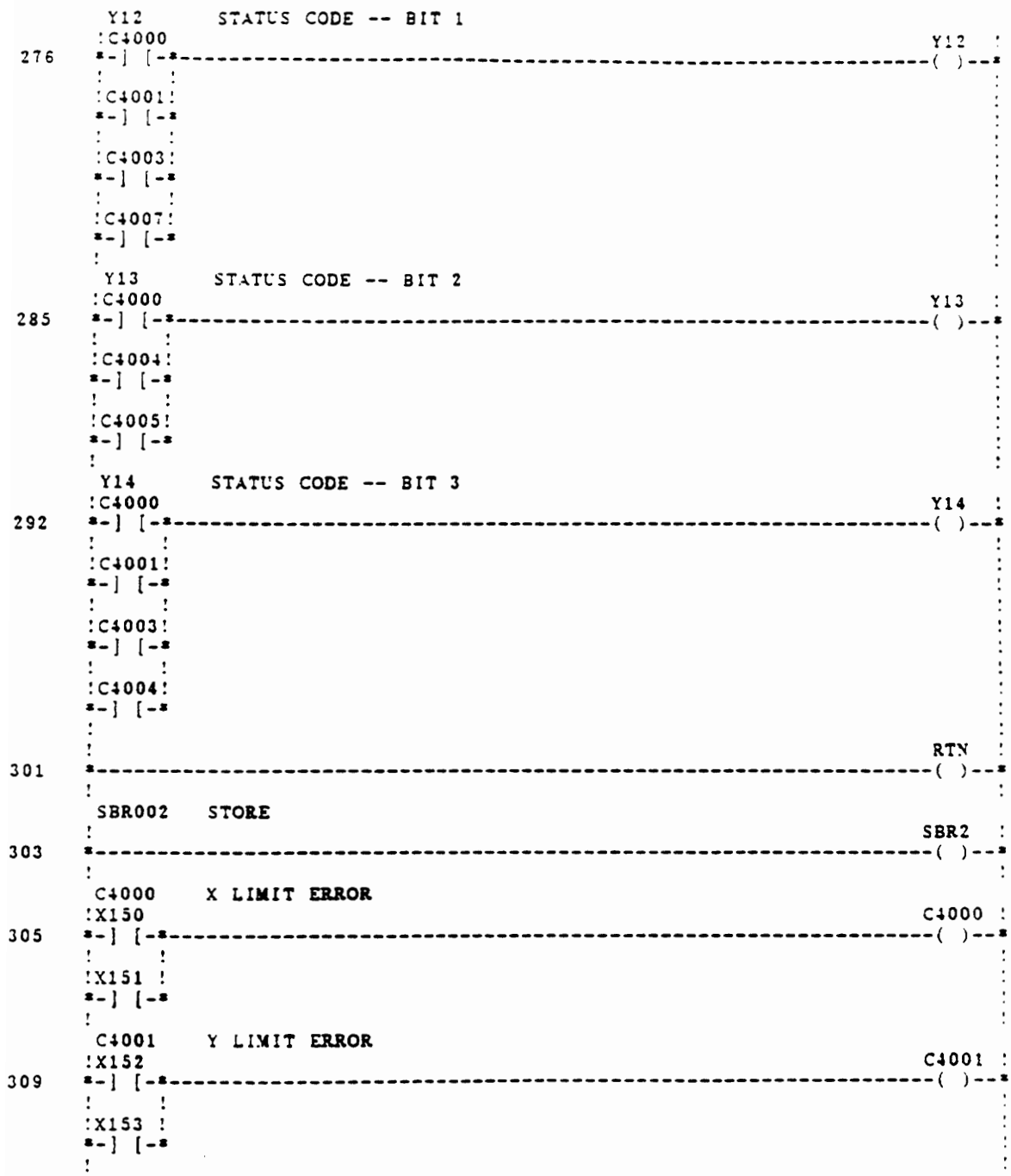

C3
HORIZONTAL ADDRESS = 0
C2
CMP12
A: V120
B: V151
LT= C999
GT= C4
C3
39
C6
VERTICAL ADDRESS = 0
C2
CMP13
A: V121
B: V151
LT= C999
GT= C5
C6
47
C4009
INVALID ADDRESS
C1091
C3
C4009
55
C1092
C3
X103
C1094
C1095
C1100
C1101
C6
C3
SKP012
INVALID ADDRESS -- SKIP OVER COMMANDS
C4009
SKP12
77
GTS002
GO TO SBR 2: STORE
X103
C1
C2
Y9
C4
C5
GTS2
80
GTS003
GO TO SBR 3: RETRIEVE
X103
C1
C2
Y9
C4
C5
GTS3
88

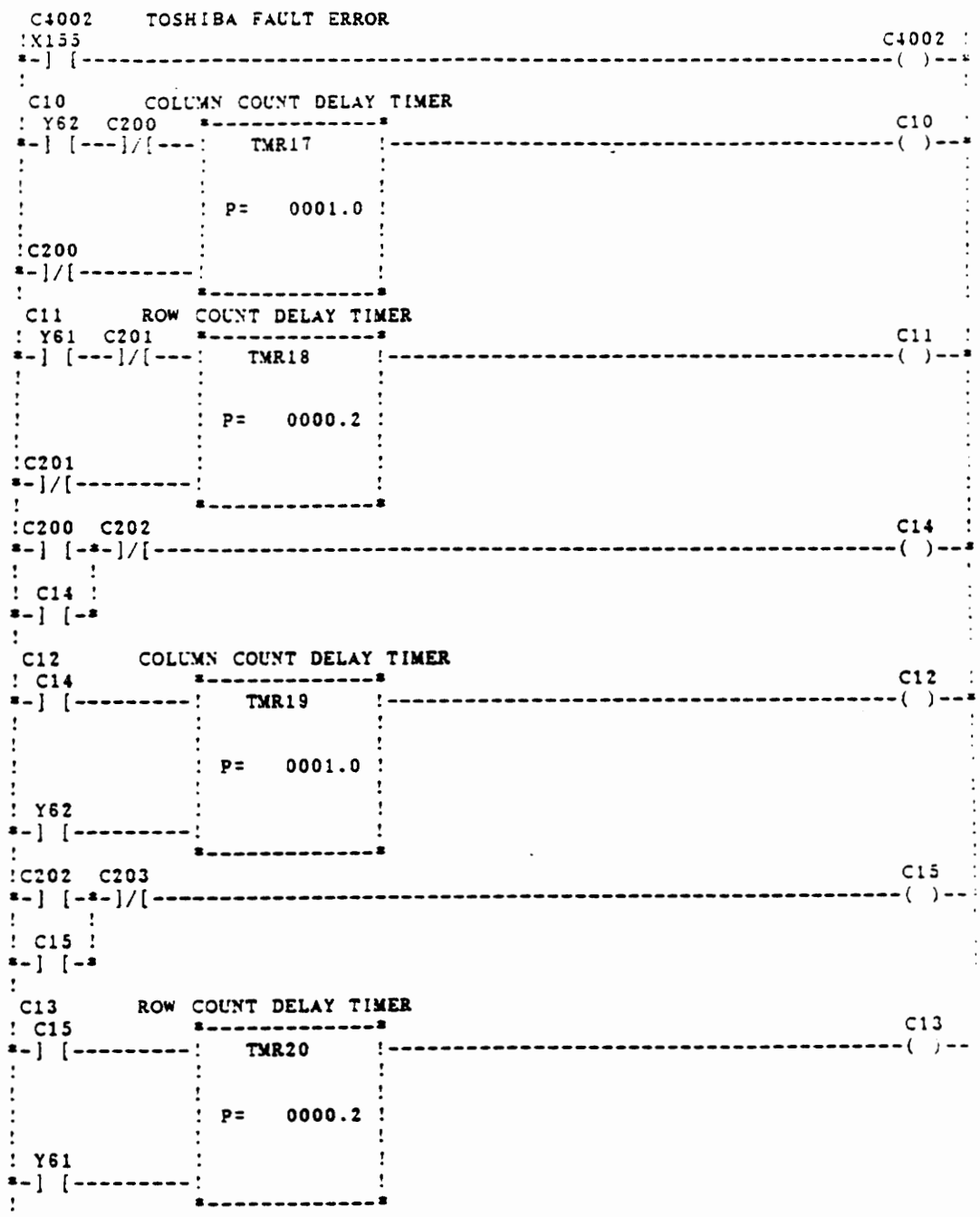
```
       GTS006   GO TO SBR 6: MOVE TO ROW
       : C1    X103    C2     C3     C5                                  GTS6  :
96     *-] [---] [---] [---] [---] [-----------------------------------( )--*
       :                                                                       :
       GTS007   GO TO SBR 7: MOVE TO COLUMN
       : C1    X103    C2     C3     C5                                  GTS7  :
103    *-] [---]/[---] [---] [---] [-----------------------------------( )--*
       :                                                                       :
       GTS008   GO TO SBR 8: SHUTDOWN
       : C1    X103    C2     C3     C6                                  GTS8  :
110    *-] [---] [---] [---] [---] [-----------------------------------( )--*
       :                                                                       :
       :                                                                 LBL12 :
117    *-----------------------------------------------------------------( )--*
       :                                                                       :
       Y14      STATUS CODE -- BIT 3
       :C4009                                                             Y14  :
118    *-] [-----------------------------------------------------------( )--*
       :                                                                       :
       END      END OF MAIN PROGRAM
       :                                                                  END  :
121    *-----------------------------------------------------------------( )--*
       :                                                                       :
       SBR001   INITIALIZATION
       :                                                                 SBR1  :
122    *-----------------------------------------------------------------( )--*
       :                                                                       :
       Y8       BUSY PERFORMING OPERATION
       :C999                                                              Y8   :
124    *-]/[-----------------------------------------------------------( )--*
       :                                                                       :
       C4000    X LIMIT ERROR
       :X150                                                            C4000 :
127    *-] [-*-------------------------------------------------------( )--*
       :     :                                                                 :
       :X151 :                                                                 :
       *-] [-*                                                                 :
       :                                                                       :
       C4001    Y LIMIT ERROR
       :X152                                                            C4001 :
131    *-] [-*-------------------------------------------------------( )--*
       :     :                                                                 :
       :X153 :                                                                 :
       *-] [-*                                                                 :
       :                                                                       :
       C4002    TOSHIBA FAULT ERROR
       :X155                                                            C4002 :
135    *-] [-----------------------------------------------------------( )--*
       :                                                                       :
```


Y56 TOSHIBA INTERLOCK (ON/OFF)
X119 C999 Y56
138
Y61
Y31
Y61 Y31 C999 Y33
144
C4004 VERTICAL MOTOR TIMER ERROR
Y61 Y31 TMR1 C4004
149
P= 0180.0
X119
END(C) END SCAN IF CONDITION MET
X119 C999 END
156
Y56 TOSHIBA INTERLOCK (ON/OFF)
X119 X118 C999 Y56
160
Y62
Y31
Y62 Y31 C999 Y33
167
C4005 HORIZONTAL MOTOR TIMER ERROR
Y62 Y31 TMR2 C4005
172
P= 0300.0
X118
C4007 CRANE IN MOTION
Y62 C4007
179
Y61

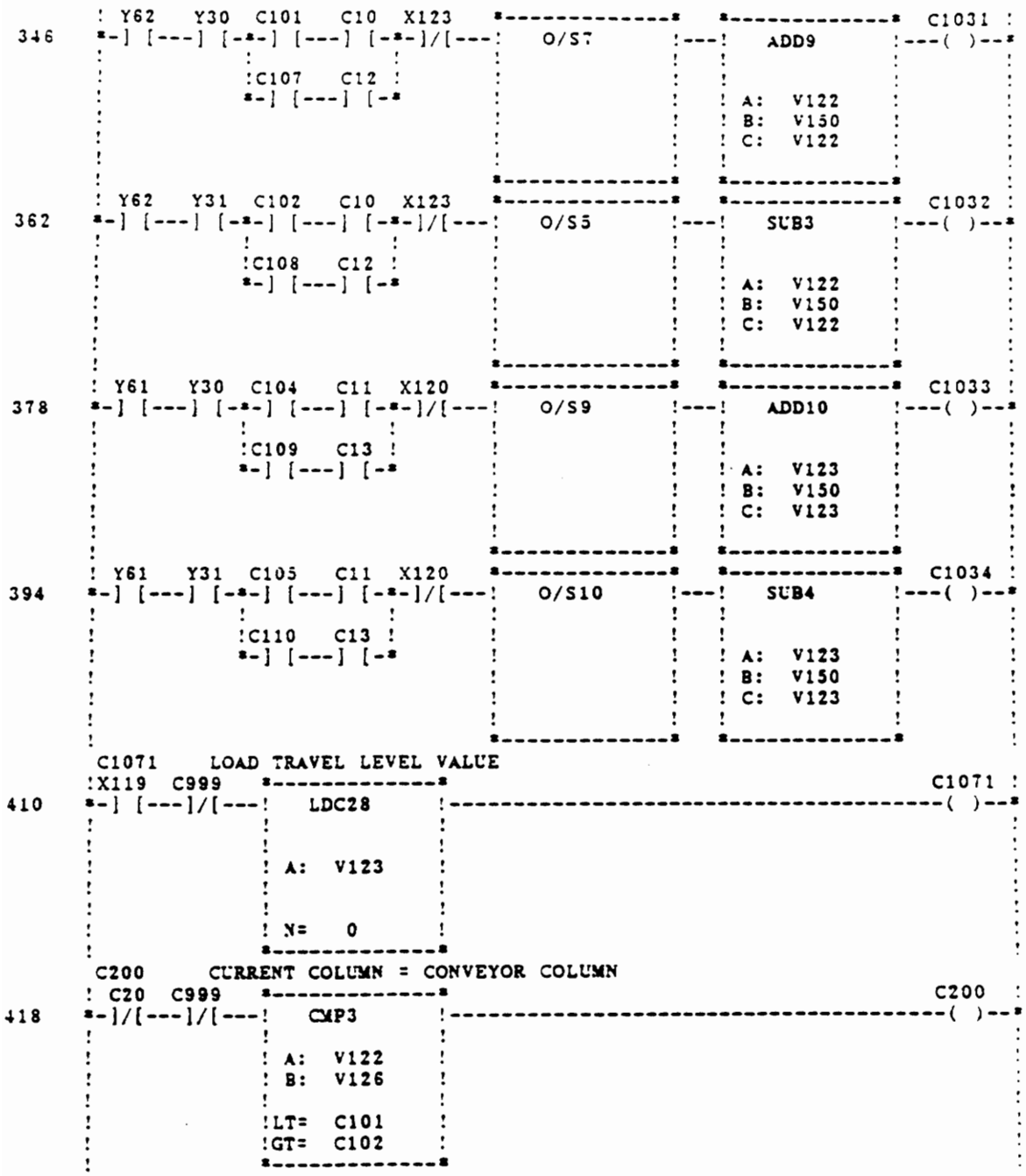

END(C) END SCAN IF CONDITION MET
X118
183
END
(C)
SKP009 IF FIRST INITIALIZATION DONE, SKIP OVER NEXT RUNG
C1
185
SKP9
C1 FIRST INITIALIZATION COMPLETE
X118 X119
187
C1
LBL9
190
C4003 AT HOME POSITION
X119 X118
191
C4003
C1069 LOAD INTO MEMORY HOME POSITION
X118 X119 C999
195
C1069
LDC17
A: V122
N= 1
LDC18
A: V123
N= 0
X118 X119 C999
207
C1025
LDC19
A: V150
N= 1
X118 X119 C999
216
C1026
LDC22
A: V151
N= 0
C1028 CONVEYOR ROW = 1
X118 X119 C999
225
C1028
LDC24
A: V125
N= 1

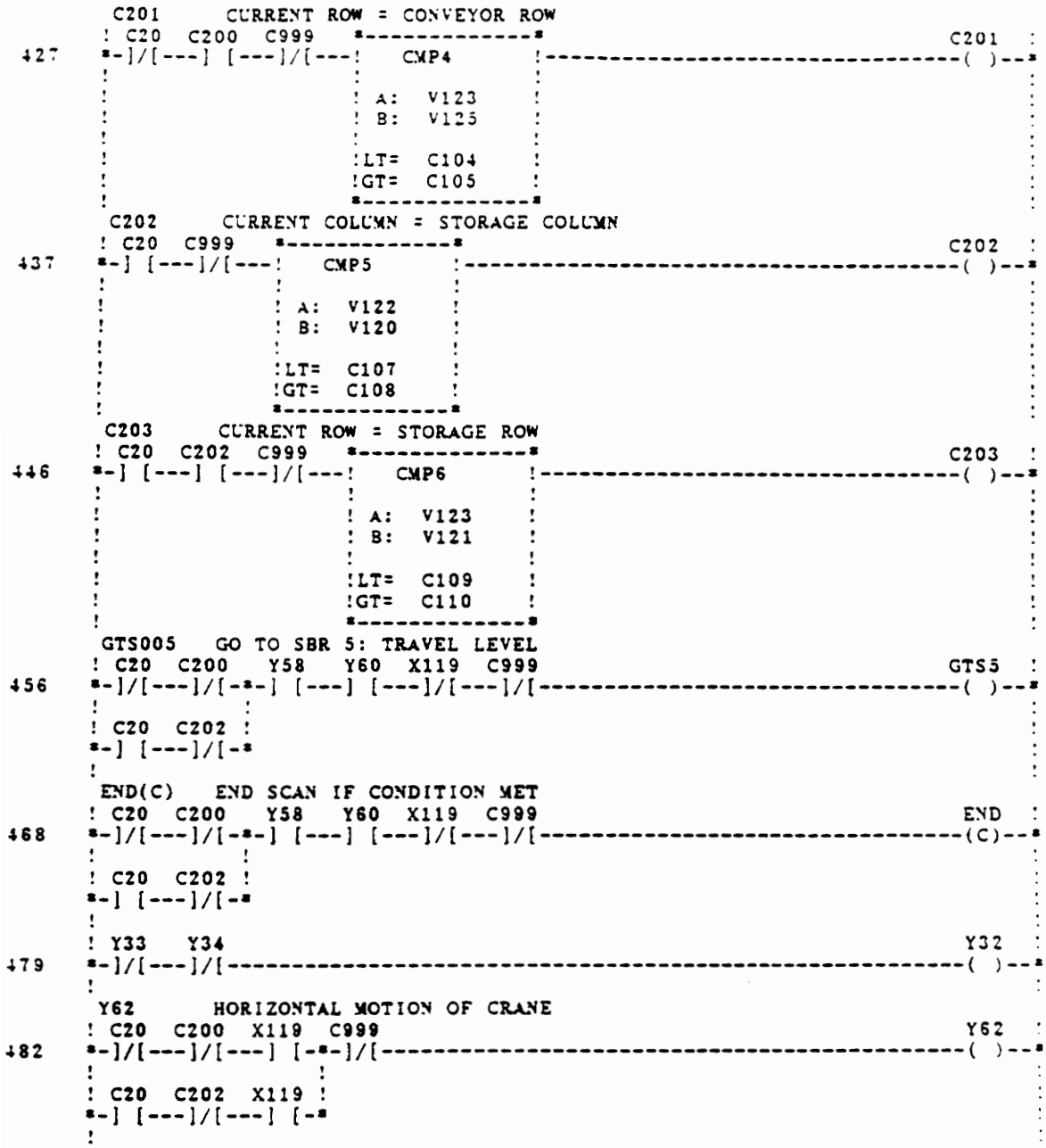

C1029 PICKUP CONVEYOR COLUMN = 14
C1028 C999
234
LDC25
A: V126
N= 14
C1029
C1030 DELIVERY CONVEYOR = 12
C1029 C999
243
LDC26
A: V127
N= 12
C1030
Y8 BUSY PERFORMING OPERATION
C2 C999
252
Y8
Y9
C999 ERROR -- STOP MOTION
C4000
257
C4001
C4002
C4004
C4005
C999
Y11 STATUS CODE -- BIT 0
C4000
269
C4001
C4002
Y11


Y12 STATUS CODE -- BIT 1
C4000
Y12
276
C4001
C4003
C4007
Y13 STATUS CODE -- BIT 2
C4000
Y13
285
C4004
C4005
Y14 STATUS CODE -- BIT 3
C4000
Y14
292
C4001
C4003
C4004
RTN
301
SBR002 STORE
SBR2
303
C4000 X LIMIT ERROR
X150
C4000
305
X151
C4001 Y LIMIT ERROR
X152
C4001
309
X153

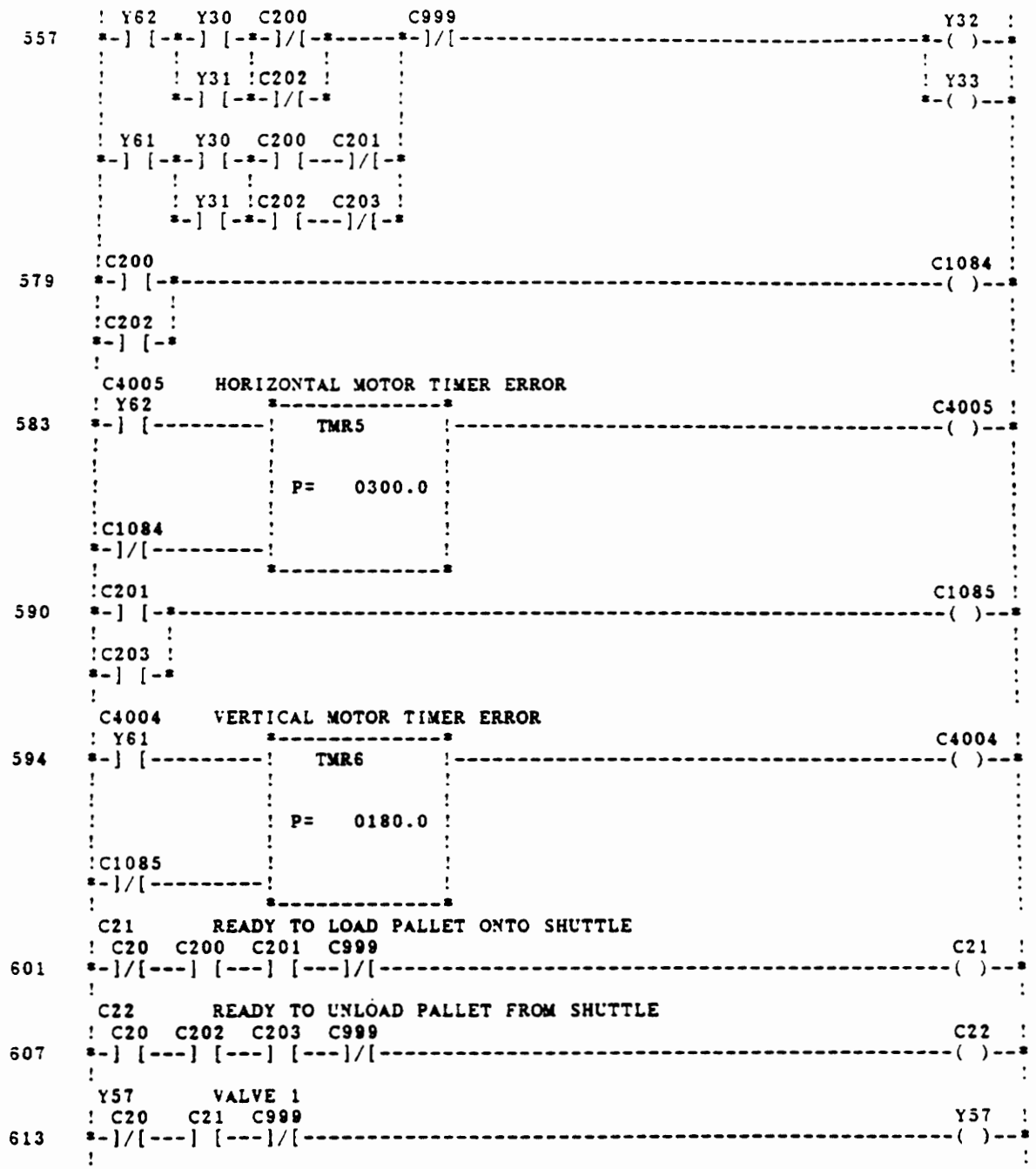

C4002 TOSHIBA FAULT ERROR
X155
C4002
313
C10 COLUMN COUNT DELAY TIMER
Y62 C200
TMR17
C10
316
P= 0001.0
C200
C11 ROW COUNT DELAY TIMER
Y61 C201
TMR18
C11
322
P= 0000.2
C201
C200 C202
C14
328
C14
C12 COLUMN COUNT DELAY TIMER
C14
TMR19
C12
332
P= 0001.0
Y62
C202 C203
C15
337
C15
C13 ROW COUNT DELAY TIMER
C15
TMR20
C13
341
P= 0000.2
Y61

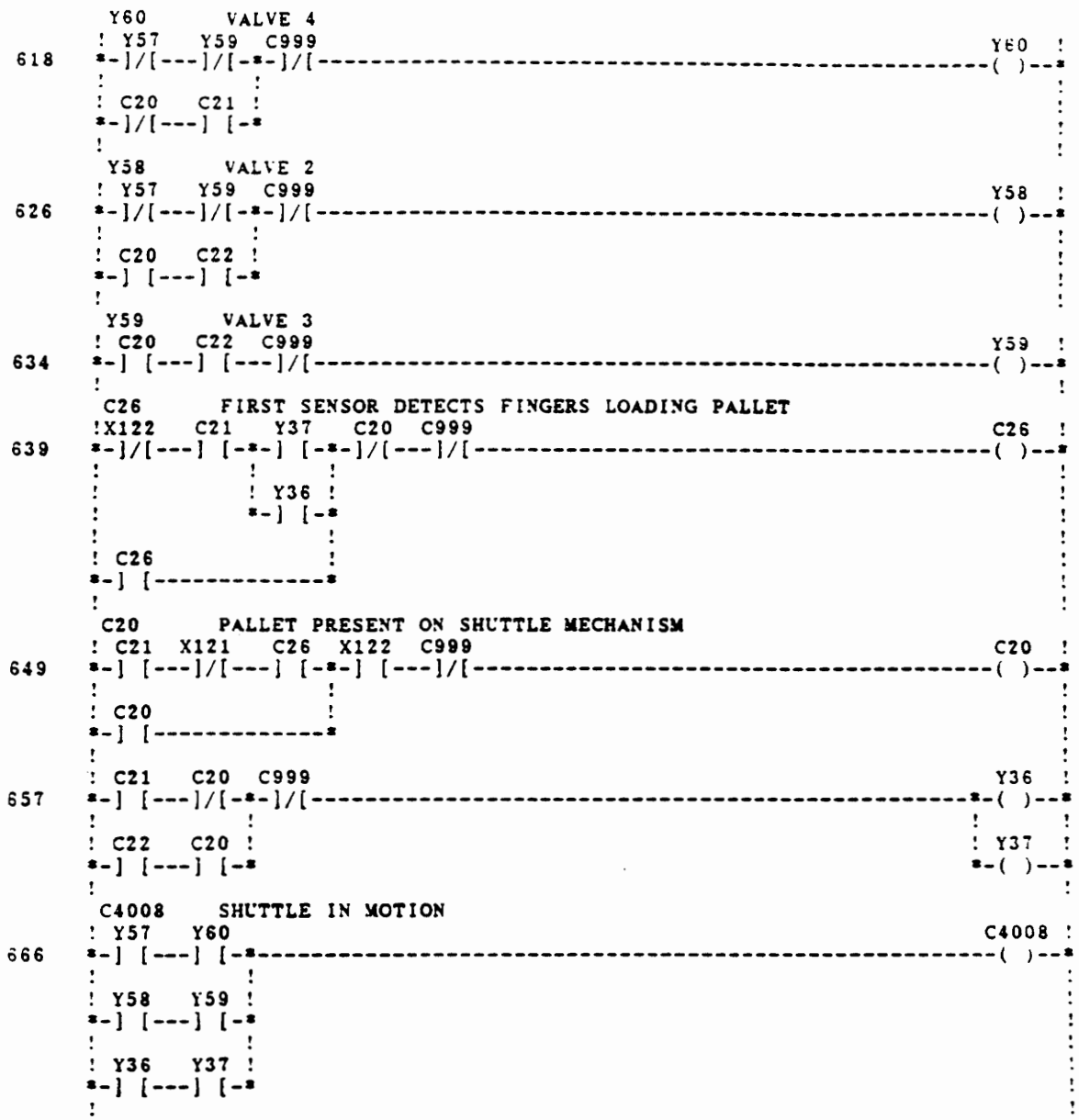
DATE= 00-00-00 OFF LINE

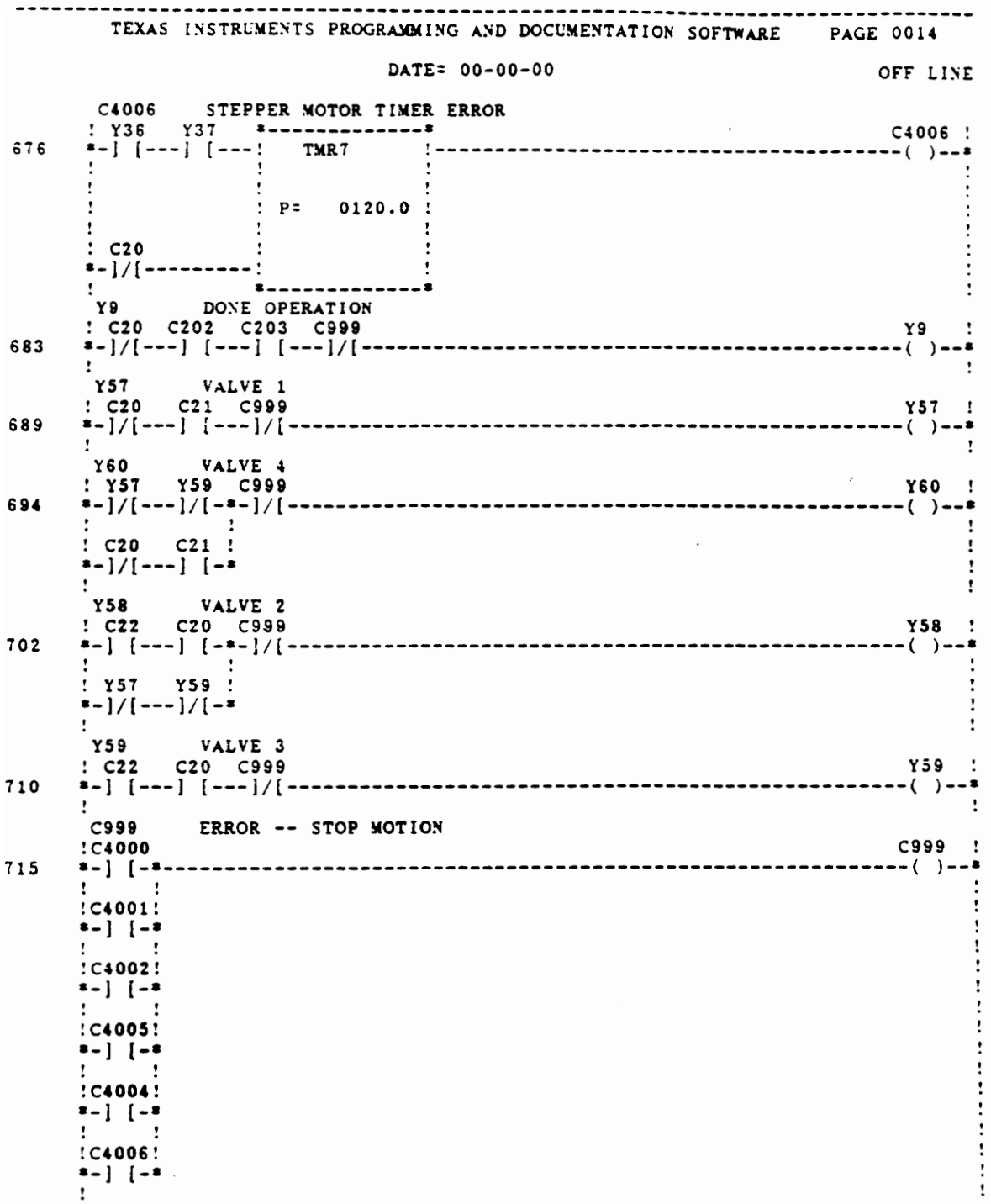



C201 CURRENT ROW = CONVEYOR ROW
C20 C200 C999
427
CMP4
A: V123
B: V125
LT= C104
GT= C105
C201
C202 CURRENT COLUMN = STORAGE COLUMN
C20 C999
437
CMP5
A: V122
B: V120
LT= C107
GT= C108
C202
C203 CURRENT ROW = STORAGE ROW
C20 C202 C999
446
CMP6
A: V123
B: V121
LT= C109
GT= C110
C203
GTS005 GO TO SBR 5: TRAVEL LEVEL
C20 C200 Y58 Y60 X119 C999
456
C20 C202
GTS5
END(C) END SCAN IF CONDITION MET
C20 C200 Y58 Y60 X119 C999
468
C20 C202
END
Y33 Y34
479
Y32
Y62 HORIZONTAL MOTION OF CRANE
C20 C200 X119 C999
482
C20 C202 X119
Y62

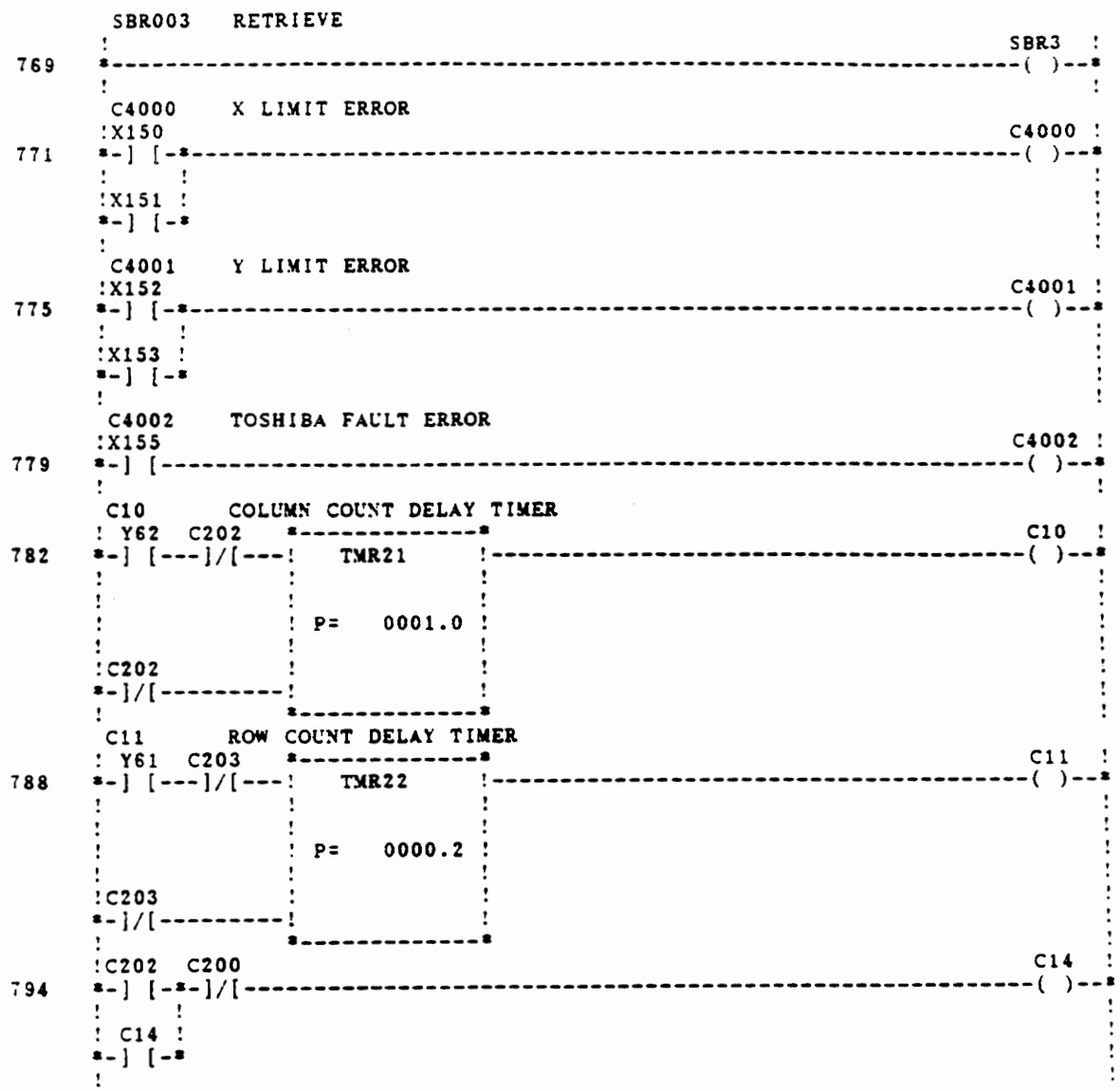
```
      Y61       VERTICAL MOTION OF SHUTTLE
      ! C20  C200  C201  C999                                                Y61  !
492   *-]/[---] [---]/[-*-]/[-----------------------------------------------( )--*
      !                   !                                                      !
      ! C20  C202  C203 !                                                       !
      *-] [---] [---]/[-*                                                      !
      !                                                                        !
      C4007     CRANE IN MOTION
      ! Y62                                                                C4007 !
502   *-] [-*-----------------------------------------------------------------( )--*
      !     !                                                                  !
      ! Y61 !                                                                  !
      *-] [-*                                                                  !
      !                                                                        !
      Y56       TOSHIBA INTERLOCK (ON/OFF)
      ! Y61  C999                                                          Y56  !
506   *-] [-*-]/[-----------------------------------------------------------( )--*
      !     !                                                                  !
      ! Y62 !                                                                  !
      *-] [-*                                                                  !
      !                                                                        !
      Y31       REVERSE DIRECTION
      ! C20  C200  C102  X119   Y62  C999                                  Y31  !
511   *-]/[---]/[---] [-*-] [---] [-*-]/[-----------------------------------( )--*
      !                   !            !                                       !
      ! C20  C202  C108 !              !                                       !
      *-] [---]/[---] [-*              !                                       !
      !                                !                                       !
      ! C20  C200  C201  C105   Y61 !                                          !
      *-]/[---] [---]/[---] [-*-] [-*                                          !
      !                      !                                                 !
      ! C20  C202  C203  C110 !                                                !
      *-] [---] [---]/[---] [-*                                                !
      !                                                                        !
      Y30       FORWARD DIRECTION
      ! C20  C200  C101  X119   Y62  C999                                  Y30  !
534   *-]/[---]/[---] [-*-] [---] [-*-]/[-----------------------------------( )--*
      !                   !            !                                       !
      ! C20  C202  C107 !              !                                       !
      *-] [---]/[---] [-*              !                                       !
      !                                !                                       !
      ! C20  C200  C201  C104   Y61 !                                          !
      *-]/[---] [---]/[---] [-*-] [-*                                          !
      !                      !                                                 !
      ! C20  C202  C203  C109 !                                                !
      *-] [---] [---]/[---] [-*                                                !
      !                                                                        !
```

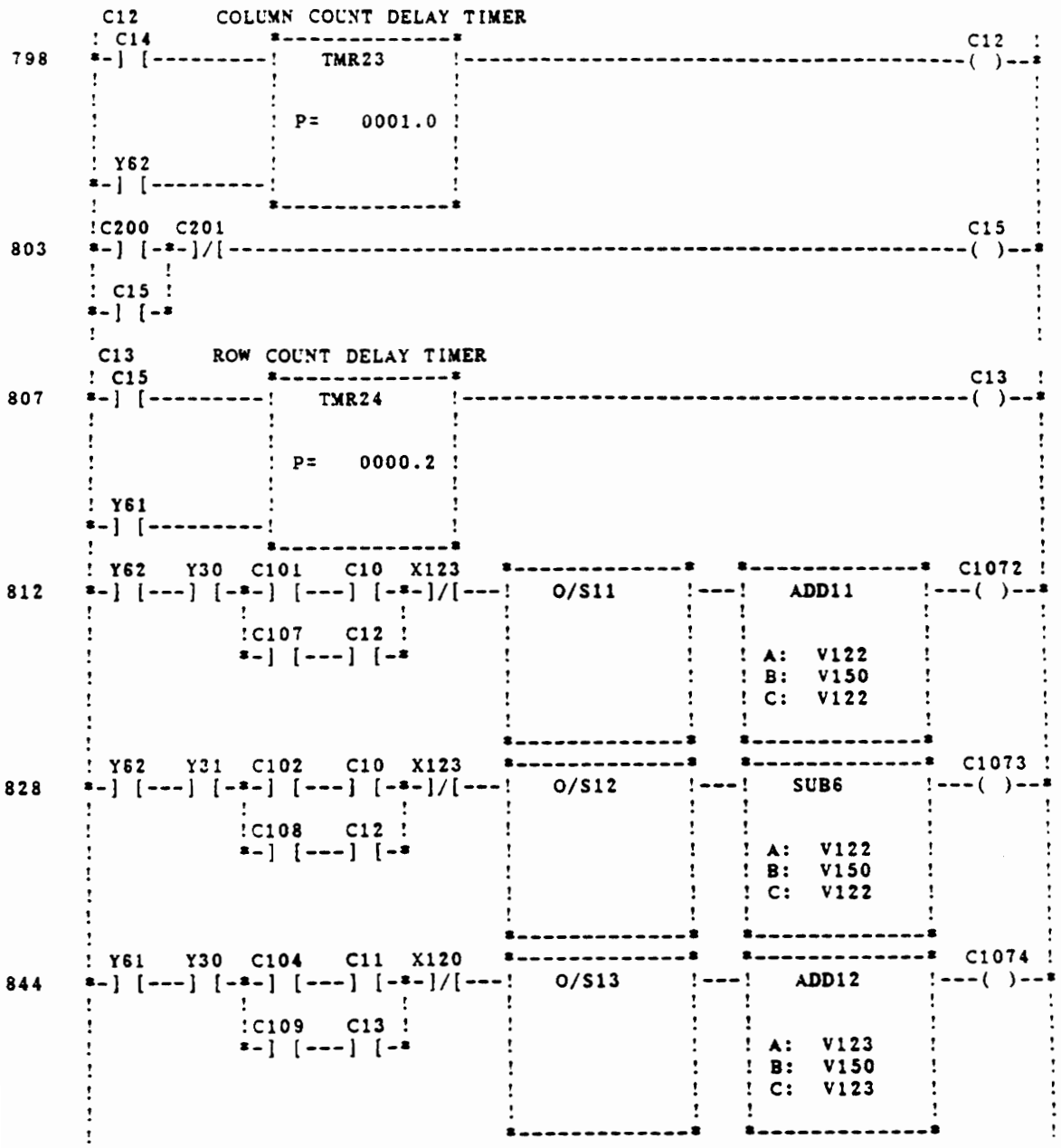

Y62 Y30 C200 C999 Y32
557
Y31 C202 Y33
Y61 Y30 C200 C201
Y31 C202 C203
C200 C1084
579
C202
C4005 HORIZONTAL MOTOR TIMER ERROR
Y62 C4005
583
TMR5
P= 0300.0
C1084
C201 C1085
590
C203
C4004 VERTICAL MOTOR TIMER ERROR
Y61 C4004
594
TMR6
P= 0180.0
C1085
C21 READY TO LOAD PALLET ONTO SHUTTLE
C20 C200 C201 C999 C21
601
C22 READY TO UNLOAD PALLET FROM SHUTTLE
C20 C202 C203 C999 C22
607
Y57 VALVE 1
C20 C21 C999 Y57
613

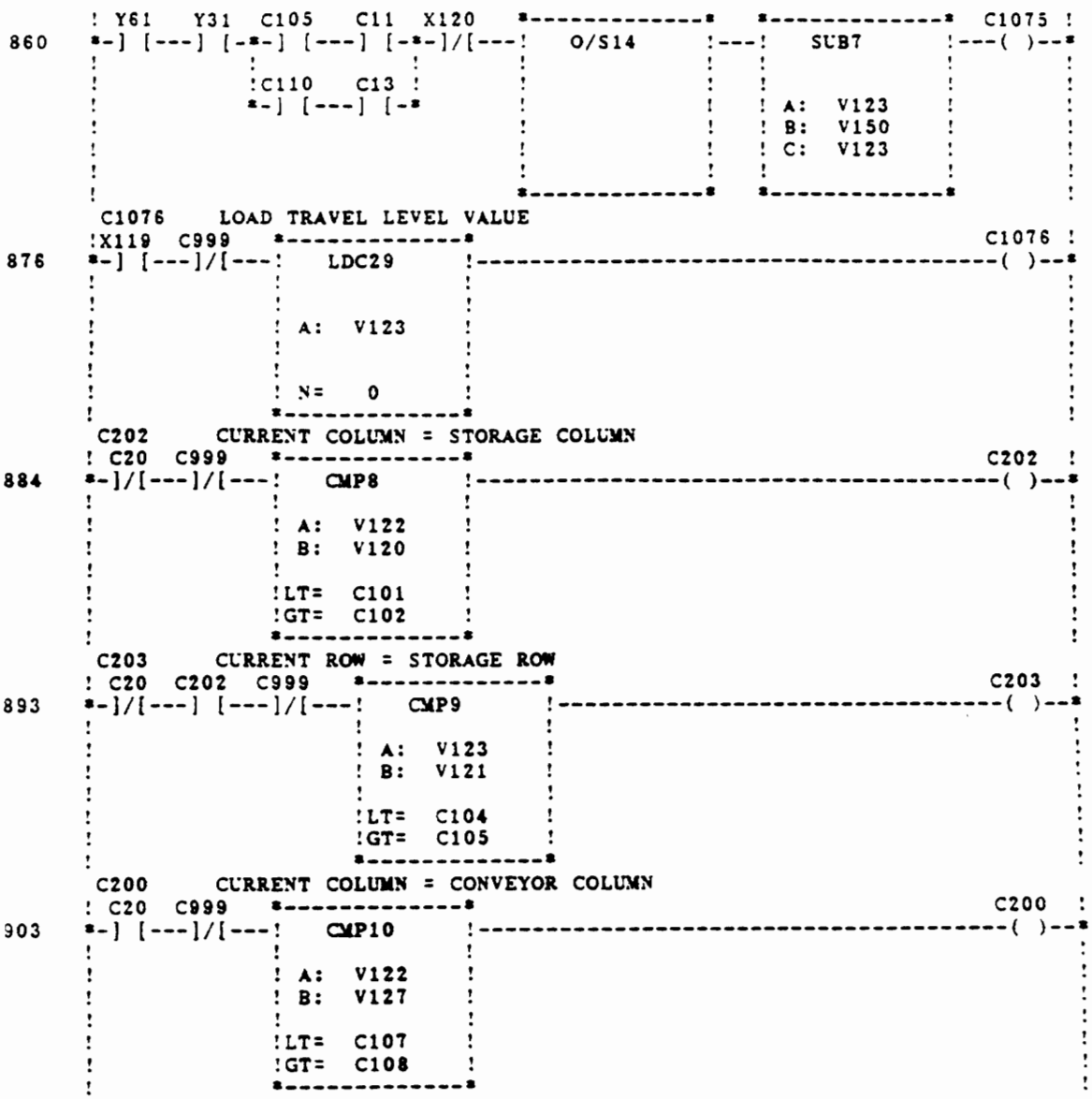

Y60 VALVE 4
Y57 Y59 C999 Y60
618
C20 C21
Y58 VALVE 2
Y57 Y59 C999 Y58
626
C20 C22
Y59 VALVE 3
C20 C22 C999 Y59
634
C26 FIRST SENSOR DETECTS FINGERS LOADING PALLET
X122 C21 Y37 C20 C999 C26
639
Y36
C26
C20 PALLET PRESENT ON SHUTTLE MECHANISM
C21 X121 C26 X122 C999 C20
649
C20
C21 C20 C999 Y36
657
C22 C20 Y37
C4008 SHUTTLE IN MOTION
Y57 Y60 C4008
666
Y58 Y59
Y36 Y37

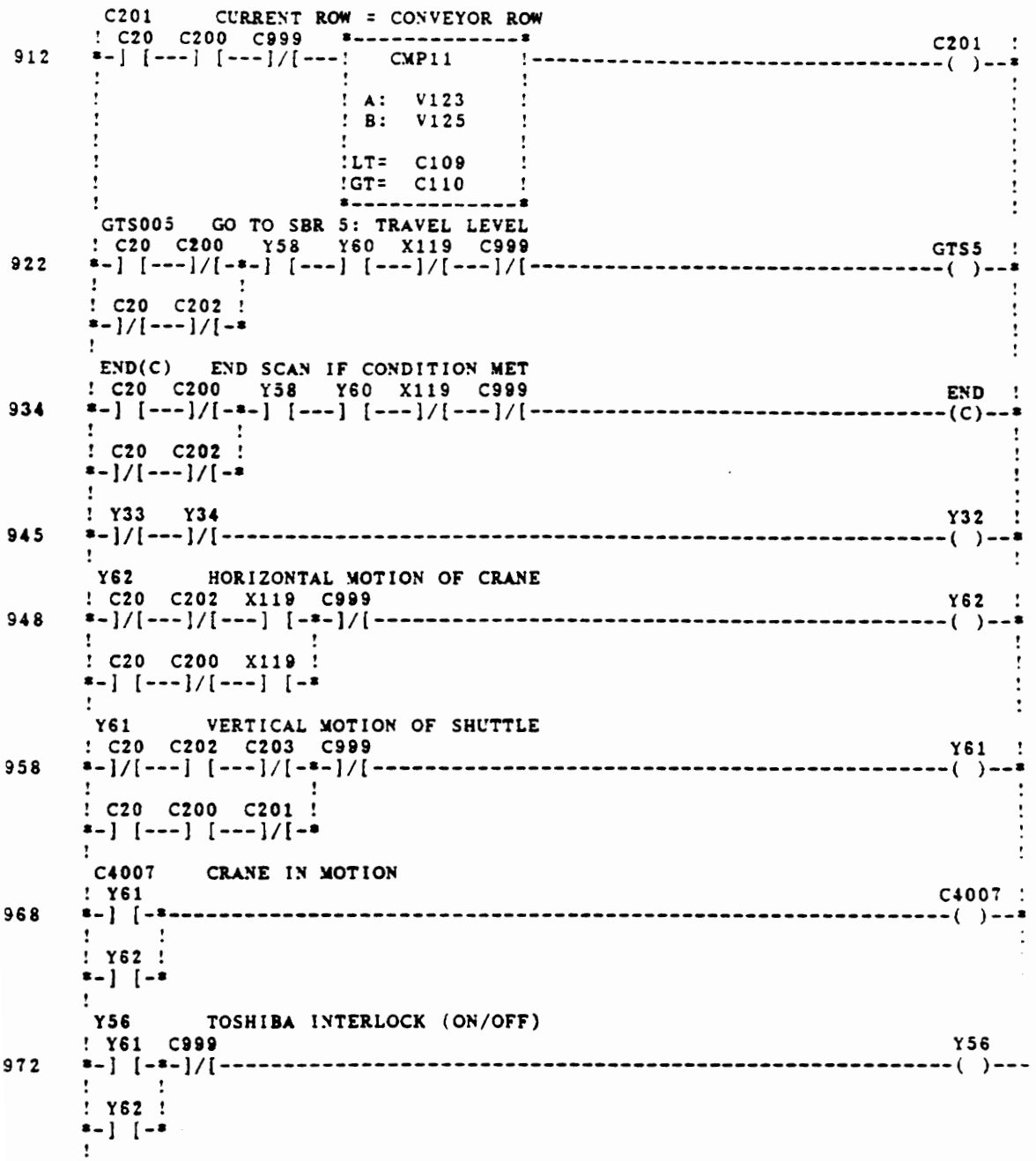

C4006 STEPPER MOTOR TIMER ERROR
Y36 Y37 TMR7 C4006
676
P= 0120.0
C20
Y9 DONE OPERATION
C20 C202 C203 C999 Y9
683
Y57 VALVE 1
C20 C21 C999 Y57
689
Y60 VALVE 4
Y57 Y59 C999 Y60
694
C20 C21
Y58 VALVE 2
C22 C20 C999 Y58
702
Y57 Y59
Y59 VALVE 3
C22 C20 C999 Y59
710
C999 ERROR -- STOP MOTION
C4000 C999
715
C4001
C4002
C4005
C4004
C4006

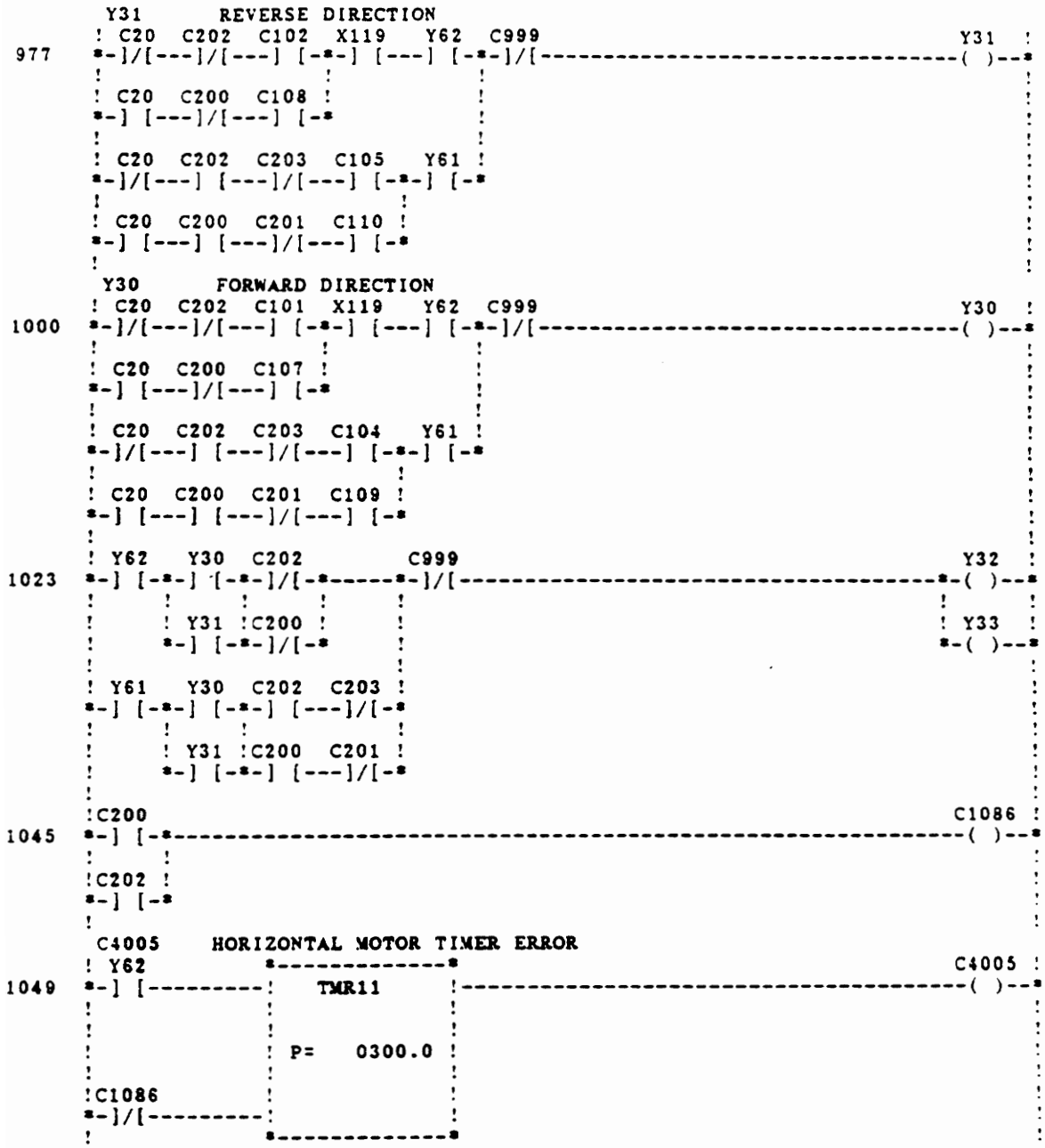
```
-----------------------------------------------------------------------------
        TEXAS INSTRUMENTS PROGRAMMING AND DOCUMENTATION SOFTWARE    PAGE 0015

                                DATE= 00-00-00                          OFF LINE

       Y11       STATUS CODE -- BIT 0
      !C4000                                                          Y11  !
729   *-] [-*-------------------------------------------------------( )--*
      !     !                                                              !
      !C4001!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4002!                                                              !
      *-] [-*                                                              !
      !                                                                    !
       Y12       STATUS CODE -- BIT 1
      !C4000                                                          Y12  !
736   *-] [-*-------------------------------------------------------( )--*
      !     !                                                              !
      !C4001!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4007!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4008!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4006!                                                              !
      *-] [-*                                                              !
      !                                                                    !
       Y13       STATUS CODE -- BIT 2
      !C4000                                                          Y13  !
747   *-] [-*-------------------------------------------------------( )--*
      !     !                                                              !
      !C4005!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4004!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4008!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4006!                                                              !
      *-] [-*                                                              !
      !                                                                    !
       Y14       STATUS CODE -- BIT 3
      !C4000                                                          Y14  !
758   *-] [-*-------------------------------------------------------( )--*
      !     !                                                              !
      !C4001!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4004!                                                              !
      *-] [-*                                                              !
      !     !                                                              !
      !C4006!                                                              !
      *-] [-*                                                              !
      !                                                                    !
      !                                                              RTN  !
767   *-------------------------------------------------------------( )--*
      !                                                                    !
```

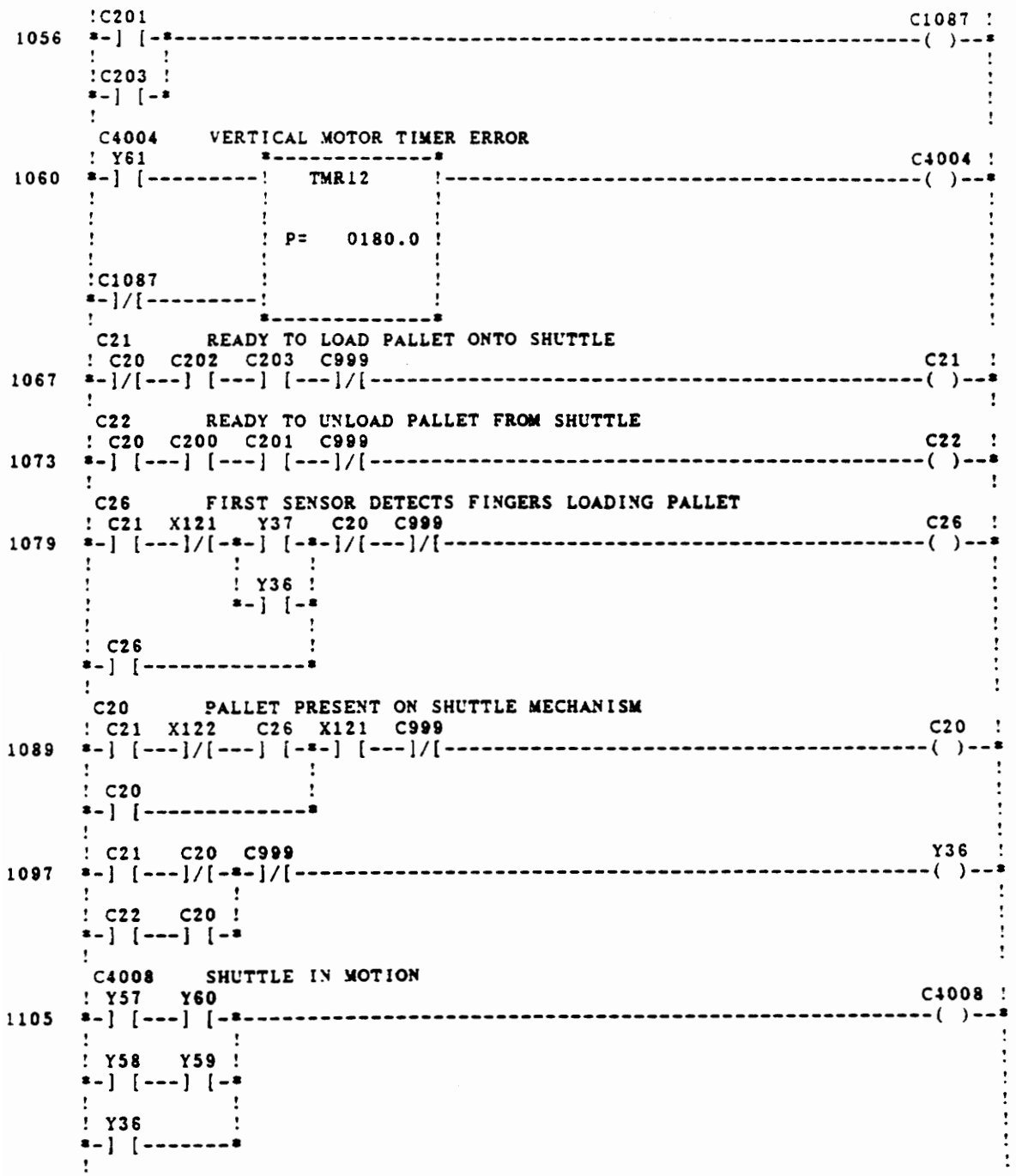

SBR003 RETRIEVE
SBR3
769
C4000 X LIMIT ERROR
X150
C4000
771
X151
C4001 Y LIMIT ERROR
X152
C4001
775
X153
C4002 TOSHIBA FAULT ERROR
X155
C4002
779
C10 COLUMN COUNT DELAY TIMER
Y62 C202
C10
782
TMR21
P= 0001.0
C202
C11 ROW COUNT DELAY TIMER
Y61 C203
C11
788
TMR22
P= 0000.2
C203
C202 C200
C14
794
C14

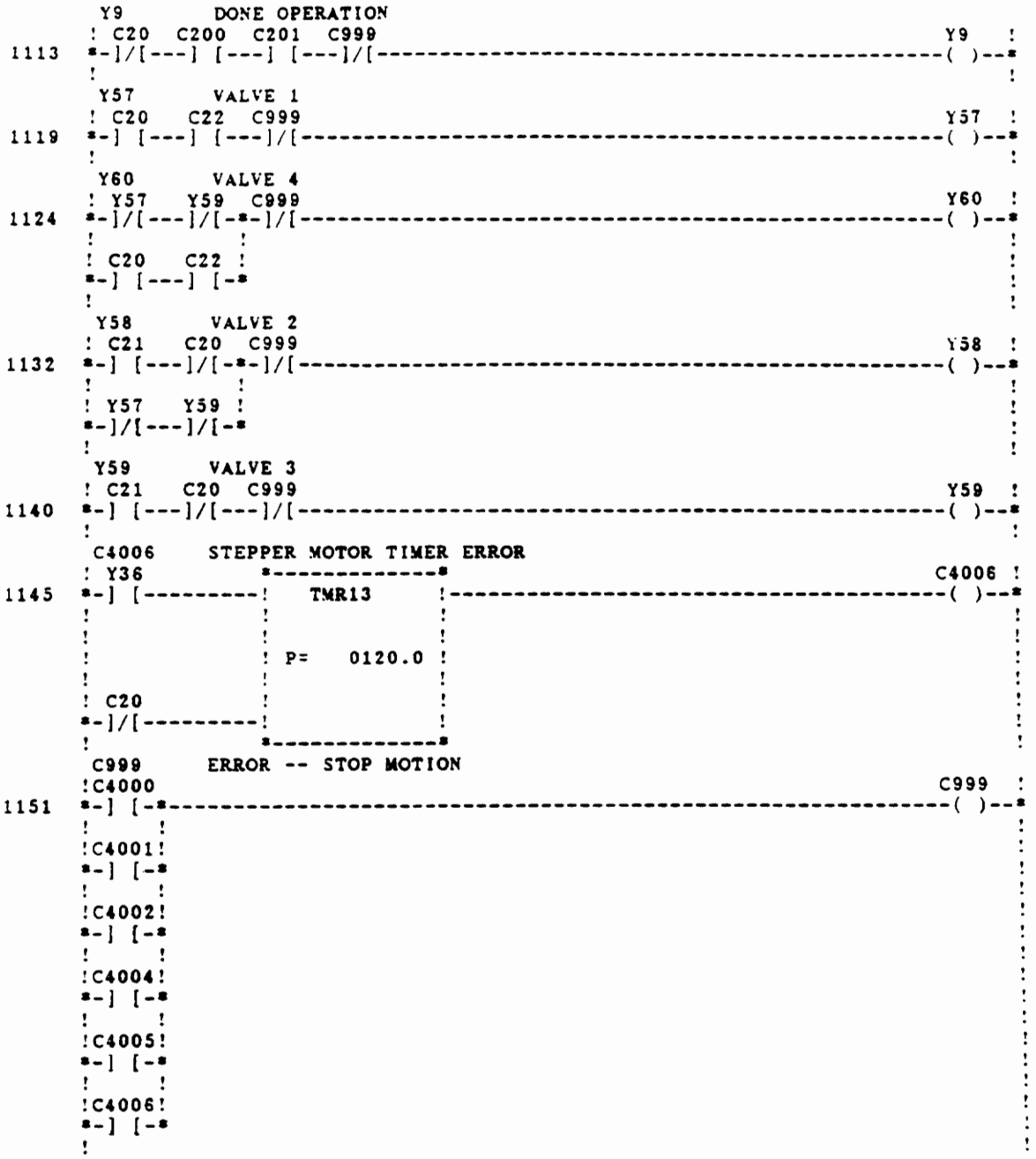

C12 COLUMN COUNT DELAY TIMER
C14
798
TMR23
P= 0001.0
Y62
C12
C200 C201
803
C15
C15
C13 ROW COUNT DELAY TIMER
C15
807
TMR24
P= 0000.2
Y61
C13
Y62 Y30 C101 C10 X123
812
C107 C12
O/S11
ADD11
A: V122
B: V150
C: V122
C1072
Y62 Y31 C102 C10 X123
828
C108 C12
O/S12
SUB6
A: V122
B: V150
C: V122
C1073
Y61 Y30 C104 C11 X120
844
C109 C13
O/S13
ADD12
A: V123
B: V150
C: V123
C1074

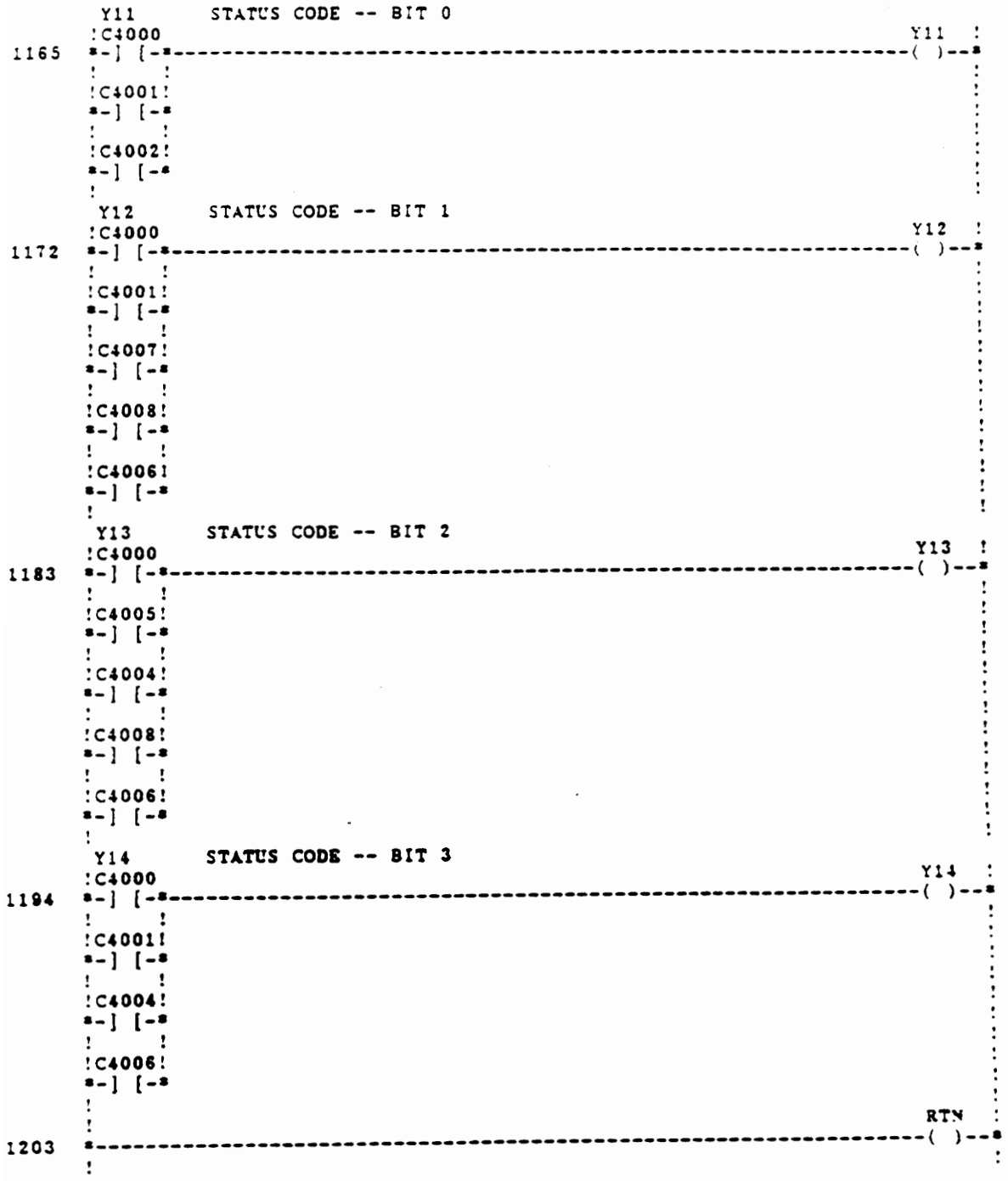

Y61 Y31 C105 C11 X120
C110 C13
O/S14
SUB7
A: V123
B: V150
C: V123
C1075
860
C1076 LOAD TRAVEL LEVEL VALUE
X119 C999
LDC29
A: V123
N= 0
C1076
876
C202 CURRENT COLUMN = STORAGE COLUMN
C20 C999
CMP8
A: V122
B: V120
LT= C101
GT= C102
C202
884
C203 CURRENT ROW = STORAGE ROW
C20 C202 C999
CMP9
A: V123
B: V121
LT= C104
GT= C105
C203
893
C200 CURRENT COLUMN = CONVEYOR COLUMN
C20 C999
CMP10
A: V122
B: V127
LT= C107
GT= C108
C200
903

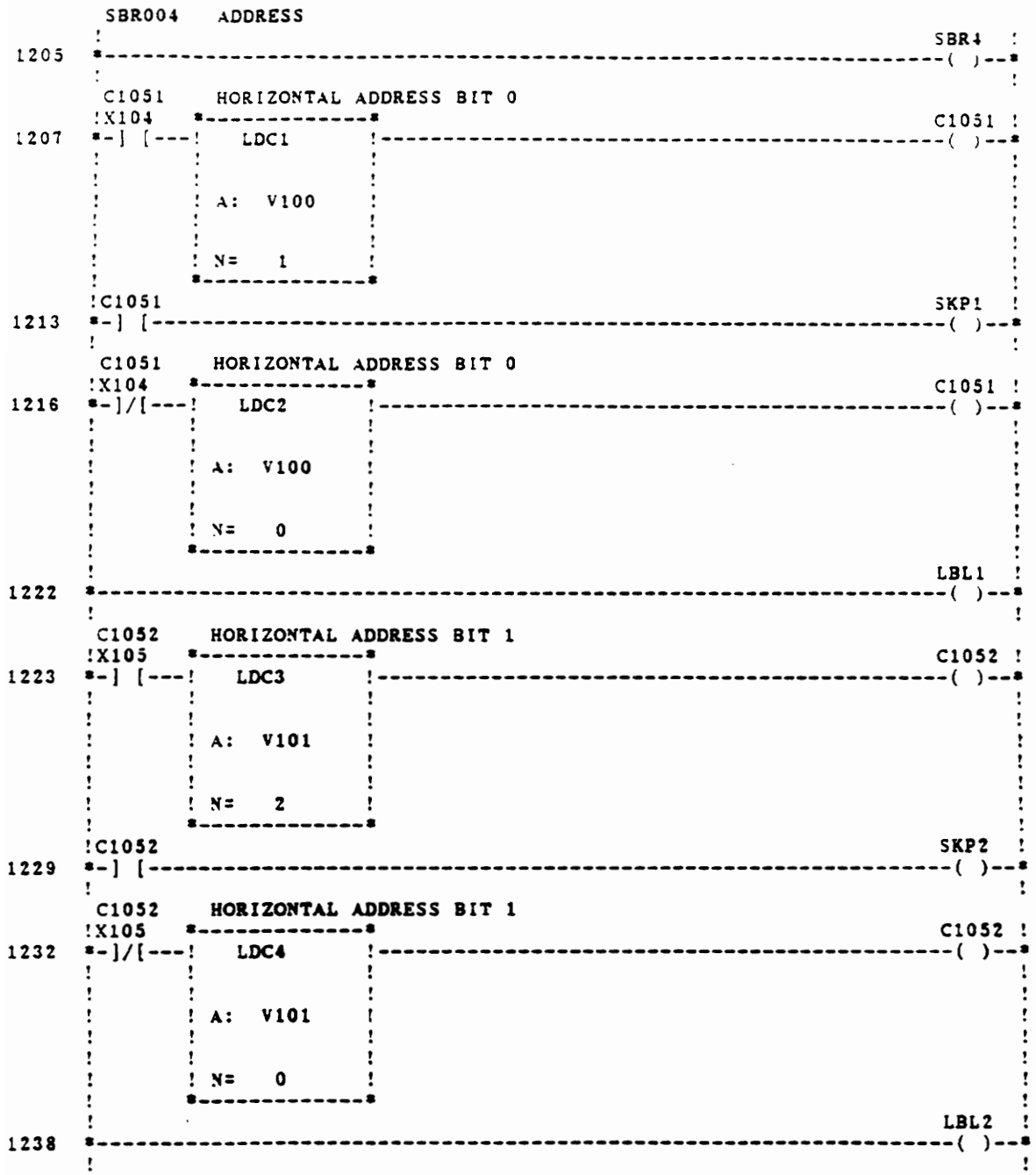

C201 CURRENT ROW = CONVEYOR ROW
912
C20 C200 C999
CMP11
A: V123
B: V125
LT= C109
GT= C110
C201
GTS005 GO TO SBR 5: TRAVEL LEVEL
922
C20 C200 Y58 Y60 X119 C999
C20 C202
GTS5
END(C) END SCAN IF CONDITION MET
934
C20 C200 Y58 Y60 X119 C999
C20 C202
END
945
Y33 Y34
Y32
Y62 HORIZONTAL MOTION OF CRANE
948
C20 C202 X119 C999
C20 C200 X119
Y62
Y61 VERTICAL MOTION OF SHUTTLE
958
C20 C202 C203 C999
C20 C200 C201
Y61
C4007 CRANE IN MOTION
968
Y61
Y62
C4007
Y56 TOSHIBA INTERLOCK (ON/OFF)
972
Y61 C999
Y62
Y56

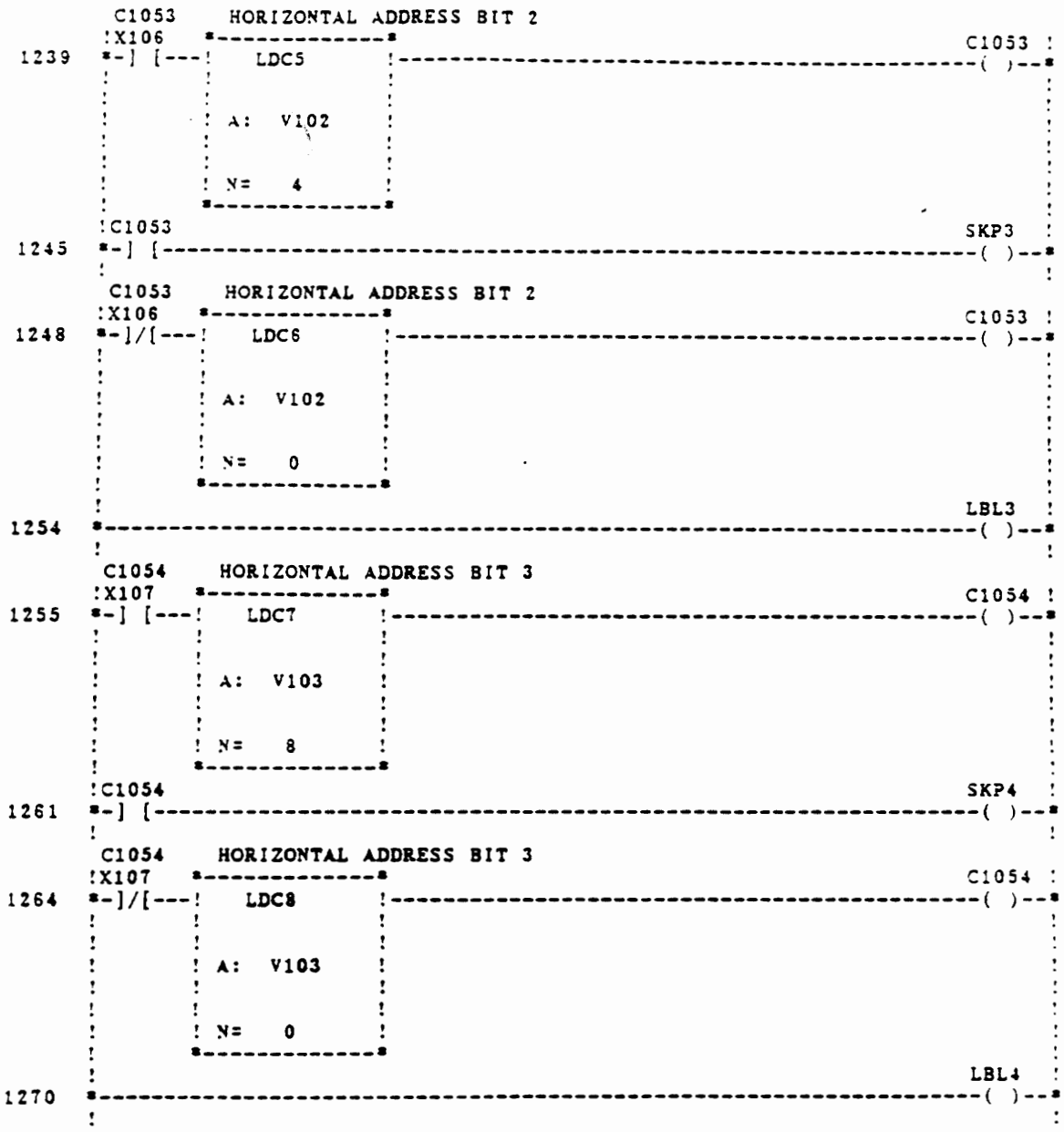

Y31 REVERSE DIRECTION
C20 C202 C102 X119 Y62 C999 Y31
977
C20 C200 C108
C20 C202 C203 C105 Y61
C20 C200 C201 C110
Y30 FORWARD DIRECTION
C20 C202 C101 X119 Y62 C999 Y30
1000
C20 C200 C107
C20 C202 C203 C104 Y61
C20 C200 C201 C109
Y62 Y30 C202 C999 Y32
1023
Y31 C200 Y33
Y61 Y30 C202 C203
Y31 C200 C201
C200 C1086
1045
C202
C4005 HORIZONTAL MOTOR TIMER ERROR
Y62 C4005
1049
TMR11
P= 0300.0
C1086

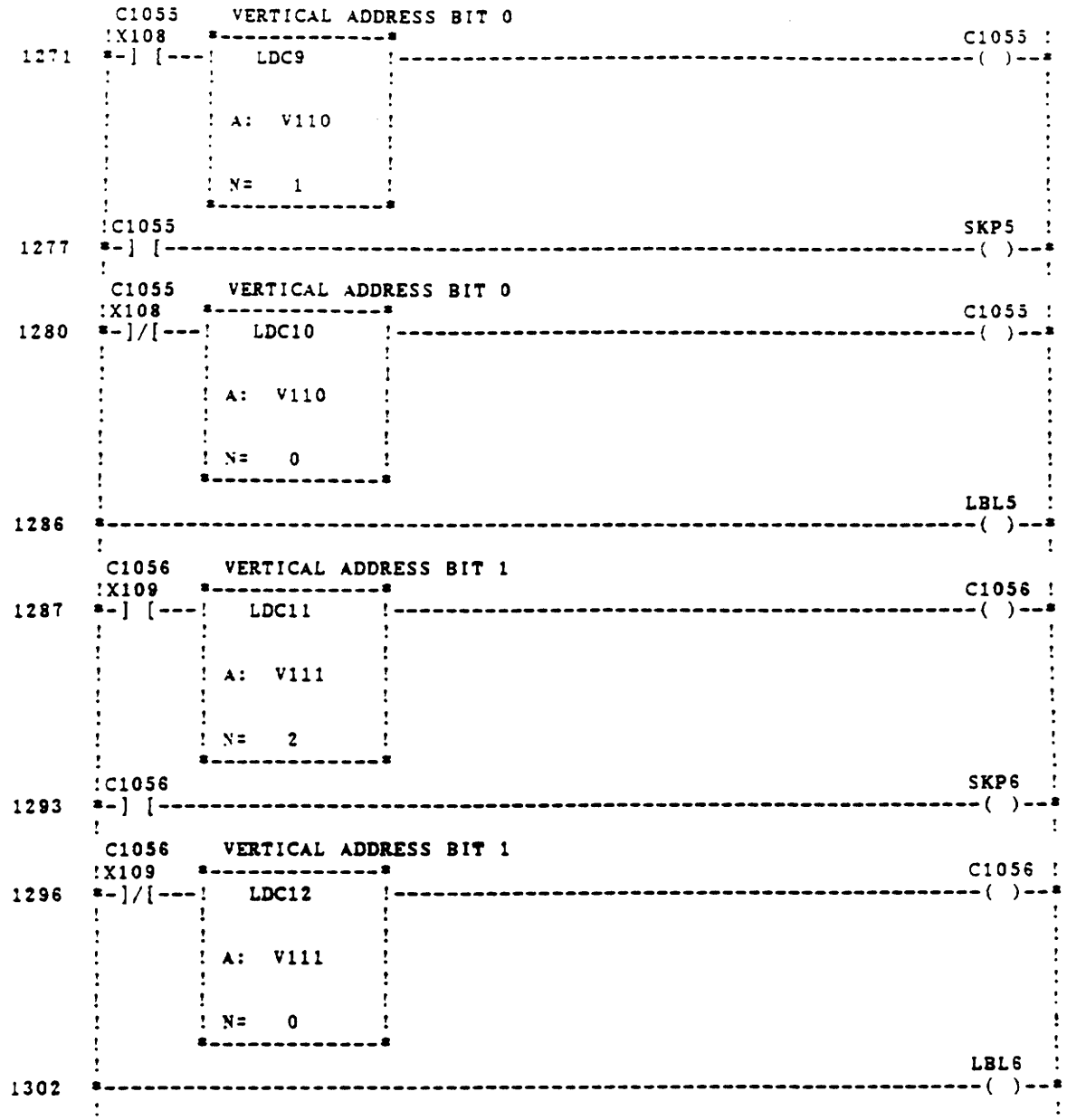

C201
C203
C1087
1056
C4004 VERTICAL MOTOR TIMER ERROR
Y61
TMR12
P= 0180.0
C4004
1060
C21 READY TO LOAD PALLET ONTO SHUTTLE
C20 C202 C203 C999
C21
1067
C22 READY TO UNLOAD PALLET FROM SHUTTLE
C20 C200 C201 C999
C22
1073
C26 FIRST SENSOR DETECTS FINGERS LOADING PALLET
C21 X121 Y37 C20 C999
C26
1079
Y36
C26
C20 PALLET PRESENT ON SHUTTLE MECHANISM
C21 X122 C26 X121 C999
C20
1089
C20
C21 C20 C999
Y36
1097
C22 C20
C4008 SHUTTLE IN MOTION
Y57 Y60
C4008
1105
Y58 Y59
Y36

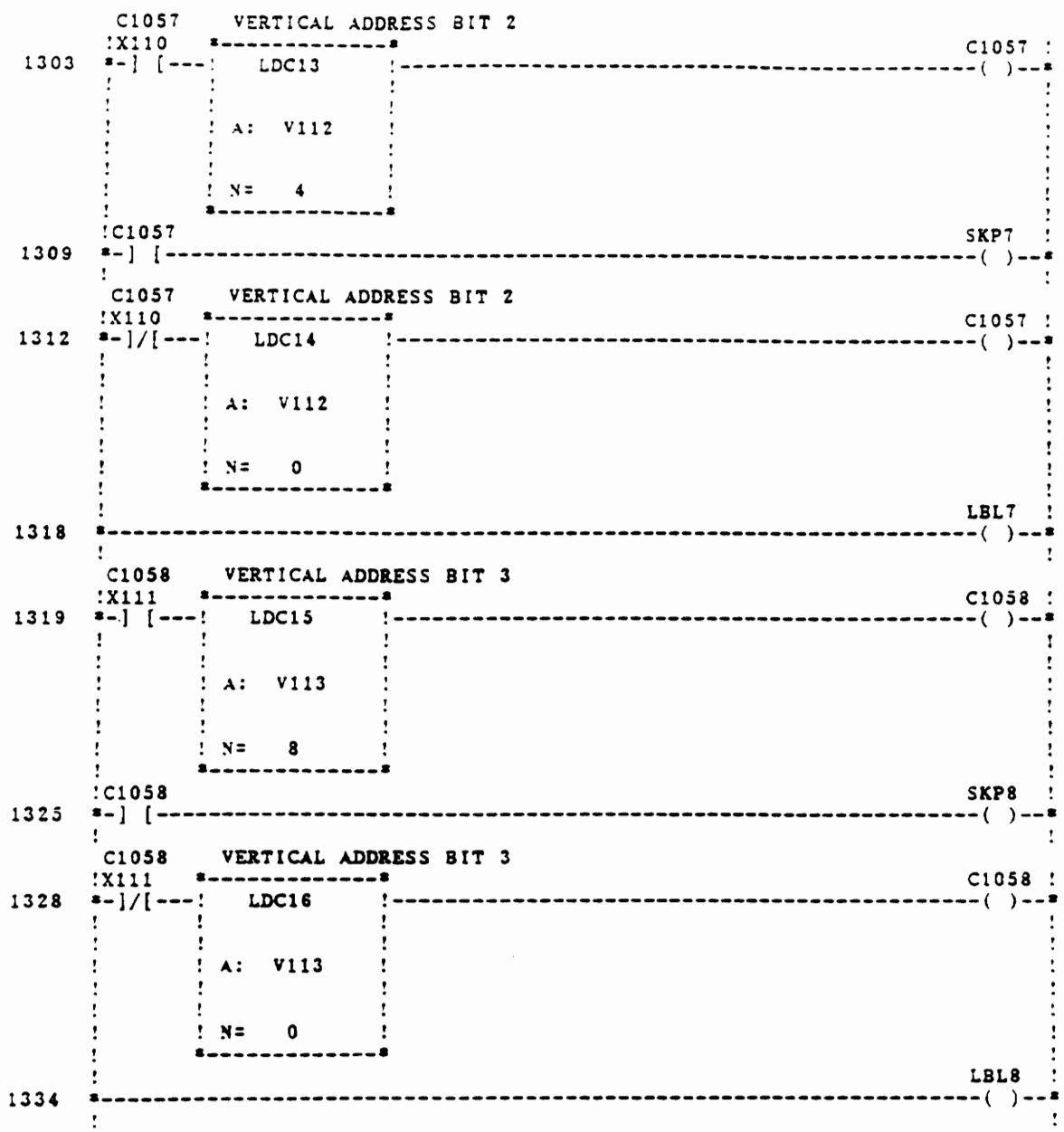

Y9 DONE OPERATION
C20 C200 C201 C999 Y9
1113
Y57 VALVE 1
C20 C22 C999 Y57
1119
Y60 VALVE 4
Y57 Y59 C999 Y60
C20 C22
1124
Y58 VALVE 2
C21 C20 C999 Y58
Y57 Y59
1132
Y59 VALVE 3
C21 C20 C999 Y59
1140
C4006 STEPPER MOTOR TIMER ERROR
Y36 TMR13 C4006
P= 0120.0
C20
1145
C999 ERROR -- STOP MOTION
C4000 C999
1151
C4001
C4002
C4004
C4005
C4006

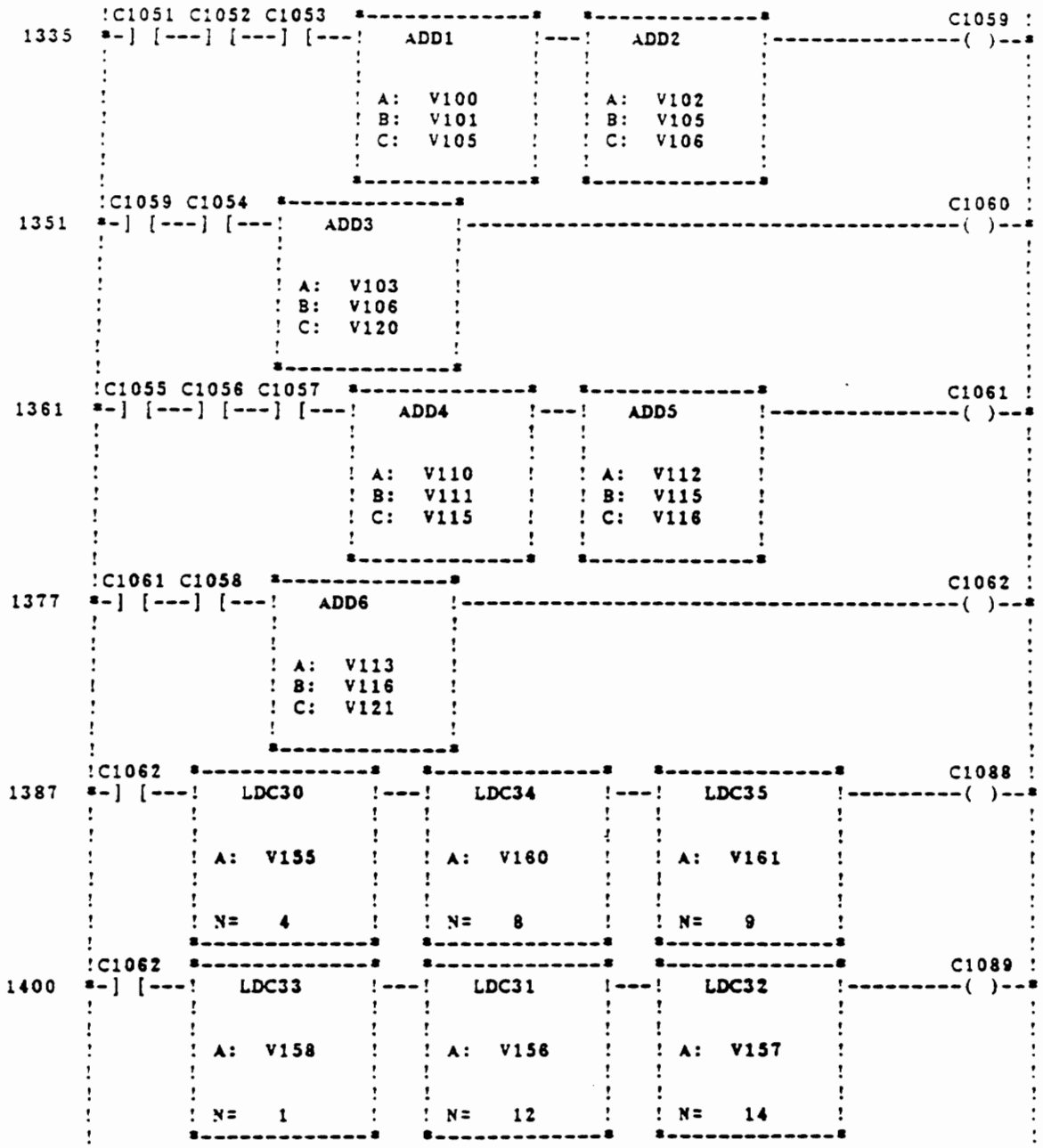

Y11 STATUS CODE -- BIT 0
C4000
C4001
C4002
Y11
1165
Y12 STATUS CODE -- BIT 1
C4000
C4001
C4007
C4008
C4006
Y12
1172
Y13 STATUS CODE -- BIT 2
C4000
C4005
C4004
C4008
C4006
Y13
1183
Y14 STATUS CODE -- BIT 3
C4000
C4001
C4004
C4006
Y14
1194
RTN
1203

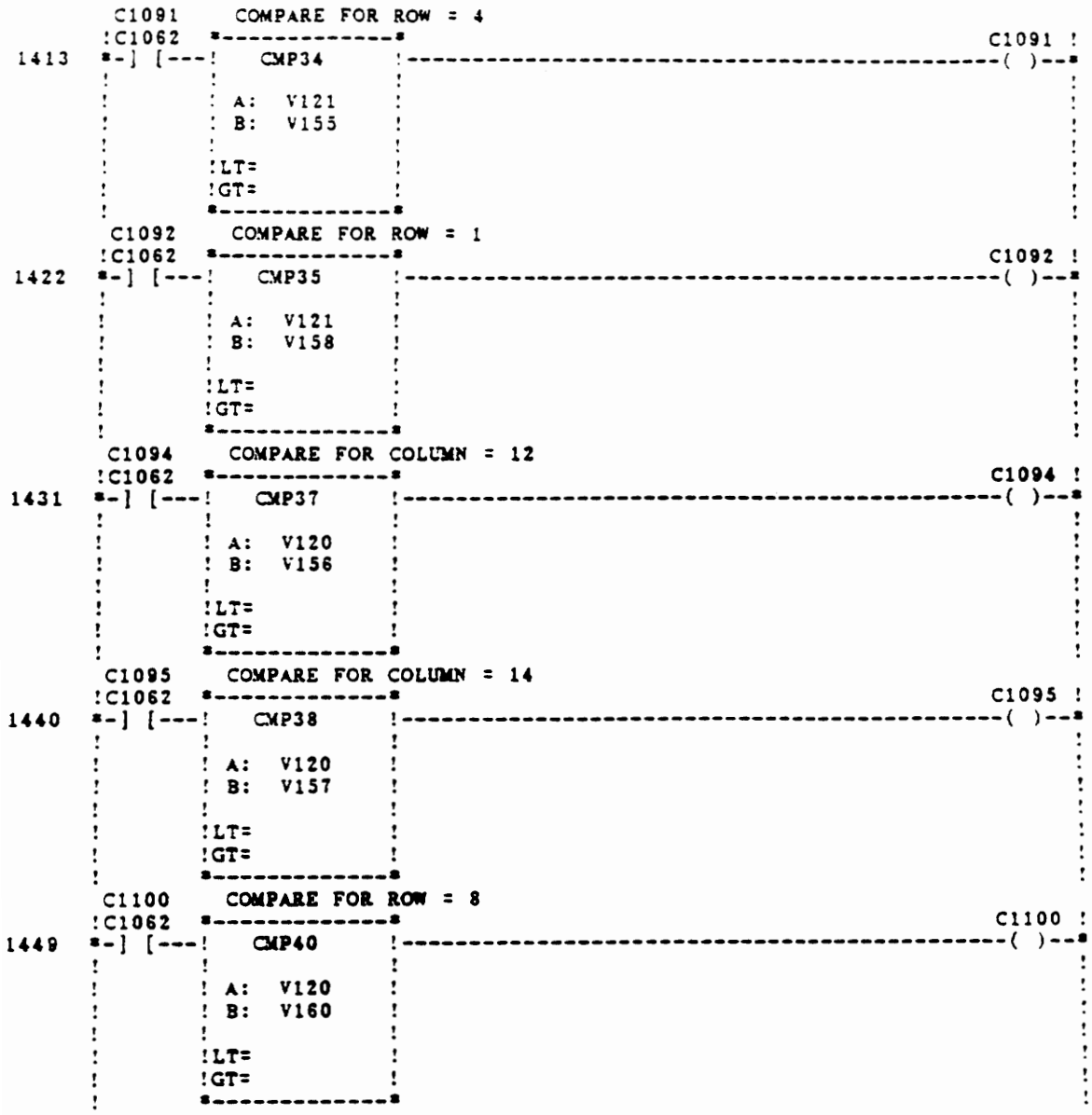

SBR004 ADDRESS
SBR4
1205
C1051 HORIZONTAL ADDRESS BIT 0
X104
C1051
1207
LDC1
A: V100
N= 1
C1051
SKP1
1213
C1051 HORIZONTAL ADDRESS BIT 0
X104
C1051
1216
LDC2
A: V100
N= 0
LBL1
1222
C1052 HORIZONTAL ADDRESS BIT 1
X105
C1052
1223
LDC3
A: V101
N= 2
C1052
SKP2
1229
C1052 HORIZONTAL ADDRESS BIT 1
X105
C1052
1232
LDC4
A: V101
N= 0
LBL2
1238

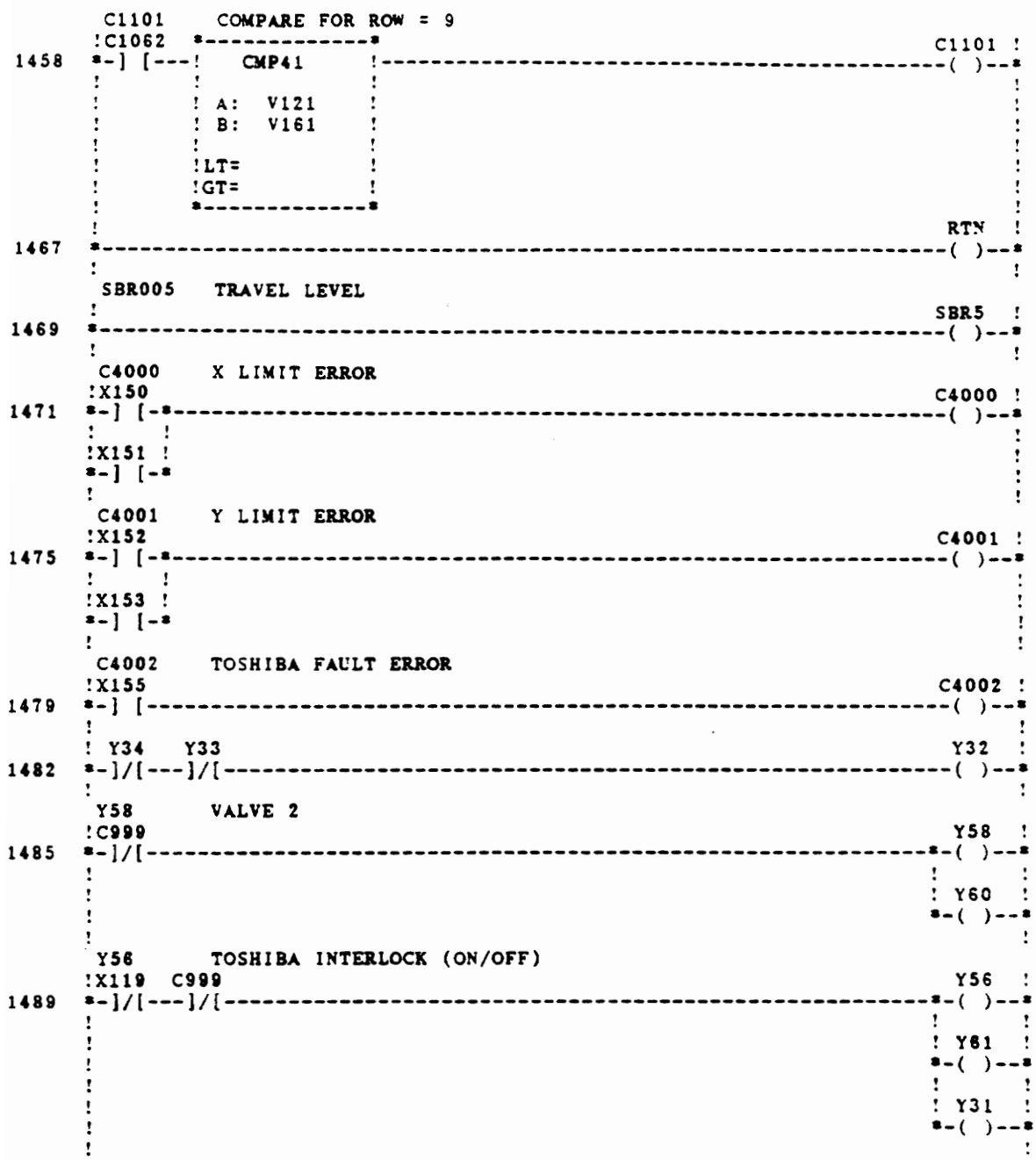

C1053 HORIZONTAL ADDRESS BIT 2
X106
LDC5
A: V102
N= 4
C1053
1239
C1053
SKP3
1245
C1053 HORIZONTAL ADDRESS BIT 2
X106
LDC6
A: V102
N= 0
C1053
1248
LBL3
1254
C1054 HORIZONTAL ADDRESS BIT 3
X107
LDC7
A: V103
N= 8
C1054
1255
C1054
SKP4
1261
C1054 HORIZONTAL ADDRESS BIT 3
X107
LDC8
A: V103
N= 0
C1054
1264
LBL4
1270

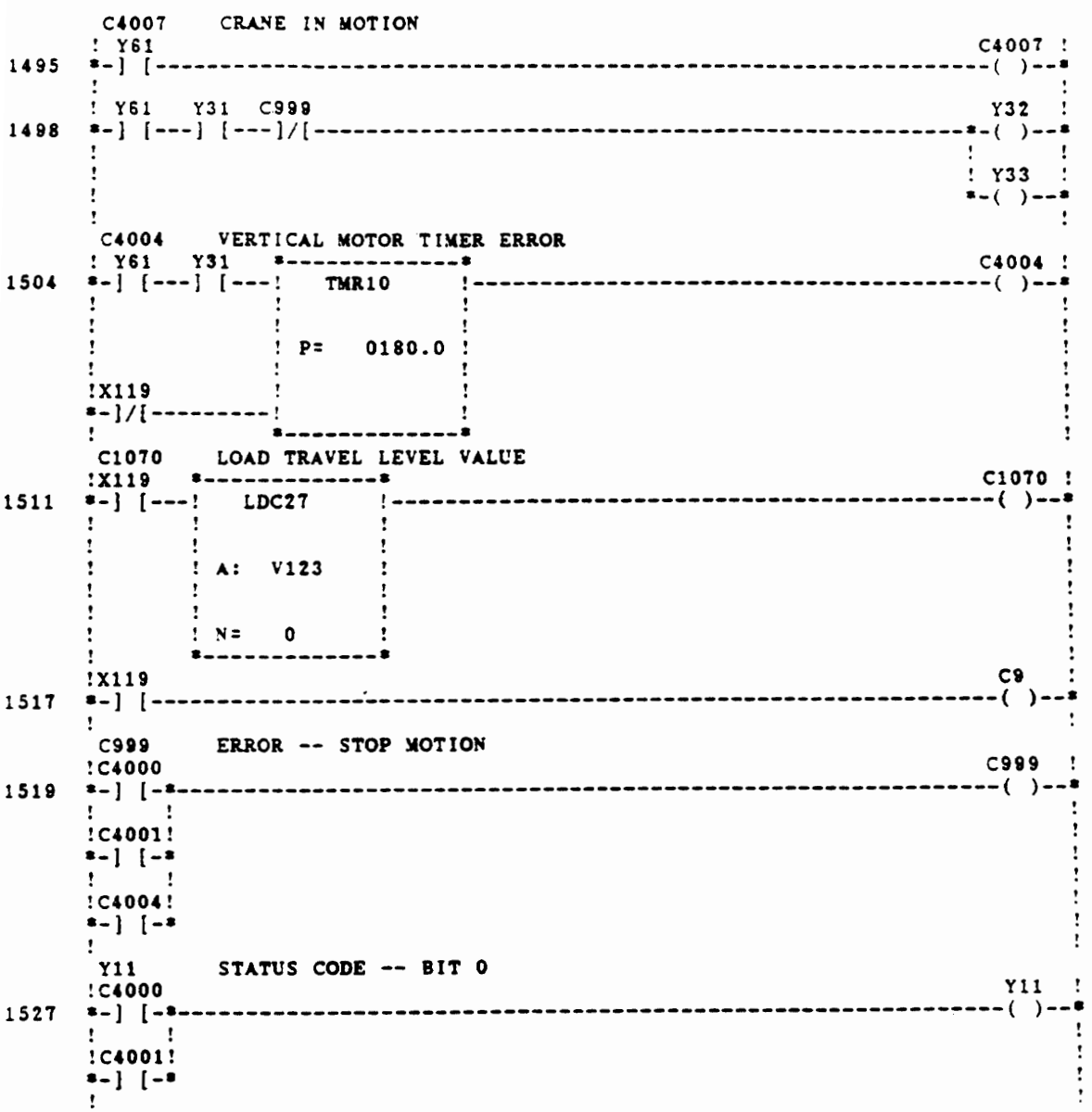

C1055 VERTICAL ADDRESS BIT 0
X108
1271
LDC9
A: V110
N= 1
C1055
C1055
1277
SKP5
C1055 VERTICAL ADDRESS BIT 0
X108
1280
LDC10
A: V110
N= 0
C1055
LBL5
1286
C1056 VERTICAL ADDRESS BIT 1
X109
1287
LDC11
A: V111
N= 2
C1056
C1056
1293
SKP6
C1056 VERTICAL ADDRESS BIT 1
X109
1296
LDC12
A: V111
N= 0
C1056
LBL6
1302


C1057 VERTICAL ADDRESS BIT 2
X110
1303
LDC13
A: V112
N= 4
C1057
1309
SKP7
C1057 VERTICAL ADDRESS BIT 2
X110
1312
LDC14
A: V112
N= 0
1318
LBL7
C1058 VERTICAL ADDRESS BIT 3
X111
1319
LDC15
A: V113
N= 8
C1058
1325
SKP8
C1058 VERTICAL ADDRESS BIT 3
X111
1328
LDC16
A: V113
N= 0
1334
LBL8

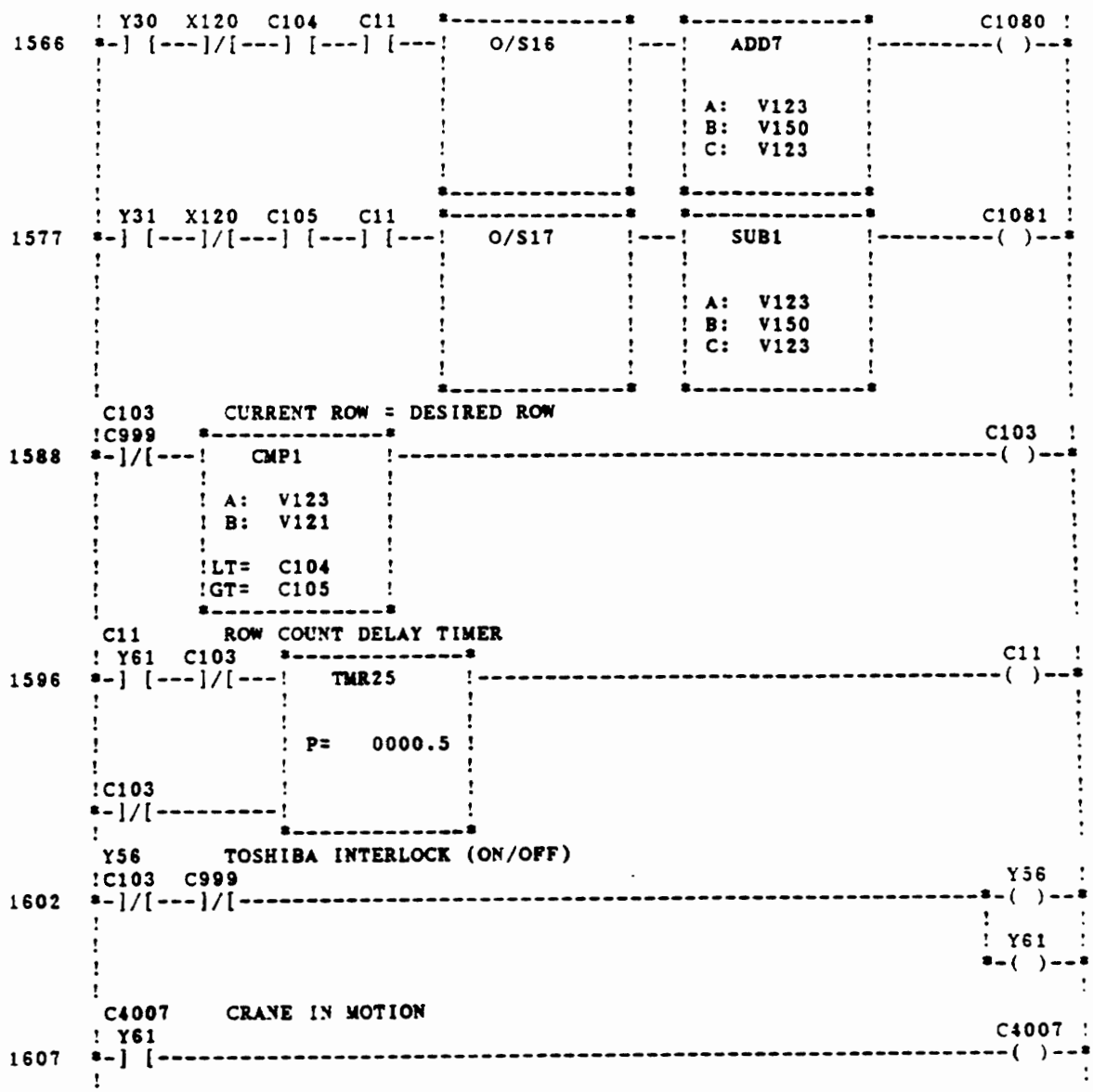

1335
C1051 C1052 C1053
ADD1
A: V100
B: V101
C: V105
ADD2
A: V102
B: V105
C: V106
C1059
1351
C1059 C1054
ADD3
A: V103
B: V106
C: V120
C1060
1361
C1055 C1056 C1057
ADD4
A: V110
B: V111
C: V115
ADD5
A: V112
B: V115
C: V116
C1061
1377
C1061 C1058
ADD6
A: V113
B: V116
C: V121
C1062
1387
C1062
LDC30
A: V155
N= 4
LDC34
A: V160
N= 8
LDC35
A: V161
N= 9
C1088
1400
C1062
LDC33
A: V158
N= 1
LDC31
A: V156
N= 12
LDC32
A: V157
N= 14
C1089

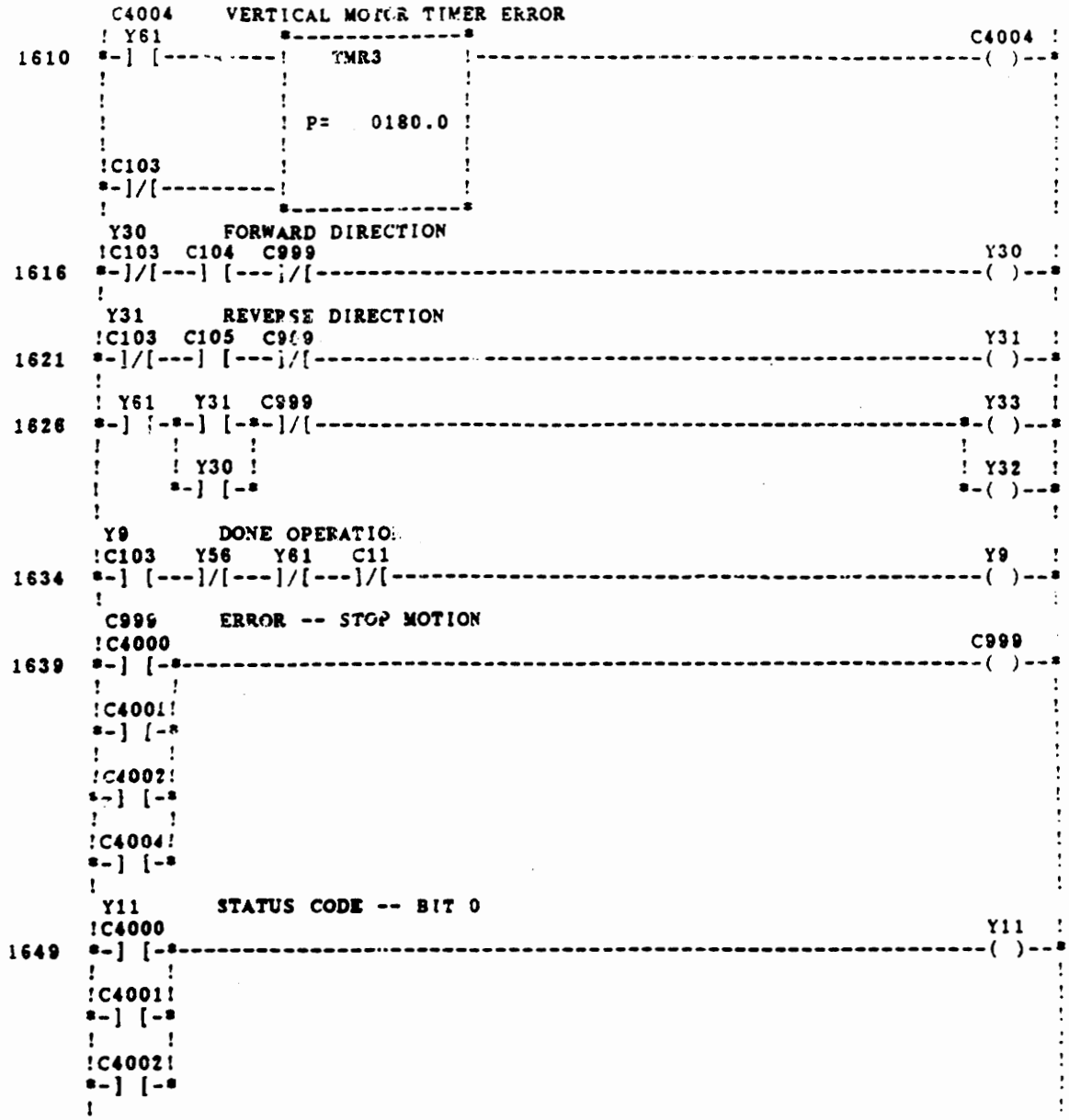

C1091
C1062
1413
CMP34
A: V121
B: V155
LT=
GT=
C1091
C1092
C1062
1422
CMP35
A: V121
B: V158
LT=
GT=
C1092
C1094
C1062
1431
CMP37
A: V120
B: V156
LT=
GT=
C1094
C1095
C1062
1440
CMP38
A: V120
B: V157
LT=
GT=
C1095
C1100
C1062
1449
CMP40
A: V120
B: V160
LT=
GT=
C1100

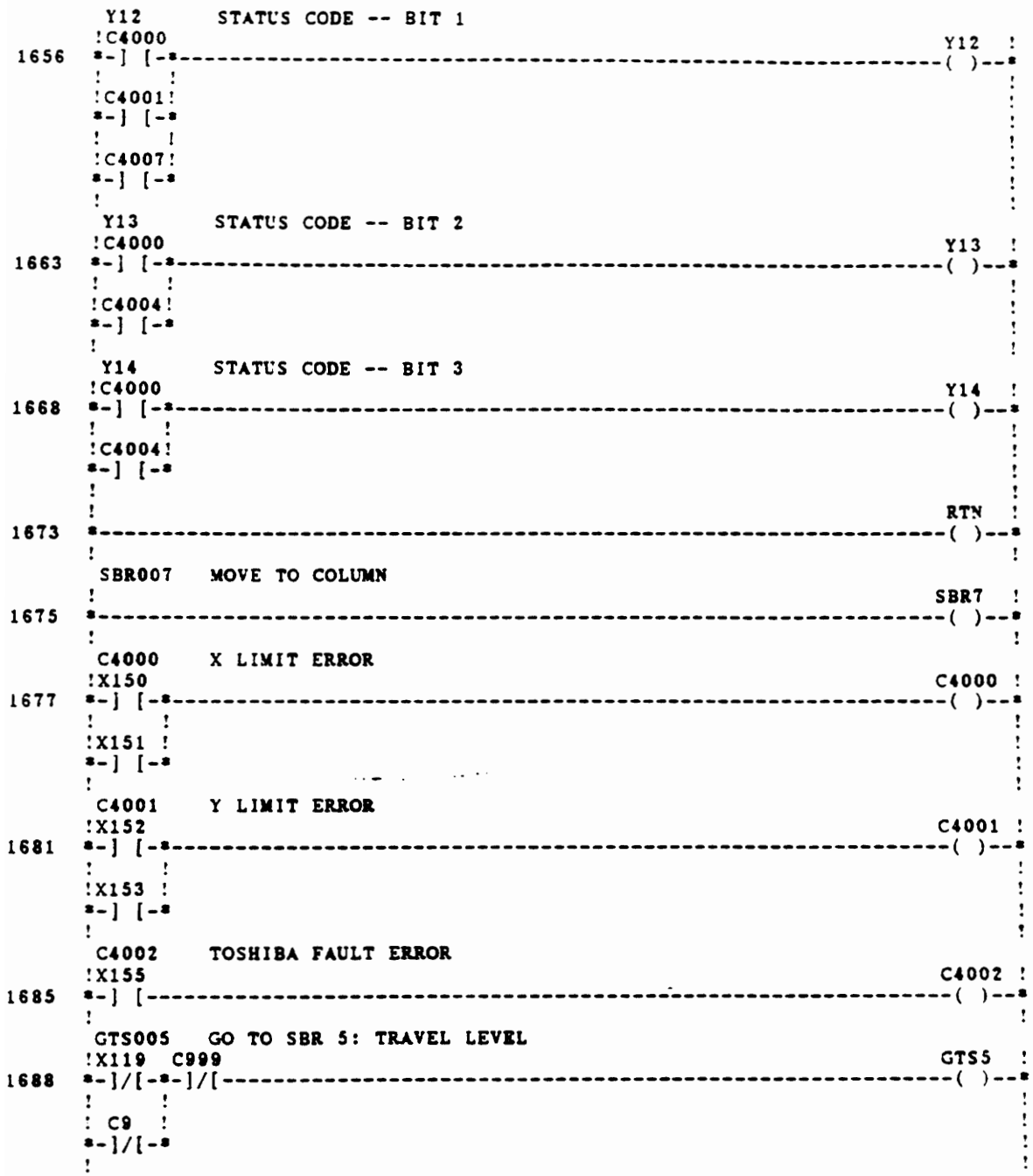

C1101 COMPARE FOR ROW = 9
C1062
1458
CMP41
A: V121
B: V161
LT=
GT=
C1101
1467
RTN
SBR005 TRAVEL LEVEL
1469
SBR5
C4000 X LIMIT ERROR
X150
1471
C4000
X151
C4001 Y LIMIT ERROR
X152
1475
C4001
X153
C4002 TOSHIBA FAULT ERROR
X155
1479
C4002
Y34 Y33
1482
Y32
Y58 VALVE 2
C999
1485
Y58
Y60
Y56 TOSHIBA INTERLOCK (ON/OFF)
X119 C999
1489
Y56
Y61
Y31

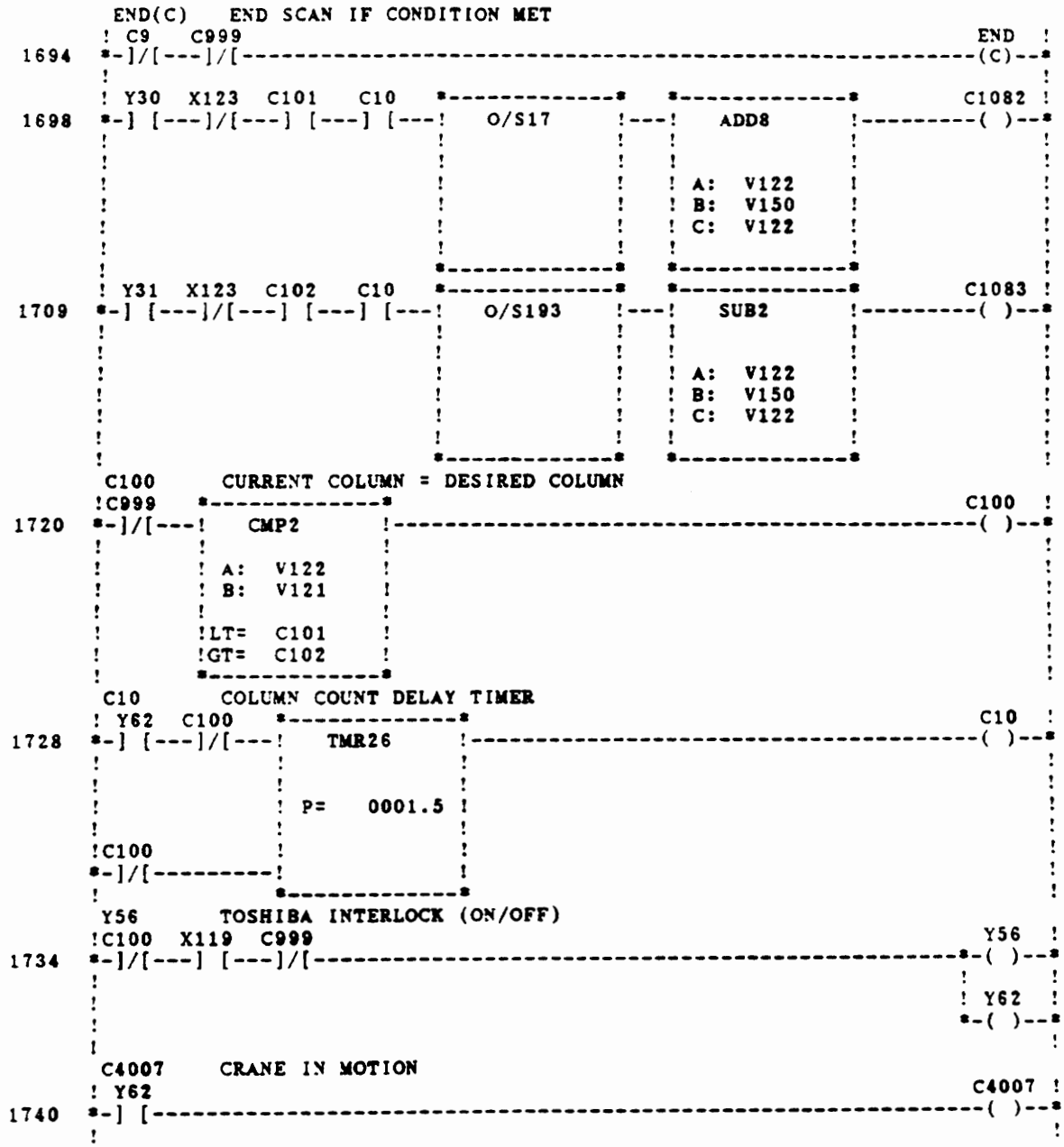

C4007 CRANE IN MOTION
Y61
1495
C4007
Y61 Y31 C999
1498
Y32
Y33
C4004 VERTICAL MOTOR TIMER ERROR
Y61 Y31
1504
TMR10
P= 0180.0
C4004
X119
C1070 LOAD TRAVEL LEVEL VALUE
X119
1511
LDC27
A: V123
N= 0
C1070
X119
1517
C9
C999 ERROR -- STOP MOTION
C4000
1519
C4001
C4004
C999
Y11 STATUS CODE -- BIT 0
C4000
1527
C4001
Y11

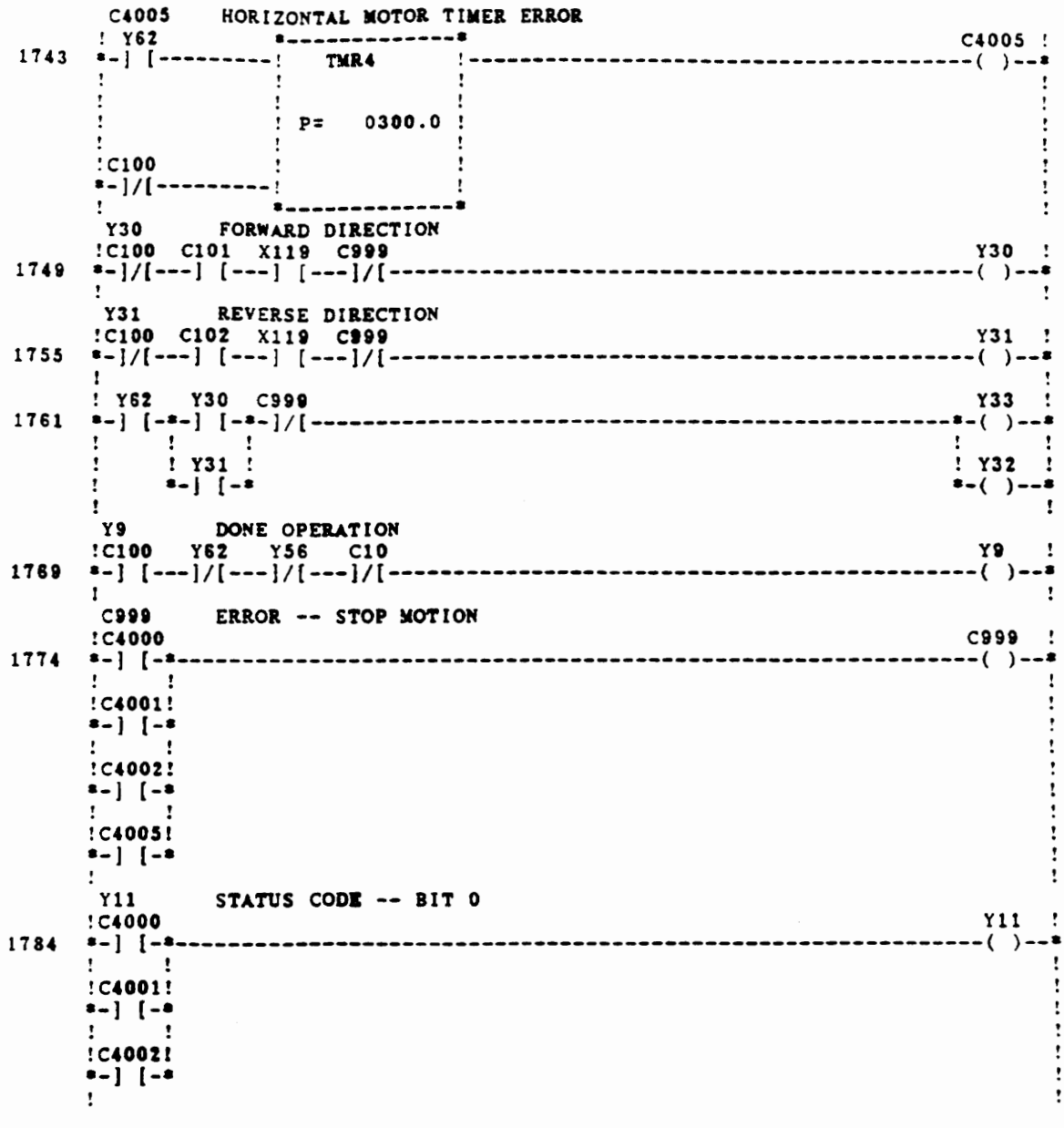
```
-----------------------------------------------------------------------------
        TEXAS INSTRUMENTS PROGRAMMING AND DOCUMENTATION SOFTWARE    PAGE 0032

                                DATE= 00-00-00                          OFF LINE

        Y12       STATUS CODE -- BIT 1
       !C4000                                                          Y12  !
1532   *-] [-*-------------------------------------------------------( )--*
       !     !                                                              !
       !C4001!                                                              !
       *-] [-*                                                              !
       !     !                                                              !
       !C4004!                                                              !
       *-] [-*                                                              !
       !                                                                    !
        Y13       STATUS CODE -- BIT 2
       !C4000                                                          Y13  !
1539   *-] [-*-------------------------------------------------------( )--*
       !     !                                                              !
       !C4004!                                                              !
       *-] [-*                                                              !
       !                                                                    !
        Y14       STATUS CODE -- BIT 3
       !C4000                                                          Y14  !
1544   *-] [-*-------------------------------------------------------( )--*
       !     !                                                              !
       !C4001!                                                              !
       *-] [-*                                                              !
       !     !                                                              !
       !C4004!                                                              !
       *-] [-*                                                              !
       !                                                                    !
       !                                                                RTN  !
1551   *-------------------------------------------------------------( )--*
       !                                                                    !
        SBR006    MOVE TO ROW
       !                                                               SBR6  !
1553   *-------------------------------------------------------------( )--*
       !                                                                    !
        C4000     X LIMIT ERROR
       !X150                                                          C4000 !
1555   *-] [-*-------------------------------------------------------( )--*
       !     !                                                              !
       !X151 !                                                              !
       *-] [-*                                                              !
       !                                                                    !
        C4001     Y LIMIT ERROR
       !X152                                                          C4001 !
1559   *-] [-*-------------------------------------------------------( )--*
       !     !                                                              !
       !X153 !                                                              !
       *-] [-*                                                              !
       !                                                                    !
        C4002     TOSHIBA FAULT ERROR
       !X155                                                          C4002 !
1563   *-] [-------------------------------------------------------( )--*
       !                                                                    !
```

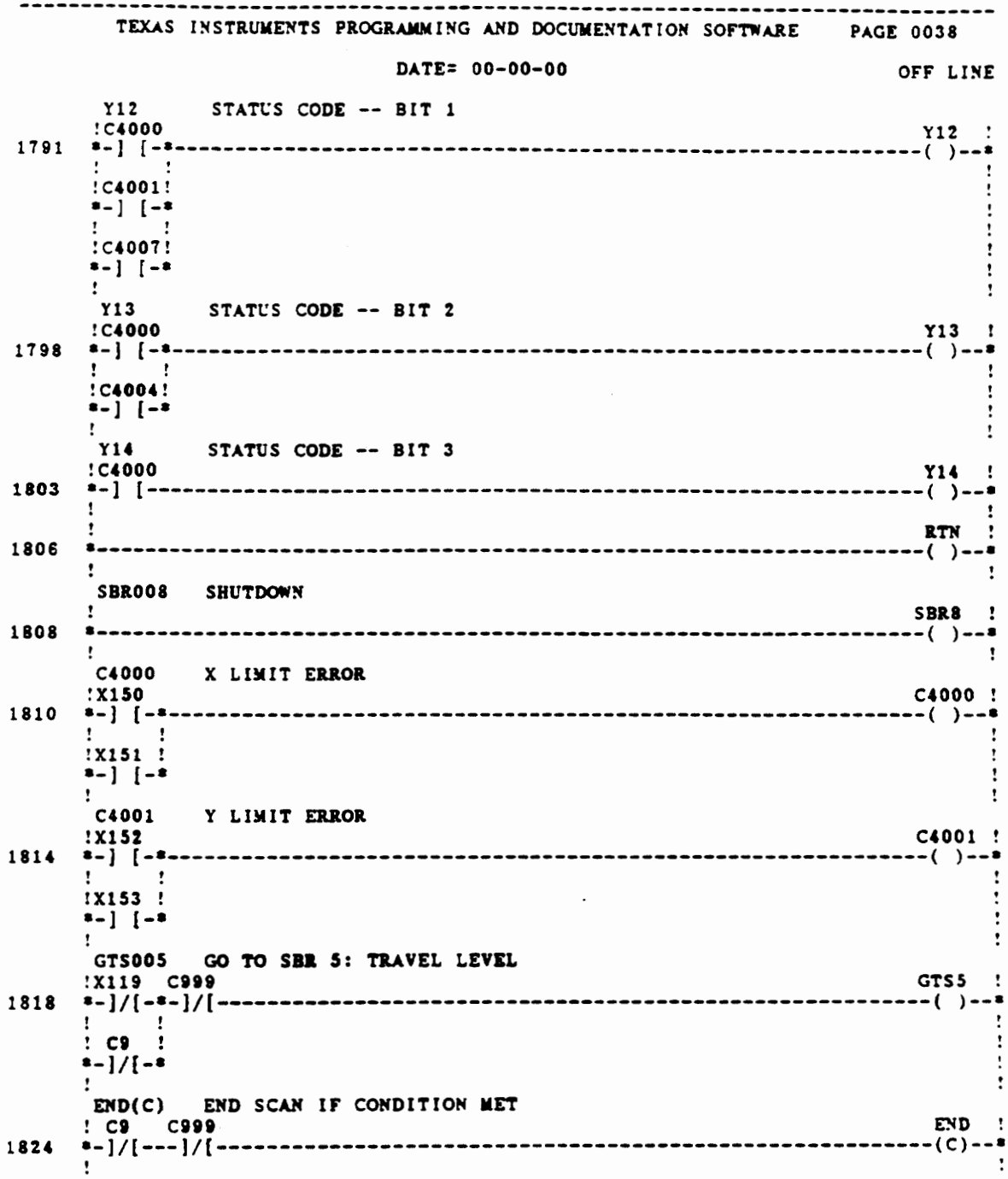

Y30 X120 C104 C11
1566
O/S16
ADD7
A: V123
B: V150
C: V123
C1080
Y31 X120 C105 C11
1577
O/S17
SUB1
A: V123
B: V150
C: V123
C1081
C103 CURRENT ROW = DESIRED ROW
C999
1588
CMP1
A: V123
B: V121
LT= C104
GT= C105
C103
C11 ROW COUNT DELAY TIMER
Y61 C103
1596
TMR25
P= 0000.5
C103
Y56 TOSHIBA INTERLOCK (ON/OFF)
C103 C999
1602
Y56
Y61
C4007 CRANE IN MOTION
Y61
1607
C4007

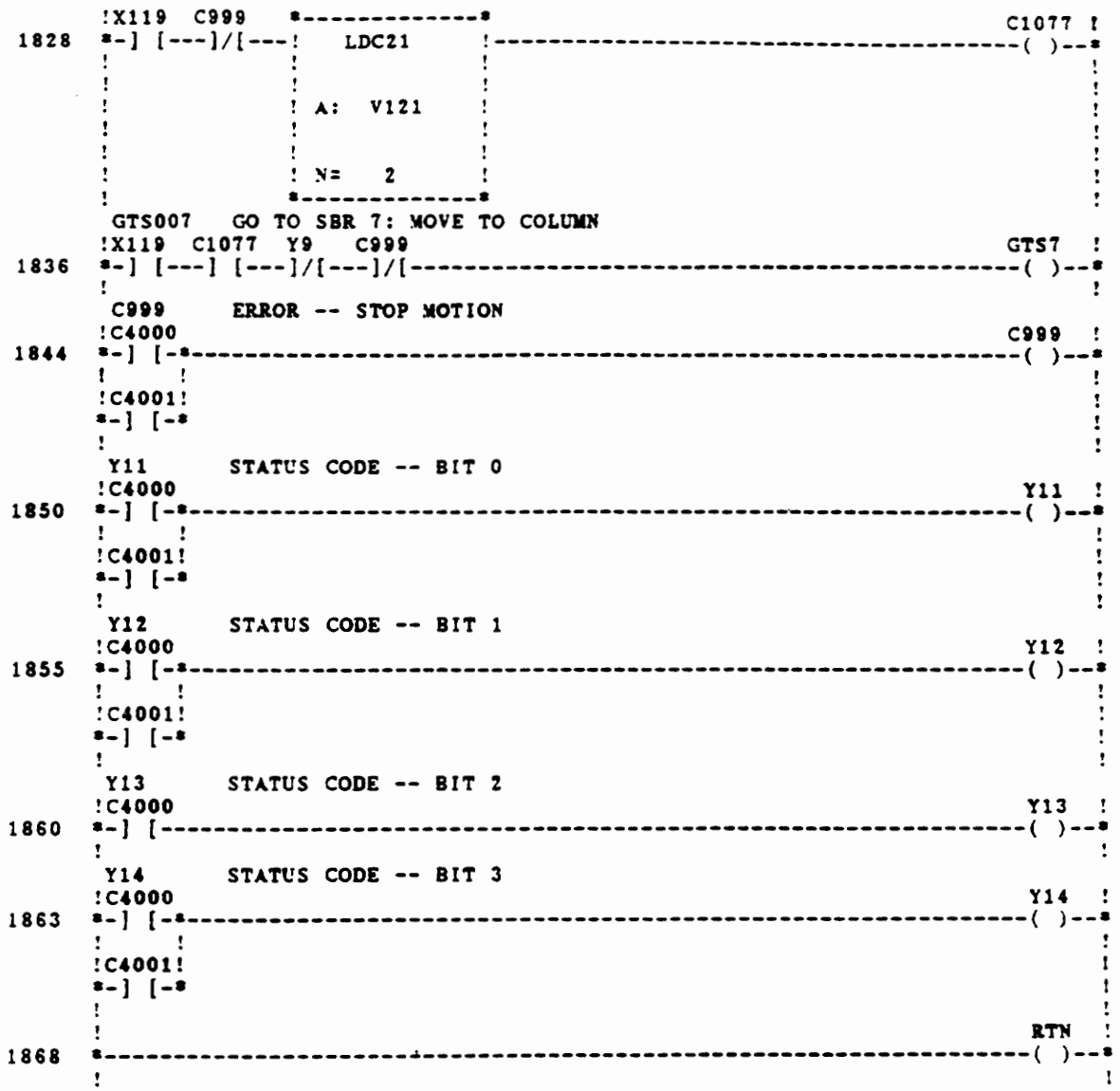

C4004 VERTICAL MOTOR TIMER ERROR
Y61
C4004
1610
TMR3
P= 0180.0
C103
Y30 FORWARD DIRECTION
C103 C104 C999
Y30
1616
Y31 REVERSE DIRECTION
C103 C105 C999
Y31
1621
Y61 Y31 C999
Y33
1626
Y30
Y32
Y9 DONE OPERATION
C103 Y56 Y61 C11
Y9
1634
C999 ERROR -- STOP MOTION
C4000
C999
1639
C4001
C4002
C4004
Y11 STATUS CODE -- BIT 0
C4000
Y11
1649
C4001
C4002


Y12 STATUS CODE -- BIT 1
C4000
C4001
C4007
Y12
1656
Y13 STATUS CODE -- BIT 2
C4000
C4004
Y13
1663
Y14 STATUS CODE -- BIT 3
C4000
C4004
Y14
1668
RTN
1673
SBR007 MOVE TO COLUMN
SBR7
1675
C4000 X LIMIT ERROR
X150
X151
C4000
1677
C4001 Y LIMIT ERROR
X152
X153
C4001
1681
C4002 TOSHIBA FAULT ERROR
X155
C4002
1685
GTS005 GO TO SBR 5: TRAVEL LEVEL
X119 C999
C9
GTS5
1688


END(C) END SCAN IF CONDITION MET
C9 C999
END
1694
Y30 X123 C101 C10
C1082
1698
O/S17
ADD8
A: V122
B: V150
C: V122
Y31 X123 C102 C10
C1083
1709
O/S193
SUB2
A: V122
B: V150
C: V122
C100 CURRENT COLUMN = DESIRED COLUMN
C999
C100
1720
CMP2
A: V122
B: V121
LT= C101
GT= C102
C10 COLUMN COUNT DELAY TIMER
Y62 C100
C10
1728
TMR26
P= 0001.5
C100
Y56 TOSHIBA INTERLOCK (ON/OFF)
C100 X119 C999
Y56
1734
Y62
C4007 CRANE IN MOTION
Y62
C4007
1740


C4005 HORIZONTAL MOTOR TIMER ERROR
Y62
1743
TMR4
P= 0300.0
C100
C4005
Y30 FORWARD DIRECTION
C100 C101 X119 C999
1749
Y30
Y31 REVERSE DIRECTION
C100 C102 X119 C999
1755
Y31
Y62 Y30 C999
1761
Y33
Y31
Y32
Y9 DONE OPERATION
C100 Y62 Y56 C10
1769
Y9
C999 ERROR -- STOP MOTION
C4000
1774
C999
C4001
C4002
C4005
Y11 STATUS CODE -- BIT 0
C4000
1784
Y11
C4001
C4002


Y12 STATUS CODE -- BIT 1
!C4000
1791 *-] [-*
!C4001!
-] [-
!C4007!
-] [-
Y12
( )
Y13 STATUS CODE -- BIT 2
!C4000
1798 *-] [-*
!C4004!
-] [-
Y13
( )
Y14 STATUS CODE -- BIT 3
!C4000
1803 *-] [
Y14
( )
RTN
1806 *
( )
SBR008 SHUTDOWN
SBR8
1808 *
( )
C4000 X LIMIT ERROR
!X150
1810 *-] [-*
!X151 !
-] [-
C4000
( )
C4001 Y LIMIT ERROR
!X152
1814 *-] [-*
!X153 !
-] [-
C4001
( )
GTS005 GO TO SBR 5: TRAVEL LEVEL
!X119 C999
1818 *-]/[-*-]/[
! C9 !
-]/[-
GTS5
( )
END(C) END SCAN IF CONDITION MET
! C9 C999
1824 *-]/[---]/[
END
(C)


!X119 C999
LDC21
C1077
A: V121
N= 2
GTS007 GO TO SBR 7: MOVE TO COLUMN
!X119 C1077 Y9 C999
GTS7
C999 ERROR -- STOP MOTION
!C4000
C999
!C4001!
Y11 STATUS CODE -- BIT 0
!C4000
Y11
!C4001!
Y12 STATUS CODE -- BIT 1
!C4000
Y12
!C4001!
Y13 STATUS CODE -- BIT 2
!C4000
Y13
Y14 STATUS CODE -- BIT 3
!C4000
Y14
!C4001!
RTN

APPENDIX D

LADDER LOGIC VARIABLES

CODE				DESCRIPTION
Y14	Y13	Y12	Y11	
0	0	0	0	AS/RS Idle
0	0	0	1	Toshiba Fault Error
0	0	1	0	Crane In Motion
0	0	1	1	Unused
0	1	0	0	Horizontal Motor Error
0	1	0	1	Unused
0	1	1	0	Shuttle In Motion
0	1	1	1	Unused
1	0	0	0	Invalid Address
1	0	0	1	General Error
1	0	1	0	At Home Position (Initialized)
1	0	1	1	Vertical Extreme Limit Error
1	1	0	0	Vertical Motor Error
1	1	0	1	Unused
1	1	1	0	Stepper Motor Error
1	1	1	1	Horizontal Extreme Limit Error