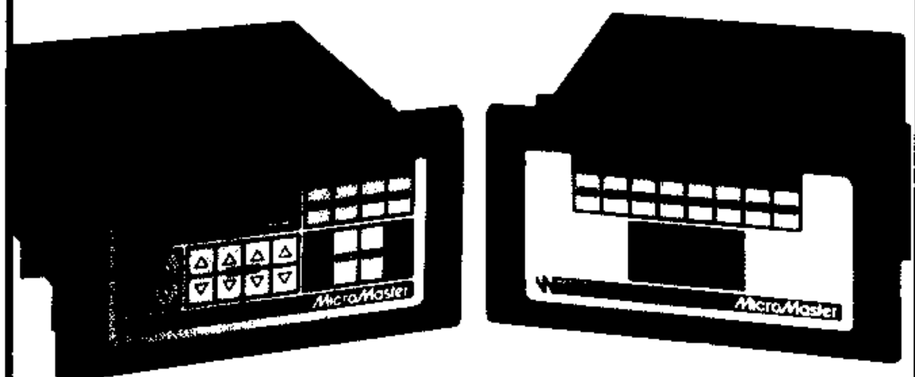
$10.00

# INSTRUCTION MANUAL

MicroMaster®

## SERIES WP6200 & WP6300

## PROGRAMMABLE CONTROLLERS, EXPANDERS AND ACCESSORIES




Minarik Masters of Motion Control

# PRODUCT DATA MANUAL WP6200 and WP6300

## INSTALLATION, PROGRAMMING and OPERATING INSTRUCTIONS FOR SERIES WP6200 AND WP6300 MicroMaster® PROGRAMMABLE CONTROLLERS, EXPANDERS and ACCESSORIES

TABLE OF CONTENTS

# SECTION 1 - INTRODUCTION

## THE MICROMASTER© CONCEPT

The microprocessor-based MicroMaster© system has a multitude of applications. A general discussion of the concept will enable the user to have a clearer picture of why so many machine and automation equipment builders are now utilizing the MicroMaster© approach. The product line is designed to solve sequential control problems that are commonly found in small machines and automated processes. It uses a sequential approach rather than a ladder diagram structure. The reason is one of economy. Prior to this strategy, small applications could not justify using state of the art technology. Compared to ladder diagram programmable controllers the sequential logic avenue drastically reduces the amount of required memory. This creates a cost effective way to replace cam timers, control relays, time delay relays, and other discrete control components. The advantage of the MicroMaster© approach is obvious. It replaces discrete components which cost more, take up more space, take longer to design in, and have greater possibility of system failure.

## SYSTEM OVERVIEW

The system is a general purpose package available in two basic versions. The fully programmable panel mount units may be used as development tools to create proper sequences, test them, and debug them before final production units are utilized. The production versions can be either the RAM based panel mountable units where program changes are frequent, or ROM based subplate versions where programs seldom, if ever, change. The basic system can accommodate up to 4 inputs and 4 outputs. If more inputs and outputs are needed, the expander products can be added to the main system for controlling a larger process.

The approach to problem solving is very much computer oriented, but the ease of programming and the simple instruction set allow easy use, even for the most uninitiated programmer.

**IMPORTANT!:** To prevent damage to internal components and the PC board etched contact fingers **NEVER** remove or install the terminal block edge connector on the MicroMaster© without first disconnecting all power from the terminal block.

## RADIO FREQUENCY INTERFERENCE WARNING

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with this instruction manual, may cause interference to radio communications. As temporarily permitted by regulation, it has not been tested for compliance with the limits for Class A computing devices pursuant to Subpart J of Part .15 of FCC rules, which are designed to provide reasonable protection against such interference. Operation of this equipment in a residential area is likely to cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

# SECTION 1 - INTRODUCTION

## THE MICROMASTER© CONCEPT

The microprocessor-based MicroMaster© system has a multitude of applications. A general discussion of the concept will enable the user to have a clearer picture of why so many machine and automation equipment builders are now utilizing the MicroMaster© approach. The product line is designed to solve sequential control problems that are commonly found in small machines and automated processes. It uses a sequential approach rather than a ladder diagram structure. The reason is one of economy. Prior to this strategy, small applications could not justify using state of the art technology. Compared to ladder diagram programmable controllers the sequential logic avenue drastically reduces the amount of required memory. This creates a cost effective way to replace cam timers, control relays, time delay relays, and other discrete control components. The advantage of the MicroMaster© approach is obvious. It replaces discrete components which cost more, take up more space, take longer to design in, and have greater possibility of system failure.

## SYSTEM OVERVIEW

The system is a general purpose package available in two basic versions. The fully programmable panel mount units may be used as development tools to create proper sequences, test them, and debug them before final production units are utilized. The production versions can be either the RAM based panel mountable units where program changes are frequent, or ROM based subplate versions where programs seldom, if ever, change. The basic system can accommodate up to 4 inputs and 4 outputs. If more inputs and outputs are needed, the expander products can be added to the main system for controlling a larger process.

The approach to problem solving is very much computer oriented, but the ease of programming and the simple instruction set allow easy use, even for the most uninitiated programmer.

While computer terms and approaches are used, they are easily related to terminology that is familiar to users of control equipment. The following examples and their related industrial equivalent are for reference.

**PROGRAM** - A computer term that describes to the MicroMaster© the machine or process sequence that is to be controlled.

**INPUT** - External machine or process switch that causes a change in the sequence. Typical terminology used to describe these switches includes ENCODERS, PHOTOCELLS, LIMIT SWITCHES, PROXIMITY SWITCHES, and MANUAL PANEL SWITCHES (start, stop, halt, up, down, etc.)

**OUTPUT** - External machine, process load or prime mover that is turned on or off by MicroMaster©. Typical terminology used to describe these loads includes SOLID STATE RELAYS, POWER RELAYS, ACTUATORS, SOLENOIDS and MOTORS.

## SAFETY WARNINGS

**INSTALLATION.....**This equipment should be installed, adjusted and serviced by qualified electrical maintenance personnel familiar with the construction and operation of the equipment and the hazards involved. It is the responsibility of the equipment manufacturer or individual installing the apparatus to take diligent care when installing equipment. The National Electrical Code (NEC), sound electrical and safety codes, and when applicable, the Occupational Safety and Health Act (OSHA) should be followed when installing the apparatus to reduce hazards to person and property.

**USE..............**The chance of electric shocks, fires or explosion can be reduced by giving proper consideration to the use of grounding, thermal and over-current protection, type of enclosure and good maintenance procedures.

SECTION 2 - PROGRAMMABLE CONTROLLER

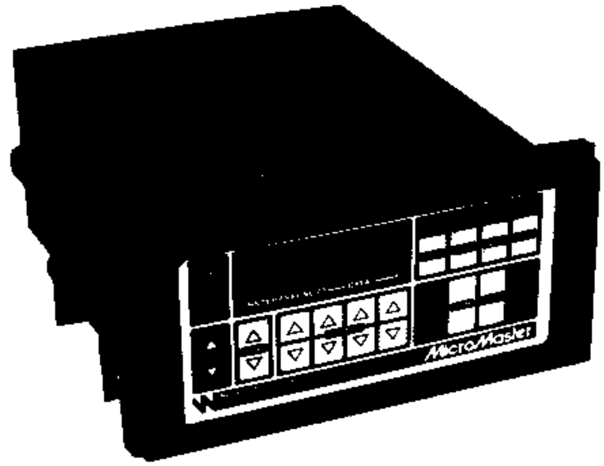



Figure 2-1 WP6200 MicroMaster®

DESCRIPTION

The Series WP6200 MicroMaster® is a completely self-contained microprocessor-based eight function industrial process controller. It is designed to simplify the implementation of time, count and logic functions in control applications. Remote inputs to the unit include RUN, HALT, RESET, COUNT and four user definable inputs. RUN, HALT and RESET can also be operated from the front panel. All inputs can be operated by a simple dry contact closure or an open collector transistor.

OPERATION

With power applied and the unit turned on, the WP6200 has five modes.

1. **RUN** - In the RUN mode, the controller is operating in sequential steps, accepting inputs and executing outputs as programmed.

2. **PROGRAM** - In the PROGRAM mode, new programs may be entered into the controller memory. Also, existing programs can be displayed, step by step, to a maximum program length of 79 steps and changed if desired.

3. **HALT** - In the HALT mode the operation of a running program is suspended or put on "hold", all outputs are returned to their normal state and the on-board timer/counter is stopped. When the controller is returned to the RUN mode, the program restarts from the exact point in the step at which it was halted.

4. **RESET** - In the RESET mode all outputs are returned to normal, the on-board timer/counter is stopped and programmed inputs are disabled.

5. **P-FAIL** - In this mode the display reads "P-FAIL" (program failure), which indicates a catastrophic loss of program memory. All outputs return to normal, the on-board timer/counter is stopped and there is no program stored in the memory. Returning to RESET mode will place the default program of "CT0000" into each of the 79 available steps. **Note!** P-FAIL in a controller does not necessarily indicate a P-FAIL in its companion expander.

Operation of the WP6200 is a two step process. First, while in the program mode, a written program is entered into the unit's memory. Second, the unit is switched to the RUN mode and the program is executed. The alphanumerical display shows a continuous status readout while the program is running. Additionally, four individual lighted rectangles indicate the output status. To test run and debug a program which uses any or all of the four user definable inputs, without the controller being connected to the user's system, appropriate switches can easily be connected to the rear terminal block to simulate the system's inputs. Errors detected during this dry-run testing are easily corrected by simply returning to the program mode and making the necessary changes.

## AUTOMATIC INDEXING

When in the PROGRAM MODE, if any of the Program Control Group switches (Figure 2-4 shown on page 7) and any of the Data Control Group switches (Figure 2-5 shown on page 8) are depressed and held for longer than one second, they will automatically start indexing the display at the rate of three times a second.

## FRONT PANEL CONTROL SWITCHES

All of the front panel switches are momentary actuated push-buttons. The front panel switches are divided into five color coded groups. The operation of each switch is described below.

### GROUP 1 - ON and OFF POWER SWITCHES (Green/Red)

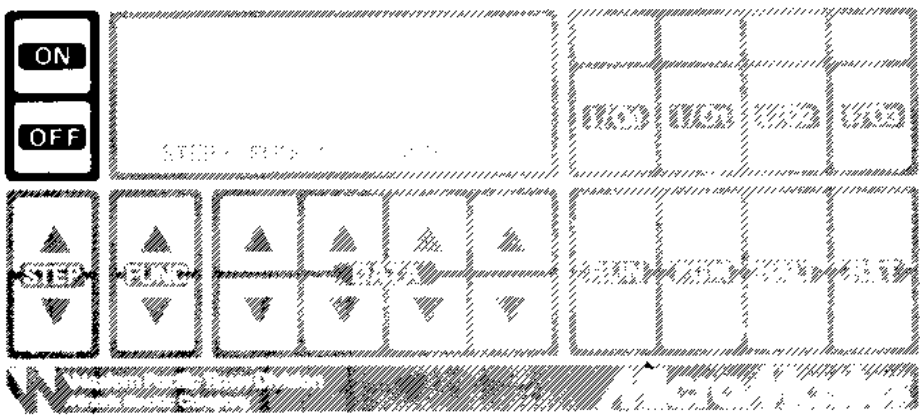



Figure 2-2 - ON and OFF POWER PUSHBUTTON SWITCHES

When power is applied to the controller input and the ON switch is depressed, all of the front panel switches become operative, the displays are turned on and the outputs can be activated as programmed. When the OFF switch is depressed, all of the front panel switches become inoperative, the displays are turned OFF, the outputs return to their normal state and the program resets. Turning OFF the controller does not disable the program memory because the WP6200 has an internal battery backup. NOTE! It takes approximately 48 hours of initial power for the battery to be fully charged on MicroMaster© products. If you program your units and remove power during the first 48 hours of use, there is a chance of program loss (P-FAIL).

**To familiarize yourself with the operation of the controller, connect the unit's terminal strip to a 115 VAC, 60 Hertz power source as shown in Figure 2-8 on page 12. In the upper left corner of the unit are the**

power ON and OFF switches. Press the ON button. The display will light up and show RESET or P-FAIL. The display shows P-FAIL if for any reason the contents of the program memory has been lost. Both front and rear panel RUN signals are disabled in this mode. A subsequent entry into PROGRAM MODE removes this restriction. This P-FAIL mode prevents running the WP6200 if the program has been inadvertently lost.

## GROUP 2 - MODE CONTROL SWITCHES

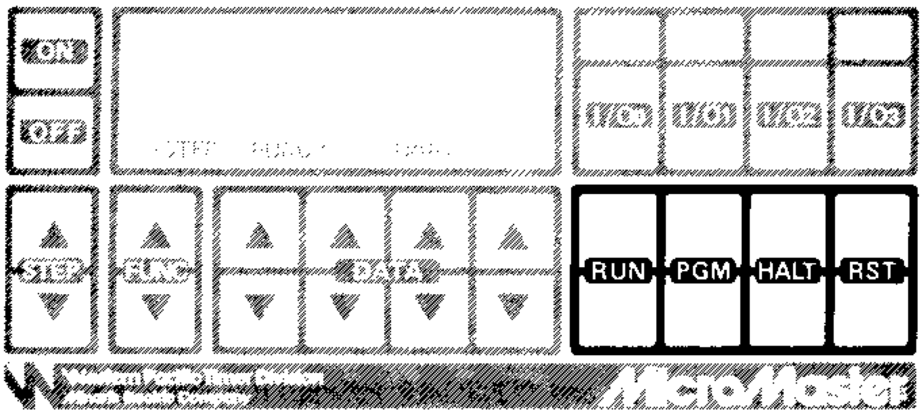



Figure 2-3 - MODE CONTROL SWITCHES

### RUN PUSHBUTTON SWITCH (Green)

This switch is active in the PGM (PROGRAM), HALT and RST (RESET) modes. It initiates execution of a stored sequence starting at step 01 if pushed from the RESET mode, or from the step currently displayed if pushed while in the PROGRAM mode. It is also used to restart the program after a HALT has been signalled.

### PGM PUSHBUTTON SWITCH (Yellow)

The PGM (PROGRAM) switch is active in the RESET or P-FAIL modes. It is used to place the controller into the PROGRAM mode, where the user can use the STEP switches to look at an existing program or use the STEP, FUNCTION and DATA switches to enter a new program.

## HALT PUSHBUTTON SWITCH (Orange)

The HALT switch is used to suspend operation of a running program. The program is stopped and the display reads "cs HALT" ("cs" represents the current step number). The program may be restarted by pushing the RUN switch, or placed in the RESET mode by pushing the RESET switch.

## RST PUSHBUTTON SWITCH (Red)

The RST (RESET) switch is used to return the controller to the RESET mode. The display reads "RESET". RESET is operational at all times.

**To continue with our familiarization of the WP6200, press the RST button and note that the display readout now shows RESET. Press the PGM button to put the controller into the PROGRAM mode.** Note that the display has changed to show STEP "01". The display shows the program in the following manner from left to right:

Characters 1 and 2 show the step number, 01 thru 79.

Characters 3 thru 8 show the step function and any numerical quantities associated with the step.

## GROUP 3 - PROGRAM CONTROL SWITCHES

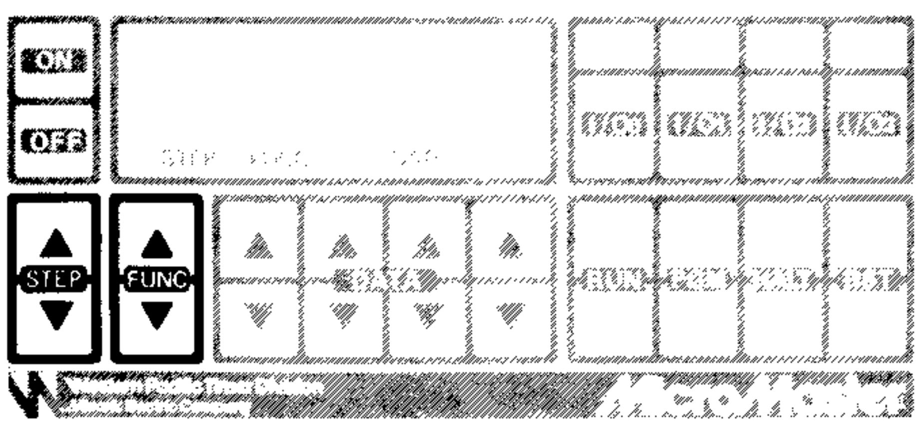



Figure 2-4 - PROGRAM CONTROL SWITCHES

At the left end of the panel, beneath the ON and OFF power switches, four pushbutton switches (Figure 2-4) form the program control group. These four switches and their functions are as follows:

## STEP PUSHBUTTON SWITCHES (Blue)

The STEP ▲ (STEP UP) switch is used to increment by one the number in the step portion of the display, characters 1 and 2. The display will change to show the contents of the step. The STEP ▲ switch is operational only in the PROGRAM mode.

The STEP ▼ (STEP DOWN) switch functions the same as the STEP ▲ switch, except the step number is decremented by one. The STEP ▼ switch is operational only in the PROGRAM mode.

## FUNCTION PUSHBUTTON SWITCHES (Gray)

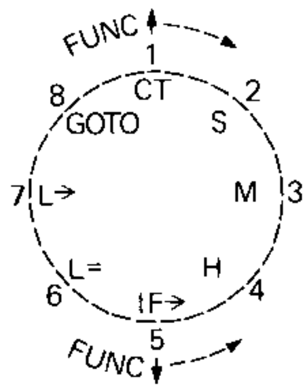



Figure 2-5
FUNCTION
SELECTION

The FUNC ▲ (FUNCTION FORWARD) switch is operational only in the PROGRAM mode and is used to select one of the eight functions available for a given step. By using the STEP switches to locate a particular sequence step, the user can then change the functions currently in RAM for that step by pushing the FUNC ▲ switch. The display will show one of the eight functions and will change the function each time the user pushes the FUNC ▲ switch. The controller will eventually return to the original function displayed or it can be stopped at any desired function. The RAM is changed each time the FUNC ▲ switch places a new function on the display. The user can automatically cycle through the eight functions by holding the FUNC ▲ switch closed. The FUNC ▲ switch selects the eight functions in a clockwise order (refer to Figure 2-5).

The FUNC ▼ (FUNCTION REVERSE) switch operates in exactly the same manner as the FUNC ▲ switch, except the eight functions are selected in a counterclockwise order (refer to Figure 2-5).

Remaining in the PROGRAM mode, press the STEP ▲ switch and notice the 01 has changed to 02. Press the FUNC ▲ switch eight times and notice that 02 does not change but characters 3 thru 8 have changed with each push. Press the FUNC ▲ switch eight more times and notice you are back to the same read-out. Press the FUNC ▼ switch eight times and notice that the functions occur in the reverse order.

Press the STEP ▼ switch and notice the step has changed back to 01. Press the STEP ▼ switch again and notice that you have returned to RESET. Press the PGM switch to return to the PROGRAM mode and then press the STEP ▲ switch seventy-eight times so that STEP number "79" appears in the display. Press the STEP ▲ switch and notice the controller has returned to RESET. You cannot enter more than 79 basic program steps. However, by using loops and nested loops (which are covered later on in this manual) you can expand your program to hundreds of steps.

## GROUP 4 - DATA CONTROL SWITCHES (Yellow)

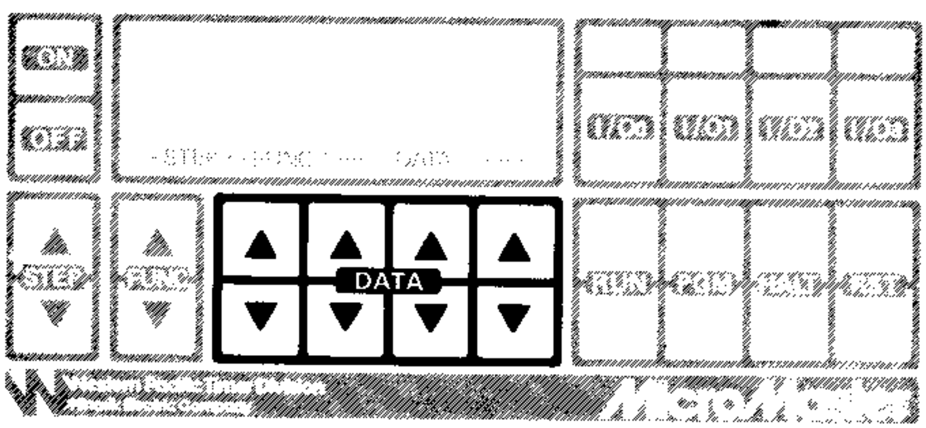



Figure 2-6 - DATA CONTROL SWITCHES

Immediately to the right of the program control group is the data control group (Figure 2-6). It consists of four pairs of yellow pushbutton switches which are active from the PROGRAM mode only. These switches are used to modify numerical quantities associated with the step. There is a pair of switches for each of the DATA display characters. The top row of yellow switches [DATA ▲ (DATA UP)] is used to increment the DATA display characters and the bottom row [DATA ▼ (DATA DOWN)] is used to decrement these same characters. These switch pairs are arranged in the same order as the display characters.

Press the PROGRAM switch to return to PROGRAM mode. Press the FUNC ▲ switch until the characters "0 1 S 9 9 . 9 9" appear on the display. Using the order of left to right, press fourth lower yellow switch (DATA ▼) and notice that the display has changed to read "0 1 S 9 9 . 9 8". Press the same yellow switch eight more times and notice as you do so, each press decrements the number by one, until you have reached zero. Press the fourth upper yellow switch (DATA ▲) nine times and notice that the same digit increases from 0 thru 9. Press the same switch one more time and notice that it returns the number to zero, but has not affected the other three numbers in the display. Any change in one column has no effect on the other columns while in PROGRAM mode. Press all eight yellow switches individually to either increment or decrement the numbers in the display until the read-out displays "0 1 S 1 2 . 3 4". Press the FUNC ▲ switch and notice the display now shows "0 1 M 9 9 : 5 9". Press the FUNC ▼ switch and notice you have returned to the previous function on the display, but the "1 2 . 3 4" has been replaced with "9 9 . 9 9". Whenever a function is removed from the display and returned, all characters always return to their maximum values for that particular function.

## GROUP 5 - OUTPUT CONTROL SWITCHES

## OUTPUT PUSHBUTTON SWITCHES (Orange)

The last group of switches are the I/O (OUTPUT) switches

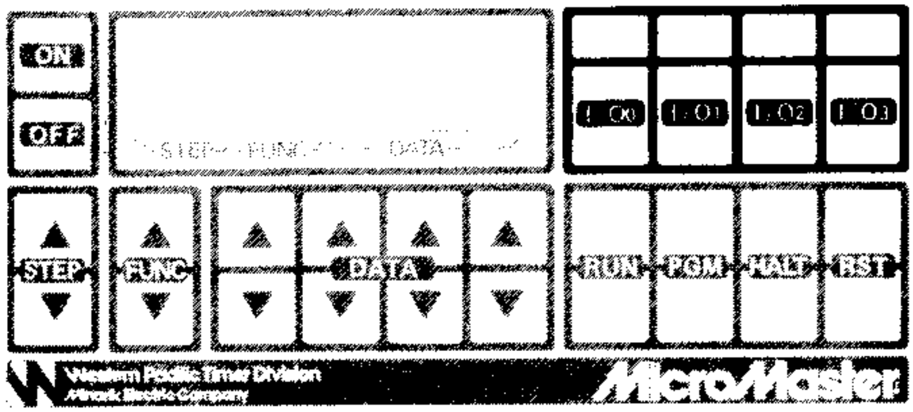



Figure 2-7 - OUTPUT SWITCHES

Figure 2-7) which are located immediately above the mode control group. These four switches are used to program the outputs.

In the PROGRAM mode each switch represents a specific output, labeled 0 thru 3. They are used to set up the output state for each step. Pressing any of these switches will cause the rectangle immediately above it to change state. If the rectangle is lighted, pushing its associated I/O switch will turn it off and vice-versa. A lighted rectangle calls for an output operation at that step. **With the controller in the PROGRAM mode, press I/O button 0. Notice that, if the rectangle were lighted, it went off or if it were off, it lighted. Individually press all of the I/O buttons (0 thru 3) and notice each push alternately turns the lighted rectangle off and on.**

In the RUN mode, the lighted rectangles change to give a continuous indication of the state of the outputs at each step.

## REAR PANEL CONFIGURATION

Extending through a slot on the back of the WP6200 is a PC board to which a terminal block edge connector is attached. The rear panel terminal block connection data is shown in

Figure 2-8 (below). Please refer to Section 7 for installation and wiring information.

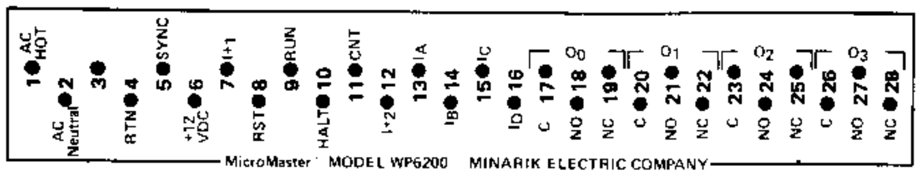



Figure 2-8 - WP6200 REAR TERMINAL BLOCK CONNECTIONS

**IMPORTANT!**: To prevent damage to internal components and the PC board etched contact fingers **NEVER** remove or install the terminal block edge connector on the MicroMaster® without first disconnecting all power from the terminal block.

## SAFETY WARNINGS

**INSTALLATION.....**This equipment should be installed, adjusted and serviced by qualified electrical maintenance personnel familiar with the construction and operation of the equipment and the hazards involved. It is the responsibility of the equipment manufacturer or individual installing the apparatus to take diligent care when installing equipment. The National Electrical Code (NEC), sound electrical and safety codes, and when applicable, the Occupational Safety and Health Act (OSHA) should be followed when installing the apparatus to reduce hazards to person and property.

**USE..............**The chance of electric shocks, fires or explosion can be reduced by giving proper consideration to the use of grounding, thermal and over-current protection, type of enclosure and good maintenance procedures.

## SECTION 3 - PROGRAMMABLE INPUT/OUTPUT EXPANDER

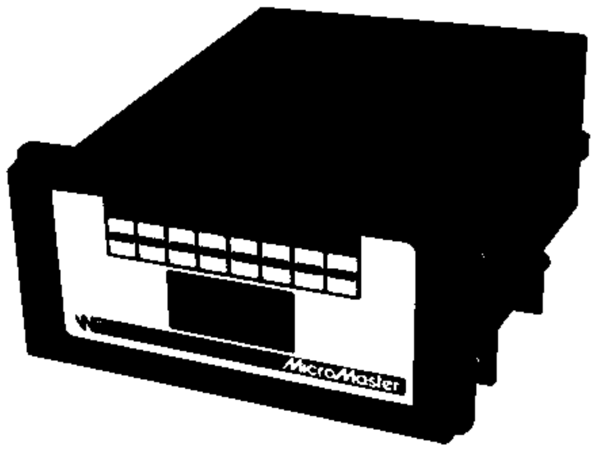



Figure 3-1 WP6300 MicroMaster® INPUT/OUTPUT EXPANDER

### DESCRIPTION

The Series WP6300 MicroMaster® is a completely self-contained microprocessor-based input/output expander. The units are similar in appearance and identical in dimensions to the WP6200 controller. The expander is used to increase or expand the number of available inputs and outputs on the WP6200 Programmable Controller. Adding one WP6300 expander increases the WP6200 controller OUTPUTS from 4 to 12 and adding a second WP6300 expander increases the outputs to 20. Also, adding a WP6300 expander increases the WP6200 controller INPUTS from 4 to 7 and if a WP6350 Input Expander Diode Matrix is used (shown in Section 4), the INPUTS are increased to 16. For quick reference see Figure 3-2, shown on page 14.

### OPERATION

With power applied and the companion WP6200 controller turned on, the WP6300 expander receives information via the sync line for the step number displayed on the controller.

1. **RUN** - In the RUN mode the expander is operating in the same sequential steps as the companion WP6200 controller and selects inputs and operates outputs as programmed.

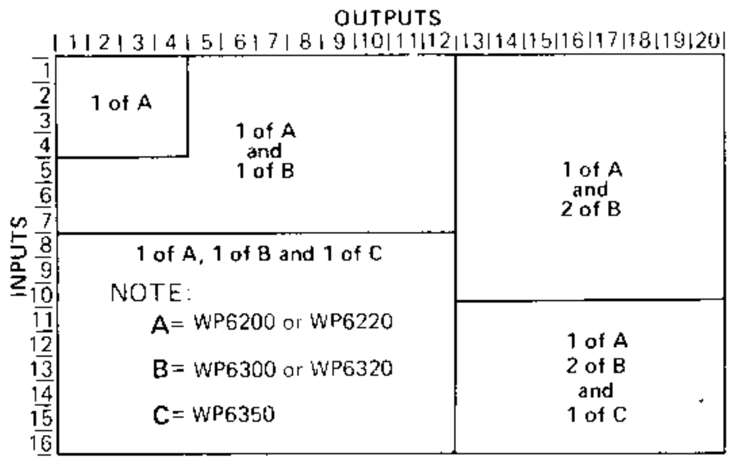



Figure 3-2 COMPONENT SELECTION CHART

2. **PROGRAM** - In the PROGRAM mode new programs may be entered into the expander memory. Also, existing programs can be displayed, step by step, and changed as desired.

3. **P-FAIL** - In the P-FAIL (program failure) mode the expander has a catastrophic loss of memory. In this mode all the outputs return to open and the programmed input groups are disabled. The only indication of an expander P-FAIL is the programming lights will not be lighted in the steps in which they were previously lighted. **IMPORTANT!** The companion controller does NOT receive any signal from the expander noting that its program has failed. Unless the controller has also had a P-FAIL, the controller will continue to run its program without the expander output. **NOTE!** Turning off the power to the expander does not disable the program memory because the WP6300 has an internal battery backup. However, it takes approximately 48 hours of initial power for the battery to be fully charged. If you program your units and remove power during the first 48 hours of use, there is a chance of program loss (P-FAIL).

## FRONT PANEL CONTROL SWITCHES

All of the front panel switches are momentary actuated pushbuttons. The front panel switches are divided into two color coded groups with a programming light above each switch. The operation of each switch is described below.

## INPUT GROUP PUSHBUTTONS (Green)

The four pushbuttons represent input groups labeled 0 thru 3, which are specified in conjunction with a selected input from a companion WP6200 controller. In the PROGRAM mode they are used to set up the input requirements for any step. Pressing any of these switches will cause the programming light immediately above it to change state. If the light is on, pushing its associated switch will turn it off and vice-versa. A lighted programming light indicates a group selection. In the RUN mode, the lights change to give a continuous indication of selected groups at each step.

## OUTPUT PUSHBUTTONS (Orange)

The eight pushbuttons represent specific outputs, labeled 4 thru 11. In the PROGRAM mode, they are used to set up desired outputs for any selected steps. Pressing any of these switches will cause the representative lights above them to change state. If the light is on, pushing its associated switch will turn it off and vice-versa. A lighted light calls for an output closure. In the RUN mode, the lights change to give a continuous indication of the state of the output at each step.

## REAR PANEL CONFIGURATION

Extending through a slot on the back of the WP6300 is a PC board to which a terminal block edge connector is attached. Please refer to Section 7 for installation and wiring information. See Figure 3-3 for illustration of rear terminal block arrangement.

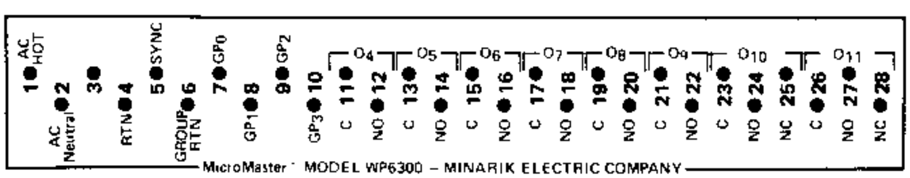



Figure 3-3 - WP6300 REAR TERMINAL BLOCK CONNECTIONS

**IMPORTANT!:** To prevent damage to internal components and the PC board etched contact fingers **NEVER** remove or install the terminal block edge connector on the MicroMaster® without first disconnecting all power from the terminal block.

## SAFETY WARNINGS

**INSTALLATION.....**This equipment should be installed, adjusted and serviced by qualified electrical maintenance personnel familiar with the construction and operation of the equipment and the hazards involved. It is the responsibility of the equipment manufacturer or individual installing the apparatus to take diligent care when installing equipment. The National Electrical Code (NEC), sound electrical and safety codes, and when applicable, the Occupational Safety and Health Act (OSHA) should be followed when installing the apparatus to reduce hazards to person and property.

**USE..............**The chance of electric shocks, fires or explosion can be reduced by giving proper consideration to the use of grounding, thermal and over-current protection, type of enclosure and good maintenance procedures.

## SECTION 4 - RELATED PRODUCTS

### WP6220 FIXED PROGRAM CONTROLLER (CHASSIS MOUNT)

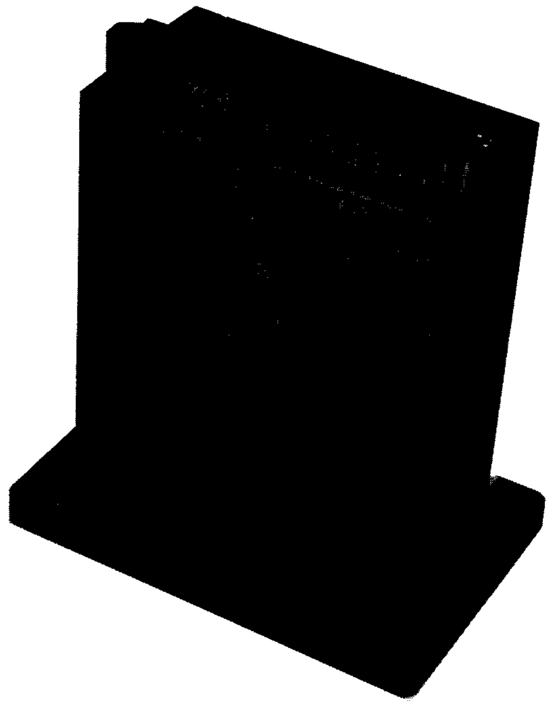


Figure 4-1 WP6220 MicroMaster® CONTROLLER

The WP6220 is a fixed program version without readout of the MicroMaster® WP6200 controller and will perform the same operations. Utilizing the WP6200 controller, a program is developed and "debugged". The program is stored in a PROM (programmable read only memory) which has been custom programmed by Minarik Electric or one of its Authorized Programming Centers. The program can only be changed by ordering a PROM with a different program.

### WP6240 FIXED PROGRAM CONTROLLER (PANEL MOUNT UNIT WITH READOUT)

The WP6240 is a panel mount version of the WP6220 fixed program controller, with readout of STEP, FUNCTION and OUTPUT STATES. It shows continuous status of program execution.

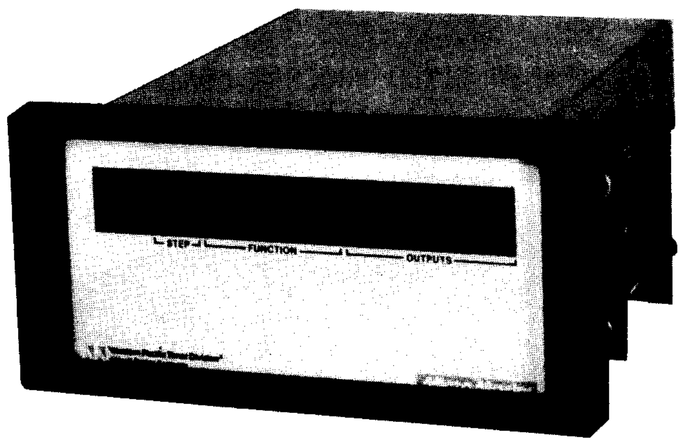


Figure 4-2 WP6240 MicroMaster® CONTROLLER WITH DISPLAY

**WP6320 FIXED PROGRAM INPUT/OUTPUT EXPANDER (CHASSIS MOUNT)**

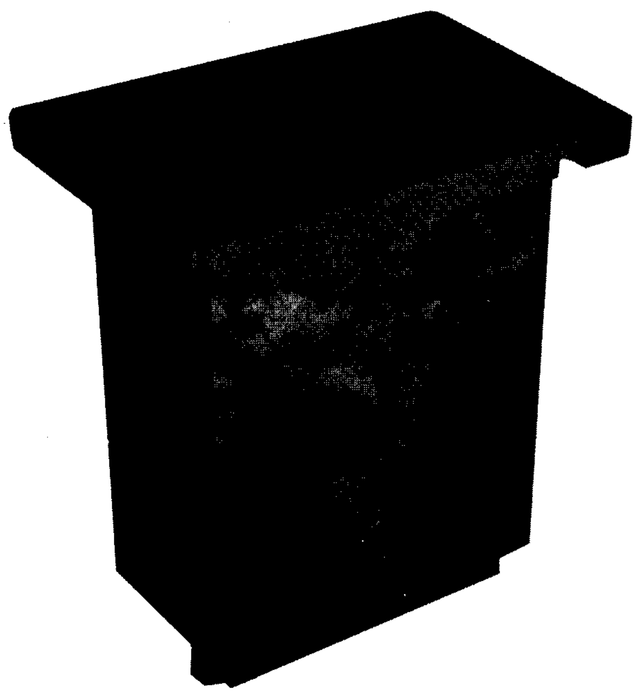


Figure 4-3 WP6320 MicroMaster® INPUT/OUTPUT EXPANDER

The WP6320 is a fixed program version of the MicroMaster® WP6300 expander. Utilizing the WP6200 controller and the

WP6300 expander, a program is developed and "debugged". The proven program is stored in a PROM which has been custom programmed by Minarik Electric or one of its Authorized Programming Centers. The program can only be changed by ordering a PROM with a different program.

## WP6350 INPUT EXPANDER DIODE MATRIX

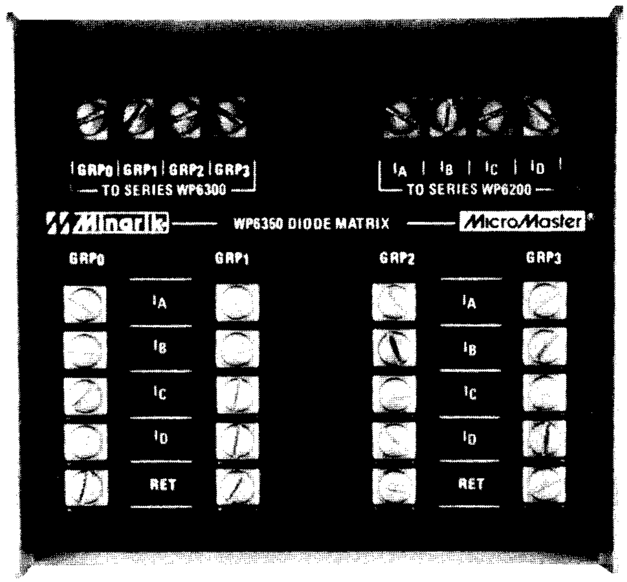



Figure 4-4 WP6350 INPUT EXPANDER DIODE MATRIX

The WP6350 Input Expander Diode Matrix is used in conjunction with a controller and expander when more than 7 user defined inputs are required. The WP6350 consists of a group of terminal blocks and internal diodes for input isolation. This device permits connection of up to 16 inputs.

## WP6012 FIXED PROGRAM TIMER MODULE

The WP6012 is an on-delay timing module, with a 20:1 knob adjustable timing range from 0.05 thru 1.0 second (other time ranges available). The output is an open collector transistor which saturates while timing and opens when timed out. The WP6012 can be utilized to supply timing variations for the fixed program WP6220 and WP6240 controllers. The timer module is used in conjunction with the controller's loopcounter to increase timing range.

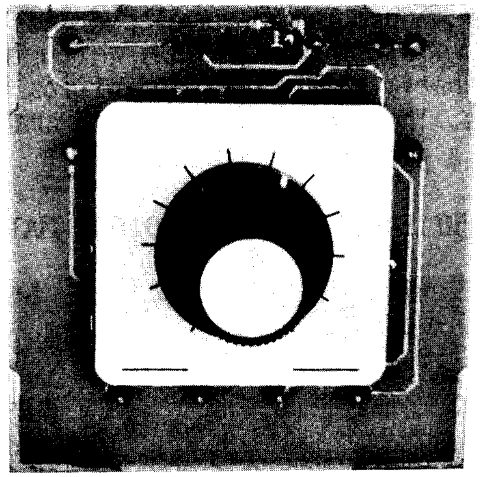


Figure 4-5 WP6012 TIMER MODULE

**SAFETY WARNINGS**

**INSTALLATION**.....This equipment should be installed, adjusted and serviced by qualified electrical maintenance personnel familiar with the construction and operation of the equipment and the hazards involved. It is the responsibility of the equipment manufacturer or individual installing the apparatus to take diligent care when installing equipment. The National Electrical Code (NEC), sound electrical and safety codes, and when applicable, the Occupational Safety and Health Act (OSHA) should be followed when installing the apparatus to reduce hazards to person and property.

**USE**..............The chance of electric shocks, fires or explosion can be reduced by giving proper consideration to the use of grounding, thermal and over-current protection, type of enclosure and good maintenance procedures.

# SECTION 5 - THE INSTRUCTION SET

## GENERAL INFORMATION

The instruction set for the WP6200 MicroMaster© controller consists of eight possible programmable functions. The program is a list of things for the controller to do. This list is made up of a combination of these eight functions. The program is executed sequentially, step by step, unless conditional or unconditional branch instructions cause jumps out of the normal sequence. The WP6200 will not allow the programming of a function it does not understand. Wherever numerical or alphabetical quantities are associated with a step, the internal programming of the WP6200 limits the range of each digit to the range acceptable to the controller. For instance, if a function refers to an input, the character in the display which refers to the input letter will only assume the values of A thru D. If an attempt is made to increment this character to E, the character will "go over the top" and return to A. If a function refers to a step number, that step number will not be allowed to be set to a value outside the range of 00 thru 79 (00 is RESET). Thus, it is impossible to enter an "illegal" command to the WP6200's program memory.

## THE EIGHT PROGRAMMABLE FUNCTIONS

Preceding the explanation of the eight programmable functions is a description of how they will appear in the alphanumerical display. The speed of operation of all non-count and non-time functions is so fast that they are not normally seen on the display. Before proceeding with the discussion of the programmable functions, it is necessary to establish some conventions about the type of notations to be used. Four such conventions will be used.

1. cs - The notation "cs" will be used to indicate the "current step" number for a function. This two digit number may have a value from 01 thru 79. It is the first two digits of the display readout (reading from left to right).

2. **xxxx** - The notation "xxxx" will be used to indicate the "numerical value" associated with a count or time figure and may have any value from 0000 thru 9999. These are the last four digits of the display.

3. **ts** - The notation "ts" will be used to indicate the "target step" number for a branch instruction. It is the step number that the program will jump to when a jump is executed and may have any value from 00 thru 79.

4. **c** - The notation "c" will be used to indicate the character for the "user definable inputs A thru D".

For example: "cs S xx.xx" could be equivalent to "01 S 99.99" and "cs IF c > ts" could be equivalent to "01 IF D > 79".

To the right of the alphanumerical display are four indicator lights that show the status of the four outputs at each step. When the indicator light is off for a given step, the output will be in its "normal" state (for instance, with a form C relay output the normally closed contact will be closed and the normally open contact will be open). When the indicator light is on, the output will assume the opposite state. The indicator lights, from left to right, correspond to outputs 0 thru 3 respectively.

**FUNCTION "CT" (COUNT)**

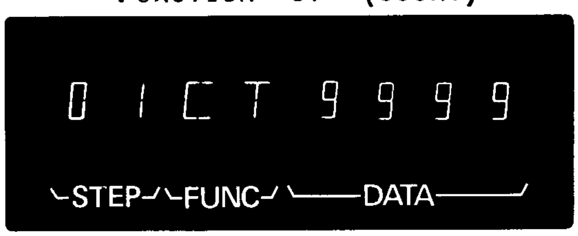



Figure 5-1 "CT" (01 is the current step number, CT represents count and 9999 is the maximum predetermined number which may be entered for this function.)

**(cs CT xxxx)** - This is the predetermining counter function and it can be set from 9999 down to 0001. The outputs are set to the desired output state, as shown by the indicator lights, and will remain in that state until "xxxx" number of counts have been applied to the external count input on the rear panel. In the RUN mode, the output state, shown by the indicator lights, will be transferred to the outputs for the number of counts necessary to complete the function. As the countdown progresses, the remaining number of counts left are shown on the alphanumerical readout display. At the completion of the function, the controller will proceed to the next higher sequential program step.

FUNCTION "S" (SECONDS)

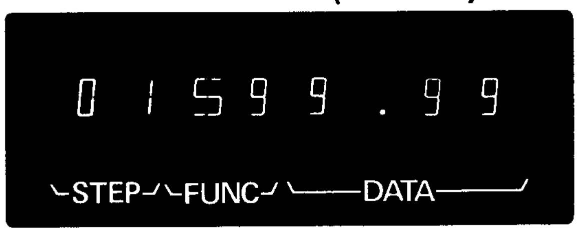



Figure 5-2 "S" (01 is the current step number, S represents seconds and 99.99 is 99 seconds 99/100 seconds, which is the maximum time that can be entered for this function.)

**(cs S xx.xx)** - This is a crystal-controlled timer function with the time base in 1/100ths of a second. Programmable range is 00.01 thru 99.99 seconds. The outputs are set to the desired state as shown by the indicator lights and will remain in that state for the length of time indicated by "xx.xx". In the RUN mode, the output state, shown by the indicator lights, will be transferred to the outputs for the length of time necessary for the completion of the function. As the time elapses, the numerals "count-down", with the remaining time at any one moment shown on the alphanumerical display. At the completion of the programmed time delay the controller will proceed to the next sequential program step.

FUNCTION "M" (MINUTES)

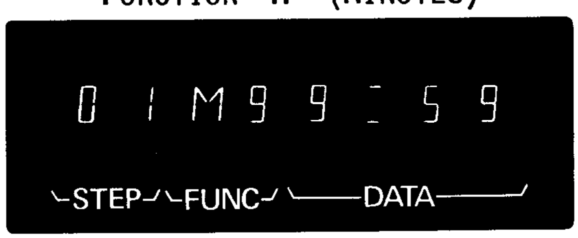



Figure 5-3 "M" (01 is the current step number, M represents minutes and 99:59 is 99 minutes 59 seconds, which is the maximum time that can be entered for this function.)

**(cs M xx:xx)** - This is a timer function similar to Function S, except the time base is in minutes and seconds, with a time range of 00 minutes 01 seconds thru 99 minutes 59 seconds.

FUNCTION "H" (HOURS)

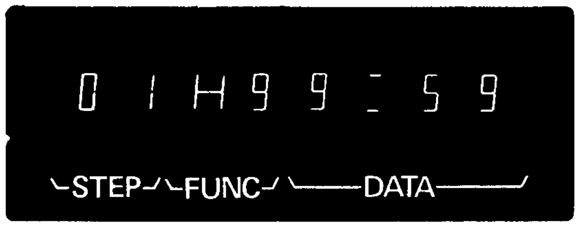



Figure 5-4 "H" (01 is the current step number, H represents hours and 99:59 is 99 hours 59 minutes, which is the maximum time that can be entered for this function.)

**(cs H xx:xx)** - This is also a timer function similar to Function M, except the time base is in hours and minutes, with a time range of 00 hours 01 minutes thru 99 hours 59 minutes.

FUNCTION "GO TO" (JUMP TO)

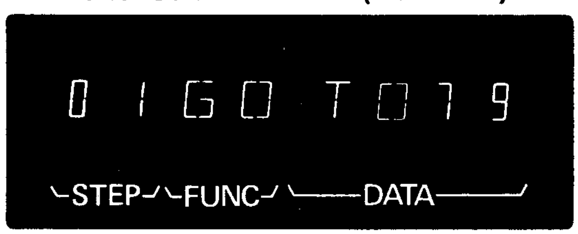



Figure 5-5 "GO TO" (01 is the current step number and GO TO 79 instructs the controller to jump to target step number 79.)

**(cs GO TO xx)** - This is the unconditional branch instruction. When encountered in the program, it tells the controller "jump to step 'ts' instead of proceeding to the next sequential program step". Execution then continues from 'ts' on. In the RUN mode, the display is continuously indicating the current program step. Step 00 is the RESET mode of the

WP6200 controller. The normal way to end a program is to tell the controller to jump to the RESET mode, or "cs GO TO 00".

FUNCTION "IF >" (IF - JUMP TO)

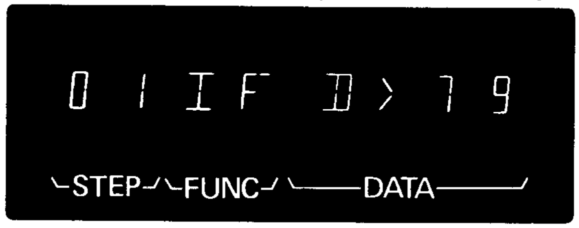



Figure 5-6 "IF >" (01 is the current step number and IF D > 79 says "if input D is closed, jump to target step 79".)

(cs IF c >ts) - This is the conditional branch instruction. It tells the controller "Go look at input 'c'. If it is closed, go to step 'ts'. If it is open, go to the next sequential program step." This is the instruction which allows the user to check the state of inputs A thru D. NOTE!: The inputs are NOT monitored continuously, but only when the program instructs the controller to check them.

## THE WP6200 LOOPCOUNTER STACK

The WP6200 loopcounter stack is a group of four registers used to store numbers for loopcounter functions L = (loop equal) and L > (loop jump to). To understand these functions, it is first necessary to understand what a "stack" is. A stack is a pile of numbers. "Pushing" a number onto the stack is simply a matter of putting that number on top of the stack. "Popping" the stack means to remove the top number from the stack and move the number below it into the top position. The only number on the stack which is accessible is the top number. Numbers "under" the top number may not be used until the stack is popped. One way to think of this is to imagine a box and several pieces of paper. The box is just big enough to hold the pieces of paper when layed flat in the box. Pushing a number onto the stack would be equivalent to writing the number on one of the pieces of paper and laying the paper face-up in the box. At this point we can still see the number on the piece of paper and so the number is accessible. If we then do a second push by writing another number on another piece of paper and lay it face-up over the first piece in the box, we can no longer see the first piece of paper so that number is inaccessible. The number on the second piece of paper is still visible, however, and thus is accessible. Popping this stack is simply a matter of removing the top sheet of paper and throwing it away. This brings the first sheet back to the top position and thus it is again accessible.

**FUNCTION "L = " (LOOP EQUAL)**

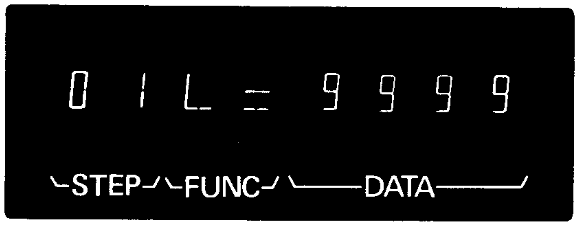



Figure 5-7 "L = " (01 is the current step number, L = 9999 sets the loopcounter to 9999 loops.)

(cs L = xxxx) - This function is used to set an internal "loopcounter". There are four loopcounters in the controller and they may be set from 0001 thru 9999. These loopcounters are maintained as a stack. The Function "cs L = xxxx" pushes the value "xxxx" on top of the stack. Since there are only four loopcounters, any attempts to push more than four values onto the stack will result in the loss of earlier pushes "out the bottom". Only the last four values will be saved.

**FUNCTION "L >" (LOOP JUMP TO)**

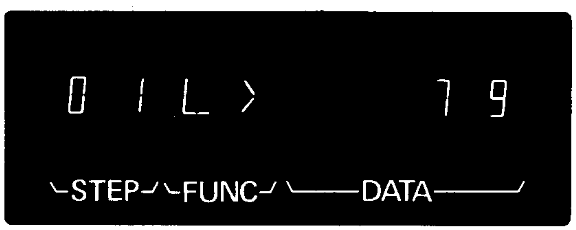



Figure 5-8 "L >" (01 is the current step number and L > 79 instructs the controller to execute the loop jump to target step 79 and follow the instruction as described on the next page.)

**(cs L > ts)** - This function is rather sophisticated. It tells the controller, "pop the top number from the stack. Subtract '1' from this value. If the value is then zero, go to the next sequential program step. If it is not zero, push the decremented value back onto the stack and go to step 'ts'." Step 'ts' is normally the step immediately following the "L = xxxx" instruction which initialized the loop being closed. When they are used this way, the group of functions or program steps between that "L = xxxx" step and its "L > ts" step will be executed "xxxx" times.

## SAFETY WARNINGS

**INSTALLATION.....**This equipment should be installed, adjusted and serviced by qualified electrical maintenance personnel familiar with the construction and operation of the equipment and the hazards involved. It is the responsibility of the equipment manufacturer or individual installing the apparatus to take diligent care when installing equipment. The National Electrical Code (NEC), sound electrical and safety codes, and when applicable, the Occupational Safety and Health Act (OSHA) should be followed when installing the apparatus to reduce hazards to person and property.

**USE.............**The chance of electric shocks, fires or explosion can be reduced by giving proper consideration to the use of grounding, thermal and over-current protection, type of enclosure and good maintenance procedures.

# SECTION 6-SOFTWARE APPLICATION INFORMATION

## INTRODUCTION

This section will explain several programming techniques that can be used to solve application problems. Before this section is read, it is important that the user understand some basic technical details of the system.

**INSTRUCTION CYCLE** - Each of the eight functions requires a finite time to perform its task. This is the instruction cycle time. The technical data section of this manual lists the instruction time for each function.

**OUTPUT STATES** - If an output is programmed to be "ON" in two or more consecutive steps, that output will be maintained continually for the duration of the steps. Consequently, there will be no "drop-out" or "bounce" during the transition from one step to the next.

## SEQUENTIAL FUNCTION DEFINITION

Sequential functions are defined as those functions which allow program flow to proceed in a normal step by step fashion. These include the count function and the three time functions.

## CONTROL FUNCTION DEFINITION

Control functions are defined as those functions which alter, perhaps conditionally, the normal step by step order of program execution. These include the GO TO, IF >, and the two loop based instructions. Loop = does not, by itself, alter the sequence. It is included here because it is always used with the L > instruction and together these functions will alter program flow.

## SEQUENTIAL FUNCTION TECHNIQUES

### CAM TIMER SIMULATION

One of the most common sequencing components is the cam timer. It has a specific cycle time and has various circuits opening and closing during that cycle time. The use of the 3 sequential timing functions (S, H or M) is all that is necessary to duplicate an electromechanical cam timer with a wide timing range and very fine resolution. The dedicated ROM-based unit becomes an economical replacement for cam timers when there are many events occurring during the cycle or when the ratio of ON time to OFF time is greater than 50:1. For developing programs to replace existing cam timer installations, a timing chart is necessary. Once the user has developed such a timing chart, programming is straight forward.

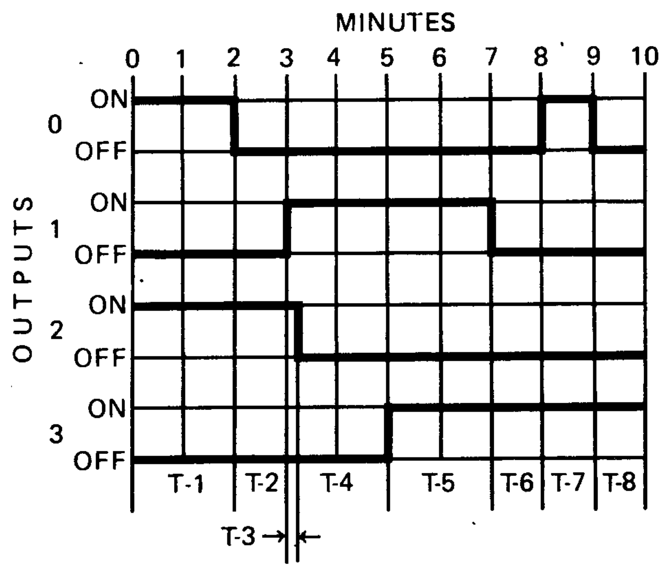



Figure 6-1 TIMING CHART

A simple timing chart (Figure 6-1) is used to help write the cam timer simulation program. In this example, a four cam timer is required with a 10 minute time cycle. The first cam switch is ON for the first 2 minutes of the time cycle, OFF for 6 minutes, ON for 1 minute and OFF for 1 minute. The second cam switch is OFF for the first 3 minutes, ON for 4 minutes and OFF for 3 minutes. The third cam switch is ON for the first 3 minutes 10 seconds and OFF for the balance of the 10 minutes. The fourth cam switch is OFF for 5 minutes and ON for 5 minutes. Outputs 0 thru 3 represent cam switches 1 thru 4. For each period between output state changes, a program step is needed with the appropriate timing function (H, S or M) and output "ON" states. T-1 thru T-8 represent these output state timing changes or steps.

The following Cam Timer Simulation Program is written by utilizing the data from the timing chart shown in Figure 6-1. The program shown is a continuously recycling cam timer simulation. To make the timer into a single cycle cam timer simulation, Step 09 would be changed to GO TO 00.

| STEP | FUNCTION | OUTPUTS 0 1 2 3 | |
|---|---|---|---|
| 01 | M 02:00 | x - x - | (T-1) |
| 02 | M 01:00 | - - x - | (T-2) |
| 03 | M 00:10 | - x x - | (T-3) |
| 04 | M 01:50 | - x - - | (T-4) |
| 05 | M 02:00 | - x - x | (T-5) |
| 06 | M 01:00 | - - - x | (T-6) |
| 07 | M 01:00 | x - - x | (T-7) |
| 08 | M 01:00 | - - - x | (T-8) |
| 09 | GO TO 01 | - - - x | |

## FIVE DECADE OR GREATER COUNTING

While the predetermining count function is rather straight forward, there are two programming techniques that will make the count function a little more versatile. The first technique allows counting beyond 4 decades. All that is required is to use a second count step with the same output state as the first to count beyond 9999.

| STEP | FUNCTION |
|---|---|
| 01 | CT9999 |
| 02 | CT0001 |

This program counts 10,000 times before continuing the process. Care must be taken in using this technique. At count frequencies above 100 Hertz, it is possible that a count can occur during the transition time from one count step to the next. Such counts will be missed by the MicroMaster©.

Count can also be used as a 5th input when a programmed wait is required. In this application, the count input would be used as a 5th normally open input. Using CT0001, the Micro-

Master© will remain on the given step until the count input closes once. Some care must be used in applying this technique. Specifically, the input must be "open" when the step is begun and then closed. If it is already closed when the step is begun, the MicroMaster© will not see the input unless it opens and closes again.

## CONTROL FUNCTION TECHNIQUES

### PROGRAMMED WAIT WITH NORMALLY OPEN INPUTS

A very common requirement in the sequential control of small machines and automated processes is the ability to leave outputs in certain states until an input or group of inputs change state. The MicroMaster© has the ability to "wait" for an input to close while maintaining a particular output state. A typical example of this is a box moving on a conveyor. The conveyor is driven by a large motor which we'll identify as output 0. The conveyor is required to move the box until it is positioned beneath a box sealing mechanism. A normally open photocell is used to detect when the box is in position. The box will break the light beam when it has reached the required position causing the photocell switch to close. For this example, we'll label the photocell input A. The photocell then becomes an input to the MicroMaster©. A program to meet these requirements involves the IF instruction and the GO TO instruction. The following program demonstrates the programmed wait technique.

Output 0 = conveyor motor control relay, normally open.

Input A = photocell wired normally open.

| STEP | FUNCTION | OUTPUTS 0 1 2 3 |
|---|---|---|
| 01 | IF A > 03 | x - - - |
| 02 | GO TO 01 | x - - - |
| 03 | PROCESS AFTER BOX POSITIONING | - - - - |

When the RUN button is pushed, MicroMaster© immediately turns output 0 on to run the conveyor and looks at input A to see if it is closed. If it is closed, it jumps to step 3 to turn the conveyor off since the box has reached the po-


Figure 6-2 DISPLAY IN RUN MODE AWAITING INPUT A TO CLOSE

sition for sealing. If input A is open, it drops to step 2. Step 2 still leaves the conveyor on (output 0) and sends the program back to step 1 (GO TO 01) to check the photocell again. The strange appearance of the display (see Figure 6-2) is caused by steps 01 and 02 being displayed on top of each other. The processor will repeat this sequence every two instruction cycles until the box finally closes input A, causing a jump to step 3.

It is important to realize that when the input closes, it must remain closed at least as long as the total instruction time of the steps involved. The timing chart in Figure 6-3 shows why a closed input may be missed if it opens in less than the total instruction time. For the two-step wait technique, the wait input must remain closed for 2 instruction cycles to insure proper performance. Refer to the technical data section for the instruction time of each function.

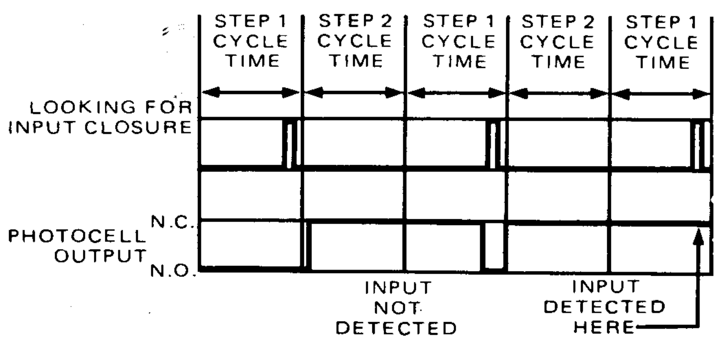



Figure 6-3 TIMING CHART FOR N.O. WAIT INPUT CLOSURE

## PROGRAM PROTECT

Since the system is not supplied with a key lock switch, the user may need to provide protection from unauthorized programming of the unit. The following program solves this problem.

Run Input - Normally Closed Keyswitch (Program Protect)

INPUT A = N.O. switch used as start sequence switch.

| STEP | FUNCTION | OUTPUTS |
|---|---|---|
| 01 | IF A > 03 | NO OUTPUTS |
| 02 | GOTO 01 | NO OUTPUTS |
| 03 | BEGIN AUTHORIZED PROGRAM | |

With the switch key locked in the N.C. position, the program will start to run at step 1 but will cycle between step 1 and 2 since input A will not close until the start sequence switch on input A is closed. With the run input closed thru the keyswitch, the operator cannot go into the program mode.

**WP6200 REAR PANEL INPUT TERMINALS**

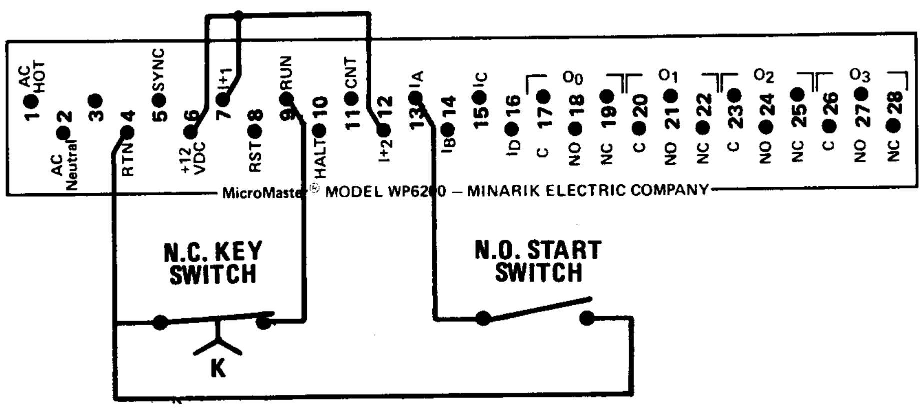



Figure 6-4 PROGRAM PROTECT WIRING DIAGRAM

## RUN PROTECT

To protect unauthorized running of a program, a key switch wired normally closed may be used as a remote reset switch (see figure 6-5).

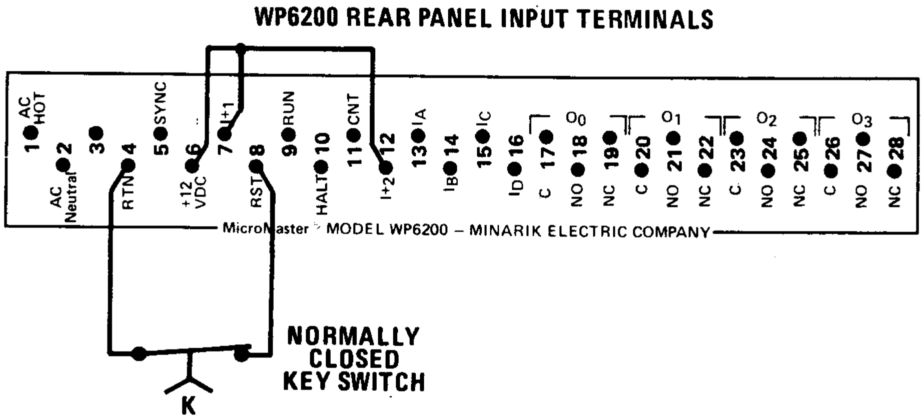



Figure 6-5 RUN PROTECT WIRING DIAGRAM

## PROGRAMMED WAIT WITH NORMALLY CLOSED INPUTS

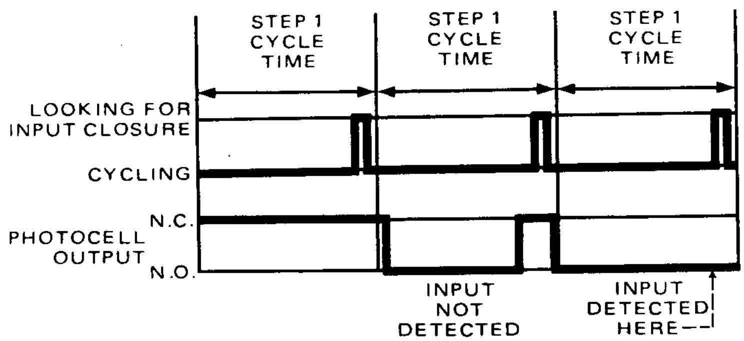
Waiting for a normally closed switch to open reduces the number of steps needed to implement the programmed wait by one. The following program is the solution to the problem described in the normally open wait example but with the photocell wired normally closed. In this example, the input must stay open for at least one instruction cycle to insure correct operation. See Figure 6-7 for timing information on normally closed inputs.

| STEP | FUNCTION | OUTPUTS 0 1 2 3 |
|---|---|---|
| 01 | IF A > 01 | x - - - |
| 02 | PROCESS AFTER BOX POSITIONING | - - - - |


Figure 6-6 DISPLAY IN RUN MODE AWAITING INPUT A TO OPEN

As long as input A is closed, the processor will jump to the target step which is step 1. Output 0 will remain on since the processor will repeat step 1 until input A opens, causing it to drop to step 2.

Due to the problems associated with breaking the oxide layer on dry contacts, it is not recommended that this technique be used when expander input group selection is required.




## MULTIPLE PROGRAM STORAGE AND SELECTION

Manual panel mounted selector switches may be used as inputs to the system to allow the operator to select one of several sequences by closing the appropriate switch. Several programs may be stored in the system as long as the total steps needed for all programs does not exceed 79, including those steps needed to scan the program selection switches. This provides for versatility in the ROM-based units, as several programs may be permanently stored to be selected by the operator.

| STEP | FUNCTION | OUTPUTS |
|---|---|---|
| 01 | IF A > 15 | ALL OFF |
| 02 | IF B > 30 | ALL OFF |
| 03 | IF C > 45 | ALL OFF |
| 04 | IF D > 60 | ALL OFF |
| 05 | GO TO 01 | ALL OFF |

-

15 Begin sequence A

-

30 Begin sequence B

-

45 Begin sequence C

-

60 Begin sequence D

In this example, input A would be labeled "sequence A start" on the operator panel. Input B would be labeled "sequence B start", input C would be labeled "sequence C start" and input D would be labeled "sequence D start."

## DRUM PROGRAMMER SIMULATION

Many applications utilize drum programming. Drum programmers provide manual operation of output sequences. Manual operation allows an operator to have complete control of when a sequence is to happen. This application thus allows machine set up before automatic operation or just general manual control of a process. The following program simulates a drum programmer with two user indexing switches. One switch (input B) is for forward indexing, the second switch (input C) is for reverse indexing.

Input B = Momentary forward sequence operator switch wired n.o.

Input C = Momentary reverse sequence operator switch wired n.o.

| STEP | FUNCTION | SEQUENCE |
|---|---|---|
| 01 | IF B > 01 | A |
| 02 | IF C > 02 | A |
| 03 | IF B > 06 | A |
| 04 | IF C > 12 | A |
| 05 | GO TO 03 | A |
| 06 | IF B > 06 | B |
| 07 | IF C > 07 | B |
| 08 | IF B > 11 | B |
| 09 | IF C > 02 | B |
| 10 | GO TO 08 | B |
| 11 | IF B > 11 | C |
| 12 | IF C > 12 | C |
| 13 | IF B > 01 | C |
| 14 | IF C > 07 | C |
| 15 | GO TO 13 | C |

If the operator continues to push only input B, the output sequence will occur in the following order with each push (ABCABCABC). If the operator pushes input C consecutively, the sequences will change as follows (ACBACBACB). Each sequence prevents a maintained closure on input B or C from causing the program to rapidly sequence through the different states.

## "AND" INPUTS

The following five step program shows how a logical "AND" is accomplished for two inputs. Before the program is allowed to arrive at step 5, input A AND input B must be closed at the same time for at least three instruction cycles to guarantee recognition.

| STEP | FUNCTION |
|---|---|
| 01 | IF A > 03 |
| 02 | GO TO 01 |
| 03 | IF B > 05 |
| 04 | GO TO 01 |
| 05 | SEQUENCE BASED ON "AND" CONDITION |

## "OR" INPUTS

The following four step program shows how a logical "OR" is accomplished for two inputs. Before the program is allowed to arrive at step 4, input A OR input B must be closed for at least three instruction cycles to guarantee proper operation.

| STEP | FUNCTION |
|---|---|
| 01 | IF A > 04 |
| 02 | IF B > 04 |
| 03 | GO TO 01 |
| 04 | SEQUENCE BASED ON "OR" CONDITION |

## INPUT WIRING FOR "OR" and "AND"

To minimize the number of inputs necessary to perform a task, wire all "AND" inputs in series and all "OR" inputs in parallel whenever possible.

## SINGLE LOOP APPLICATION

The following program shows the basic use of the looping instructions. This technique repeats a sequence as many times as defined by the L = number which allows a sequence to be repeated and then continue. In this example, a buzzer represented by output 0 will be toggled on and off at one second intervals twenty-five times before continuing at step 5. The beginning of the sequence starts at step 2, hence the L > target step is step 2. If the loop back target step was step 1, the L = would continually be redeclared and would therefore never be decremented to 0. This would prevent the program from going beyond step 4.

| STEP | FUNCTION | OUTPUTS |
|---|---|---|
| 01 | L = 0025 | |
| 02 | S 01.00 | BUZZER ON |
| 03 | S 01.00 | BUZZER OFF |
| 04 | L > 02 | |
| 05 | PROCESS AFTER LOOP COMPLETES | |

## LOOP TIMER

The following two step program shows how a timer is created using instruction time as loop counting. This approach to timing is much more functional than the S, M and H instructions because steps can be incorporated between the L = and the L > functions that will be executed each loop cycle until timed out. Step 1 of the program sets the loop counter to 9999. Step 2 decrements the loop counter by 1 and sends the program again to step 2. Step 3 will not be executed until the total time of 9999 x 1 instruction cycle has completed.

| STEP | FUNCTION |
|---|---|
| 01 | L = 9999 |
| 02 | L > 02 |
| 03 | TIMED OUT PROCESS |

## NESTED LOOPS

Please refer to the discussion of the loop stack when reviewing this section. The loop stack will keep track of up to four L = 's at any one time in the program. Nested loops are loops within loops and serve mainly to increase the total time needed when used as a loop timer. Each loop declare is limited to four decades. Nesting of loops expands total available time by a rather large factor. The following two loop nests expand the total loop timing from the previous example by a factor of 9999 x (9999 x 1 instruction cycle).

| STEP | FUNCTION |
|---|---|
| 01 | L = 9999 |
| 02 | L = 9999 |
| 03 | L > 03 |
| 04 | L > 02 |
| 05 | TIMED OUT PROCESS |

Let's go through the stack to see what is occurring. Step 1 pushes a number 9999 on top of the stack. Step 2 pushes another number 9999 onto the stack making the push in step 1 unaccessable until the push in step 2 has been popped. Step 3 is a loop compare of the most recently declared loop number which is in step 2 and at the top of the stack. The target step for the L > in step 3 is step 3. Therefore, the program will stay on step 3 (jump to itself) until the number at the top of the stack (declared at step 2) has been decremented to 0 (popped). When this occurs, the L = in step 1 will move to the top of the stack and become accessable. This will also allow the program to arrive at step 4 which compares and decrements the loop created in step 1. The target step in step 4 sends the program back to step 2 which redeclares or pushes another number onto the stack again making the current value of the remaining loop number in step 1 unaccessable (current value is 9998). The cycle repeats until the value remaining in the loop register set at step 1 is 0 (finally popping the initial L =) allowing the program to continue at step 5. As indicated in the Single Loop Application shown on the previous page, sequential functions may be inserted between L = and L > steps. When this is done, the target step for L > is usually the step immediately following L =.

## ADVANCED TECHNIQUES AND APPLICATIONS

### INPUT SAVING APPROACH TO MULTIPLE PROGRAM STORAGE

Earlier in this section a technique was shown for loading more than one program into the MicroMaster® memory and then using inputs as program selection switches. While useful, this technique used one input for each program. Another method, which looks at combinations of inputs to determine which program should be run, can be used to reduce the number of inputs required. This example shows this technique being used to select one of four programs using only two inputs. It is easily expanded to three inputs and eight programs or four inputs and sixteen programs. To be practical, the program selection switch should be of the rotary type with one pole for each input used and one position for each program in the MicroMaster®. The switch must be set before the "run" button is pushed. The method is demonstrated graphically in the flow chart in Figure 6-8. Input A is checked first (step 01). The state of this input is used to select one of two pairs of possible programs. Input B is then checked (steps 02 and 04) to determine which of the two programs in the selected pair will be run.

INPUT TRUTH TABLE

1 = CLOSED, 0 = OPEN

| INPUT A | INPUT B | |
|---|---|---|
| 0 | 0 | SEQUENCE A |
| 0 | 1 | SEQUENCE B |
| 1 | 0 | SEQUENCE C |
| 1 | 1 | SEQUENCE D |

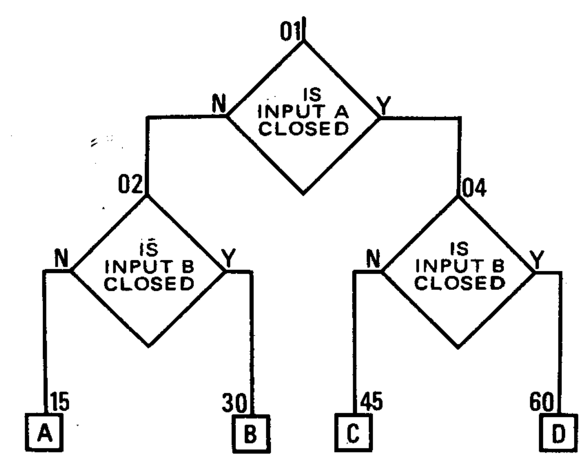



Figure 6-8 INPUT SAVING FLOW CHART

PROGRAMMING SOLUTION OF INPUT TRUTH TABLE

| STEP | FUNCTION | OUTPUTS |
|---|---|---|
| 01 | IF A > 04 | NO OUTPUTS |
| 02 | IF B > 30 | NO OUTPUTS |
| 03 | GO TO 15 | NO OUTPUTS |
| 04 | IF B > 60 | NO OUTPUTS |
| 05 | GO TO 45 | NO OUTPUTS |

15 BEGIN SEQUENCE A

-

30 BEGIN SEQUENCE B

-

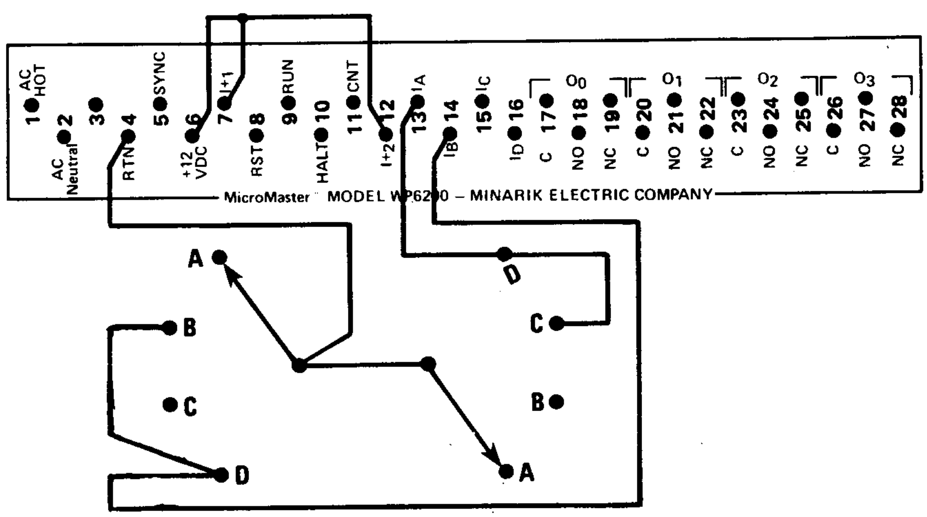
45 BEGIN SEQUENCE C

-

60 BEGIN SEQUENCE D

The only difference with this approach is that the sequence must be selected prior to pushing the run button versus a continuous scan of four inputs until the operator closes one as in the previous·example.

**WP6200 REAR PANEL INPUT TERMINALS**




**2 pole, 4 position ROTARY SWITCH**

Figure 6-9 INPUT SAVING WIRING DIAGRAM

## INTERRUPTIVE INPUT SCANNING

The following program uses the L = and L > to generate a time delay which may be interrupted by an external input.

The loop counter is initially set to 7500. The L > at step 03 will normally send the program back to step 02 7500 times. The scan time for each loop is 4 ms (2 ms for step 02 and 2 ms for step 03). 7500 x 4 = 30000 ms or 30 seconds. If input A does not close during this time, the program will go on to step 04 when this time delay is complete. If input A does close during this time, the program will branch to the interrupt process at step 10. The loop may be expanded to include other "IF >" statements if more than one interrupting signal is required. The loop number would have to be adjusted to allow for the new loop time.

INPUT A - INTERRUPT TIMING IF CLOSED

| STEP | FUNCTION | OUTPUTS |
|---|---|---|
| 01 | L = 7500 | ALL OFF |
| 02 | IF A > 10 | ALL OFF |
| 03 | L > 02 | ALL OFF |
| 04 TIMED OUT PROCESS | | |
| -- | | |
| -- | | |
| 10 INTERRUPTED PROCESS | | |

With the loop counter set to 7500, the loop back (step 03) will send the program to step 02 7500 times unless input A changes state causing a jump to step 04. The scan time for each cycle is 4 ms (2 ms for step 02 and 2 ms for step 04). 4 x 7500 = 30000 ms total or 30 seconds. If input A does not change state for that 30 seconds, the program will go to step 04 when the loop counter finally has been decremented to 0. With this approach, the 30 second timer has the ability to check an input every 2 ms while timing. The open loop timer version of this without the interrupt ability would simply contain a step of "S 30.00". For each step added within the loop, the additional scan time must be calculated to determine the right loop number for the total cycle time. For instance, it may be required to check two or more inputs during this cycle time with two or more possible interrupt sequences based on which one is closed.

## MAINTAINING CONSTANT CYCLE TIME IN TWO PROCESSES

Earlier in this section, a technique was shown which made it possible to allow an input to interrupt a time delay. An expansion of this technique is sometimes useful when the original time delay must be maintained. An example will help illustrate this. Suppose a certain manufacturing process requires that water be delivered from a tank for a fixed period of time whenever a demand switch is activated. It is also necessary to maintain the water in the tank between two levels which will be indicated by float switches.

The necessary inputs and their input numbers are as follows:

IA - DEMAND SWITCH, CLOSE TO START CYCLE
IB - LOWER FLOAT SWITCH, CLOSE WHEN LOW
IC - UPPER FLOAT SWITCH, CLOSE WHEN FULL

Outputs are assigned as follows:

00 - TANK OUTLET VALVE
01 - TANK INLET VALVE

| STEP | FUNCTION | OUTPUTS 0 1 2 3 | COMMENTS |
|---|---|---|---|
| 01 | IF A > 07 | - - - - | NO DEMAND |
| 02 | IF B > 04 | - - - - | NO FILL |
| 03 | GO TO 01 | - - - - | |
| 04 | IF A > 11 | - x - - | NO DEMAND |
| 05 | IF C > 01 | - x - - | FILL |
| 06 | GO TO 04 | - x - - | |
| 07 | L = XXXX | x - - - | DEMAND |
| 08 | IF B > 13 | x - - - | NO FILL |
| 09 | L > 08 | x - - - | |
| 10 | GO TO 01 | x - - - | |
| 11 | L = XXXX | x x - - | DEMAND |
| 12 | IF C > 09 | x x - - | FILL |
| 13 | L > 12 | x x - - | |
| 14 | GO TO 04 | x x - - | |

The process first starts by using a programmed wait on two inputs. Nothing will occur beyond step 03 until either the demand switch (input A) is closed or the low float switch (input B) closes indicating that water in the tank has reached the lower level.

If the demand switch closes before the low float switch does, the program is sent to step 07 to set a loop timer (L = XXXX) for the duration of the process. It then goes to step 08 to look at the lower float valve (input B) while pumping water out of the tank (output 0). Should the lower switch close during this timed process, the program is sent to step 13 which continues the loop timing and also allows more water into the tank by turning output 1 on. The program will time out in steps 12 and 13 unless more water is coming in than is going out causing the upper float switch to eventually close (input C) which will send the program back to step 09; completing the timing process without allowing any more water in the tank (output 1 off).

If the low float switch closes before the demand switch is pressed, the process will start at step 04 allowing water to enter until the demand switch is pressed (input A) which starts the loop timer at step 11 or until the upper float switch (input C) turns off the inlet valve (output 1).It then would send the program back to await input A or B to close at steps 01 and 02. Assuming that the demand switch is pressed at step 4 before water reaches the upper limit, the process then starts at step 11 which sets the loop timer and allows water to enter starting at step 12 until either the timer times out or the upper float switch closes causing the timing to continue at step 09.

The loop timer for the process will be decremented either by L > at step 09 or the L > at step 12 depending on the level of water during the total cycle time as sensed by the upper and lower float switches.

This process allows one of two processes to occur within a given cycle time by using one of two L > instructions alternately based on the condition of two inputs.

## TWO HANDED MACHINE START (ANTI TIE-DOWN CODE)

Manufacturers of machinery are often required to provide their equipment with a "two handed" anti-tiedown start. This consists of two physically separated buttons which must be pressed simultaneously to start the equipment. This is to insure that the operators hands are clear of the machine before it can be operated. The anti-tiedown feature prevents the operator from defeating the device by jamming one of the buttons. This program shows a method of implementing this function. The sequence can only be started if the two buttons are pressed within .2 seconds of each other.

The following program suggests an approach to use. It does not guarantee safety; it establishes a technique for safer machine operation. The designer of the machine or process is ultimately responsible for safe operation by designing both machine hardware and software with operator safety in mind.

INPUT A - LEFT HAND START BUTTON
INPUT B - RIGHT HAND START BUTTON

NO OUTPUTS ON FOR THIS SEQUENCE

| STEP | FUNCTION | COMMENTS |
|---|---|---|
| 01 | IF A > 04 | WAIT FOR LEFT OR |
| 02 | IF B > 09 | RIGHT HAND INPUT |
| 03 | GO TO 01 | TO CLOSE |
| 04 | L = 50 | WAIT FOR RIGHT INPUT |
| 05 | IF B > 14 | TO BE PUSHED WITHIN |
| 06 | L > 5 | 202 MS OF LEFT HAND PUSH |
| 07 | IF A > 07 | WAIT FOR RELEASE OF I-A |
| 08 | GO TO 01 | START OVER |
| 09 | L = 50 | WAIT FOR LEFT INPUT |
| 10 | IF A > 14 | TO BE PUSHED WITHIN |
| 11 | L > 10 | 202 MS OF RIGHT HAND PUSH |
| 12 | IF B > 12 | WAIT FOR RELEASE OF I-B |
| 13 | GO TO 01 | START OVER |
| 14 | EXECUTE PROCESS | |

## PRESET INDEXING OF STEPPING MOTORS

Stepping motors and preset indexers are often used in positioning applications. By using Type 2 outputs from the MicroMaster©, and by utilizing both scan time and loop instructions, the MicroMaster© system can be programmed to run stepping motors at specific speeds and to preset index the stepping motor shaft to precise locations. Due to the relatively slow scan time of the system, the standard maximum pulse rate available is 250 hz. However, special software is also available to increase this rate at the expense of other standard features. The following program and wiring diagram is based on the use of a Slo-Syn© stepping motor system.

**TYPE 2 OUTPUT REQUIRED FOR STEPPER MOTOR INTERFACE**

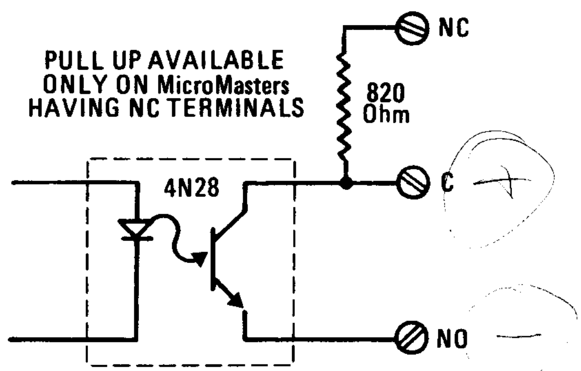



Figure 6-10 TYPE 2 OUTPUT

### STEPPING MOTOR PROGRAM FOR SLO-SYN© SYSTEM

OUTPUT 0 = MOTOR CW
OUTPUT 1 = MOTOR CCW

| STEP | FUNCTION | OUTPUTS 0 1 2 3 |
|---|---|---|
| 01 | L = 200 | - - - - |
| 02 | GO TO 03 | x - - - |
| 03 | L > 02 | - - - - |
| 04 | S 02.00 | - - - - |
| 05 | L = 100 | - - - - |

| | | |
|---|---|---|
| 06 | S 00.02 | - x - - |
| 07 | L > 06 | - - - - |
| 08 | S 00.50 | - - - - |
| 09 | GO TO 01 | - - - - |

The above program will pulse the stepping motor 200 steps CW at 250 HZ, then stop for 2 seconds and pulse the motor 100 steps CCW at 34 HZ, then stop for 1/2 second, then repeat. The pulse train frequency is determined by the scan time or delay time of the step that turns on the output. In step 2, the GO TO instruction is used because it has the quickest scan time (about 2 ms or 250 HZ.). To slow the speed, a time step or slower scan instruction can be placed in the step that generates a pulse as shown in step 6. S 00.50 would generate a pulse train of about 34 HZ. For each pulse, the stepping motor will move its normal step angle. In the case of a Slo-Syn© with the STM-103 translator, the normal step angle is 1.8 degrees of arc, Therefore, by setting the loop counter to 200 or 200 pulses before going to step 4, the stepping motor will rotate exactly 1 revolution before stopping. (200 x 1.8 degrees = 360 degrees). With other types of translator modules that allow microstepping for greater resolution, the same MicroMaster© program will allow even greater positioning control of the stepping motor system.

**WIRING TO Slo-Syn TRANSLATOR**
**MODEL STM-103**

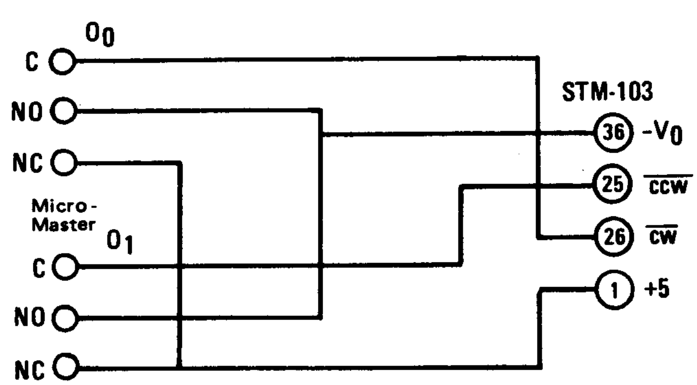



Figure 6-11 WIRING DIAGRAM FOR SLO-SYN© TRANSLATOR

## SUBROUTINE DEFINED

A subroutine is defined as a set of instructions that can be executed from any point within a program and then return to the point of call to continue once the subroutine has been executed. Since the number of available steps in the Micro-Master® is limited to 79, the use of subroutines may allow a program to be executed which otherwise might be too large to fit in the 79 available steps. The subroutine must be at least four steps long to provide a reduction in the number of steps required since it takes three steps to call for a subroutine. The following program uses the L = and L > functions to create subroutine calls. A review of the operation of these instructions will be helpful in understanding the program operation.

The L = value will be equal to the total amount of subroutine calls needed + 1.
In this example, we'll call for a subroutine 3 times, therefore L = 4.

| STEP | FUNCTION | COMMENTS |
|---|---|---|
| 01 | | INITIAL EXECUTION POINT |
| -- | | OF A GIVEN PROGRAM. |
| -- | | |
| -- | | |
| 10 | L = 4 | FIRST SUBROUTINE CALL |
| 11 | L > 21 | |
| 12 | | CONTINUE PROGRAM AFTER |
| -- | | SUBROUTINE HAS COMPLETED |
| -- | | FIRST EXECUTION |
| -- | | |
| 20 | L = 4 | SECOND SUBROUTINE CALL |
| 21 | L > 31 | |
| 22 | | CONTINUE PROGRAM AFTER |
| -- | | SUBROUTINE HAS COMPLETED |
| -- | | SECOND EXECUTION |
| -- | | |
| 30 | L = 4 | THIRD SUBROUTINE CALL |
| 31 | L > 50 | |
| 32 | | CONTINUE PROGRAM AFTER |
| -- | | SUBROUTINE HAS COMPLETED |
| -- | | THIRD EXECUTION |

```
--
45      GO TO 00
--
--
50                        BEGIN SUBROUTINE AND
--                         RETURN TO THE POINT
--                         THAT IT WAS CALLED
--                        FOR AFTER COMPLETION
--
60      GO TO 11
```

Now for a point by point explanation. The subroutine in this program starts at step 50 and ends at step 60. The first call is at step 10 which pushes a number 4 onto the loop stack. Step 11 is the comparison of the number at the top of the stack (the most recently declared L = which is at step 10). It decrements the number at the top of the stack by 1. If the remaining number is greater than 0, it causes a jump to the target step. In the first execution, the number at the top of the stack was 4. Subtracting 1 gives 3 which is greater than 0. Therefore, it will jump to the target step. The target step in this case is step 21. Had the remaining loop number been 0, it would not jump to step 21, but would go to the next sequential step. In this case that would be step 12. Step 21 requires another decrement and comparison of the number in the register (reduced to 2) and jumps to the target step 31. Step 31 requires another decrement and comparison of the number in the register (reduced to 1) and jumps to the target step 50. Step 50 is the starting point of the subroutine. The last step in the subroutine sends the processor back to step 11 which makes another comparison to the top of the stack. The number at the top before the comparison is 1 since it has been decremented 3 times already. Now, when the processor decrements the number at the top of the stack again the remaining number is 0. The L = declared at step 10 is finally "popped" and the processor will now send execution to step 12 rather than step 21. This process allows steps 50 through 60 to be executed and the program to be continued from the point of the initial subroutine call.

Now, let's see what happens when we call for the subroutine again. The subroutine is required again starting at step 20.

This second call pushes a number 4 onto the loop stack. Just as it did at step 11. The program execution proceeds as it did in the first call, jumping first to step 31 and then to step 50. When the subroutine completes, the program then jumps to step 11, however, the loop will have only been decremented to two, step 11 will decrement it to one and send it on to step 21. Step 21 will finally decrement it to 0 and the program will continue at step 22. The third call at steps 30-31 will operate in a similar fashion.

## TECHNIQUES CONCLUSION

This completes the discussion of popular programming techniques. There are other techniques being used and discovered every day. These basic techniques will allow the potential user to create programming solutions to small scale problems and enhance his knowledge of programmed control.

## PROGRAMMING WITH THE EXPANDER

## BASIC PROGRAM ENTRY

The use of expander products become necessary when more than four inputs or more than four outputs are needed. The expander system must operate with the main MicroMaster© system and uses two wire serial data lines for receiving data from the WP6200. Program entry to the expander is similar to the WP6200. The expander is an extension of the main system which allows more outputs to be controlled and expands the use of the IF instruction thru input group selection.

The orange buttons control the eight outputs in exactly the same manner as the I/O buttons on the WP6200 controller. The green buttons control input group selection in a similar fashion. The step being programmed is controlled by the WP6200 and is the step displayed on the WP6200.

## INPUT PROGRAMMING INSTRUCTIONS WHEN USING EXPANDERS

The WP6200 MicroMaster© is limited to 4 user definable INPUTS. These user definable INPUTS can be increased to 7 by adding a WP6300 Expander and if a WP6350 Input Expander Diode Matrix is also used, the INPUTS are increased to 16.

The following listings illustrate how to program these additional inputs by showing what should be displayed on the WP6200's alphanumerical display and which lights should be lighted on the WP6300's group indicator lights for each individual input.

### 7 INPUTS USING A WP6200 AND A WP6300

| INPUT | WP6200 DISPLAY | WP6300 GROUP LIGHTS | | | |
|---|---|---|---|---|---|
| | | G/0 | G/1 | G/2 | G/3 |
| A | cs IF A > ts | OFF | OFF | OFF | OFF |
| B | cs IF B > ts | OFF | OFF | OFF | OFF |
| C | cs IF C > ts | OFF | OFF | OFF | OFF |
| D-G/0 | cs IF D > ts | ON | OFF | OFF | OFF |
| D-G/1 | cs IF D > ts | OFF | ON | OFF | OFF |
| D-G/2 | cs IF D > ts | OFF | OFF | ON | OFF |
| D-G/3 | cs IF D > ts | OFF | OFF | OFF | ON |

### 16 INPUTS USING A WP6200, A WP6300 AND A WP6350

| INPUT | WP6200 DISPLAY | WP6300 GROUP LIGHTS | | | |
|---|---|---|---|---|---|
| | | G/0 | G/1 | G/2 | G/3 |
| A-G/0 | cs IF A > ts | ON | OFF | OFF | OFF |
| B-G/0 | cs IF B > ts | ON | OFF | OFF | OFF |
| C-G/0 | cs IF C > ts | ON | OFF | OFF | OFF |
| D-G/0 | cs IF D > ts | ON | OFF | OFF | OFF |
| A-G/1 | cs IF A > ts | OFF | ON | OFF | OFF |
| B-G/1 | cs IF B > ts | OFF | ON | OFF | OFF |
| C-G/1 | cs IF C > ts | OFF | ON | OFF | OFF |
| D-G/1 | cs IF D > ts | OFF | ON | OFF | OFF |
| A-G/2 | cs IF A > ts | OFF | OFF | ON | OFF |
| B-G/2 | cs IF B > ts | OFF | OFF | ON | OFF |
| C-G/2 | cs IF C > ts | OFF | OFF | ON | OFF |
| D-G/2 | cs IF D > ts | OFF | OFF | ON | OFF |
| A-G/3 | cs IF A > ts | OFF | OFF | OFF | ON |
| B-G/3 | cs IF B > ts | OFF | OFF | OFF | ON |
| C-G/3 | cs IF C > ts | OFF | OFF | OFF | ON |
| D-G/3 | cs IF D > ts | OFF | OFF | OFF | ON |

## EXPANDER PROGRAM LOSS

The loss of a program loaded into RAM is possible for the same reasons as discussed for the "P-FAIL" display to occur on the main system. Since there is no alphanumeric display on the expander units, the only visual sign of a program loss on the expander is a lack of any display lights for both the internal group select switches and the outputs. All outputs will go to their off state if a memory loss occurs. The RAM in the expander only keeps track of the additional outputs and the internal group selection switches.

## TESTING PROGRAMS

Testing of the program can be accomplished in the same manner as the WP6200. Refer to the fourth paragraph on page 4 for details.

## INPUT "OR" TECHNIQUES

As discussed earlier, the MicroMaster© system can be programmed for logical "OR" and logical "AND" input scanning. Where inputs are to be "ORed", some program space may be conserved if the inputs are connected to like input numbers in different groups. It is then possible to check both inputs using one IF > instruction by selecting their associated input groups simultaneously. The following example will illustrate this technique.

X = Programmed on

| STEP | FUNC | GROUP LIGHTS 0 | 1 | 2 | 3 |
|---|---|---|---|---|---|
| 01 | IF B > 10 | X | X | X | X |

Any one of four independant inputs all wired into input B terminals will cause a jump to step 10 if closed.

## PROGRAMMED WAIT LIMITATIONS

Implementation of the programmed wait using normally closed switches is not recommended when using expander products. Refer to page 83, "Normally Closed Contacts Used With Expanders" for explanation.

## PROBLEM SOLVING TECHNIQUES

In order to solve a systems problem with a MicroMaster®, it is important to approach the solution in the proper manner. This approach should be used in even elementary problems, as it will minimize errors and establish documentation that will make future modifications easy and clear.

The MicroMaster® is a Sequential Process Controller. Other process controllers are generally defined as "Ladder Diagram" Processors. In this type of system, the complete ladder diagram is scanned in a fixed time, which is a function of the length of the program. States of inputs and outputs as well as timers and counters are stored in memory and then acted upon as the processor reaches the end of the program in the ladder diagram.

The MicroMaster® differs from this in that it is a sequentially programmed device. It acts upon, and completes, each instruction as it is encountered in each step of the program. In some cases, this is a decided advantage. For example, a ladder type process controller typically cannot do loops, or nested loops as can a MicroMaster®. This means that each set of instructions that could be handled by L = and L > in the MicroMaster®, must be programmed individually in a ladder type process controller. In this case, the number of program steps is much less in the MicroMaster®.

Since the MicroMaster® is a sequential device, the approach to solving a system problem is to break down the machine operation into a series of sequential operations. Generally, this is how machines are designed. It is only after a machine is designed that the designer creates a ladder diagram to serve as a connection diagram for the user. Therefore, a listing of the machine sequence is the first step in problem solving for a MicroMaster® application.

At this point, identification of inputs and outputs is possible. It is critical that specific identification, such as Input A = Start Switch and Output 0 = Main Drive Motor, be made for each input and output of the system. Next, using "The Instruction Set" and "Programming Techniques" explained earlier in this manual, a list of instructions, or program may be developed. Often, it is useful to write this in the form of a flow chart. An example of a flow chart is shown in figure 6-13 on page 59.

Finally, the program should be listed on the MicroMaster© programming form enclosed with each MicroMaster© product. It is a good idea to leave spaces in the list of instructions, so that changes or corrections can be made without rewriting the entire program. For example:

| STEP | FUNCTION | OUTPUTS<br>0 1 2 3 |
|---|---|---|
| 01 | IF B > 03 | - - - - |
| 02 | GO TO 01 | - - - - |
| 03 | S 01.00 | x - - - |
| 04 | GO TO 07 | - - - - |
| 05 | GO TO 05 | - - - - |
| 06 | GO TO 06 | - - - - |
| 07 | S 02.00 | - x - - |

If it is found that another time delay is required after Step 03, it can be inserted without rewriting all the instructions that follow. For example:

| STEP | FUNCTION | OUTPUTS<br>0 1 2 3 |
|---|---|---|
| 01 | IF B > 03 | - - - - |
| 02 | GO TO 01 | - - - - |
| 03 | S 01.00 | x - - - |
| 04 | S 05.00 | - - - x |
| 05 | GO TO 07 | - - - - |
| 06 | GO TO 06 | - - - - |
| 07 | S 02.00 | - x - - |

The following example illustrates a point by point procedure that will help in the development of an application program:

POINT 1 - Define the overall function that is to be accomplished by the control system:

A conveyor control is required to keep track of the movement of two conveyors that are being used to transfer stereo speakers into a baking oven for curing. The speakers are randomly spaced on one conveyor and must be evenly spaced on a second conveyor which is positioned at a right angle to the first conveyor for proper indexing into the oven by a pusher mechanism. The process is continuous unless stopped by the operator or unless the pusher mechanism jams. If the pusher mechanism jams, an alarm must sound.

POINT 2 - Describe a step by step operation (refer to figure 6-12 shown on page 58 for a graphic illustration):

The sequence is started by the pushing of a start button by the machine operator. Motor "A" is started and drives conveyor "A" until the first of randomly spaced speakers drops off conveyor "A" onto conveyor "B". This is sensed by photocell "A". At this point, motor "A" is stopped. The limit switch at the end of conveyor "B" is checked to see if the line is full. If it is, the pusher solenoid is activated for five seconds, moving the speakers into the oven. A three second time delay at this point allows the pusher solenoid to retract. Limit switch "C" is checked during this time delay to verify that the pusher solenoid has retracted. If it has not retracted at the end of three seconds, the process is stopped and an alarm will sound. If the pusher solenoid has retracted or the line was not full when limit switch "B" was checked, conveyor "B" is activated for 1.3 seconds. The whole process is then repeated. There is also a requirement for an emergency stop push button.

POINT 3 - Define all inputs and outputs that are needed.

From the description in Point 2, the following inputs and outputs may be defined:

**INPUTS**

Start, Emergency Stop, Photocell "A", Limit Switch "B", and Limit Switch "C".

## OUTPUTS

Conveyor Motor "A", Conveyor Motor "B", Pusher Solenoid, and Alarm

We now assign the inputs and outputs as follows:

## INPUTS

| | |
|---|---|
| Start | = RUN terminal on MicroMaster® |
| Emergency Stop | = HALT terminal on MicroMaster® |
| Photocell "A" | = INPUT A on MicroMaster® |
| Limit Switch "B" | = INPUT B on MicroMaster® |
| Limit Switch "C" | = INPUT C on MicroMaster® |

## OUTPUTS

| | |
|---|---|
| Pusher Solenoid | = OUTPUT 0 on MicroMaster® |
| Conveyor Motor "A" | = OUTPUT 1 on MicroMaster® |
| Conveyor Motor "B" | = OUTPUT 2 on MicroMaster® |
| Alarm | = OUTPUT 3 on MicroMaster® |

POINT 4 - Utilizing the data from Points 2 and 3, write a step to step program description which will be used to write the final program.

In step 1, we look at Input A which is a normally closed photocell at the end of conveyor "A". Output 1, which is the motor for conveyor "A", is turned on during this step. When the photocell opens, we drop to step 2, which is a 1/2 second delay for allowing the speaker to settle on conveyor "B". In this step, all of the outputs are off. After 1/2 second has elapsed, we move to step 3 which checks Input B, the limit switch at the end of conveyor "B" to see if the line is full. Since this limit switch is also normally closed and if the line is not full, we will move to step 9. Step 9 is a 1.3 second time delay with Output 2 on. Output 2 is the motor for conveyor "B" and this indexes the speakers along conveyor "B". We now drop to step 10, which tells us to repeat this cycle again. We will continue to repeat it until Input B opens, indicating that a speaker has reached the end of conveyor "B". In step 4, we activate the pusher mechanism for 5 seconds to push speakers into the oven. At this point, Output 0 means the pusher solenoid is on. At steps 5, 6 and 7, we scan the switch Input C, on the pusher mechanism, to see if it has retracted. The time required to scan this in steps 5, 6 and 7 is approximately 3 seconds. At step 6, there's a 2 millisecond instruction time and also at step 7. Both of these total 4 milliseconds times 750 which is 3 seconds. Finally, if 3 seconds, or 750 loops elapse without Input C closing we go to step 8 which sounds an alarm. The program will stay on step 8 until the RESET button is pushed on MicroMaster®.

POINT 5 - Utilizing data in Point 4 write the final program:

| STEP | FUNCTION | OUTPUTS | | | |
|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 |
| 01 | IF A > 01 | - | x | - | - |
| 02 | S 00.50 | - | - | - | - |
| 03 | IF B > 09 | - | - | - | - |
| 04 | S 05.00 | x | - | - | - |
| 05 | L = 750 | - | - | - | - |
| 06 | IF C > 09 | - | - | - | - |
| 07 | L > 06 | - | - | - | - |
| 08 | GO TO 08 | - | - | - | x |
| 09 | S 01.30 | - | - | x | - |
| 10 | GO TO 01 | - | - | - | - |

A completed Programming Form for the above program is shown in figure 6-14 on page 60. This approach is only one of several possible solutions to this problem. As long as the 79 steps are not exceeded, any given solution to a problem is as good as another.

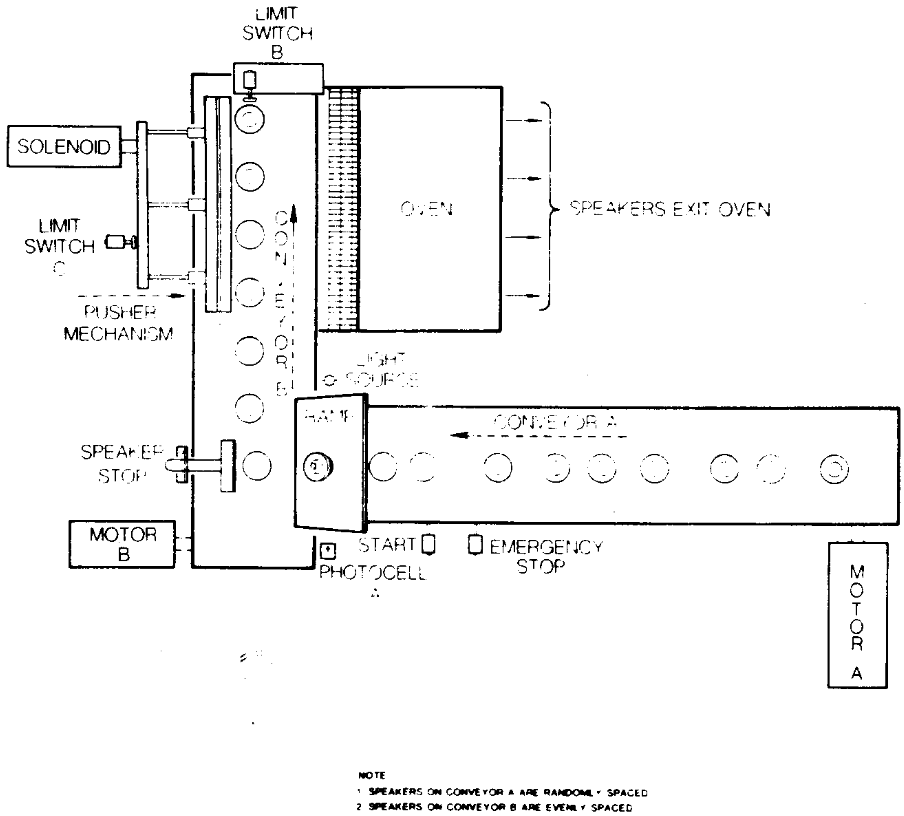



Figure 6-12 CONVEYOR GRAPHIC

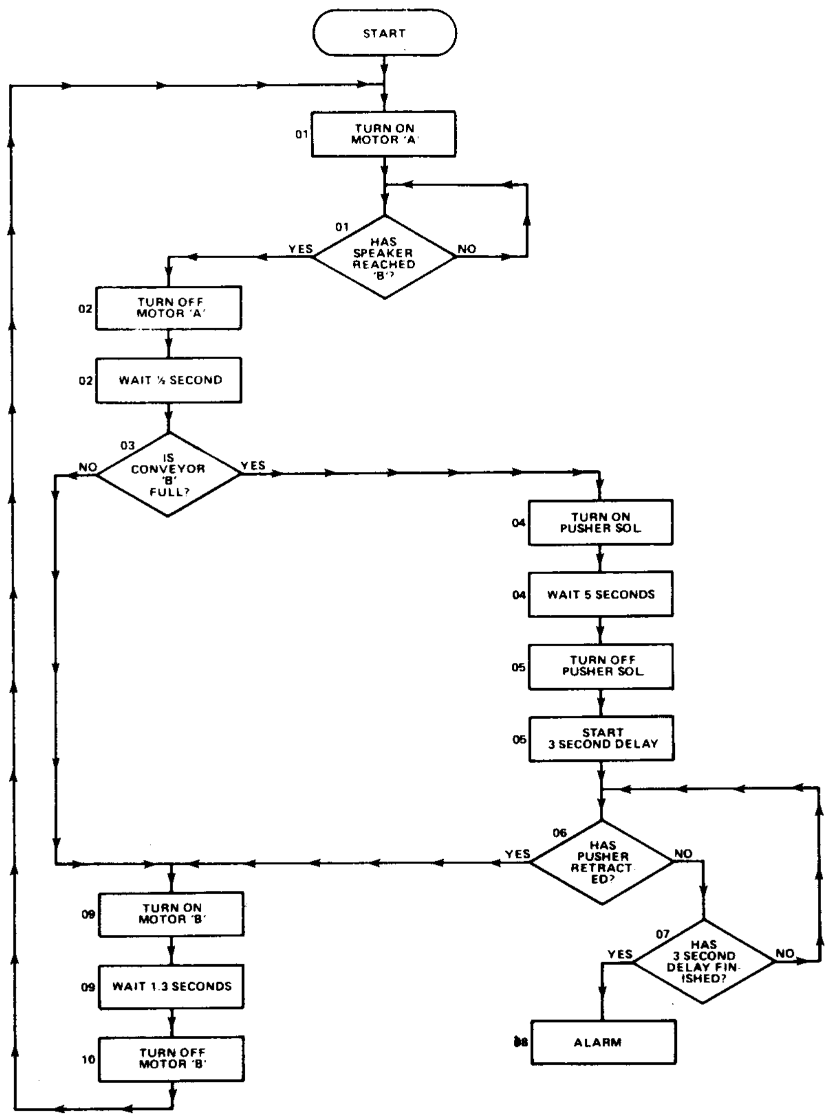



Figure 6-13 FLOW CHART

**MicroMaster** PROGRAMMING FORM FOR SERIES WP6200 and WP6300

PROGRAM CONVEYOR CONTROL DATE ............ PAGE 1 of 1 PAGES

................ SERIES WP6200 ONLY ................

INPUT A PHOTOCELL "A" INPUT B SWITCH "B" INPUT C SWITCH "C" INPUT D ............

OUTPUT 0 PUSHER SOL OUTPUT 1 MOTOR "A" OUTPUT 2 MOTOR "B" OUTPUT 3 ALARM

................ SERIES WP6200 and WP6300 ................

INPUTS (Example = INPUT A-1 DISPLAY = -̇ꜰ L GROUP =)

INPUT A-0 ............ INPUT A-1 ............ INPUT A-2 ............ INPUT A-3 ............

INPUT B-0 ............ INPUT B-1 ............ INPUT B-2 ............ INPUT B-3 ............

INPUT C-0 ............ INPUT C-1 ............ INPUT C-2 ............ INPUT C-3 ............

INPUT D-0 ............ INPUT D-1 ............ INPUT D-2 ............ INPUT D-3 ............

OUTPUTS

OUTPUT 0 ............ OUTPUT 1 ............ OUTPUT 2 ............ OUTPUT 3 ............

OUTPUT 4 ............ OUTPUT 5 ............ OUTPUT 6 ............ OUTPUT 7 ............

OUTPUT 8 ............ OUTPUT 9 ............ OUTPUT 10 ............ OUTPUT 11 ............

| STEP | FUNCTION | WP6200 OUTPUTS | | | | WP6300 OUTPUTS | | | | | | | | GROUP | | | | COMMENTS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 0 | 1 | 2 | 3 | |
| Ø1 | IF A → Ø1 | — | X | — | — | | | | | | | | | | | | | MOVE "A" UNTIL SPEAKER DROPS |
| Ø2 | SS ØØ.5Ø | — | — | — | — | | | | | | | | | | | | | SETTLING TIME |
| Ø3 | IF B → Ø9 | — | — | — | — | | | | | | | | | | | | | SEE IF LINE IS FULL |
| Ø4 | SS ØØ.5Ø | X | — | — | — | | | | | | | | | | | | | PUSH SPEAKER TO OVEN |
| Ø5 | L=2ØØ | — | — | — | — | | | | | | | | | | | | | WAIT 3 SECONDS FOR RETRACT |
| Ø6 | IF C → Ø9 | — | — | — | — | | | | | | | | | | | | | RETRACT O.K., CONTINUE |
| Ø7 | L → Ø6 | — | — | — | — | | | | | | | | | | | | | END OF 3 SECOND DELAY LOOP |
| Ø8 | GOTO Ø8 | — | — | — | X | | | | | | | | | | | | | PUSHER JAMMED ALARM |
| Ø9 | SS Ø1.3Ø | — | — | X | — | | | | | | | | | | | | | MOVE SPEAKER ON "B" FOR 1.5 SECONDS |
| 1Ø | GOTO Ø1 | — | — | — | — | | | | | | | | | | | | | REPEAT SEQUENCE |

Figure 6-14 COMPLETED PROGRAMMING FORM

## SECTION 7 - HARDWARE APPLICATION INFORMATION

### SYSTEM CONFIGURATION

A typical MicroMaster® system consists of a series WP6200 product which provides connections for 8 inputs, 4 outputs, 115 vac 50/60 Hertz power and if necessary, expansion series WP6300 products. The expander products are connected via serial data lines (2 wire, sync and return). The following information is to be used to reduce the possibility of incorrect operation caused by spurious signals and general noise caused by mixing different types of signals in a given installation.

### INSTALLATION INSTRUCTIONS

### GENERAL INFORMATION

The MicroMaster® system has two basic packages for machine or process control. The ROM based chassis models (WP6220 and WP6320) are designed to be surface mounted inside control cabinets. NEMA rated cabinets may be necessary depending on the particular enviornment. The location surrounding the chassis system should be free of mechanical shock, heavy vibration, atmospheric contaminants such as water spray, dust, oil mist etc., and extreme temperatures.

The other type of package offered in the MicroMaster® line is the panel mount products (WP6240, WP6200, AND WP6300). These products fit into panel cavities with the following dimensions:

3" High x 5-3/8" Wide x 7-1/4" Deep

If airborne contaminants are near the process or machine, it is recommended that panels be protected in appropriate NEMA rated cabinets with subpanels. Remote operation of the run, halt, and reset inputs can be accomplished using appropriate switches connected to the switch terminations on the rear of the product.

### CLEARANCES

When installing the MicroMaster® dedicated chassis system, provisions should be made for easy access to the unit so the PC board can easily be removed from its case for replacement and/or changes in the PROM chip, the plug-in input modules and the plug-in output modules. For panel mount products, there should be adequate room behind the product to allow easy connection of all external devices to the terminal block. Dimensional data is shown in Section 8 on pages 103 and 104.

## WIRING PRECAUTIONS

All MicroMaster® products should be wired in accordance with the National Electrical Code, federal, state, and local codes, and/or any other applicable safety codes. In addition, the user should turn off power and take other necessary precautions during installation, service and repair to prevent personal injury and equipment damage. All MicroMaster® products are internally fused to help protect their circuits from abnormal current flows due to incorrect hookup and/or internal component malfunctions.

To connect wiring to the MicroMaster® terminal blocks:

1. Strip or remove approximately 5/16" of the wire's insulation.

2. Using a small screwdriver, loosen the appropriate numbered terminal screw by turning the screw counter-clockwise.

3. Insert the uninsulated end of the wire into the hole directly beneath the terminal screw.

4. Hold the wire in place and tighten the terminal screw by turning it clockwise.

**IMPORTANT!:** To prevent damage to internal components and the PC board etched contact fingers **NEVER** remove or install the terminal block edge connector on the MicroMaster® without first disconnecting all power from the terminal block.

A power switch is recommended on all ROM-based units to control power to all elements. In the case of RAM-based units, in order to prevent program loss due to a discharged or faulty battery, it is recommended that AC power be continuously applied to the product. Even if the panel mounted power switch is turned off, applied AC power is still charging the battery and holding the RAM memory. In this instance, the master power switch should only interrupt the load power.

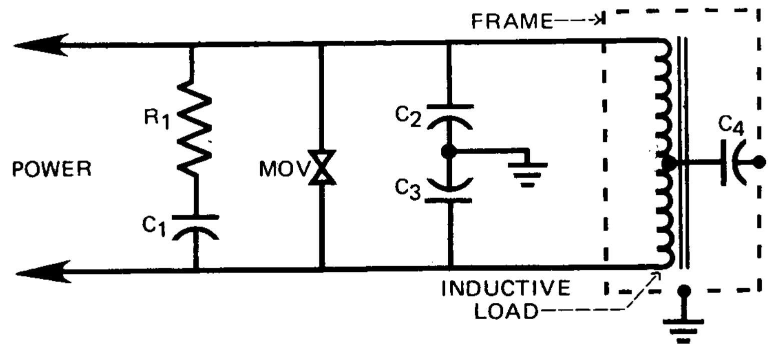
## INSTALLATION PRACTICES

Whenever a process controller such as the MicroMaster® is used, the equipment designer must keep in mind that such controllers are a collection of very sensitive and complex electronic circuits which must be operated in a generally hostile electrical environment. Unless precautions are taken in the installation of these devices, noise of sufficient amplitude to cause equipment malfunction can and will enter the equipment. A little planning in the installation can go a long way toward guaranteeing fault-free operation.

For this discussion, the connections to the MicroMaster® can be broken down into three major groups. These are power input connections, output load connections, and the signal input and sync connections.

The first rule to be followed in the installation of the MicroMaster® is **"the input, output, and power groups must NOT be cabled together"**. Keep all signal input leads in a separate bundle and keep that bundle physically separated from the load and power bundles. Normally, an inch or two of separation is all that is required, but where long parallel runs must be used, greater separation or separate conduits for signal and load lines is desirable. Never run signal and load lines in the same conduit. The importance of keeping these groups separated cannot be stated too strongly. Whenever a load is switched, rapidly varying electric and magnetic fields are set up around the conductors connected to the load. These fields can and will induce fault signals into the power leads and into the signal input leads to the MicroMaster® if steps are not taken to alleviate the problem.

Rule number two for fault-free operation is **"suppress all inductive loads"**. By far the largest single cause of improper MicroMaster® operation is the switching of unsuppressed inductive loads. Aside from causing erratic operation, unsuppressed inductive loads greatly increase the rate of wear on the output relay contacts and will cause premature failure of the output relays. The reason inductive loads cause such problems is that when the current through an inductor is interrupted, a very high voltage spike is generated by the inductor. These spikes can easily reach values of several thousand volts unless some method is employed to effectively limit the value which they may attain. This spike voltage may appear between the leads which feed power to the device, or between either or both power leads and chassis ground, depending on the physical construction of the device in question. Normally, a voltage limiting device such as a metal oxide varistor (MOV) should be placed across the inductor right at the inductor itself. This helps in two ways. First it will limit the maximum amplitude that the spike voltage may obtain. Second, since there is some capacity between the terminals of the varistor, it will help limit the rate of change of current through the inductor. Since the voltage of the spike is directly proportional to this rate of change ($V = L \times di/dt$), this will help limit the amplitude and rise time as well. In extreme cases, it may be necessary to supply an additional capacitor across the inductor to further limit this rate of change. The value of this capacitor should be approximately 1.0 mfd per amp of load current. To avoid damaging the contacts, this capacitor should be placed in series with a resistor which limits the capacitive inrush current to less than the relay contact rating. Thus, a 1 amp 115 vac load switched through a 6 amp relay would require a 1.0 mfd non-polarized capacitor in series with a 20 ohm resistor (120 volts/6 amps) connected directly across its terminals. A larger value resistor would be desirable from the standpoint of contact life, but, since the resistor reduces the effectiveness of the suppressor, it cannot be increased without limit. While these techniques will normally limit the spike voltage appearing between the leads to an acceptable level, they do nothing to suppress the spike which often appears between the leads and the metal frame of the device. This spike can result in a high frequency ground current which may interfere with processor operation. If this spike becomes a problem, it can be eliminated by bypassing both leads of the device to its frame with capacitors. For a 115 vac device, a .05 mfd capacitor rated for this kind of service will usually prove sufficient. Figure 7-1 (shown on page 65) will help to illustrate these suppression techniques.




MOV - Metal Oxide Varistor (G.E. #V130LAIOA for 115 VAC operation).
R1 - Value Greater than V-Load/I-contact.
C1 - 1.0 mfd/amp of Load Current, Non-Polarized.
C2, C3 - .05 mfd, High Voltage Bypass
C4 - Stray Capacity - Coil to Frame.

Figure 7-1 HIGH VOLTAGE SPIKE SUPPRESSION TECHNIQUE

Rule number three is **"eliminate loops in the input and load wiring".** The coupling of noise via magnetic fields between circuits is achieved through the mutual inductance of the two circuits. This mutual inductance is largely dependent on the area of the loop through which the current must flow in its journey from the source through the load and back to the source. By limiting the size of the loop, the available coupling is also limited. Figure 7-2 (shown on page 66) will help to explain the principle. In practice, this means that both conductors connected to a load or input device should follow the same physical path. Since the field from the two conductors will be equal and opposite, they will cancel and little or no magnetic coupling can occur. In the case of input leads, keeping the input and return lines close together will prevent any differential signal from appearing between them since any induced signal will be the same on both leads. It is not really necessary to run separate return lines for each input providing that the return and input lines follow the same path. For instance, the input lead bundle could contain all 8 input lines and 1 return line. The same applies to load bundles, where only 1 power common is used for all loads. It is necessary, however, that the switched (hot) lead for each load follow the same physi-

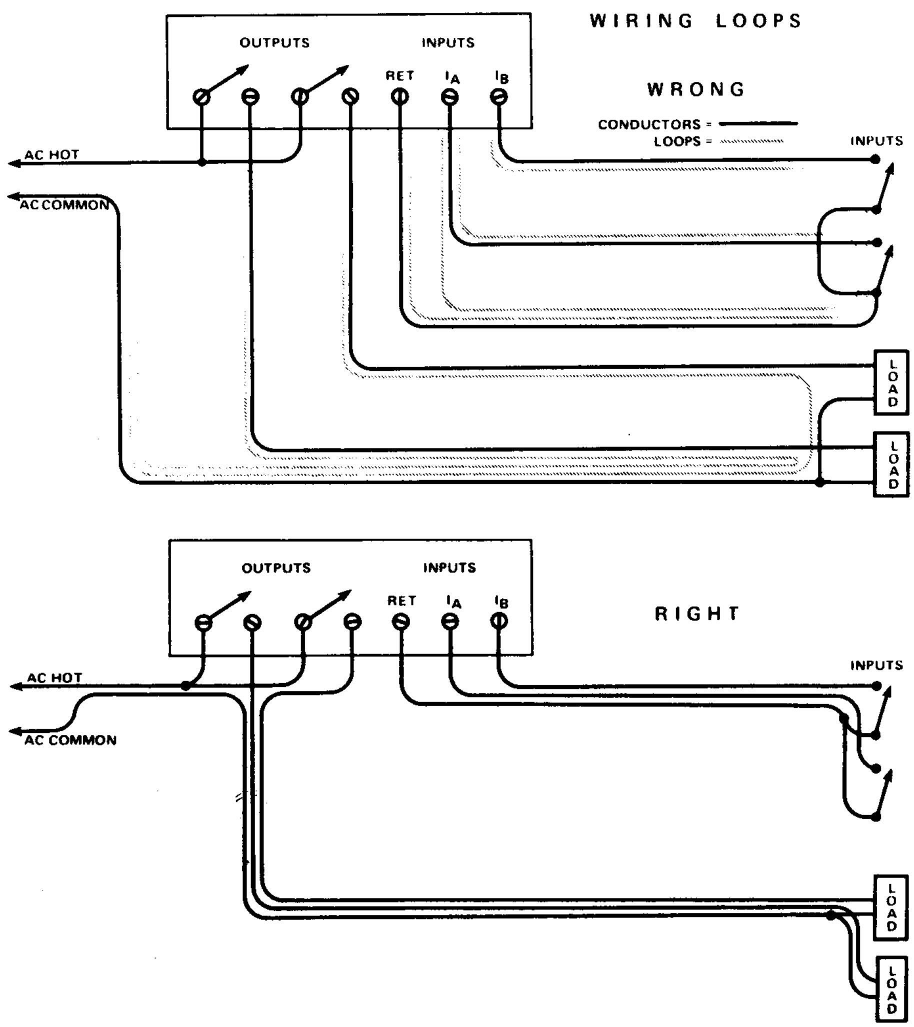



Figure 7-2 LOOP COUPLING LIMITING TECHNIQUE

cal path as the power common. If possible, the hot and common or signal and return should be twisted together. Where several hots and 1 common or several signal and 1 return line are bundled together, twisting the entire bundle will help reduce this form of coupling.

Rule number 4 is "**keep all wiring as close to ground as is practical**". Where the equipment is mounted in a metal cabinet, the wiring should be dressed along the surface of the cabinet and not just run by the shortest possible route to the switch or load device. This reduces considerably the likelihood that electric field coupling will interfere with operation of the controller since the capacity between the input leads and ground will be much higher than the capacity between the input leads and the noisy conductor. The capacitive voltage division this introduces will tend to reduce the induced fault voltage.

Adherence to these guidelines will usually result in fault-free operation of the MicroMaster® and its associated system. Problems can still arise, however, due to other equipment not related to the MicroMaster® but sharing a common power line. Generally, the interfering device can be identified and, by using the methods of suppression discussed above, the problem can be eliminated. Identification of the interfering device is usually fairly easy. It generally is observed that the MicroMaster® exhibits erratic operation when the device is switched on or off. When the device cannot be identified, a commercial EMI filter can be installed in the MicroMaster® power line to suppress the interference. The output of the filter should connect only to the MicroMaster® power terminals. It should be as close to the MicroMaster® as is practical. Where the interference is radiated into MicroMaster® rather than conducted via the power lines, shielding of the radiating leads will probably be the most effective. If that is not possible, shielding of the leads into the MicroMaster® may be attempted, but will not be as effective as shielding of the interfering device. In dealing with any interference problem, it will generally be found that suppression or shielding of the interfering device is more effective and easier to accomplish than trying to suppress all possible paths of noise coupling into the MicroMaster®.

## SAFETY DEVICES

The equipment designer who utilizes electronic process controllers must be constantly aware of the hazards involved in using these devices. The failure modes of these devices are completely random and unpredictable. Virtually anything can happen when the device fails. It could start unexpectedly, turn on all outputs, execute steps out of program sequence or simply fail to run. In view of this, the designer is warned not to rely on the process controller to activate devices or respond to signals which involve the safety of the machine operator. **This point cannot be stated too strongly.** Some positive and independent means must be provided to protect the operator of any machine utilizing process controllers. Some form of positive disconnect for load power should be provided to shut down the machine for maintenance or emergency shut-off. Do not count on the process controller reset or power switch to provide this function. Switching circuits designed to protect the operator's hands in the event the machine is inadvertently activated should directly control the machine and not be passed through the controller as an input. In general, a machine should be designed such that the failure of any one component will not result in the existence of a hazardous condition. When using process controllers, the controller itself should be considered as a single component.

**REMEMBER:**

1. Keep input, power and load conductors separated.
2. Suppress all inductive loads.
3. Avoid loops in input and output wiring.
4. Keep wiring close to ground.
5. Do not rely on the controller to activate or respond to safety devices.

## INPUT CIRCUITS

### INTERNAL INPUT CIRCUIT

The MicroMaster® Series WP6200 controllers utilize two internal plug-in input modules. Each module contains four input circuits. The first input module is identified as

Input Module Number 1 and contains the inputs for RST (RESET), RUN, HALT and CNT (COUNT). The second input module is identified as Input Module Number 2 and contains the four user definable inputs IA, IB, IC and ID. These plug-in modules contain circuitry identified as Type "A" Inputs and are operated with a DC voltage in the range of 9 thru 15 VDC. Figure 7-3 illustrates one of the Type "A" Input circuits contained on the internal plug-in input modules. An optocoupler provides photo isolation of the internal processor. The MicroMaster® power supply is an unregulated one. This power supply is normally used to supply the switching voltage for the inputs and its output appears at terminal 6 (+12 VDC) on the external terminal block. It has a range of 10 to 15 VDC depending on the input voltage to the controller and transformer loading. Therefore, the open circuit voltage level at terminal 6 can vary from 10 to 15 VDC. The switching voltage is not internally connected to the plug-in input modules and must be externally connected with jumper wires on the terminal block between terminal 6 (+12 VDC) and terminals 7 (I+1) and 12 (I+2). Figure 7-4 shows the connections of these jumpers.

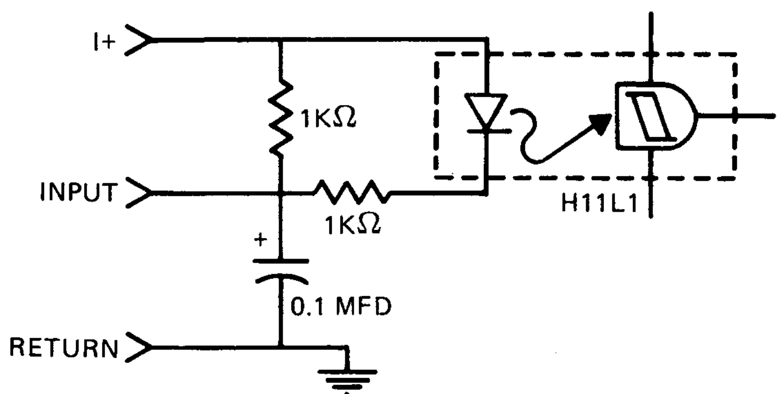



Figure 7-3 MicroMaster® TYPE "A" INPUT CIRCUIT

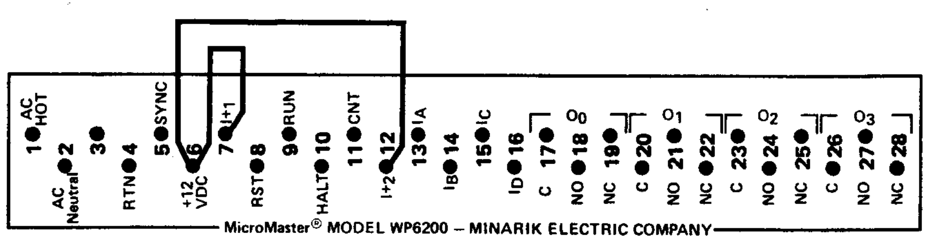



Figure 7-4 SERIES WP6200 INPUT SWITCHING VOLTAGE JUMPERS

## TYPICAL EXTERNAL INTERFACES

The basic interface to the input terminal is a switch rated for dry contact use. The MicroMaster© input circuit will indicate open if 100 microamps or less is flowing. It will indicate closed if 4 milliamps or greater is flowing. If electronic switching is utilized in lieu of dry contact switches, caution must be taken to insure proper performance. If external voltages are less than the open circuit voltage at the input terminal, enough current flow could occur to falsely sense a closed input. As long as the open circuit voltage level from the external source is equal to or greater than the terminal level, proper operation should occur. The ideal electronic interface for the system is an open collector NPN transistor connected as a current sink on the input. It represents an electronic equivalent to the dry contact. The following circuits are commonly found in industrial input devices such as TDR's, limit switches, encoders, photocells, proximity detectors, counters, and manual control switches.

### ELECTROMECHANICAL DRY CONTACT

A typical electromechanical form A dry contact is the simplest input circuit (Figure 7-5). It may be necessary to connect a capacitor across the count input to eliminate mechanical contact bounce. Generally, 2 mfd at 35 vdc has been found to be sufficient.

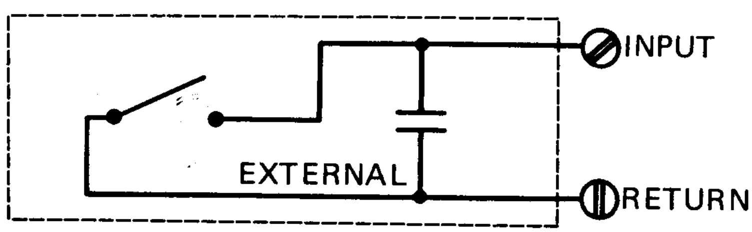



Figure 7-5 DRY CONTACT CONNECTION

### NPN OPEN COLLECTOR

Generally, no special precautions need be taken with this circuit (Figure 7-6 on page 71) as leakage through the transistor when off is typically well below 100 microamps.

However, the transistor should be able to withstand 18 volts collector to emitter in the off state and 20 milliamps collector current in the on state.

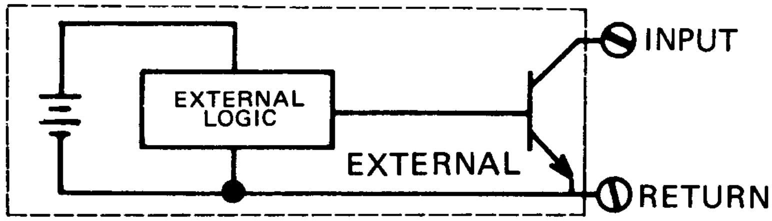



Figure 7-6 NPN OPEN COLLECTOR CONNECTION

## OPEN COLLECTOR with CLAMPING DIODE

Some manufacturers of proximity switches and other sensing devices include a diode in the internal circuitry to prevent the output from rising above the positive supply rail. This diode is normally connected between the output terminal and the positive supply terminal (See Figure 7-7). If the power supply for the external sensor is at a lower potential with respect to return than the internal MicroMaster® supply, sufficient current could flow through the clamping diode to cause the input to appear closed. It is not recommended that this type output be connected to MicroMaster® unless the sensor is powered from the MicroMaster's power supply. If the sensor requires a different voltage from that supplied by the MicroMaster® then a special input module must be used which is compatible with the sensor's power supply.

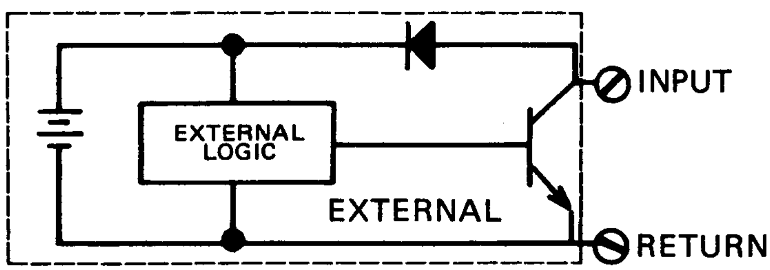



Figure 7-7 OPEN COLLECTOR WITH CLAMPING DIODE

## HIGH VOLTAGE SWITCHING INPUTS (115, 230 VAC, ETC.)

While the MicroMaster® input circuit requires low level voltage, open collector transistors or dry contact form A switches as shown in the previous discussion, the applications for MicroMaster® will involve the retrofitting of existing discrete control systems that use a wide variety of input devices. Those devices switch a multitude of both ac and dc voltage levels including power line voltages. It is recommended that commercially available optically isolated interface devices be utilized for proper sensing by MicroMaster®. Contact the Minarik factory for further details.

## MicroMaster® AS AN INPUT POWER SUPPLY

Where the use of input devices requiring 12 VDC of unregulated power are needed, it is recommended that the MicroMaster® Series WP6200 internal power supply be used. This 12 VDC supply is rated at up to 200 Milliamperes and is available at terminal 4 (RTN) and terminal 6 (+12 VDC) on the external terminal block. By using this approach, the user will save the expense of an additional power supply and will also guarantee the input voltage level to equal the MicroMaster® level.

## OUTPUT VARIATIONS AND NOMENCLATURE

The MicroMaster® Series WP6200 controllers utilize two internal plug-in output modules and the Series WP6300 expanders utilize four internal plug-in output modules. Each module contains two output circuits. These modules are identified as Output Modules Number 1 and 2 for the controllers and Output Modules Number 1, 2, 3 and 4 for the expanders. There are presently three types of output circuits available. The stock output is a relay (see Figure 7-8 on page 73). This output is defined as a type 1 circuit. The second type of output is an open collector transistor in a 4N28 optocoupler. If the MicroMaster® product normally has form C relay contacts, an 820 Ohm pullup resistor is provided as shown in Figure 7-9 on page 73. This output is defined as a type 2 circuit. The last output type is an unregulated 12 vdc programmable power supply that can source up to 200 Milliamperes maximum divided into all outputs. This output is defined as a type 3 circuit and is shown in figure 7-10 on page 73. Since the user may want to mix outputs for a given product and since all three outputs are considered standard, the chart at the top of the next page defines the output types and their description.

| OUTPUT | DESCRIPTION |
|---|---|
| TYPE A | 2 - EACH TYPE 1 CIRCUITS |
| TYPE B | 2 - EACH TYPE 2 CIRCUITS |
| TYPE C | 2 - EACH TYPE 3 CIRCUITS |
| TYPE D | 1 - EACH TYPE 1 and 1 - EACH TYPE 2 CIRCUIT |
| TYPE E | 1 - EACH TYPE 1 and 1 - EACH TYPE 3 CIRCUIT |
| TYPE F | 1 - EACH TYPE 2 and 1 - EACH TYPE 3 CIRCUIT |

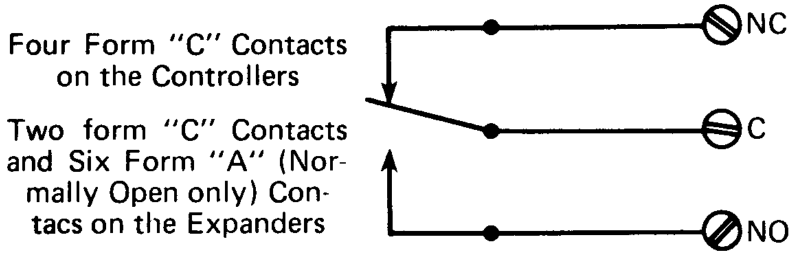



Figure 7-8 - TYPE 1 OUTPUT CIRCUIT

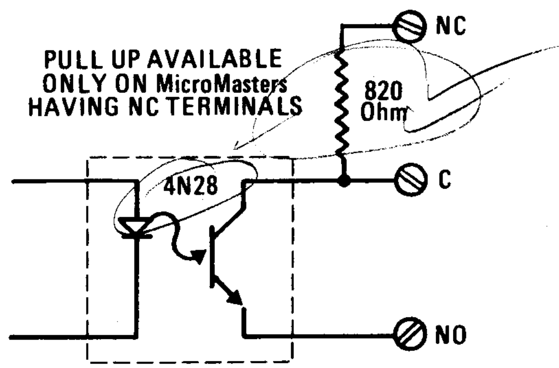



Figure 7-9 - TYPE 2 OUTPUT CIRCUIT

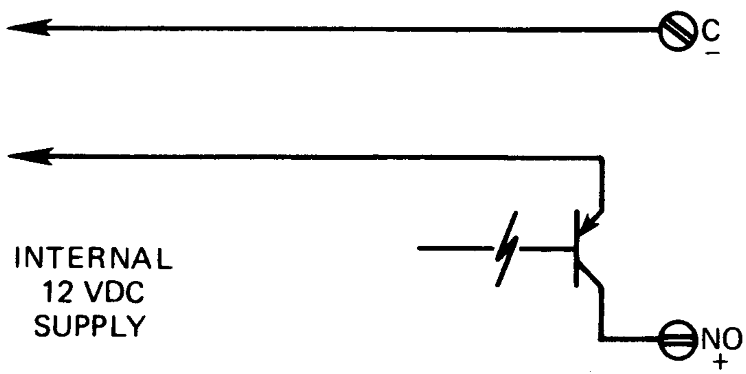



Figure 7-10 - TYPE 3 OUTPUT CIRCUIT

The following part number system identifies which input and output modules are installed in the MicroMaster® units. The controllers use the product model number followed by two, two digit groups of letters. The first two digit group identifies the internal plug-in input modules and the second two digit group identifies the internal plug-in output modules. The expanders, which have a fixed input group circuit, use the product number followed by a four digit group of letters to identify which internal plug-in output modules are installed. The internal plug-in modules are installed sequentially in the same order as the part number indicates.

Example #1:-

WP6200-AA-AA is a panel mount controller with two type "A" input modules and two type "A" output modules installed.

Example #2:-

WP6300-EAAD is a panel mount expander with two type "A", one type "D" and one type "E" internal plug-in output modules installed. Refering to the terminal block connections (expander terminal block connections shown in Figure 7-14 on page 83) output terminals 11 and 12 have a type 1, Form "A" relay contact circuit, terminals 13 and 14 have a type 3, 12 VDC output circuit, terminals 15 thru 22 have four type 1, Form "A" relay contact circuits, terminals 23 thru 25 have a type 1 Form "C" relay contact circuit and terminals 26 thru 28 have a type 2 circuit with internal pull-up resistor.

## HOW TO REMOVE and REPLACE PC BOARD ASSEMBLIES

The MicroMaster® printed circuit board assemblies must be removed from their cases in order to change the internal input and output plug-in modules and to change the PROM in ROM based units.

To remove and replace the PC board assembly from the case follow these instructions . . .

1. Disconnect input power going to the unit.

2. Remove the two screws that hold the terminal block to the case.

3. Hold the case with one hand and grasp the terminal block with the other. Using a slight rocking motion from end to end, free the terminal block from the PC board and pull it straight away from the end of the case.

4. The PC board assembly is held into the case by the front bezel on the panel mount units and by a flat plate with mounting ears on the surface mounted units. Remove the four screws from the case flange and while being careful not to let the PC board assembly fall out, remove the front bezel or the surface mounting plate.

5. Slide the PC board assembly out of the case.

6. After performing the required operations on the PC board assembly (replacing the plug-in modules or the ROM, etc.), slide it back into the case, making sure that the edges of the board are placed between the internal "runners" on each side of the case.

7. Replace the front bezel or the surface mounting plate and replace the four screws. To insure easy alignment of the screw holes, do not tighten the screws until all four of the screws have been started.

8. The terminal block, which is also a socket for the PC board edge connector, has a key and the PC board assembly edge connector has a matching slot to prevent the terminal block from being assembled incorrectly. Replace the terminal block by aligning the slot and key with the PC board and carefully press the terminal block straight down until its ends are flat against the end of the case.

9. Replace the two screws which hold the terminal block to the case.

## REPLACING INTERNAL PLUG-IN INPUT AND OUTPUT MODULES

In order to replace the internal plug-in modules, the PC board assembly must be removed from the case. Step-by-step instructions for removing and replacing the PC board assembly are described starting on page 74 under the heading "How to Remove and Replace PC Board Assemblies".

To remove and replace the internal plug-in input and output modules from the PC board carefully follow these instructions . . .

1. To remove a plug-in module, hold the PC board assembly down to a flat surface by grasping the power transformer with one hand.

2. With the other hand, grasp the plug-in module by the outer edges of its PC board and using a slight rocking motion from end to end, carefully pull it straight away from the PC board assembly.

3. To replace a plug-in module, hold the PC board assembly down to a flat surface by grasping the power transformer with one hand.

4. With the other hand, grasp the plug-in module by the outer edges of its PC board and line up the holes in the plug-in module socket with the pins on the PC board assembly, making sure that the locking ramp on the module faces the friction lock on the PC board assembly (see Figure 7-11 shown below). After lining up the holes, press the plug-in module down until it snaps into place.

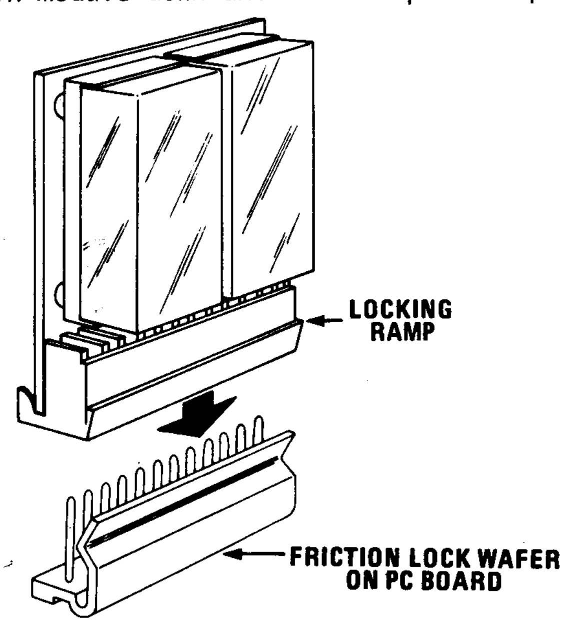



Figure 7-11 INSTALLING PLUG-IN MODULE

## APPLICATION OF DEDICATED UNITS (WP6220 and WP6320)

Generally, the creation of ROM-based units is accomplished at the factory or authorized programming center. The typical approach to programming of the ROM-based units assumes that the program has already been debugged using fully programmable RAM-based units. Programs that are fixed for a given application can then be placed on ROM and the MicroMaster© becomes a control component in a machine or process using the less expensive "dedicated" OEM versions. The programming forms supplied with RAM-based units are used to document final programs and become the sole document needed to create ROMs with appropriate programs "burned in". These ROMs are then inserted into the ROM-based products, creating a control device that has a specific function.

## INSTALLING THE PROM CHIP INTO THE WP6220 and WP6320 DEDICATED UNITS

Most of the time, the installation of the PROM integrated circuit into the system will occur before shipment from the factory or programming center. However, where installation of dedicated products is made prior to the program being finished or changes in the program are necessary, carefully follow these instructions . . .

1. Remove the PC board assembly as described in the step-by-step instructions starting on page 74.

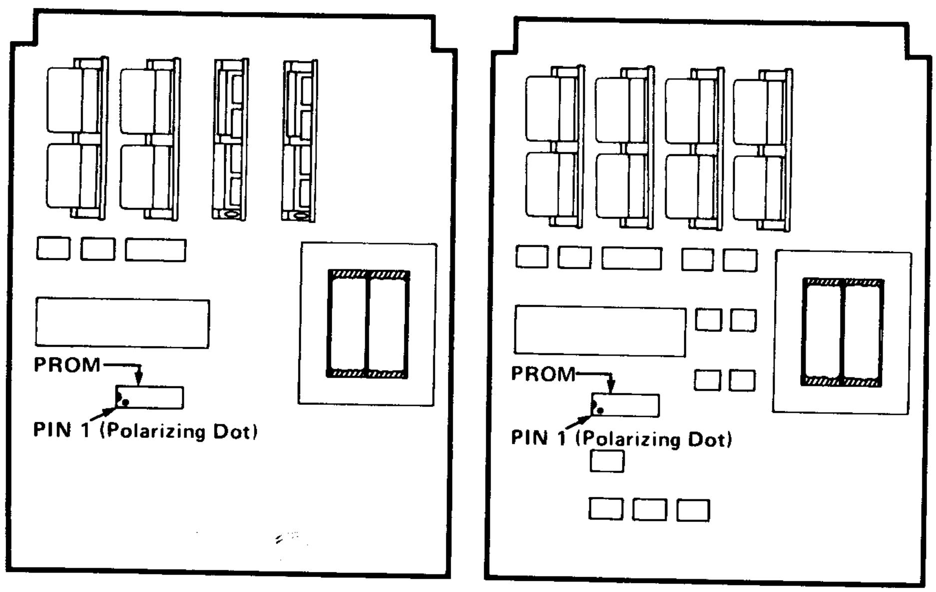
2. Remove the PROM by gently prying up with a screwdriver blade between the PROM and its socket.

3. Insert the new PROM into the socket, making sure that Pin 1 (identified by the polarizing dot on top) is oriented correctly into the socket as shown in Figure 7-12 on page 78. **CAUTION!:-** Position the pins on the PROM directly over the holes in the socket and gently apply pressure to the top of the PROM. Apply only enough pressure to seat the PROM completely. If the PROM does not slide into the socket easily, remove it, reposition the pins, then apply gentle pressure until it is seated. A very slight end-to-end wiggle may be used to help seat the PROM. But, any lateral movement may damage a pin so avoid wiggling if

you can. If you must wiggle the PROM to seat it fully, be sure that the pins are lined up properly first, then wiggle only very slightly and slowly.

4. Replace the PC board assembly as described starting with step 7 on page 75.

**NOTE!:-** Under most circumstances, the creation of custom PROMs for a given application is done either at the factory or at an authorized programming center. However, where applications are many and varied, it may become more convenient for end users to "burn in" their own PROMs using commercially available PROM programming equipment and modified MicroMaster® hardware.




WP6220 CONTROLLER WP6320 EXPANDER

Figure 7-12 MicroMaster® PROM LOCATIONS

## UTILIZING A WP6012 TIMER MODULE

While the WP6220 and the WP6320 are a very economical way of sequential control, they are limited to a pre-defined sequence that is normally developed on the fully programmable MicroMaster® products. If applications require timing variations within a fixed sequence of outputs, the dedicated units can be utilized by using the WP6012 Timer Module. The WP6012 is a panel mount, one second on-delay, time delay relay with a 10:1 timing range. It must be connected to a type "3" output on the MicroMaster© units (see Figures 7-15, 7-16 and 7-17 shown on pages 84 and 85). In operation, the output from the WP6012 is an open collector transistor which saturates while timing and opens when timed out.

The following examples demonstrate how the MicroMaster© units can be programmed:

PROGRAM FOR MAXIMUM DELAY OF 1 SECOND

The program will stay on step 1 as long as the WP6012 is timing (0.1 thru 1 second). When the preset time has elapsed the program will advance to step 2.

| STEP | FUNCTION | OUTPUTS 0 1 2 3 |
|---|---|---|
| 01 | IF A > 01 | - - - x |
| 02 | Timed-out process | - - - - |

PROGRAM FOR MAXIMUM DELAY OF 10 SECONDS

The same WP6012 is used and by using a loop back 10 times program it becomes a 10 second timer with a range of 1 second thru 10 seconds. When the preset time has elapsed the program will advance to step 4.

| STEP | FUNCTION | OUTPUTS 0 1 2 3 |
|---|---|---|
| 01 | L = 0010 | - - - x |
| 02 | IF A > 02 | - - - x |
| 03 | L > 02 | - - - - |
| 04 | Timed-out process | - - - - |

## SYSTEM TROUBLESHOOTING TECHNIQUES

This section will point out some of the more common problems experienced in MicroMaster© system applications and show some troubleshooting techniques useful in locating their cause.

## COUNT INPUT

One of the most frequent causes of problems in MicroMaster® installations is the count input. These problems stem from two major sources. The first is contact bounce. Mechanical contacts do not generally close just one time, but "bounce" several times when they are closed. If this bouncing continues long enough, the MicroMaster® will see more counts than are actually intended. There is a technique to determine whether contact bounce exists which will prove useful in many cases. Enter a program to the MicroMaster® consisting of one step:

| STEP | FUNCTION | OUTPUTS<br>0 1 2 3 |
|---|---|---|
| 01 | CT 0010 | - - - - |

Put the MicroMaster® into run mode and activate the count input using whatever input device is connected to it in such a manner as to generate one count. For instance, if the count device is a switch, manually close and re-open the switch. Observe the display. If no "bounce" was present, it will read:

| STEP | FUNCTION | OUTPUTS<br>0 1 2 3 |
|---|---|---|
| 01 | CT 0009 | - - - - |

If it shows less than nine, then a bounce was present. With a mechanical switch, it can usually be eliminated by shunting the switch with a capacitor of approximately 2 mfd, 25 volts. If polarized, the " + " end should go to the count input and the " - " end should go to return. The value is somewhat dependent on the maximum count rate in the application. If it is too large, the capacitor will filter out desired closures as well as undesired ones.

The second problem commonly experienced with the count input is due to the finite time needed for the MicroMaster® to advance from one step to the next. The count input will respond accurately to count rates as high as 1,000 counts per second. Once the counter has been decremented to zero, however, it takes the MicroMaster® as much as 2 milliseconds to get to the next step. After that, it can take as long as 6 to 8 milliseconds for the output relays to change to the next state. Thus, where count rates in excess of approximately 75 to 100 counts per second are being received, extra counts can go by unnoticed during this transition period. There is no real solution to this problem. A special "chip" is available which improves these times considerably, but it will not work if the system uses an expander, and the display will go blank during the execution of the count step. Program mode appears normal when using this "chip". If problems of this nature are experienced and these limitations are of no consequence in the application, contact the factory and ask for information on the fast count EPROM.

## LINE VOLTAGE

Another frequent problem encountered in MicroMaster© applications is caused by low line voltage. Power is applied to the MicroMaster© and either nothing happens (no display at all) or the display will light with letters or other strange symbols. The letters or symbols will be the same in all eight digit positions. This is usually caused by low line voltage. The reset generator has detected that the line is not high enough to maintain proper operation and has disabled the processor and display. This usually occurs at approximately 90 VAC input. There is considerable hysteresis in the reset generator. While the reset generator drops out at 90 Volts, the line must be raised to a voltage between 100 and 105 Volts to get the generator to pick back up.

This line voltage problem can be even more mysterious when the line dips during run mode. This can happen when a large load is switched on at the same time as the MicroMaster©, causing a momentary drop below 90 Volts. The processor will reset. The only cure is to stabilize the line or to move the MicroMaster© to some quieter, stabler power source.

## LOW BATTERY

The MicroMaster© battery circuit is constantly charged when power is applied to the unit, regardless of the state of the front panel power switch. Under normal circumstances, this is enough to keep the battery charged continuously. In applications where power is removed for long periods of time relative to the length of time that power is applied, it is conceivable that the battery could become discharged. This will result in "P-FAIL". It is usually possible to avoid this circumstance by keeping power applied to the Micro-Master® whenever plant power is on. A master power switch for the machine should be wired to de-energize only the load power, and not the MicroMaster®.

## INPUT NOISE

Noise on the inputs to the MicroMaster® is one of the most troublesome areas in MicroMaster® applications. It usually occurs when switching an inductive load in the vicinity of the input wiring. Refer to the section on installation practices for ways to minimize these problems. It is possible to use the MicroMaster® to localize the source of these problems. By entering a program similar to the following, the offending input-load pair can often be determined.

| STEP | FUNCTION | OUTPUTS<br>0 1 2 3 |
|---|---|---|
| 01 | S 00.50 | - - - - |
| 02 | IF A > 06 | x - - - |
| 03 | IF A > 06 | x - - - |
| 04 | IF A > 06 | x - - - |
| 05 | GO TO 01 | x - - - |
| 06 | GO TO 06 | x - - - |

This program checks for interference between input A and output 0. It will continually recycle unless the noise generated by switching output 0 ON causes a faulty signal to be seen on input A. In that case it will jump to step 06. Another test would involve switching output 0 ON at step 1 and OFF at all the other steps. This would check for noise when the output was de-energized. Similar tests could be run for other input-output pair combinations to determine if they showed any interaction. It may be possible to overcome such problems by writing the program in such a way that the output state does not change immediately before an input statement. For instance:

| STEP | FUNCTION | OUTPUTS<br>0 1 2 3 |
|---|---|---|
| 01 | S 01.00 | x - - - |
| 02 | IF A > 05 | - - - - |

would be more susceptible to noise than:

| STEP | FUNCTION | OUTPUTS 0 1 2 3 |
|---|---|---|
| 01 | S 01.00 | x - - - |
| 02 | S 00.05 | - - - - |
| 03 | IF A > 06 | - - - - |

Since in the second example a settling time of approximately 50 milliseconds is provided at step 2 during which the noise is allowed to die out before the input is checked. Whether this technique can be effectively used will depend on the application.

## SYNC LINE NOISE

The "sync line" is used to transmit step and mode information from the MicroMaster© to the system expanders. If noise is generated which interferes with the "sync line", the expanders will see incorrect data and will set up incorrect output states. This is usually characterized by erratic expander operation and normal MicroMaster© operation although if enough noise is present, the MicroMaster© may also act erratically. The solution to the problem can usually be found in the section on installation practices.

## NORMALLY CLOSED CONTACTS USED WITH EXPANDERS

In a previous section, it was noted that a problem sometimes exists when an attempt is made to wait for a normally closed contact to open when such contact is wired through a group select on the expander. The problem is that the contact may have closed when its associated group was de-selected. Since there is no "wetting current" available when this occurs, the oxide on the contacts may not be cleaned away. The contact is thus electrically open even though it is physically closed. If such a configuration must be implemented, the contact should be rated for this type of service. This means either a new gold plated contact or some form of solid state device with an open collector type of output. The problem does not exist when using only the MicroMaster© since wetting current is always available.

## CONNECTION DATA

The illustrations shown on the following pages show terminal connections and connection diagrams for MicroMaster© products.

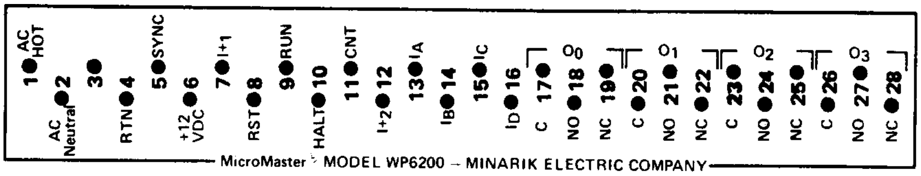



Figure 7-13 WP6200, WP6220 and WP6240
CONTROLLERS REAR TERMINAL BLOCK CONNECTIONS

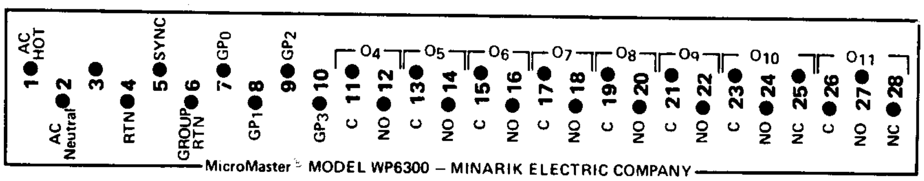



Figure 7-14 WP6300 and WP6320
EXPANDERS REAR TERMINAL BLOCK CONNECTIONS

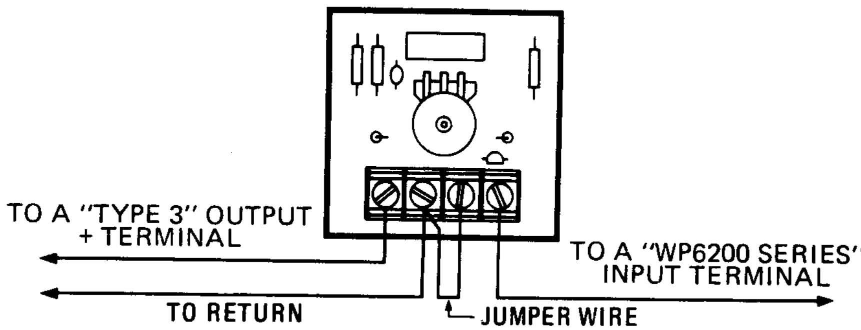



Figure 7-15 WP6012 TIMER MODULE
CONNECTED TO A WP6200 SERIES CONTROLLER

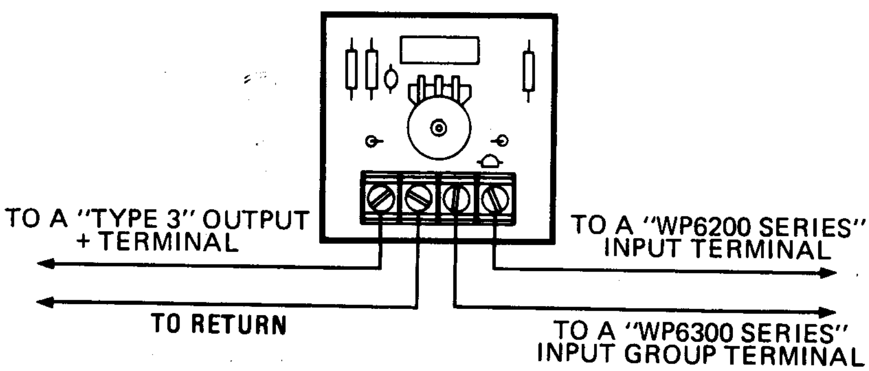



Figure 7-16 WP6012 TIMER MODULE
CONNECTED TO A WP6200 SERIES CONTROLLER
AND A WP6300 SERIES EXPANDER

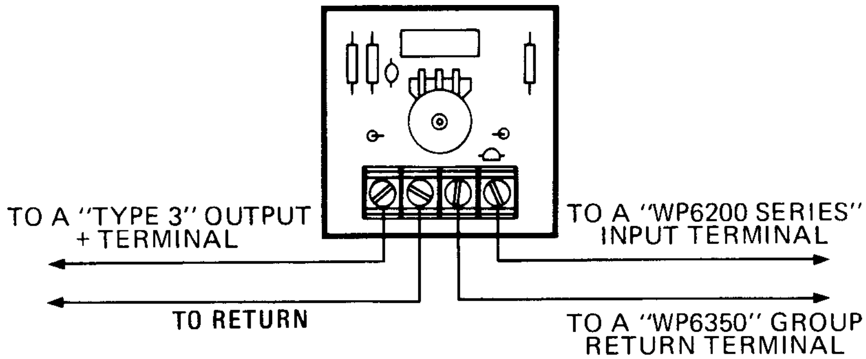



Figure 7-17 WP6012 TIMER MODULE
CONNECTED TO A WP6200 SERIES CONTROLLER AND A
WP6300 SERIES EXPANDER WITH A WP6350 DIODE MATRIX

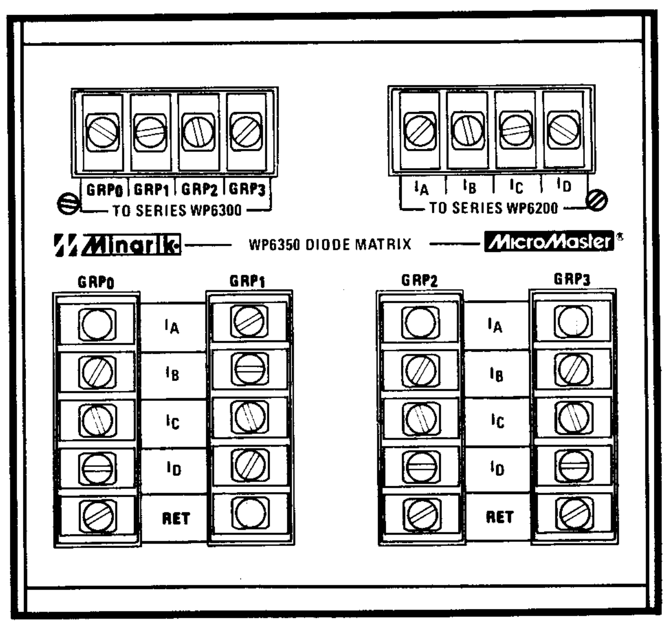



Figure 7-18 WP6350 INPUT EXPANDER
DIODE MATRIX TERMINAL BLOCKS

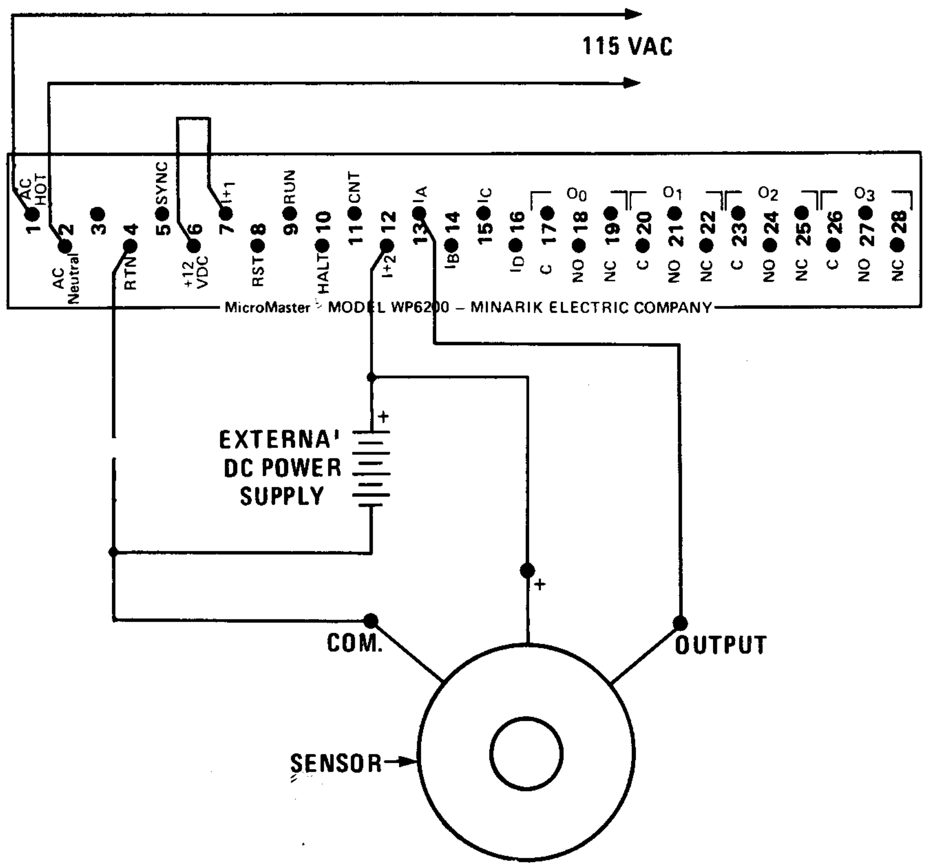



Figure 7-19 SERIES WP6200 CONTROLLER USING INPUT SWITCHING VOLTAGE SUPPLIED BY EXTERNAL SENSOR POWER SUPPLY FOR USER DEFINED INPUTS IA, IB, IC AND ID WITH SENSOR CONNECTED TO INPUT IA.

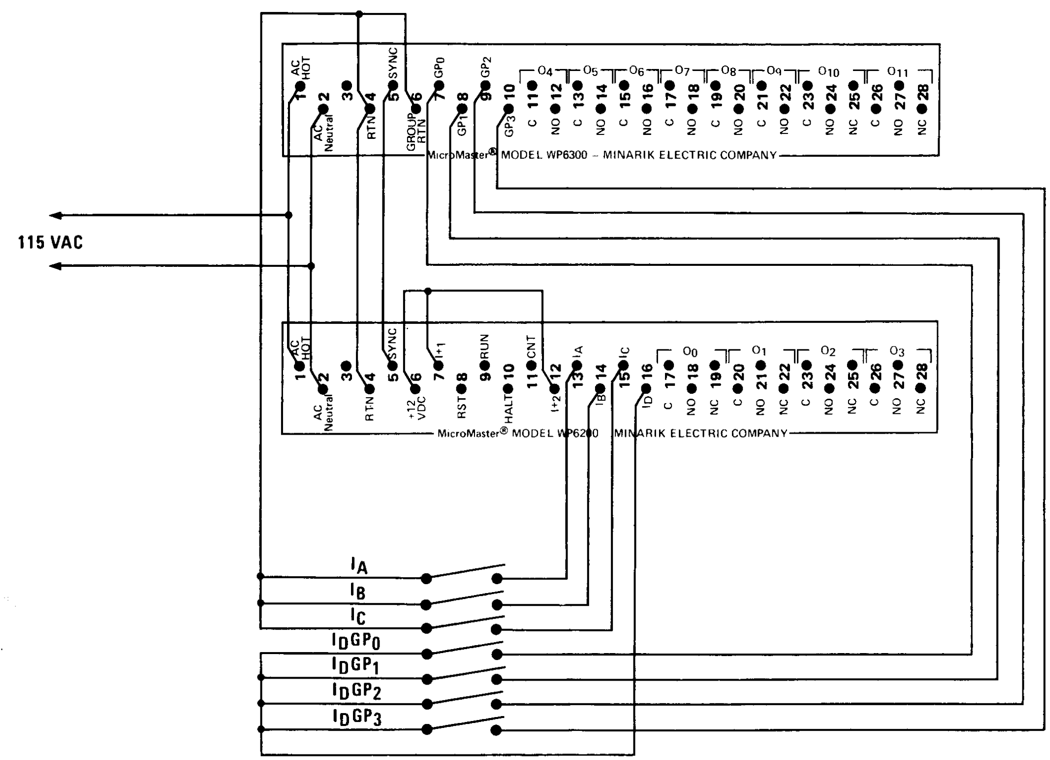



Figure 7-20 SEVEN INPUTS AND TWELVE OUTPUTS USING A SERIES WP6200 CONTROLLER AND A SERIES WP6300 EXPANDER.

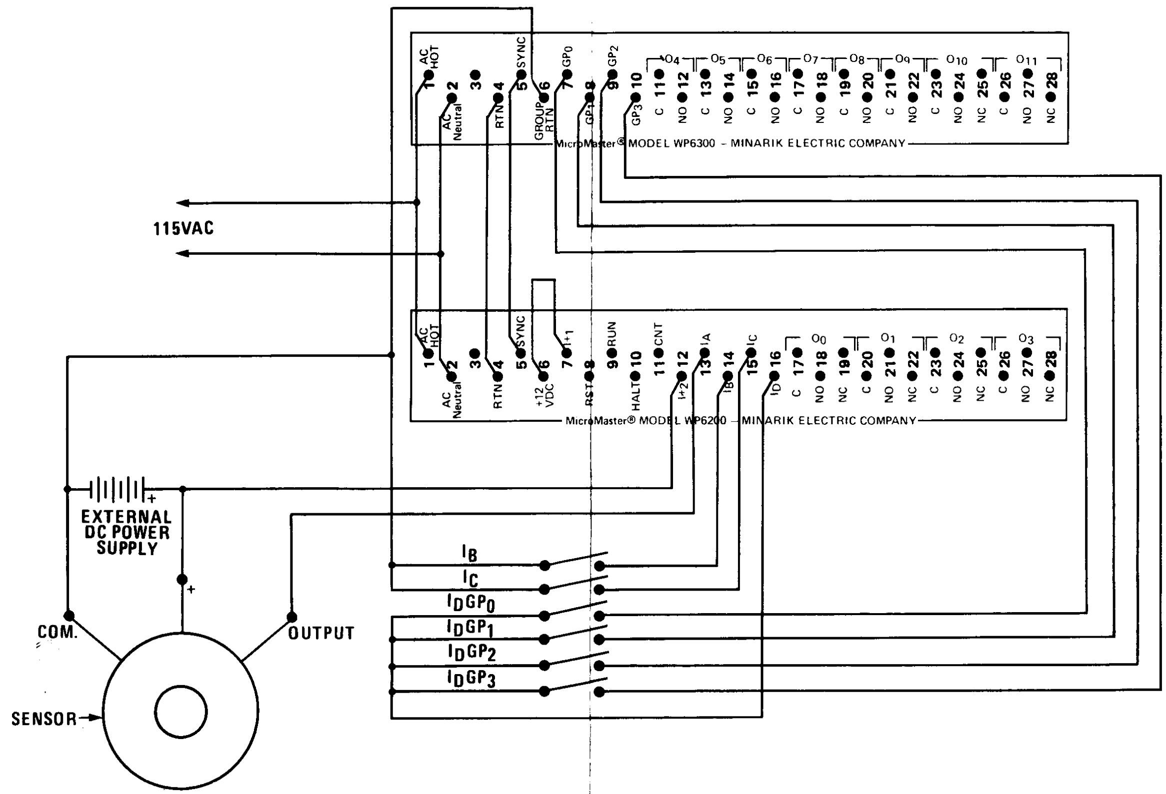



Figure 7-21 SEVEN INPUTS AND TWELVE OUTPUTS USING A SERIES WP6200 CONTROLLER AND A SERIES WP6300 EXPANDER WITH INPUT SWITCHING VOLTAGE SUPPLIED BY EXTERNAL POWER SUPPLY FOR USER DEFINED INPUTS IA THRU IC AND ID GROUP 0 THRU GROUP 3, WITH A SENSOR CONNECTED TO INPUT IA.

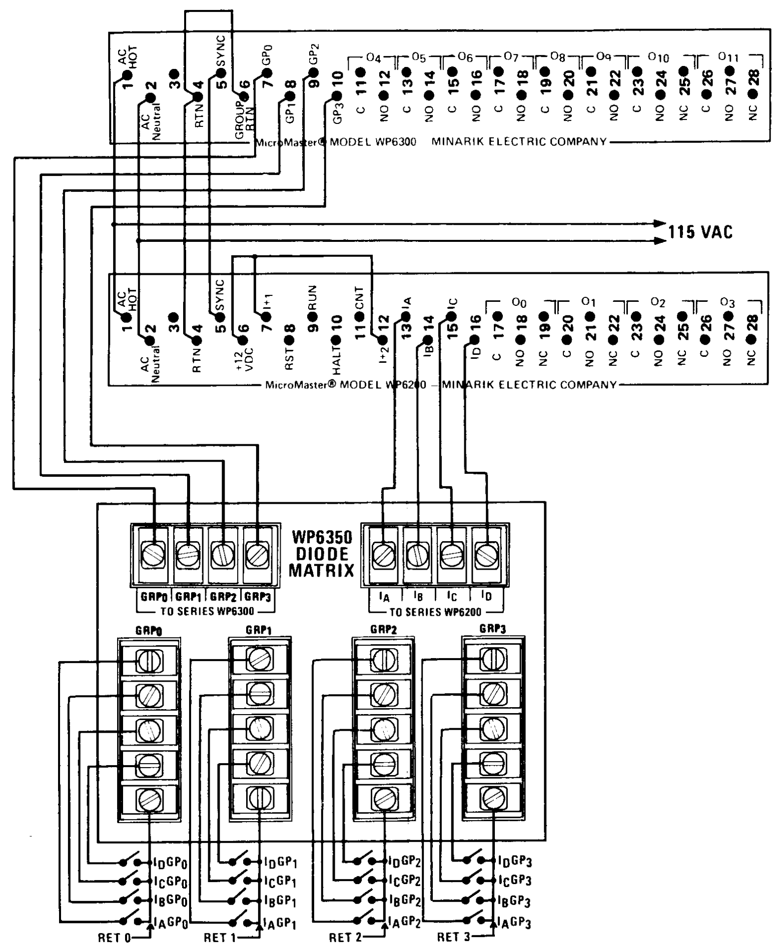



Figure 7-22 SIXTEEN INPUTS AND TWELVE OUTPUTS USING A SERIES WP6200 CONTROLLER, A SERIES WP6300 EXPANDER AND A WP6350 DIODE MATRIX.

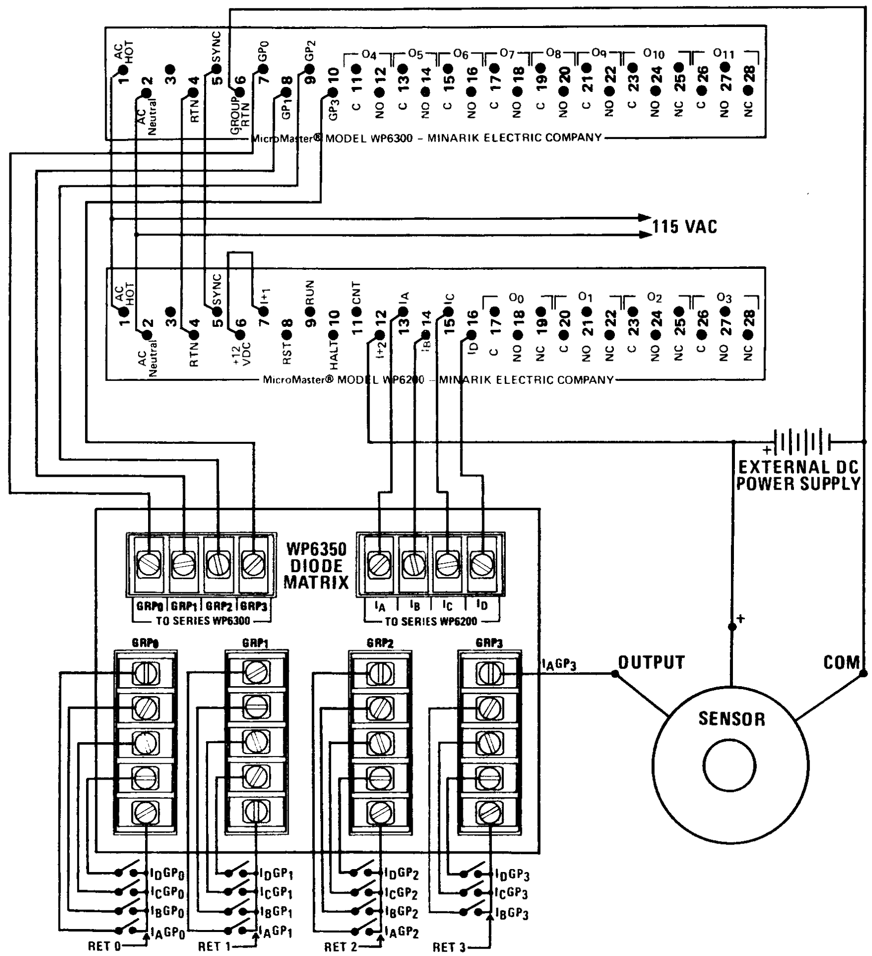



Figure 7-23 SIXTEEN INPUTS AND TWELVE OUTPUTS USING A SERIES WP6200 CONTROLLER, A SERIES WP6300 EXPANDER AND A WP6350 DIODE MATRIX. INPUT SWITCHING VOLTAGE SUPPLIED BY EXTERNAL POWER SUPPLY FOR USER DEFINED INPUTS IA GROUP 0 THRU ID GROUP 3, WITH A SENSOR CONNECTED TO INPUT IA GROUP 3.

## SECTION 8 - TECHNICAL DATA

### WP6200, WP6220 and WP6240 CONTROLLERS

**ENVIRONMENTAL**

INPUT POWER ............ 105-125 VAC, 50/60 HZ, 12 WATTS
OPERATING TEMPERATURE ................. 0-50° CENTIGRADE
RELATIVE HUMIDITY ................ 5%-95% NON-CONDENSING

**INPUTS**

INPUT VOLTAGE ........................... 18 VDC MAXIMUM
INPUT CURRENT ............................ 30 MA MAXIMUM

**OUTPUTS**

TYPE 1 .................. FORM "C" RELAY CONTACTS, RATED 115 VAC, 6 AMPS RESISTIVE
TYPE 2 ...................... OPEN COLLECTOR TRANSISTOR, RATED 30 VDC, 25 MA
TYPE 3 ................ MicroMaster® SUPPLIED 10-15 VDC, RATED UP TO 200 MA PER OUTPUT, 200 MA TOTAL

**INSTRUCTION TIMES**

........................................... 2 MS

### WP6300 and WP6320 EXPANDERS

**ENVIRONMENTAL**

INPUT POWER ............ 105-125 VAC, 50/60 HZ, 12 WATTS
OPERATING TEMPERATURE ................. 0-50° CENTIGRADE
RELATIVE HUMIDITY ................ 5%-95% NON-CONDENSING

**OUTPUTS**

TYPE 1 ..................... FORM "A" and FORM "C" RELAY CONTACTS RATED 115 VAC, 6 AMPS RESISTIVE*
TYPE 2 ...................... OPEN COLLECTOR TRANSISTOR, RATED 30 VDC, 25 MA
TYPE 3 ................ MicroMaster® SUPPLIED 10-15 VDC, RATED UP TO 200 MA PER OUTPUT, 200 MA TOTAL

*Outputs 04 thru 09 can only have Form "A" Relay Contacts and Outputs 010 and 011 can have Form "C" Relay Contacts.

## SECTION 8 - TECHNICAL DATA (CONTINUED)

### WP6350 DIODE MATRIX

**ENVIRONMENTAL**

OPERATING TEMPERATURE ................. 0-50° CENTIGRADE
RELATIVE HUMIDITY ................ 5%-95% NON-CONDENSING

### WP6012 TIMER MODULE

**ENVIRONMENTAL**

INPUT POWER ................... SUPPLIED BY MicroMaster®
OPERATING TEMPERATURE ................. 0-50° CENTIGRADE
RELATIVE HUMIDITY ................ 5%-95% NON-CONDENSING

**OUTPUTS**

SOLID STATE ....... COMPATIBLE WITH MicroMaster® INPUTS

### SAFETY WARNINGS

**INSTALLATION.....**This equipment should be installed, adjusted and serviced by qualified electrical maintenance personnel familiar with the construction and operation of the equipment and the hazards involved. It is the responsibility of the equipment manufacturer or individual installing the apparatus to take diligent care when installing equipment. The National Electrical Code (NEC), sound electrical and safety codes, and when applicable, the Occupational Safety and Health Act (OSHA) should be followed when installing the apparatus to reduce hazards to person and property.

**USE..............**The chance of electric shocks, fires or explosion can be reduced by giving proper consideration to the use of grounding, thermal and over-current protection, type of enclosure and good maintenance procedures.

## WP6012 PARTS LIST

| REFERENCE | PART NUMBER | DESCRIPTION |
|---|---|---|
| C1 | 010-0008 | 270 PF 50 VDC DISC CERAMIC |
| IC1 | 060-0037 | MC14541 OSCILLATOR-DRIVER |
| P1 | 120-0013 | 1 MEGOHM POTENTIOMETER |
| Q1 | 070-0004 | 2N6424 NPN DARLINGTON |
| R1 | 030-0038 | 2.2 MEGOHM 1/4 WATT |
| R2 | 030-0021 | 10K OHM 1/4 WATT |
| R3 | 030-0013 | 1K OHM 1/4 WATT |
| R4 | 030-0021 | 10K OHM 1/4 WATT |
| Z1 | 071-0005 | 1N4744 15V ZENER DIODE |

## WP6200 PARTS LIST

| | | |
|---|---|---|
| B1 | 062-0005 | NI CAD BATTERY |
| BR1 | 073-0005 | 1 AMP BRIDGE RECTIFIER |
| C1-C2 | 010-0044 | 0.47 MFD 250 VAC |
| C3 | 011-0040 | 4700 MFD 16 VDC |
| C4-C5 | 010-0002 | 27 PF DISC CERAMIC |
| C6 | 011-0001 | 1 MFD 15 VDC |
| C7-C18 | 010-0027 | 0.1 MFD 50 VDC CERAMIC |
| C20 | 011-0008 | 6.8 MFD 35 VDC TANTALUM |
| D1 | 071-0012 | 1 AMP 600 VOLT DIODE |
| D3 | 071-0026 | SCHOTTKY DIODE |
| D4-D5 | 071-0012 | 1 AMP 600 VOLT DIODE |
| D6-D10 | 071-0017 | 1N914 SILICON DIODE |
| D11-D14 | 071-0024 | 1N914 SILICON DIODE |
| FU1 | 050-0001 | 5 AMP PIGTAIL FUSE #175-005 |
| IC1 | 061-0011 | LM7805 5V REGULATOR |
| IC2-IC5 | 060-0050 | MC75453 RELAY DRIVER |
| IC6 | 060-0012 | ICL8211 RESET GENERATOR |
| IC7 | 060-0051 | 74LS257 INPUT MULTIPLEXER |
| IC8-IC9 | 060-0041 | P5101L CMOS RAM CHIP |
| IC10 | 060-0016 | 8748 MICROPROCESSOR (PGM P1304) |
| IC11-IC1 | 040-0009 | DL2416 DISPLAY MODULE |
| K1 | 020-0038 | NC2D-JPL2-DC12V ARROW M RELAY |
| L1-L4 | 240-0010 | 47 MICROHENRY CHOKE |
| LED1-LED4 | 040-0025 | RED LITE BAR |
| MOV | 075-0002 | V130LA10A TRANSIENT SUPPRESSOR |
| Q1-Q3 | 070-0006 | MPS-U64 PNP DARLINGTON |

## WP6200 PARTS LIST (CONTINUED)

| REFERENCE | PART NUMBER | DESCRIPTION |
|---|---|---|
| R1 | 030-0002 | 33 OHM 1/4 WATT |
| R2 | 030-0049 | 243K OHM 1/4 WATT 1% |
| R3 | 030-0033 | 470K OHM 1/4 WATT |
| R4 | 030-0050 | 27.4K OHM 1/4 WATT 1% |
| R5-R10 | 030-0021 | 10K OHM 1/4 WATT |
| RN1 | 034-0016 | 220 OHM 7 RESISTOR SIP |
| RN2 | 034-0006 | 4.7K OHM 7 RESISTOR SIP |
| RN3 | 034-0003 | 10K OHM 7 RESISTOR SIP |
| T1 | 230-0037 | ST5-20 SIGNAL TRANSFORMER |
| X1 | 062-0001 | 6.00 MHZ CRYSTAL |

## WP6220 PARTS LIST

| | | |
|---|---|---|
| BR1 | 073-0005 | 1 AMP BRIDGE RECTIFIER |
| C1-C2 | 010-0044 | 0.47 MFD 250 VAC |
| C3 | 011-0040 | 4700 MFD 16 VDC |
| C4-C5 | 010-0002 | 27 PF DISC CERAMIC |
| C6 | 011-0001 | 1 MFD 15 VDC |
| C7-C16 | 010-0027 | 0.1 MFD 50 VDC CERAMIC |
| D1 | 071-0012 | 1 AMP 600 VOLT DIODE |
| DN1 | 071-0027 | 7 DIODE NETWORK |
| FU1 | 050-0001 | 5 AMP FUSE NO. 175-005 |
| IC1 | 061-0011 | LM7805 5V REGULATOR |
| IC2-IC3 | 060-0050 | MC75453 RELAY DRIVER |
| IC4 | 060-0012 | ICL8211 RESET GENERATOR |
| IC5 | 060-0016 | 8748 MICROPROCESSOR |
| IC6 | 060-0047 | 512X4 PROM |
| L1-L4 | 240-0010 | 47 MICROHENRY CHOKE |
| MOV | 075-0002 | V130LA10A TRANSIENT SUPPRESSOR |
| Q1 | 070-0006 | MPS-U64 PNP DARLINGTON |
| R1 | 030-0050 | 27.4K OHM 1/4 WATT 1% |
| R2 | 030-0049 | 243K OHM 1/4 WATT 1% |
| R3 | 030-0033 | 470K OHM 1/4 WATT |
| R4-R7 | 030-0021 | 10K OHM 1/4 WATT |
| T1 | 230-0037 | ST5-20 SIGNAL TRANSFORMER |
| X1 | 062-0001 | 6.00 MHZ CRYSTAL |

# WP6300 PARTS LIST

| REFERENCE | PART NUMBER | DESCRIPTION |
|---|---|---|
| B1 | 062-0005 | NI CAD BATTERY |
| BR1 | 073-0005 | 1 AMP BRIDGE RECTIFIER |
| C1-C2 | 010-0044 | 0.47 MFD 250 VAC |
| C3 | 011-0001 | 1 MFD 15 VDC |
| C4-C9 | 010-0002 | 27 PF DISC CERAMIC |
| C10 | 011-0040 | 4700 MFD 16 VDC |
| C11 | 010-0014 | 0.01 MFD 50 VDC |
| C12-C28 | 010-0027 | 0.1 MFD 50 VDC CERAMIC |
| D1-D2 | 071-0012 | 1 AMP 600 VOLT DIODE |
| D3 | 071-0026 | SCHOTTKY DIODE |
| DN1 | 071-0028 | 8 DIODE NETWORK |
| FU1 | 050-0001 | 5 AMP FUSE NO. 175-005 |
| IC1 | 062-0011 | LM7805 5V REGULATOR |
| IC2-IC5 | 060-0046 | TIL-113 OPTO-COUPLER |
| IC6-IC9 | 060-0050 | MC75453 RELAY DRIVER |
| IC10 | 060-0016 | 8748 MICROPROCESSOR (PGM P1305) |
| IC11 | 060-0012 | ICL8211 RESET GENERATOR |
| IC12-IC13 | 060-0042 | 8T97 HEX BUFFER |
| IC14 | 060-0045 | HCPL-2600 OPTO-COUPLER |
| IC15 | 060-0041 | P5101L CMOS RAM CHIP |
| L1-L4 | 240-0010 | 47 MICROHENRY CHOKE |
| LED1-LED12 | 040-0025 | RED LITE BAR |
| MOV | 075-0002 | V130LA10A TRANSIENT SUPPRESSOR |
| R1 | 030-0002 | 33 OHM 1/4 WATT |
| R2 | 030-0049 | 243K OHM 1/4 WATT 1% |
| R3 | 030-0033 | 470K OHM 1/4 WATT |
| R4 | 030-0050 | 27.4K OHM 1/4 WATT 1% |
| R5-R7 | 030-0021 | 10K OHM 1/4 WATT |
| R8-R11 | 030-0038 | 2.2 MEGOHM 1/4 WATT |
| R12-R13 | 030-0010 | 270 OHM 1/4 WATT |
| R14 | 030-0021 | 10K OHM 1/4 WATT |
| RN1 | 034-0014 | 47 OHM 4 RESISTOR SIP |
| RN2 | 034-0017 | 220 OHM 15 RESISTOR DIP |
| RN3 | 034-0006 | 4.7K OHM 7 RESISTOR SIP |
| T1 | 230-0037 | ST5-20 SIGNAL TRANSFORMER |
| X1 | 062-0001 | 6.00 MHZ CRYSTAL |

## WP6320 PARTS LIST (CONTINUED)

| REFERENCE | "W" NUMBER | DESCRIPTION |
|---|---|---|
| C10 | W1063 | 4700 MFD 16 VDC |
| C11 | W1021 | 0.01 MFD 50 VDC |
| C12-C27 | W1020 | 0.1 MFD 50 VDC CERAMIC |
| D1 | W7000 | 1 AMP 600 VOLT DIODE |
| DN1 | W12043 | 8 DIODE NETWORK |
| FU1 | W17095 | 5 AMP FUSE NO. 175-005 |
| IC1 | W12089 | LM7805 5V REGULATOR |
| IC2-IC5 | W12075 | TIL-113 OPTO-COUPLER |
| IC6-IC11 | W12082 | MC75453 RELAY DRIVER |
| IC12 | W12066 | 8748 MICROPROCESSOR |
| IC13 | W12078 | ICL8211 RESET GENERATOR |
| IC14 | W12073 | HCPL-2600 OPTO-COUPLER |
| IC15 | W12076 | 512X4 PROM |
| L1-L4 | W5017 | 47 MICROHENRY CHOKE |
| MOV | W7012 | V130LA10A TRANSIENT SUPPRESSOR |
| R1 | W2070 | 243K OHM 1/4 WATT 1% |
| R2 | W2006 | 470K OHM 1/4 WATT |
| R3 | W2071 | 27.4K OHM 1/4 WATT 1% |
| R4 | W2008 | 27K OHM 1/4 WATT |
| R5 | W2007 | 10K OHM 1/4 WATT |
| R6-R9 | W2029 | 2.2 MEGOHM 1/4 WATT |
| R10 | W2050 | 470 OHM 1/4 WATT |
| RN1 | W2065 | 47 OHM 4 RESISTOR SIP |
| T1 | W5016 | ST5-20 SIGNAL TRANSFORMER |
| X1 | W7034 | 6.00 MHZ CRYSTAL |

## WP6350 PARTS LIST

| | | |
|---|---|---|
| IC1-IC2 | W12043 | DIODE NETWORK |

## TYPE A PLUG-IN INPUT MODULE PARTS LIST

| | | |
|---|---|---|
| C1-C2 | W1020 | 0.1 MFD 50 VDC CERAMIC |
| CN1 | W1065 | 0.01 MFD 8 CAPACITOR NETWORK |
| IC1-IC4 | W12088 | H11L1 OPTO-COUPLER |
| RN1 | W2038 | 1K OHM 8 RESISTOR DIP |
| RN2 | W2033 | 3.9K OHM 7 RESISTOR SIP |

## TYPE A PLUG-IN INPUT MODULE PARTS LIST

| REFERENCE | PART NUMBER | DESCRIPTION |
|---|---|---|
| C1-C2 | 010-0027 | 0.1 MFD 50 VDC CERAMIC |
| CN1 | 013-0001 | 0.01 MFD 8 CAPACITOR NETWORK |
| IC1-IC4 | 060-0015 | H11L1 OPTO-COUPLER |
| RN1 | 034-0005 | 1K OHM 8 RESISTOR DIP |
| RN2 | 034-0006 | 3.9K OHM 7 RESISTOR SIP |

## TYPE A PLUG-IN OUTPUT MODULE PARTS LIST

| | | |
|---|---|---|
| K1-K2 | 020-0024 | 12 VDC SCHRACK RELAY |

## TYPE B PLUG-IN OUTPUT MODULE PARTS LIST

| | | |
|---|---|---|
| OC1-OC2 | 060-0005 | 4N28 OPTO-COUPLER |
| R1-R2 | 031-0024 | 220 OHM 1/2 WATT |
| R3-R4 | 030-0013 | 1K OHM 1/4 WATT |

## TYPE C PLUG-IN OUTPUT MODULE PARTS LIST

| | | |
|---|---|---|
| Q1-Q2 | 070-0006 | MPS-U64 PNP DARLINGTON |
| R1-R4 | 030-0021 | 10K OHM 1/4 WATT |

--------- N O T E S ---------

## DIMENSIONAL DATA

All Dimensions in Inches

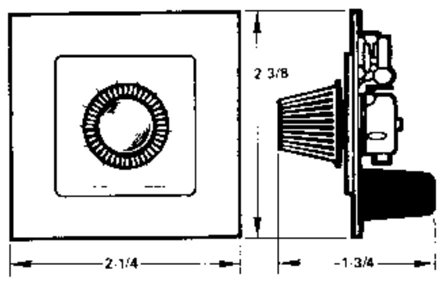



Figure 8-1 WP6012 DIMENSIONS

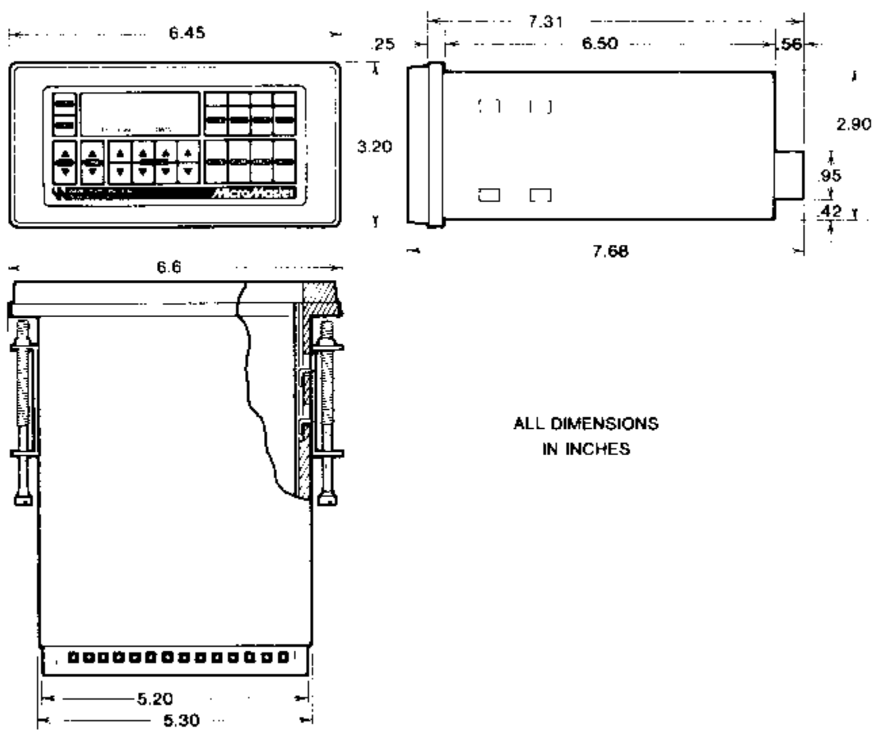



Figure 8-2 WP6200, WP6240 AND WP6300 DIMENSIONS

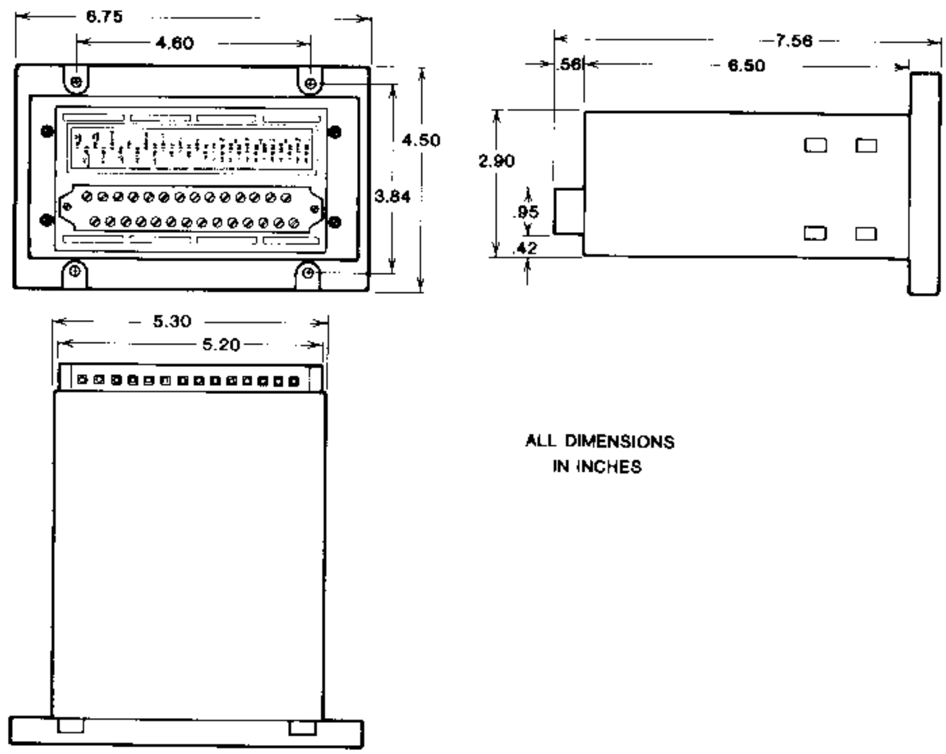



Figure 8-3 WP6220 and WP6320 DIMENSIONS

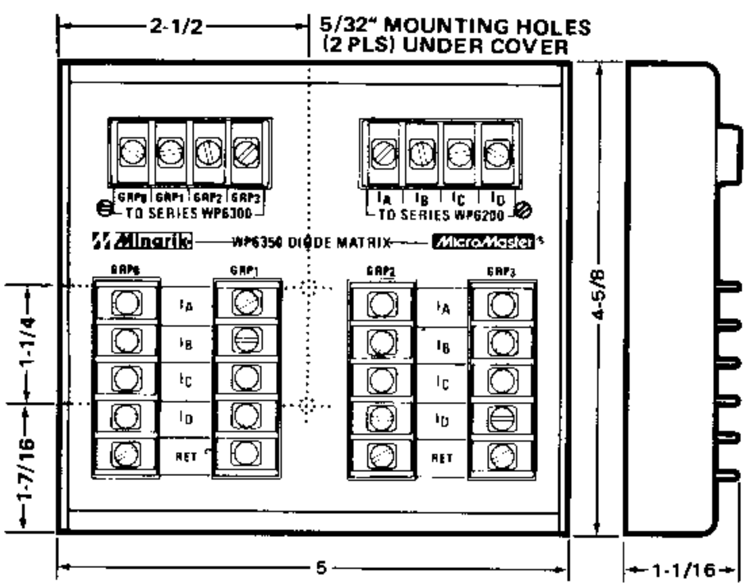



Figure 8-4 WP6350 DIMENSIONS

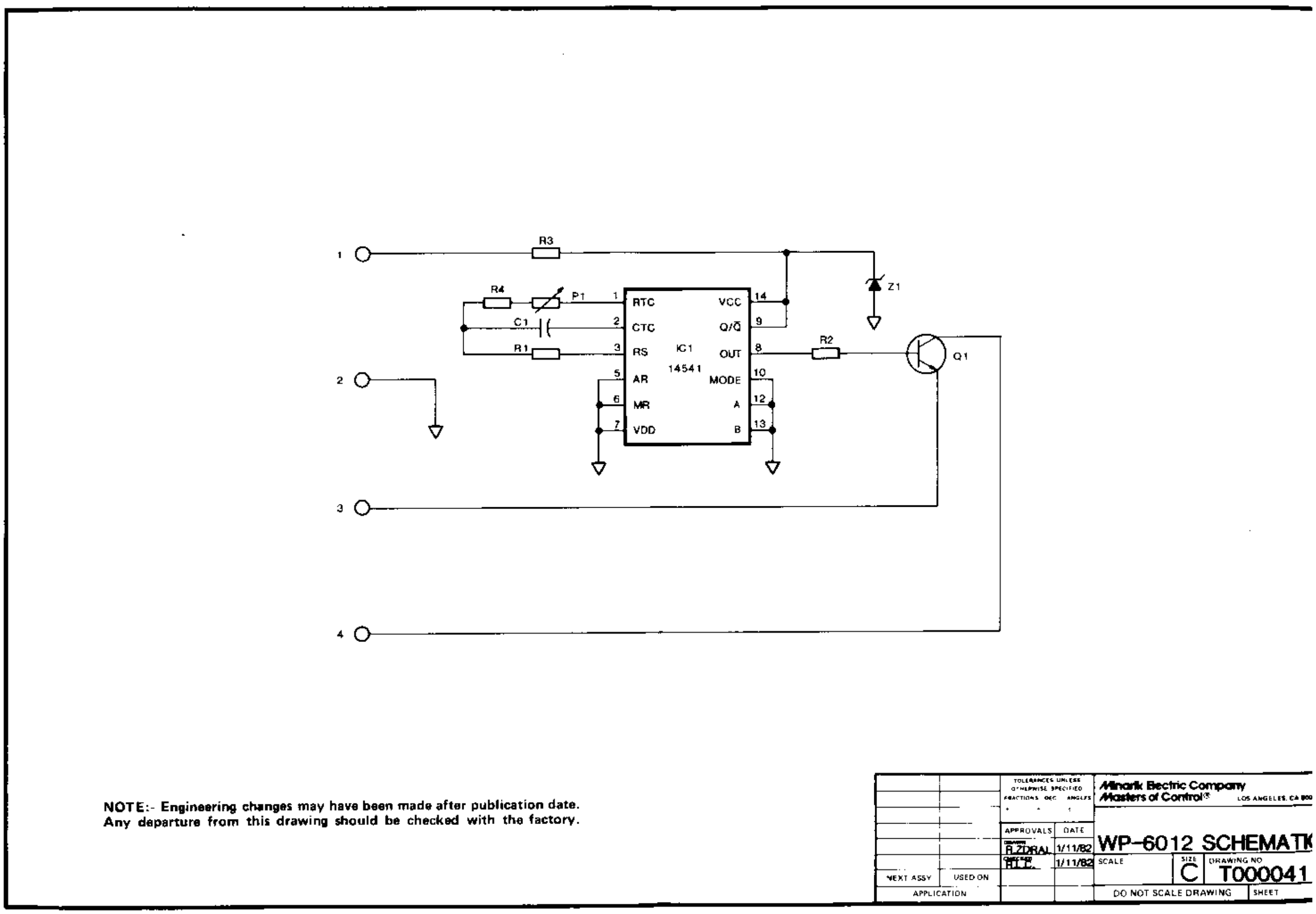



Figure 8-5 WP6012 SCHEMATIC DIAGRAM

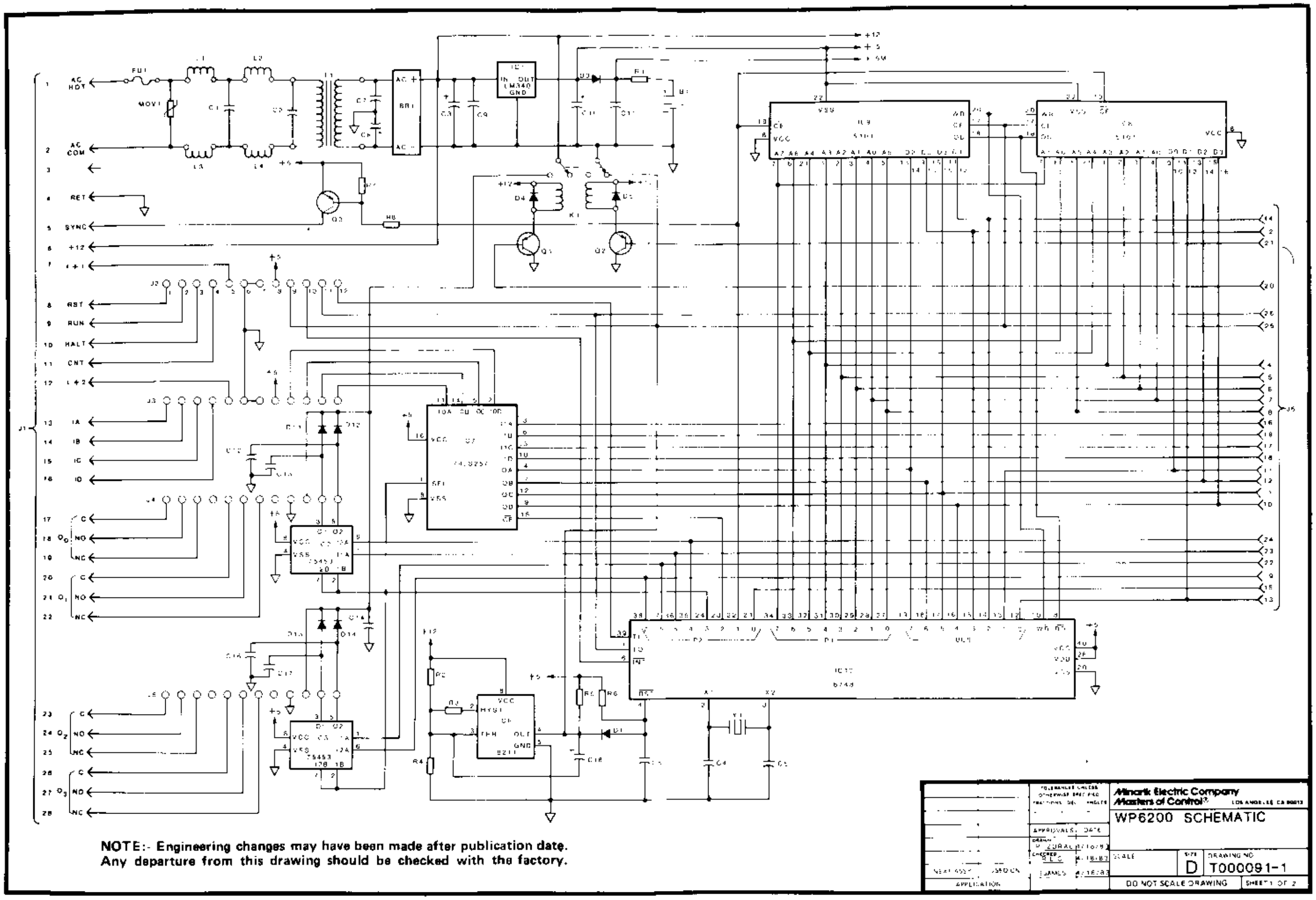
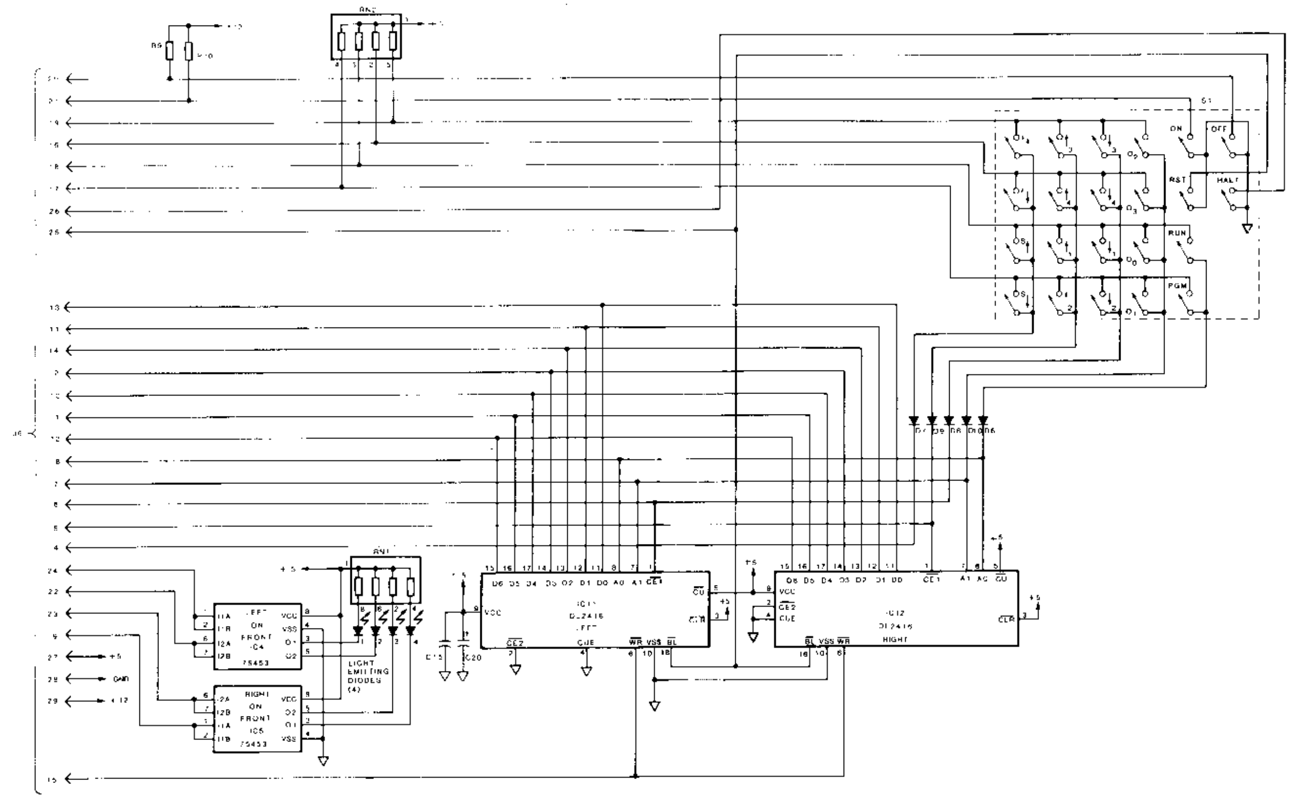


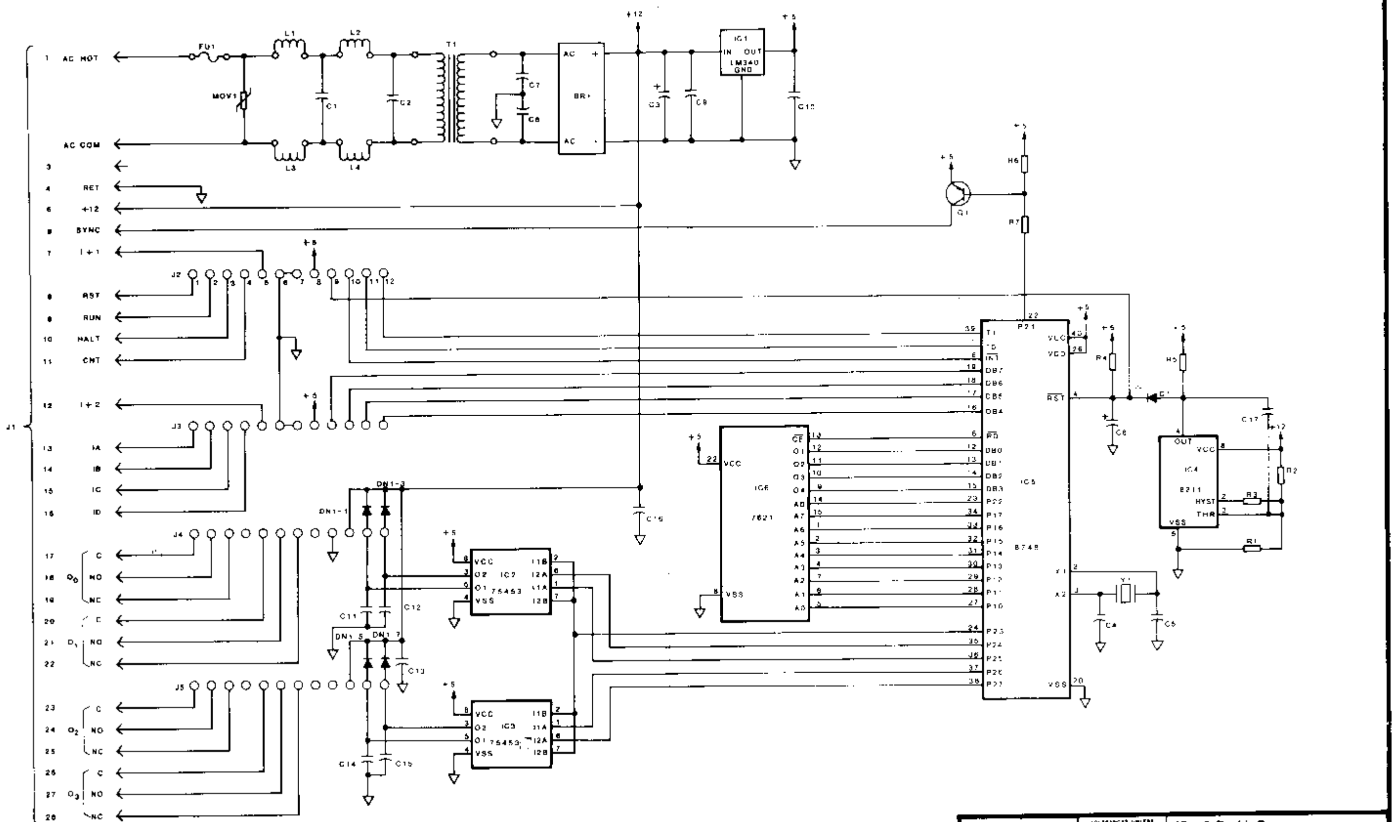

Figure 8-6 WP6200 SCHEMATIC DIAGRAM, PART 1




NOTE:- Engineering changes may have been made after publication date.
Any departure from this drawing should be checked with the factory.

| | |
|---|---|
| Minarik Electric Company Masters of Control® | |
| WP6200 SCHEMATIC | |
| FRONT PANEL | |
| APPROVALS: R. ZORAL 4/18/83; B.L.C. 4/18/83; G. JAMES 4/18/83 | SIZE D, DRAWING NO T000091 |
| DO NOT SCALE DRAWING | |

Figure 8-7 WP6200 SCHEMATIC DIAGRAM, PART 2




NOTE:- Engineering changes may have been made after publication date.
Any departure from this drawing should be checked with the factory.

Minarik Electric Company
Masters of Control®

WP6220 SCHEMATIC

WP6200 SERIES

SIZE D DRAWING NO T000084

DO NOT SCALE DRAWING

Figure 8-8 WP6220 SCHEMATIC DIAGRAM

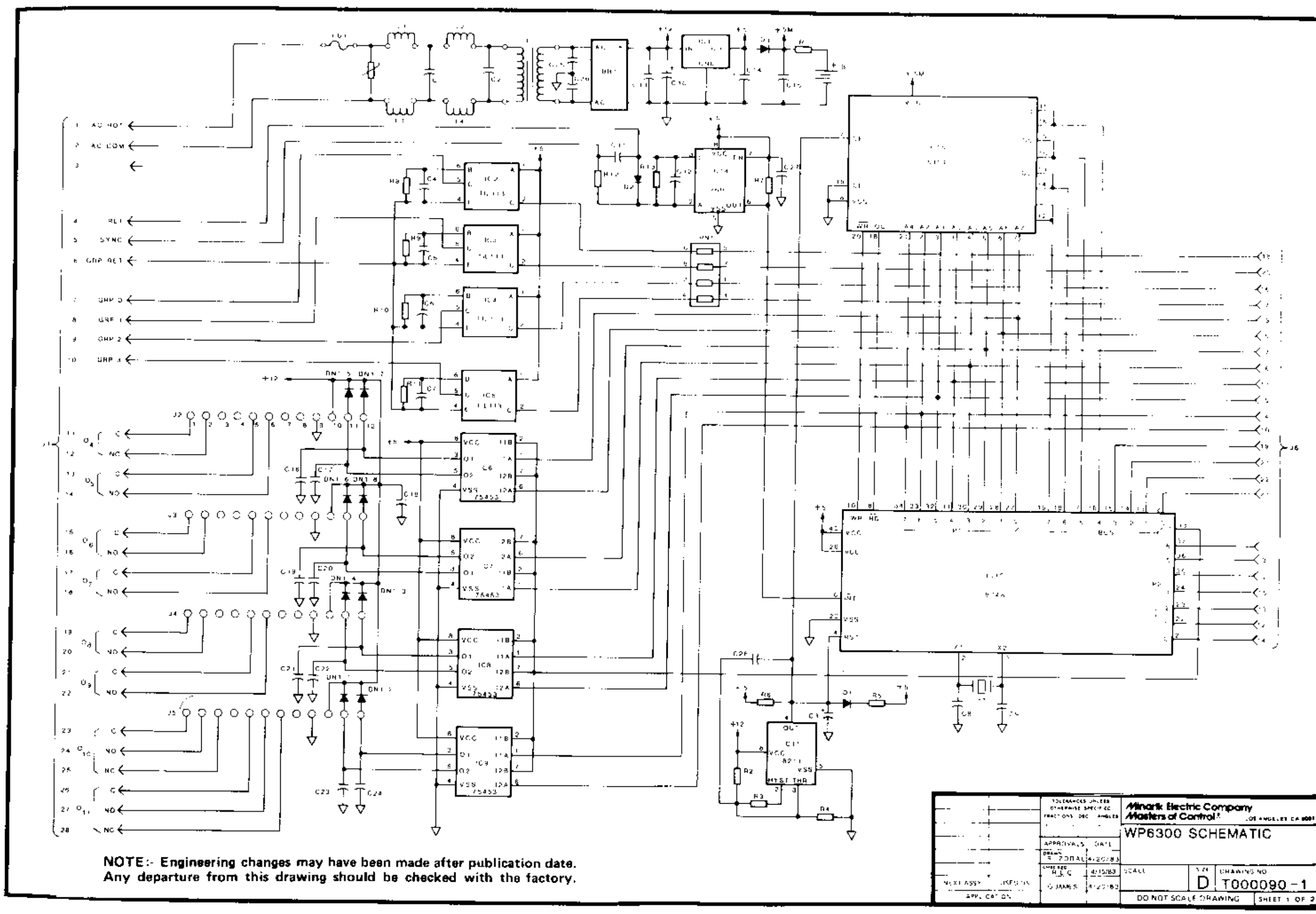



Figure 8-9 WP6300 SCHEMATIC DIAGRAM, PART 1

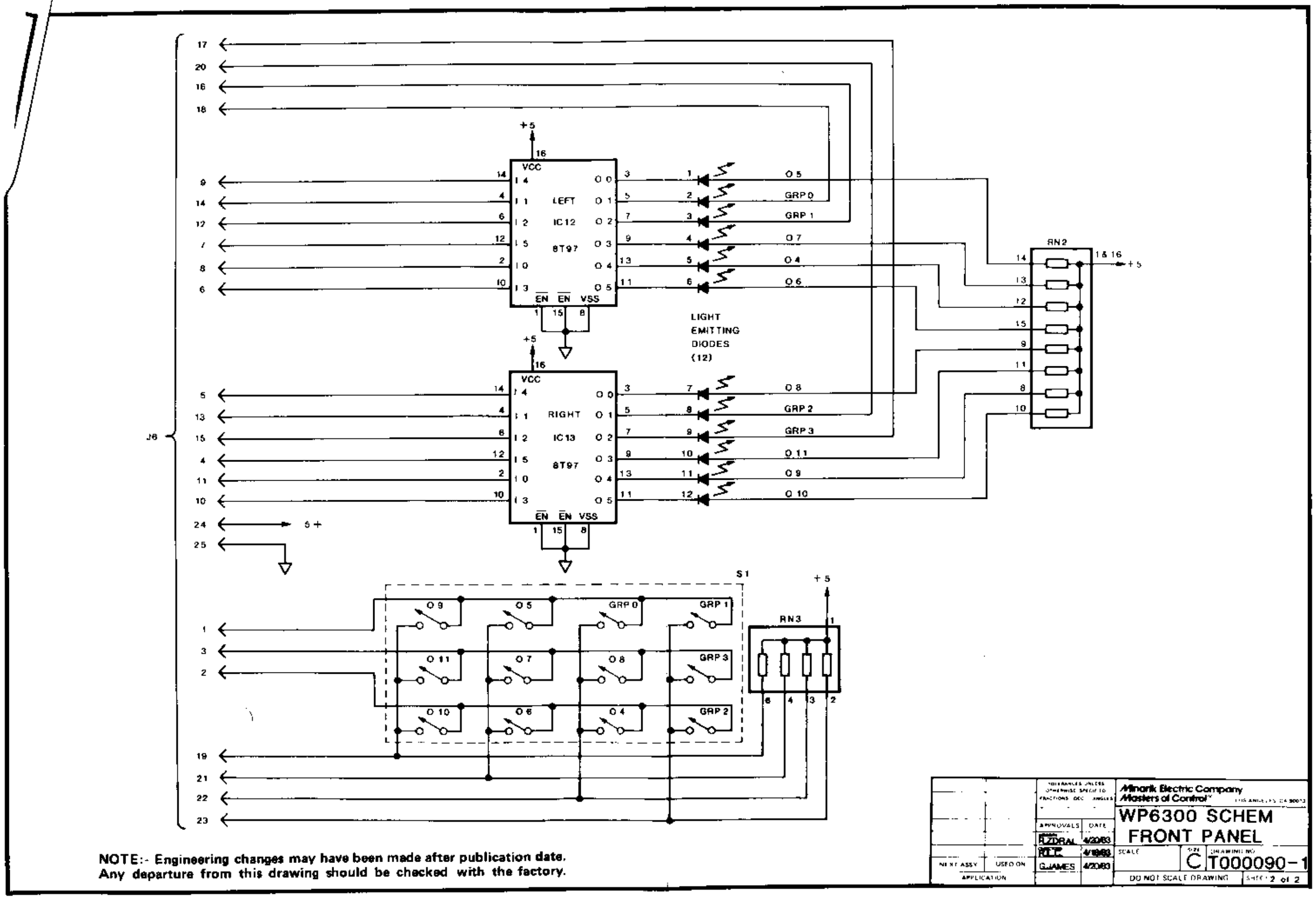



Figure 8-10 WP6300 SCHEMATIC DIAGRAM, PART 2

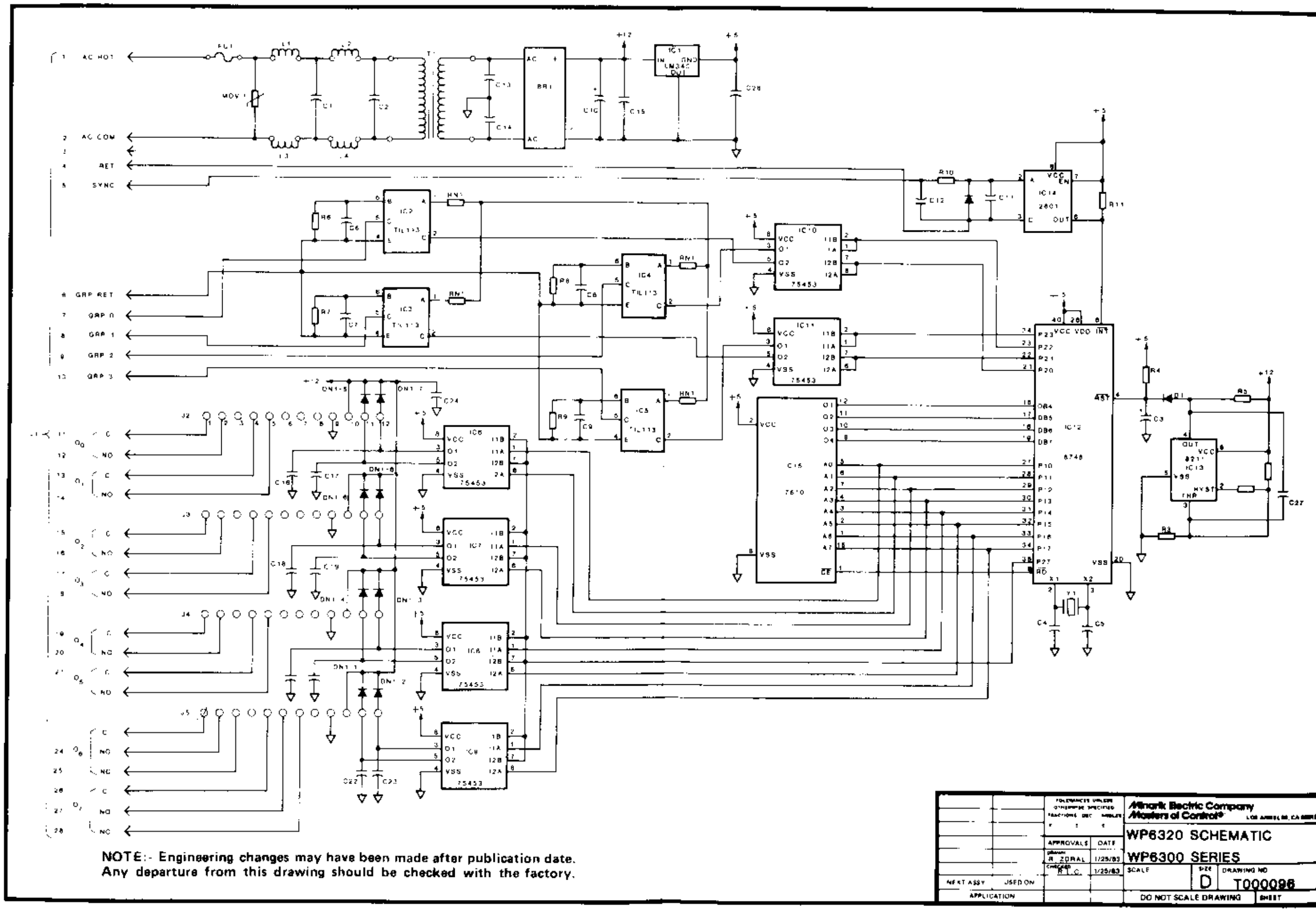



Figure 8-11 WP6320 SCHEMATIC DIAGRAM

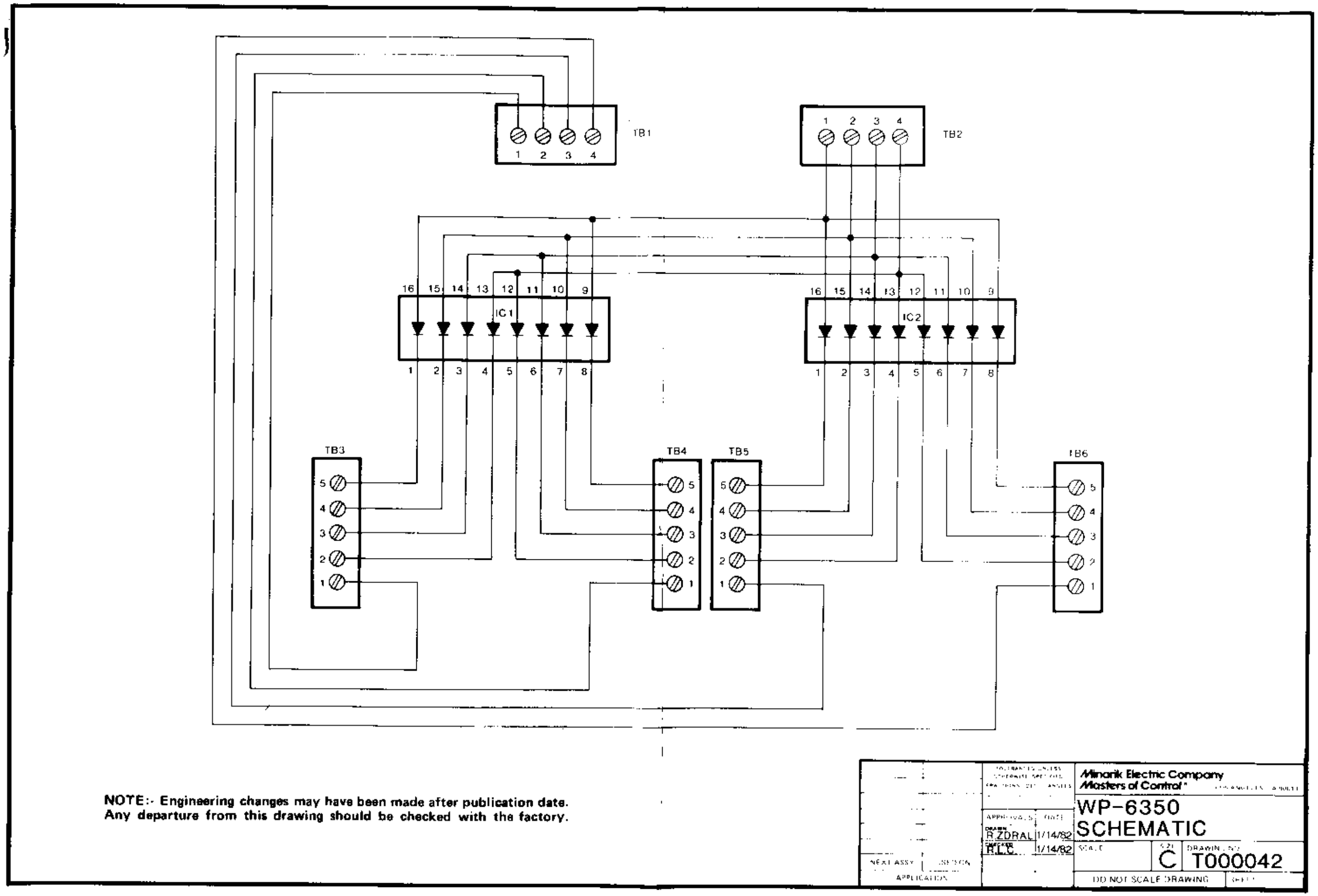



Figure 8-12 WP635Q SCHEMATIC DIAGRAM

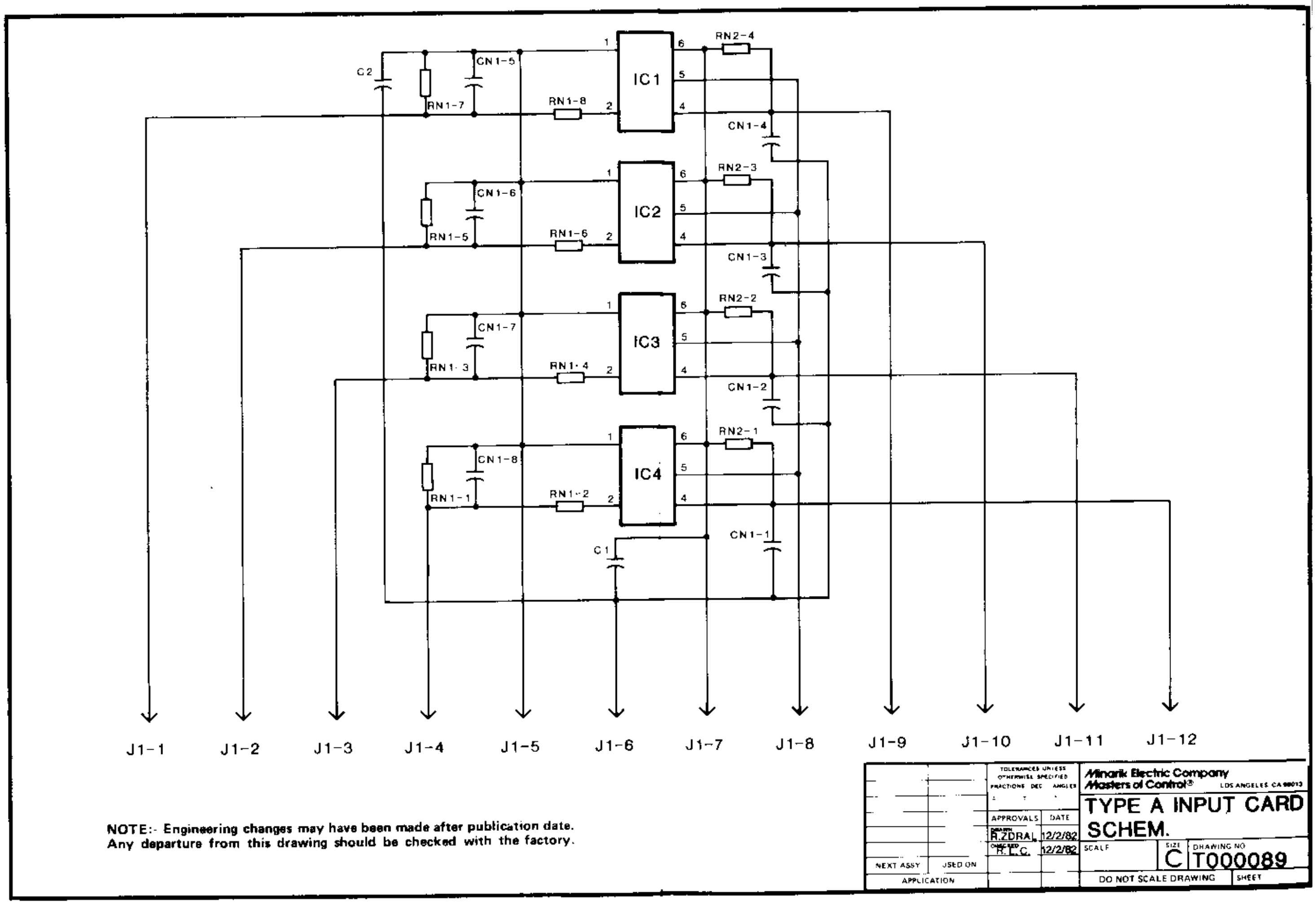



Figure 8-13 TYPE A PLUG-IN INPUT MODULE SCHEMATIC DIAGRAM

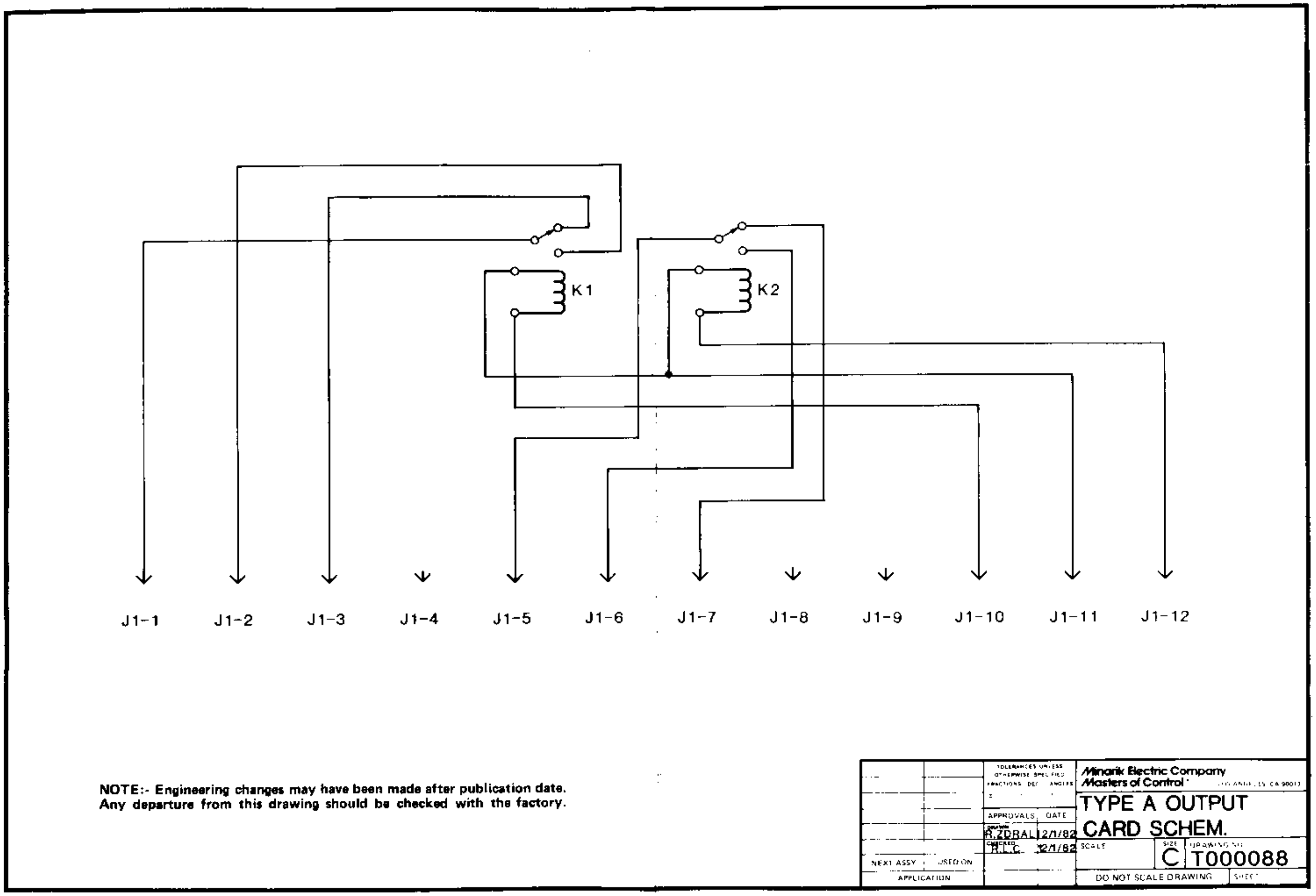
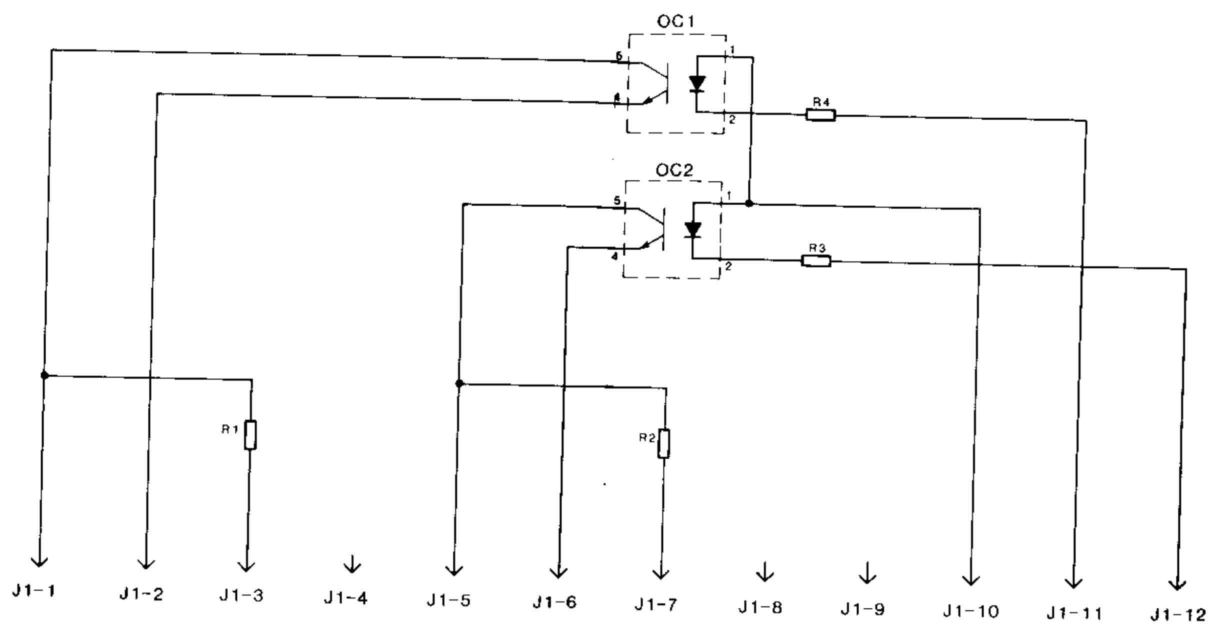



Figure 8-14 TYPE A PLUG-IN OUTPUT MODULE SCHEMATIC DIAGRAM




NOTE:- Engineering changes may have been made after publication date.
Any departure from this drawing should be checked with the factory.

Minarik Electric Company
Masters of Control®

TYPE B OUTPUT
CARD SCHEM.

| APPROVALS | DATE |
|---|---|
| DRAWN R.ZDRAL | 1/22/83 |
| CHECKED B.T.C. | 1/22/83 |

SIZE C DRAWING NO. T0000

DO NOT SCALE DRAWING

NEXT ASSY | USED ON | APPLICATION

Figure 8-15 TYPE B PLUG-IN OUTPUT MODULE SCHEMATIC DIAGRAM

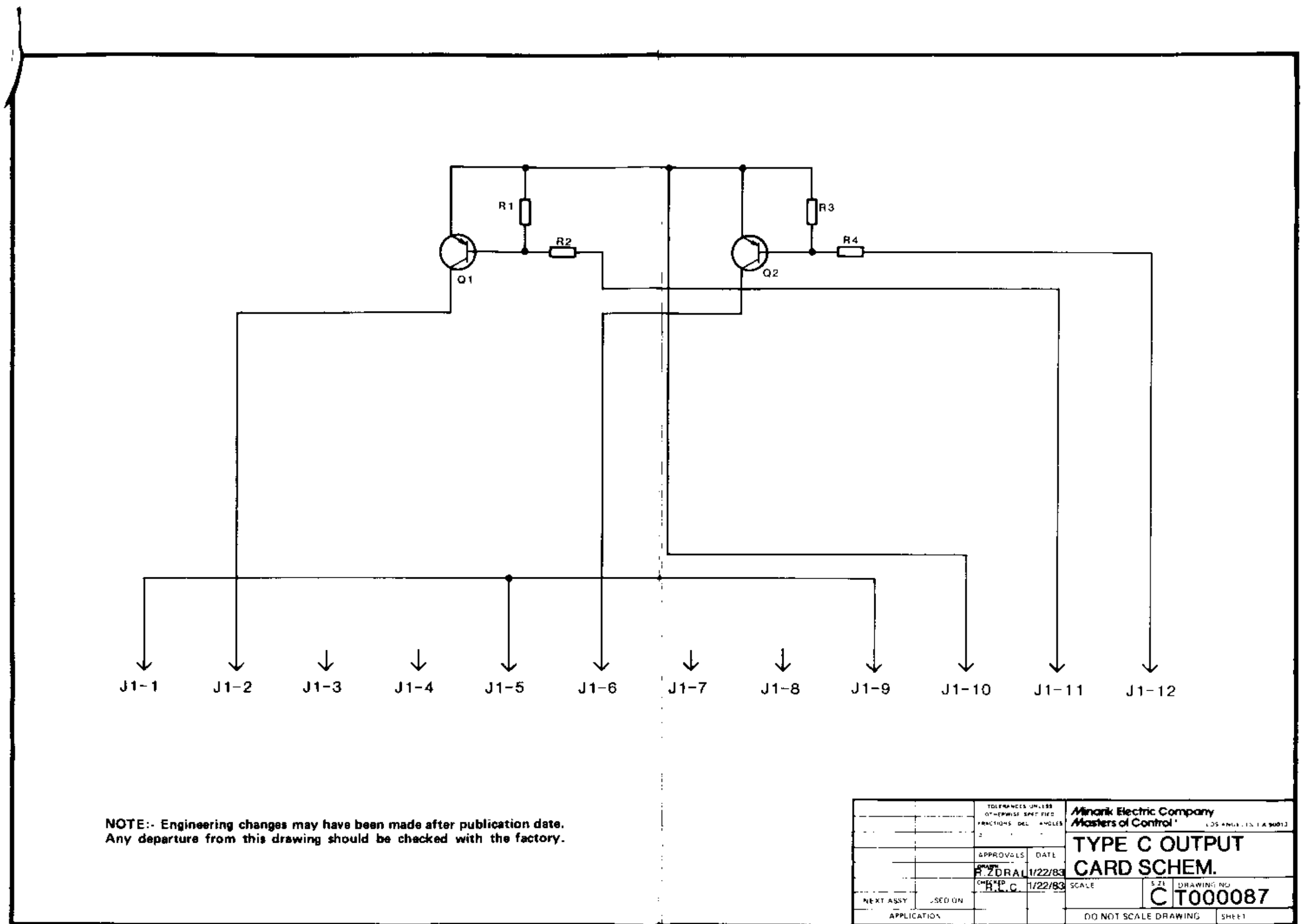



Figure 8-16 TYPE C PLUG-IN OUTPUT MODULE SCHEMATIC DIAGRAM

## DISCLAIMER

The information and technical data in this manual are subject to change without notice. Minarik Corporation and its Divisions make no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Minarik Corporation and its Divisions assume no responsibility for any errors that may appear in this manual and make no commitment to update or to keep current the information in this manual.

Minarik Masters of Motion Control

Area Code 818 - 502-1528

FAX 1-818-502-9002

901 East Thompson Avenue - Glendale, CA 91201-2011

DOCUMENT NUMBER 250-0024, REVISION 3
LITHO U.S.A. - 3/88

## LIMITED WARRANTY

**A. WARRANTY:–** Minarik Corporation warrants that their products will be free from defects in material and workmanship for a period of two (2) years from date of shipment thereof. Within the warranty period Minarik Corporation will repair or replace such products which are determined by us to be defective and which are returned to Minarik Corporation, 901 East Thompson Avenue, Glendale, CA 91201-2011 or to the nearest Minarik Authorized Service Station, with shipping charges prepaid. At our option, all return shipments are F.O.B. Minarik Corporation or its Authorized Service Station. This warranty will not apply to any product which has been subjected to misuse, negligence or accident; or misapplied; or repaired by unauthorized persons; or improperly installed. Minarik is not responsible for removal, installation or any other incidental expenses incurred in shipping the product to or from the repair point.

**B. DISCLAIMER:–** The provisions of paragraph 'A' are Minarik's sole obligation and exclude all other warranties of MERCHANTABILITY or use, express or implied. We further disclaim any responsibility whatsoever to the customer or to any other person for injury to person, or damage to or loss of property of value, caused by any product which has been subjected to misuse, negligence or accident; or misapplied; or modified or repaired by unauthorized persons; or improperly installed.

**C. LIMITATION OF LIABILITY:–** In the event of any claim for breach of any of Minarik's obligations, whether express or implied, and particularly in the event of any claim of a breach of the warranty contained in paragraph 'A', or of any other warranties, express or implied, or claim of liability, which might, despite paragraph 'B', be decided against us by any lawful authority, Minarik Corporation shall under no circumstances be liable for any consequential damages, losses or expense arising in connection with the use of, or inability to use, our product for any purpose whatsoever. An adjustment made under the warranty does not void the warranty, nor does it imply an extension of the orginal two (2) years warranty period. Products serviced and/or parts replaced on a no charge basis during the warranty period carry the unexpired portion of the original warranty only.

If for any reason any of the foregoing provisions shall be ineffective, the corporation's liability for damages arising out of its manufacture or sale of equipment, or use thereof, whether such liability is based on warranty, contract, negligence, strict liability in tort or otherwise, shall not in any event exceed the full purchase price of such equipment.

Any action against the Corporation based upon any liability or obligation arising hereunder or under any law applicable to the sale of equipment or the use thereof, must be commenced within one year after the cause of such action arises.

Minarik Masters of Motion Control