Addressing

Addressing Format

The M221 Logic Controller addressing uses the following format:

%<Type>[<Identifier>]<Location>

An address must always begin with the % character. This tells the Logic Controller that it is an address, not some other piece of information.

This is followed by one of five letters identifying the type of address:

I	The address is a physical input on either the controller or an expansion module.
K	The address is an internal memory location within the controller. The value is fixed and can NOT be changed by the program.
M	The address is an internal memory location within the controller. This value can be changed by the program.
Q	The address is a physical output on either the controller or an expansion module.
S	Internal system locations that are used to perform various functions and monitor the controller

The type may then be followed by a type identifier which can be one of the following:

(none)	The address contains a value that is a single bit having a value of either 0 or 1.
W	The address contains a value that is a word and has a value between 0 and 65535.
D	The address contains a value that is a double word and has a value between 0 and 4294967295.
F	The address contains a value that is in floating point format and has a value between 0 and 65535.

The numeric part of the address contains the location. This location can have one of two formats depending on the address type. The format nnnn is used for internal memory locations. The format x.y is used for inputs and outputs where the first part (x) identifies the module position in the rack. Slot zero is the left most slot and is the processor.

The second part (y) identifies the input or output number on that module.

Thus an address of %I1.2 refers to the third input on the first expansion module (Numbering starts at zero).

Exercise - Identifying Addresses

Learning Outcomes

By the completion of this exercise you will:

- Understand the addressing format used by SoMachine Basic

1 Explain the following addresses.

Address	Int/Ext	Location	Data Type
%I0.4			
%MW2			
%Q0.6			
%IW1			
%S6			
%MF10			
%KW2			
%SW100			

2 Create an address for the following.

i. The second digital input of the logic controller

ii. An internal memory location that can be used to store the value 17.4

iii. System word 8

iv. The third output of the second expansion module (assuming that it is a digital output)

v. The first input of the second digital output module

Addressing (cont.)

I/O List

The following I/O will be used in the application.

Inputs:

Address	Description
%I0.2	Conveyor 1 Fault Signal
%I0.3	Conveyor 1 Fault Signal
%I0.4	Conveyor 1 Fault Signal
%I0.6	Stop Button
%I0.7	Start Button

Outputs:

Address	Description
%Q0.2	Conveyor 1 Run Control
%Q0.3	Conveyor 2 Run Control
%Q0.4	Conveyor 3 Run Control
%Q0.6	Run Indicator
%Q0.7	Fault Indicator

Analog:

Address	Description
%IW0.0	Conveyor Speed

Addressing (cont.)

Register List

The following internal registers will be used in the application.

Address	Description
%M100	Run Relay
%M101	Conveyor 1 Stop Sequence
%M102	Conveyor 2 Stop Sequence
%M103	Conveyor 3 Stop Sequence
%M104	Not Used
%M105	Conveyor 1 Fault
%M106	Conveyor 2 Fault
%M107	Conveyor 3 Fault
%M108	Not Used
%M109	Stop Sequence Relay
%M110	Remote Start
%M119	Remote Stop
%M120	Variable Speed OK
%M121	Variable Speed High
%M122	Variable Speed Low

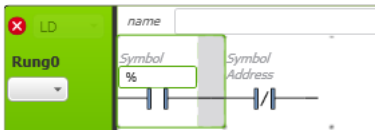
Address	Description
%MW1	Low Setpoint
%MW2	High Setpoint
%MW3	Scaled Speed

Addressing (cont.)

How to Assign Objects to I/O and Memory

When a program is created, the objects used in the program must be assigned to either an I/O or memory address. If this is not done, the program cannot be compiled and downloaded to the Logic Controller.

To assign an object to an I/O or memory address, first double-click the address above the object. An entry field will appear.



Enter the desired address into this entry field.

Note:

The % character is already in the entry box as this identifies the entry as an address (see previous section).