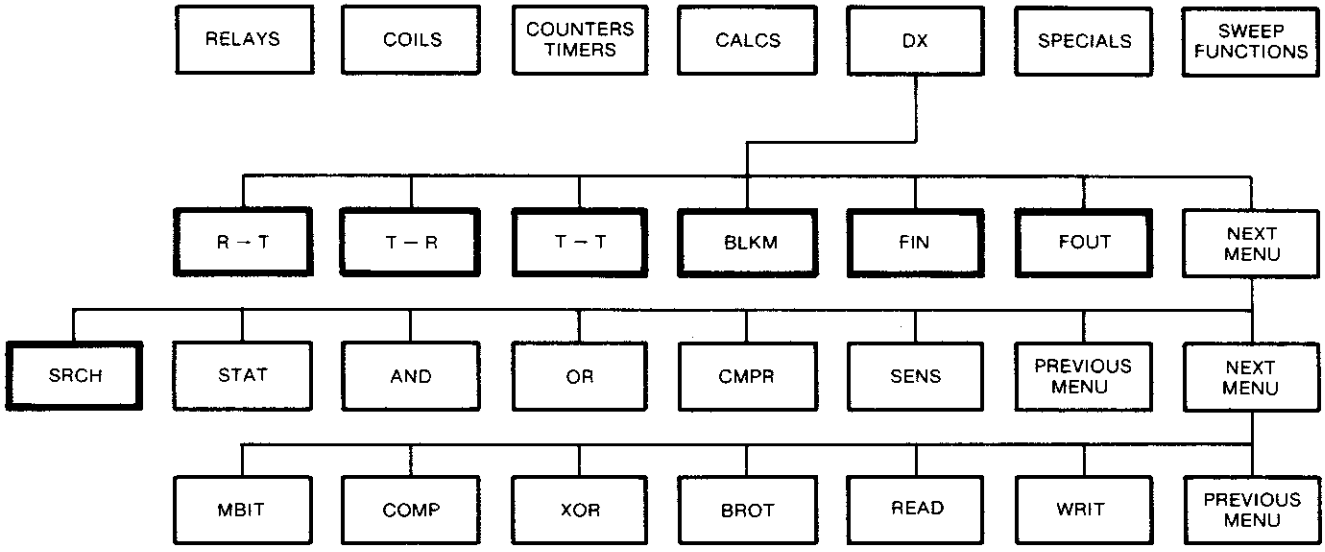
DATA TRANSFER (DX) MOVE FUNCTIONS



DATA TRANSFER (DX) MOVE FUNCTIONS

Data transfer or "DX" move functions copy data (16 bit words) from one memory area to another. The data can then be operated on without altering the original data. These functions are available under the DX portion of the programming menu and include:

register to table move (R → T)
table to register move (T → R)
table to table move (T → T)
block move (BLKM)
first in (FIN)
first out (FOUT)
table search (SRCH)
system status (STAT)

Table

A group of consecutive 16 bit words is referred to as a table. Several examples of tables are shown below, each has a "table length" of 5. Note, however, that each discrete table location contains the status ("1 = ON, "0" = off) of 16 coils or inputs.

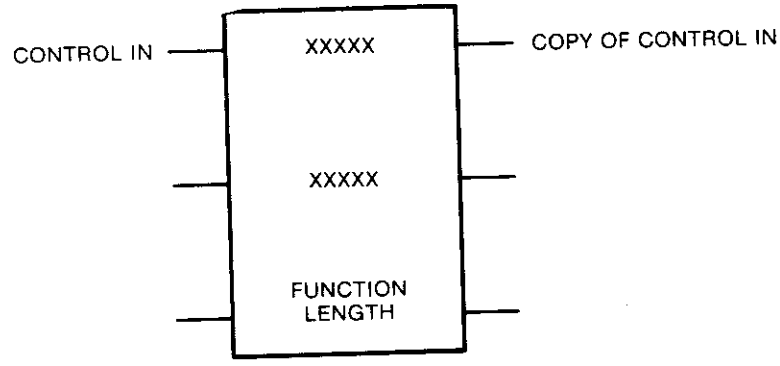
Input Register Table
30001
30002
30003
30004
30005

Holding Register Table
40001
40002
40003
40004
40005

Output Table
00001 to 00016
00017 to 00032
00033 to 00048
00049 to 00064
00065 to 00080

Input Table
10001 to 10016
10017 to 10032
10033 to 10048
10049 to 10064
10065 to 10080

Format



The figure above illustrates the general format of a data transfer move function block. The block consumes 1 horizontal and 3 vertical nodes (except system status which consumes 1 horizontal and 2 vertical nodes). The bottom node within the block specifies the function and the table length.

The top is always the "control in" input and the top output is always a copy of the "control in" input. This allows DX blocks to be cascaded.

The top and middle nodes within the block, and all other inputs and outputs vary with each function.

DATA TRANSFER (DX) MOVE FUNCTIONS

Length

Minimum table length is one (one 16 bit word or register). Maximum table length is dependent on the DX function and the controller type (16 or 24 bit CPU). See figure below.

Controller Model	R → T	T → R	T → T	BLKM	FIN	FOUT	SRCH	STAT
*584A level 1 & 2	255	255	255	100	100	100	100	71
*584M level 1 & 2	255	255	255	100	100	100	100	71
*584L level 1	255	255	255	100	100	100	100	75
**584A level 3	999	999	999	100	100	100	100	71
**584L level 2	999	999	999	100	100	100	100	75

* = 16 bit CPU
** = 24 bit CPU

Pointer

Some of the DX move functions dedicate a register which indicates which table position the data was:

copied from (T → R, T → T & FOUT)
written to (R → T & FIN)
found (SRCH)

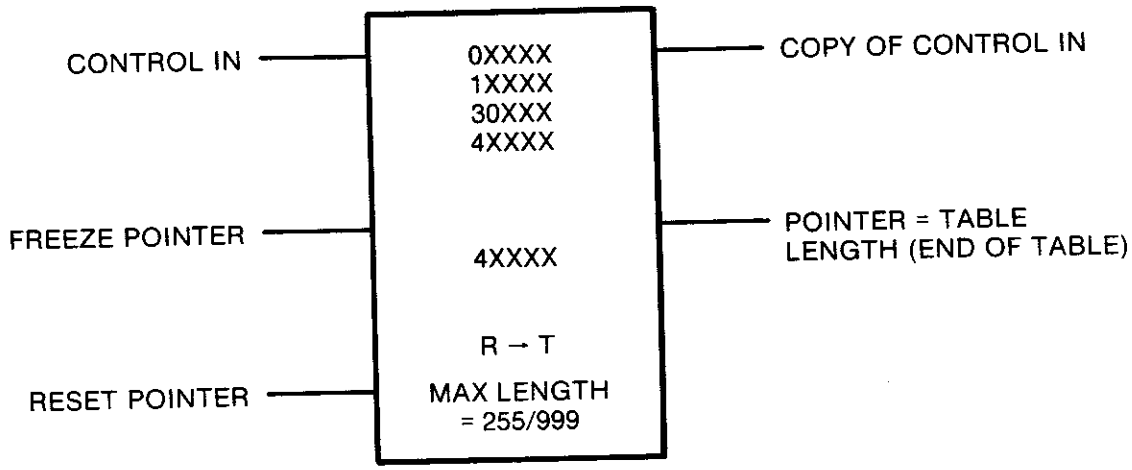
This register is termed the "pointer register". The pointer register value will never exceed the table length. Zero is a valid pointer value, and typically indicates that the next operation of the DX block will copy data from or write data to the first table position. The pointer register may be manipulated by other logic functions (timers, counters, arithmetic operations, etc.).

Discrete Tables

Discrete references are used in groups of 16. The reference number used is the first in the group; the other 15 references are implied. Additionally, only reference numbers which appear in the "first of 16" table shown on page 12-12 may be used, e.g. 1, 17, 33, etc.

REGISTER TO TABLE MOVE

This function copies the bit pattern of any register or 16 discretes to a specific register located within a table.



Function Block

The top node is the data source and can one of the following:

- the first of 16 logic coils (0XXXX)
- the first of 16 inputs (1XXXX)
- an input register (30XXX)
- a holding register (4XXXX)

The middle node must a holding register (4XXXX). This register is the pointer to the destination table which starts with the next consecutive register (4XXXX + 1).

The bottom node contains the symbol R → T and a number specifying table length. Table length may range from:

1 to 255 in 16 bit controllers
1 to 999 in 24 bit controllers

Inputs

Top: Control in. Every scan this node is powered, the R → T move is performed, and the pointer is incremented until the pointer value = the table length. A transitional contact should be used if single operations are desired.

Middle: When powered, prevents pointer from increasing.

Bottom: When powered, resets pointer to zero

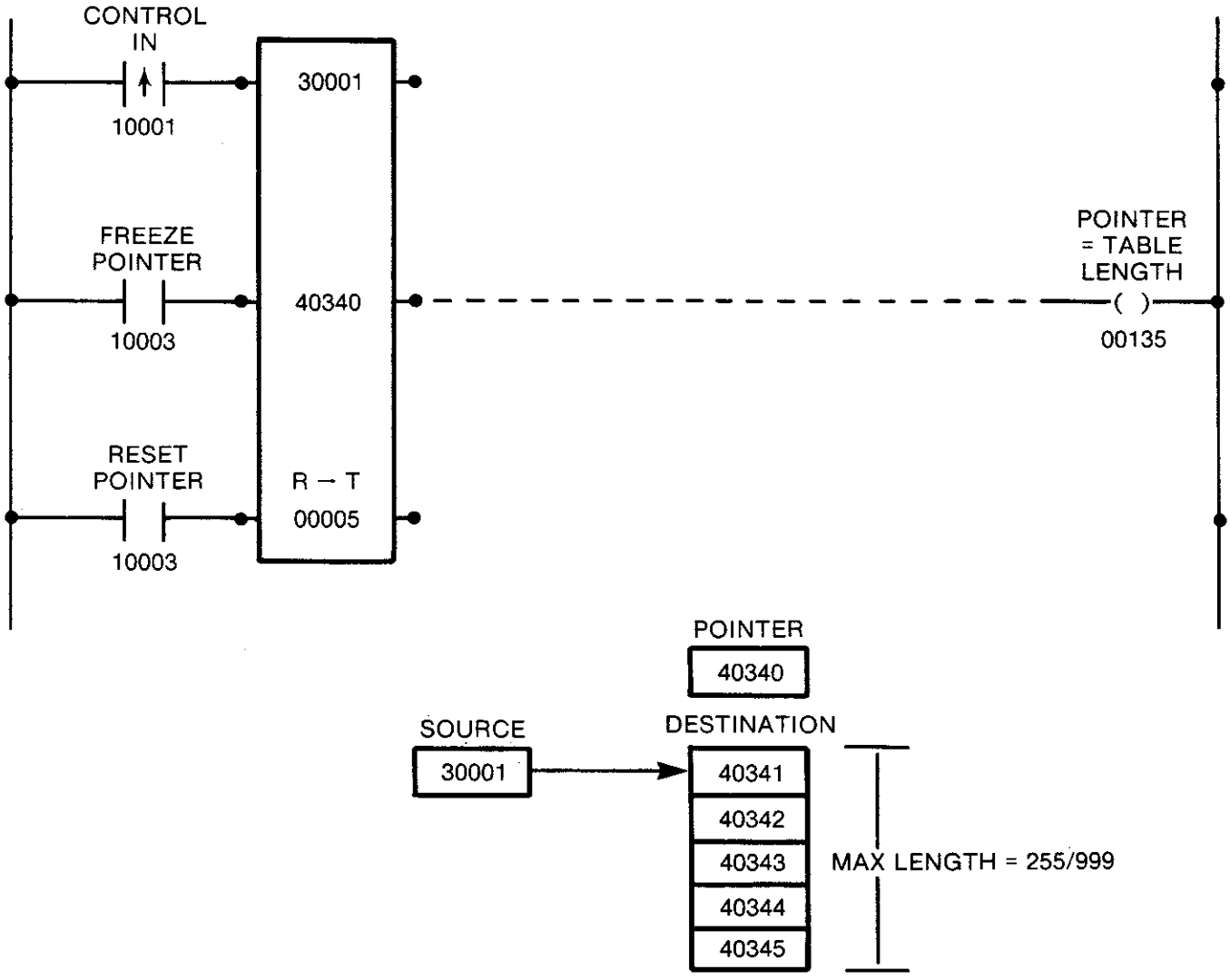
Outputs

Top: Passes power when top input is powered.

Middle: Passes power when the pointer = table length (end of table). No R → T operations are possible while the pointer value = the table length.

REGISTER TO TABLE MOVE

Network #44



Network Description

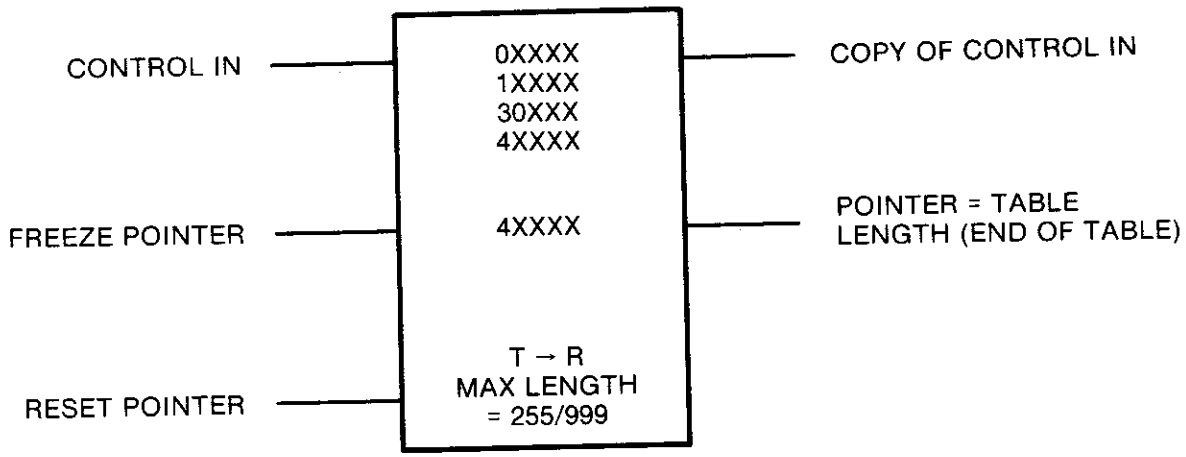
The 1st transition of 10001 copies 30001 to 40341, increments pointer to 1.
The 2nd transition of 10001 copies 30001 to 40342, increments pointer to 2.
The 3rd transition of 10001 copies 30001 to 40343, increments pointer to 3.
The 4th transition of 10001 copies 30001 to 40344, increments pointer to 4.
The 5th transition of 10001 copies 30001 to 40345, increments pointer to 5, and, since the pointer value = table length, the middle output pases power, energizing coil 00135. No R → T operations are possible while the pointer value = the table length.

If, after the 2nd transition of 10001, 10002 were energized, this would prevent the pointer value from changing, and all subsequent transitions of 10001 would cause the value in 30001 to be copied to 40343.

The pointer register will be reset to zero when 10003 is energized.

TABLE TO REGISTER MOVE

This function copies the bit pattern of any register or 16 discretes located within a table to a specific holding register.



Function Block

The top node is the data source and can be one of the following:

- the first 0XXXX reference number in a table of output references
- the first 1XXXX reference number in a table of input references
- the first 30XXX reference number in a table of input registers
- the first 4XXXX reference number in a table of holding registers

The middle node must be a holding register (4XXXX). This register is the pointer to the source table specified in the top node. The next consecutive register (4XXXX +1) is the destination register.

The bottom node contains the symbol T → R and a number specifying table length. Table length may range from:

- 1 to 255 in 16 bit controllers
- 1 to 999 in 24 bit controllers

Inputs

Top: Control in. Every scan this node is powered, the T → R move is performed, and the pointer is incremented until the pointer value = the table length. A transitional contact should be used if single operations are desired.

Middle: When powered, prevents pointer from increasing

Bottom: When powered, resets pointer to zero

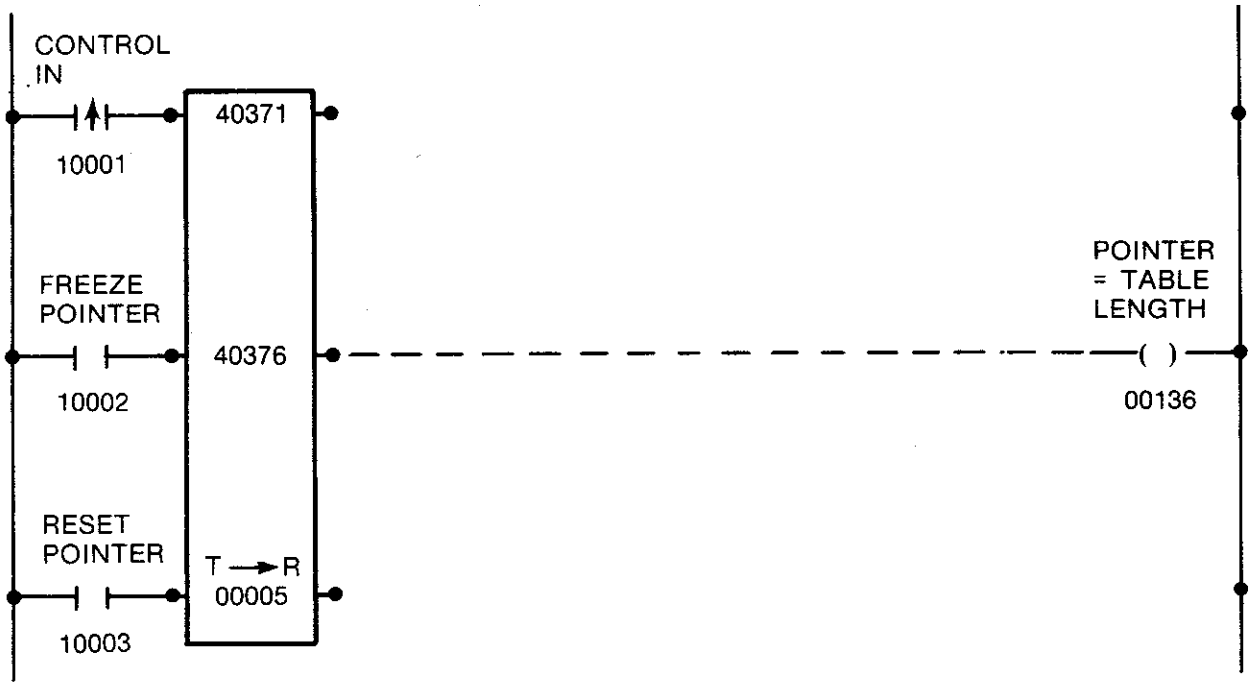
Outputs

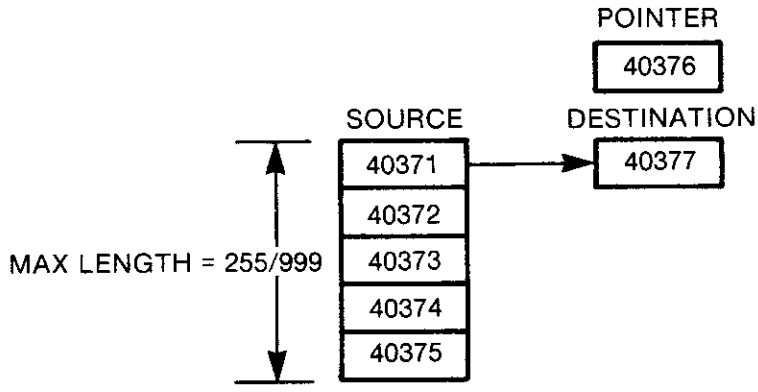
Top: Passes power when top input is powered

Middle: Passes power when the pointer = table length (end of table). No T→ R operations are possible while the pointer value = the table length.

TABLE TO REGISTER MOVE

Network # 46





Network Description

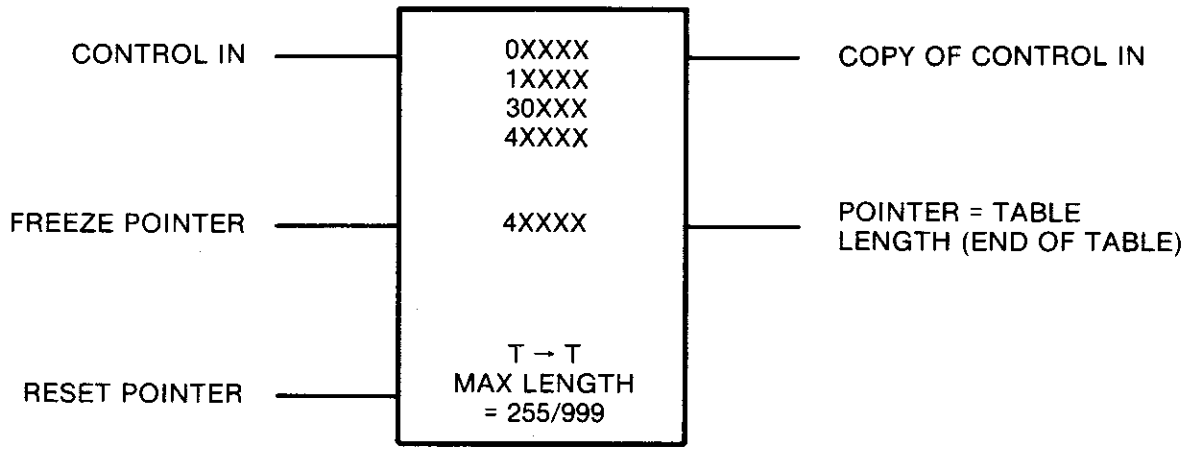
The 1st transition of 10001 copies 40371 to 40377, increments pointer to 1.
The 2nd transition of 10001 copies 40372 to 40377, increments pointer to 2.
The 3rd transition of 10001 copies 40373 to 40377, increments pointer to 3.
The 4th transition of 10001 copies 40374 to 40377, increments pointer to 4.
The 5th transition of 10001 copies 40375 to 40377, increments pointer to 5, and, since the pointer value = table length, the middle output passes power, energizing coil 00136. No T → R operations are possible while the pointer value = the table length.

If, after the 2nd transition of 10001, 10002 were energized, this would prevent the pointer value from changing, and all subsequent transitions of 10001 would cause the value in 40343 to be copied to 40377.

The pointer register will be reset to zero when 10003 is energized.

TABLE TO TABLE MOVE

This function copies the bit pattern of any register or 16 discretes from a position within one table to the same position in a second table of holding registers.



Function Block

The top node is the data source and can be one of the following:

- the first 0XXXX reference number in a table of output references
- the first 1XXXX reference number in a table of input references
- the first 30XXX reference number in a table of input registers
- the first 4XXXX reference number in a table of holding registers

The middle node must be a holding register (4XXXX). This register is the pointer to both the source and destination tables. The next consecutive register (4XXXX +1) is the first register in the destination table.

The bottom node contains the symbol T → T and a number specifying table length. Table length may range from:

1 to 255 in 16 bit controllers
1 to 999 in 24 bit controllers

Inputs

Top: Control in. Every scan this node is powered, the T → T move is performed, and the pointer is incremented until the pointer value = the table length. A transitional contact should be used if single operations are desired.

Middle: When powered, prevents pointer from increasing

Bottom: When powered, resets pointer to zero

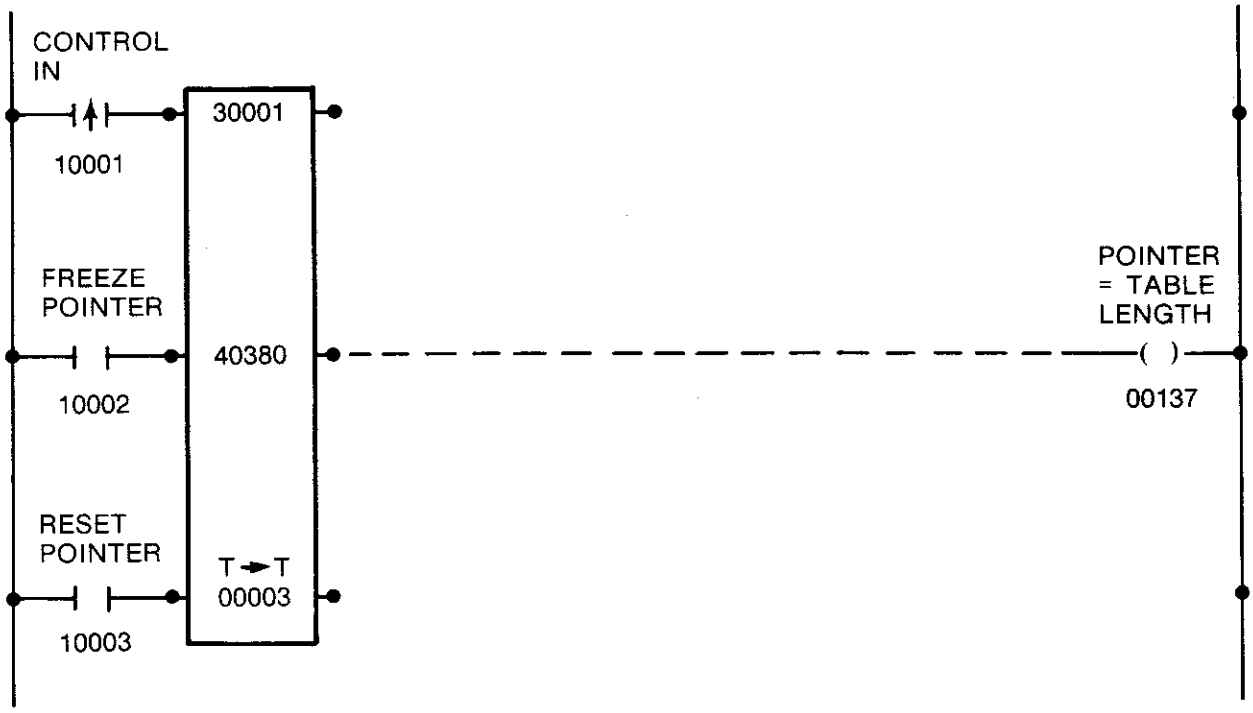
Outputs

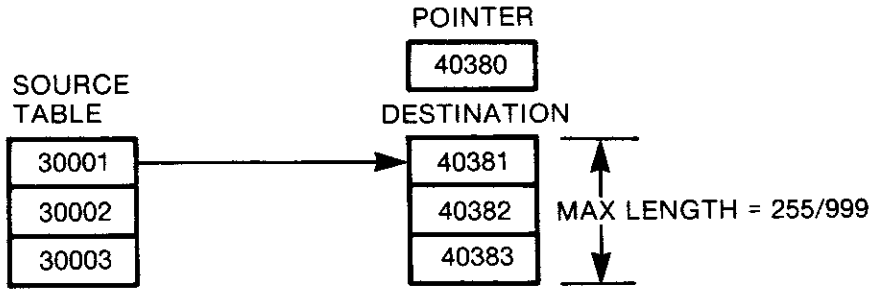
Top: Passes power when top input is powered

Middle: Passes power when the pointer = table length (end of table). No T→ T operations are possible while the pointer value = the table length.

TABLE TO TABLE MOVE

Network # 47





Network Description

The 1st transition of 10001 copies 30001 to 40381, increments pointer to 1.

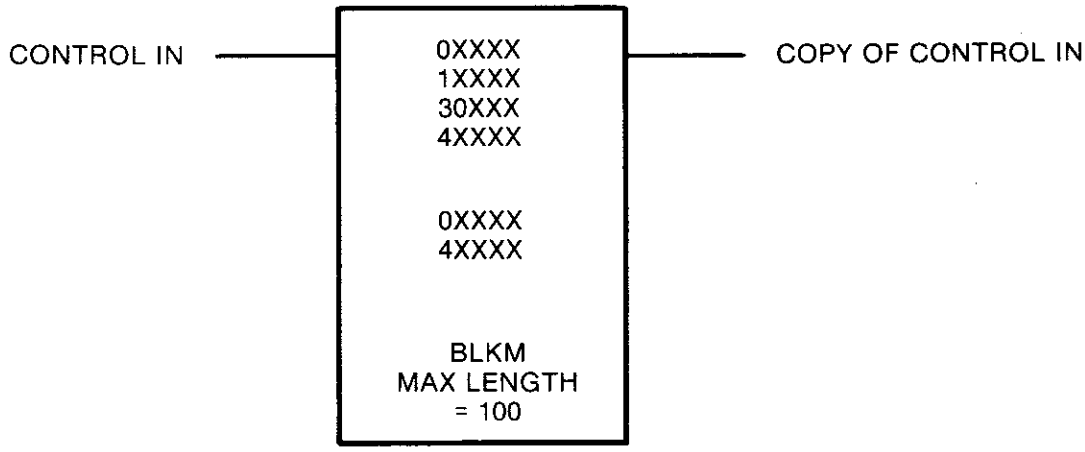
The 2nd transition of 10001 copies 30002 to 40382, increments pointer to 2.

The 3rd transition of 10001 copies 30003 to 40383, increments pointer to 3, and, since the pointer value = table length, the middle output passes power, energizing coil 00137. No T → T operations are possible while the pointer value = the table length.

If, after the 2nd transition of 10001, 10002 were energized, this would prevent the pointer value from changing, and all subsequent transitions of 10001 would cause the value in 30003 to be copied to 40383.

BLOCK MOVE

This function copies, during one scan, the entire contents of any table to a table of outputs or holding registers.



Function Block

The top node is the source table and can be one of the following:

- the first 0XXXX reference number in a table of output references
- the first 1XXXX reference number in a table of input references
- the first 30XXX reference number in a table of input registers
- the first 4XXXX reference number in a table of holding registers

The middle node is the destination table and can be one of the following:

- the first 0XXXX reference number in a table of output registers or
- the first 4XXXX reference number in a table of holding registers.

The bottom node contains the symbol BLKM and a number specifying table length. Table length may range from 1 to 100.

Inputs

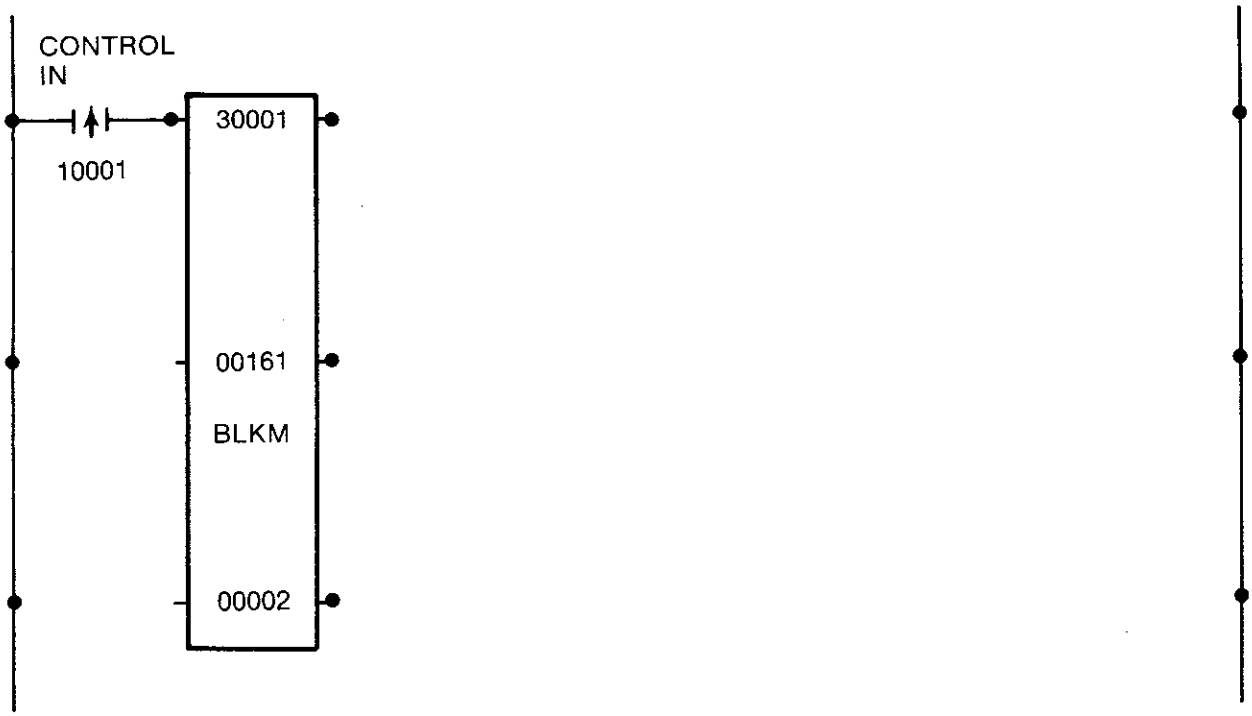
Top: Control in. Every scan this node is powered, the block move is performed. A transitional contact should be used if single operations are desired.

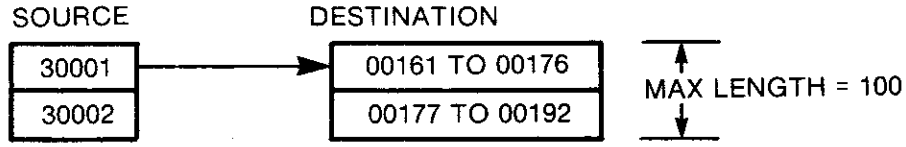
Outputs

Top: Passes power when top input is powered

BLOCK MOVE

Network # 48





Network Description

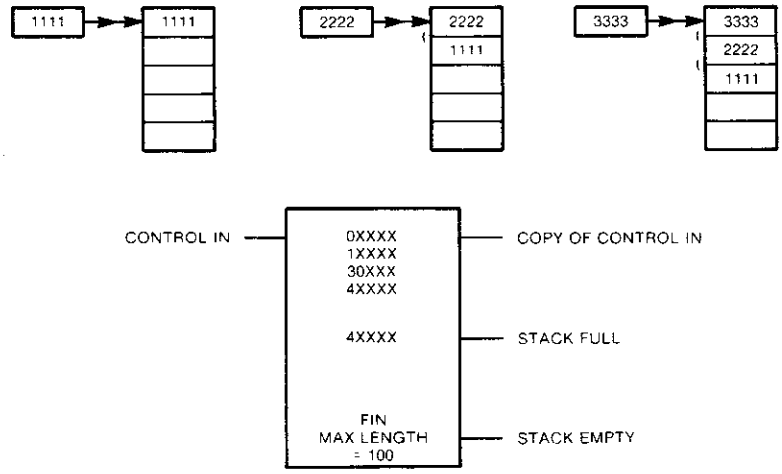
When 10001 is energized, the contents of the source table will be copied into the destination table.

In the case shown above, the destination is a table of outputs. These coils, like all coils, may only be used once. Attempts, for example, to program coil 165 in another network will result in a P190 error message: "coil used". A "search" for coil 165 via the P190 will result in the display of the network the block move is programmed within.

WARNING: The destination for this function can be a table of outputs (0XXXX). This function will override any disabled coils within this table without enabling them.

FIRST IN

This function copies the bit pattern of any register or 16 discretes to the first register in a table of holding registers. This register is the start of a queue or stack. Each entry to the stack causes the previous entry to move to the next position in the stack. This block typically is used with the first out block. Together they form a fifo (first in, first out) stack which is analogous to a paper cup dispenser - cups are removed in the same order they were inserted. Three successive first in operations are illustrated below.



Function Block

The top node is the data source and can be one of the following:

- the first of 16 logic coils (0XXXX)
- the first of 16 inputs (1XXXX)
- an input register (3XXXX)
- a holding register (4XXXX)

The middle node must be a holding register (4XXXX). This register is the pointer to the destination table which starts with the next consecutive register (4XXXX +1). The value contained in the pointer register = the current number of entries in the table or stack.

The bottom node contains the symbol FIN and a number specifying table length, which should = FOUT block. Table length may range from 1 to 100.

Inputs

Top: Control in. Every scan this node is powered, the first in move is performed, and the pointer is incremented until the pointer value = the table length. A transitional contact should be used if single operations are desired.

Outputs

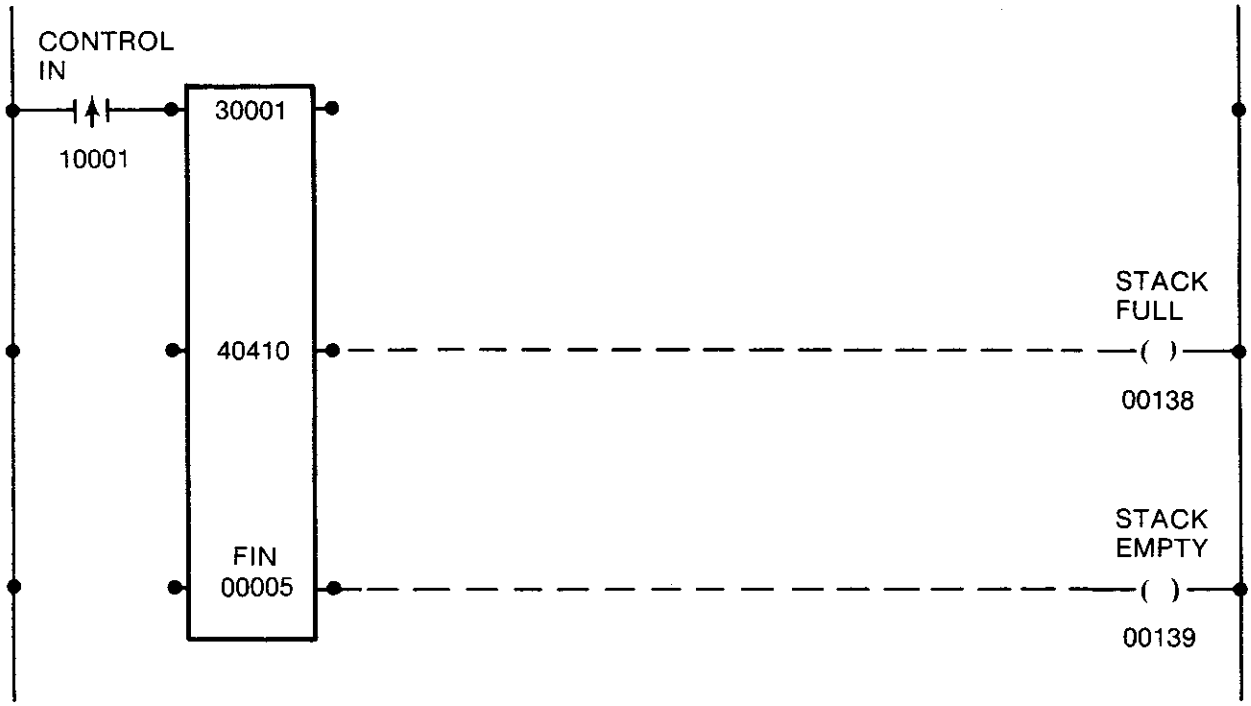
Top: Passes power when top input is powered

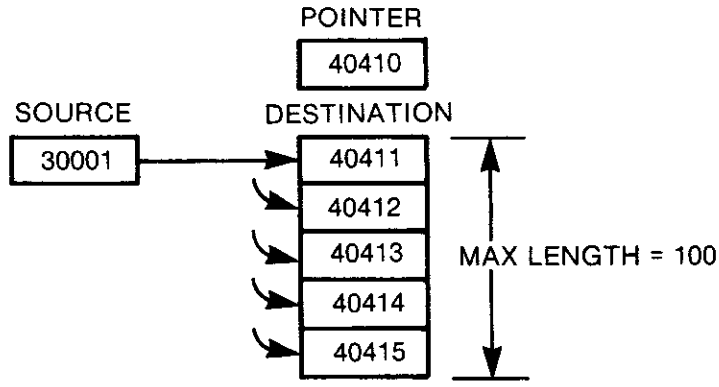
Middle: Passes power when the stack is full. No additional entries to the stack are possible while this output is on.

Bottom: Passes power when the stack is empty.

FIRST IN

Network # 49





Network Description

The 1st transition of 10001 copies 30001 to 40411, increments pointer to 1, de-energizes 00139 (stack not empty)

The 2nd transition of 10001 moves contents of 40411 to 40412, copies 30001 to 40411, increments pointer to 2.

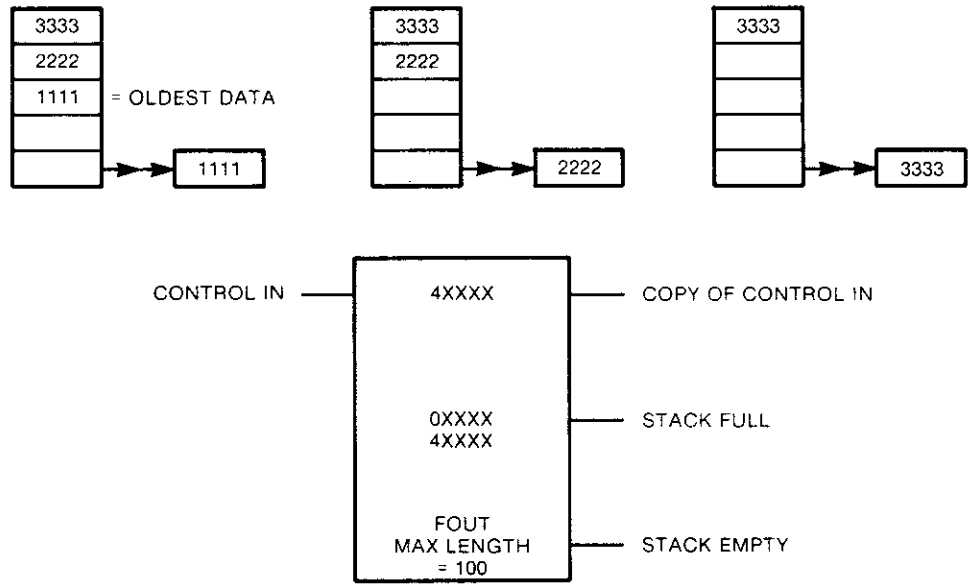
The 3rd transition of 10001 moves contents of 40412 to 40413, moves contents of 40411 to 40412, copies 30001 to 40411, increments pointer to 3.

The 4th transition of 10001 moves contents of 40413 to 40414, moves contents of 40412 to 40413, moves contents of 40411 to 40412, copies 30001 to 40411, increments pointer to 4.

The 5th transition of 10001 moves contents of 40414 to 40415, moves contents of 40413 to 40414, moves contents of 40412 to 40413, moves contents of 40411 to 40412, copies 30001 to 40411, increments pointer to 5, and energizes 00138 (stack full – no more entries possible while pointer = table length).

FIRST OUT

This function copies the bit pattern of a register within a table of holding registers. The register from which the data will be copied is the register with the oldest data (the highest register number in the table with data in it). This block typically is used with the first in block. Together, they form a fifo (first in, first out) stack which is analogous to a paper cup dispenser - cups are removed in the same order they were inserted. Three successive first out operations are illustrated below.



Function Block

The top node must be a holding register (4XXXX). This register is the pointer to the source table which starts with the next consecutive register (4XXXX +1). The number contained in the pointer register indicates which table position the data will be removed from (the current number of entries in the stack).

The middle node is the data destination and can be one of the following:

- the first of 16 logic coils (0XXXX)
- a holding register (4XXXX)

The bottom node contains the symbol FOUT and a number specifying table length. Table length may range from 1 to 100.

Inputs

Top: Control in. Every scan this node is powered, the first out move is performed, and the pointer is decremented until the pointer value = the table length.

Outputs

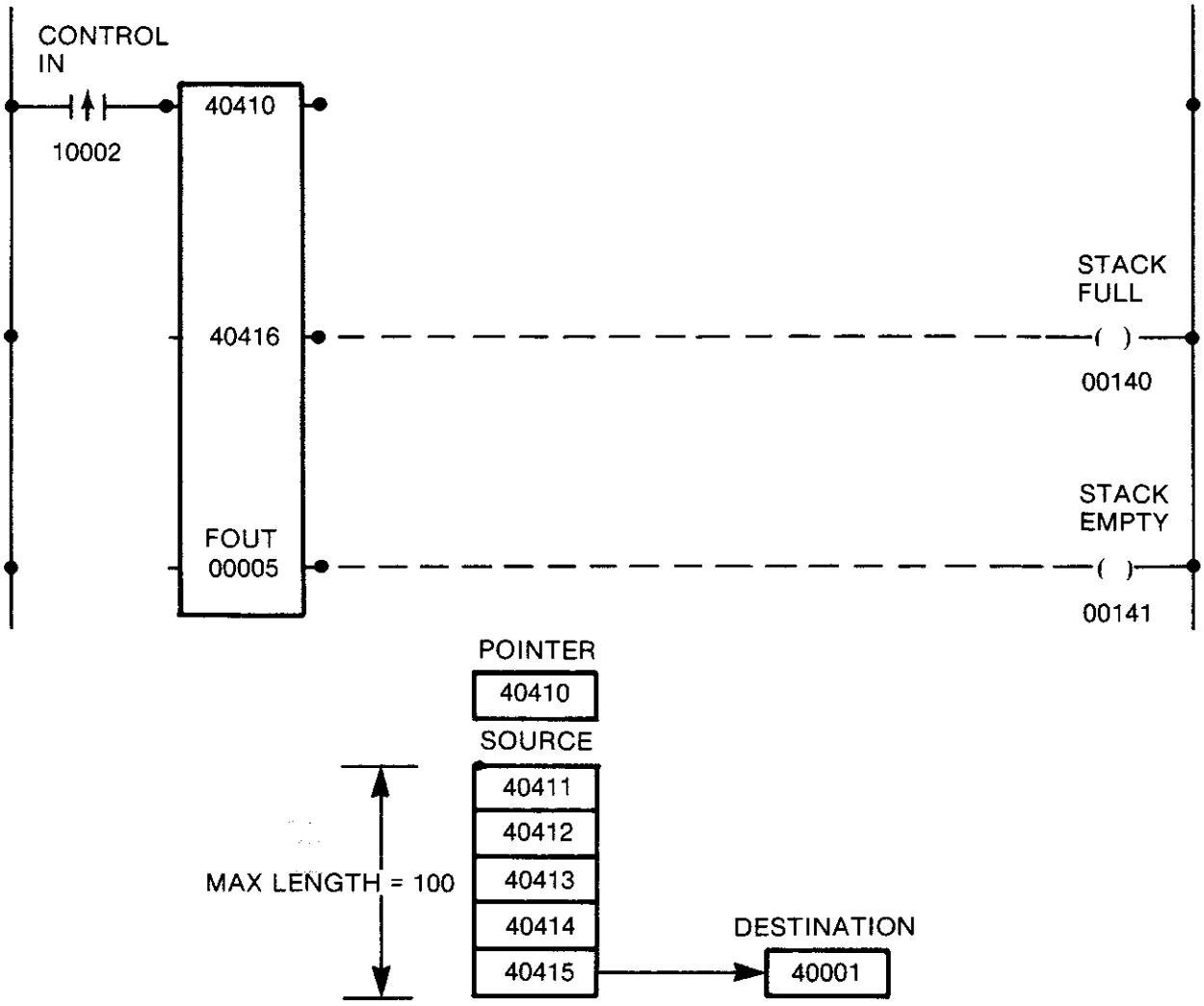
Top: Passes power when top input is powered

Middle: Passes power when the stack is full. No additional entries to the stack are allowed while this output is on.

Bottom: Passes power when the stack is empty.

FIRST OUT

Network # 50



Network Description

If stack is full, pointer value = 5, and coil 00140 is energized, then:

The 1st transition of 10002 copies 40415 to 40001, decrements pointer to 4, de-energizes coil 00140 (stack not full).

The 2nd transition of 10002 copies 40414 to 40001, decrements pointer to 3.

The 3rd transition of 10002 copies 40413 to 40001, decrements pointer to 2.

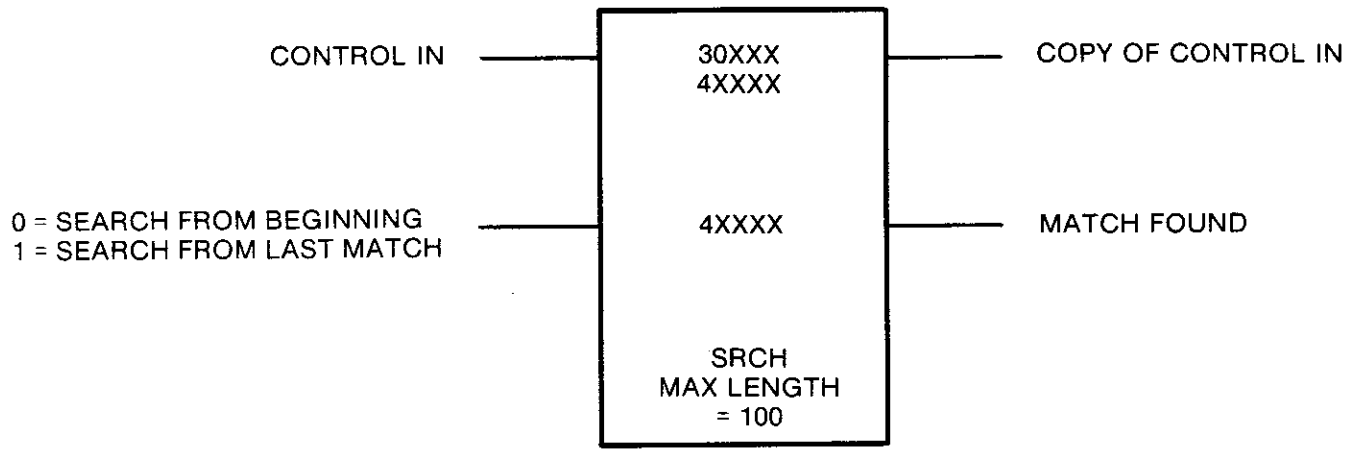
The 4th transition of 10002 copies 40012 to 40001, decrements pointer to 1

The 5th transition of 10002 copies 40011 to 40001, decrements pointer to 0, energizes 00141 (stack is empty).

WARNING: The destination for this function can be a group of 16 outputs. This function will override any disabled coils within this group without enabling them.

SEARCH

This function searches a table of registers for a specific bit pattern.



Function Block

The top node is the table to be searched and can be one of the following:

- the first 30XXX reference number in a table of input references
- the first 4XXXX reference number in a table of holding registers.

The middle node must be a holding register (4XXXX). This register is the pointer to the table specified in the top node. The next consecutive register 4XXXX +1 contains the value or bit pattern to be searched for.

The bottom node contains the symbol SRCH and a number specifying table length. Table length may range from 1 to 100.

Inputs

Top: Control in. Every scan this node is powered, a search is performed. A transitional contact should be used if single operations are desired.

Middle: When powered, table will be searched from the last match when not powered, table will be searched from the beginning

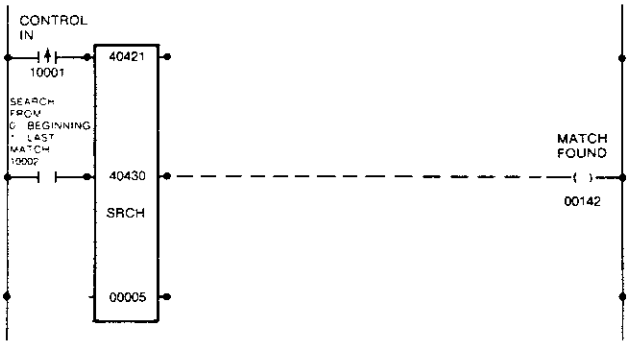
Outputs

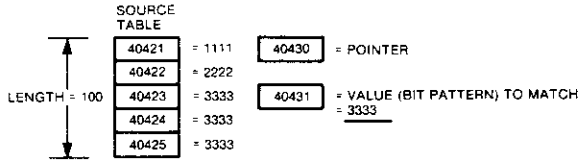
Top: Passes power when top input is powered

Middle: Passes power when a match is found

SEARCH

Network # 51





Network Description

If 10002 is not energized, every scan that the "control in" node receives power, the source table will be searched for a "3333". The search will:

find a match in register 40423
stop searching for the remainder of scan
set the pointer value to 3 for 1 scan, indicating that a match exists in table position 3.
energize coil 00142 for one scan

If 10002 is energized and 10001 transitions from off to on, the source table will be searched for a "3333".

The search will:

find a match in register 40423
stop the search
set the pointer value to 3 indicating that a match exists in table position 3.
energize coil 00142 for 1 scan

The 2nd transition of 10001 will:

find a match in register 40424
stop the search
set the pointer value to 4
energize coil 00142 for 1 scan

The 3rd transition of 10001 will:

find a match in register 40425
stop the search
for 1 scan, set the pointer to 5 for 1 scan, then 0
energize coil 00142 for 1 scan