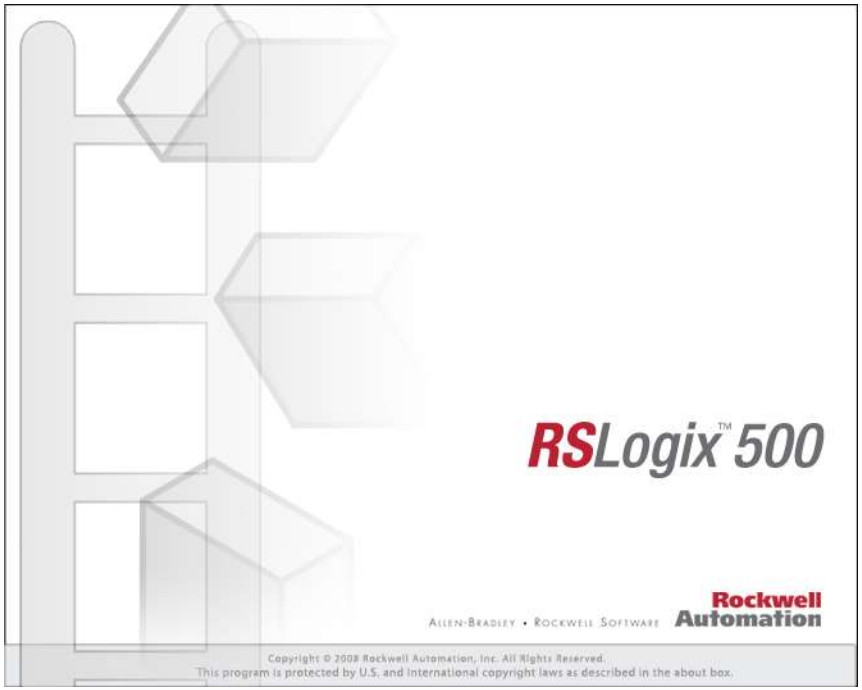
RSLogix Micro Project Report



Processor Information

Processor Type: Bul.1763 MicroLogix 1100 Series A

Processor Name: UNTITLED

Total Memory Used: 616 Instruction Words Used - 204 Data Table Words Used

Total Memory Left: 6040 Instruction Words Left

Program Files: 5

Data Files: 12

Program ID: 2474

I/O Configuration

0	Bul.1763	MicroLogix 1100 Series A
1		
2		
3		
4		

Channel Configuration

CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex Edit Resource/Owner Timeout: 60
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex Passthru Link ID: 1
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex Write Protected: No
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex Comms Servicing Selection: Yes
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex Message Servicing Selection: Yes
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex 1st AWA Append Character: \d
CHANNEL 0 (SYSTEM) - Driver: DF1 Full Duplex 2nd AWA Append Character: \a

Source ID: 1 (decimal)
Baud: 19200
Parity: NONE
Control Line : No Handshaking
Error Detection: CRC
Embedded Responses: Auto Detect
Duplicate Packet Detect: Yes
ACK Timeout(x20 ms): 50
NAK Retries: 3
ENQ Retries: 3

CHANNEL 1 (SYSTEM) - Driver: Ethernet
CHANNEL 1 (SYSTEM) - Driver: Ethernet Edit Resource/Owner Timeout: 60
CHANNEL 1 (SYSTEM) - Driver: Ethernet Passthru Link ID: 1
CHANNEL 1 (SYSTEM) - Driver: Ethernet Write Protected: No
CHANNEL 1 (SYSTEM) - Driver: Ethernet Comms Servicing Selection: Yes
CHANNEL 1 (SYSTEM) - Driver: Ethernet Message Servicing Selection: Yes

Hardware Address: 00:0F:73:01:72:04
IP Address: 192.168.1.112
Subnet Mask: 255.255.255.0
Gateway Address: 192.168.1.1
Msg Connection Timeout (x 1mS): 15000
Msg Reply Timeout (x mS): 3000
Inactivity Timeout (x Min): 30
Bootp Enable: No
Dhcp Enable No
SNMP Enable: No
HTTP Enable: Yes
Auto Negotiate Enable: Yes
Port Speed Enable: 10/100 Mbps Full Duplex/Half Duplex
Contact:
Location:

Program File List

Name	Number	Type	Rungs	Debug	Bytes
[SYSTEM]	0	SYS	0	No	0
	1	SYS	0	No	0
CONTINUOUS	2	LADDER	7	No	150
LINEAR	3	LADDER	9	No	362
BINARY	4	LADDER	9	No	602

Data File List

Name	Number	Type	Scope	Debug	Words	Elements	Last
OUTPUT	0	O	Global	No	12	4	O:3
INPUT	1	I	Global	No	18	6	I:5
STATUS	2	S	Global	No	0	66	S:65
BINARY	3	B	Global	No	1	1	B3:0
TIMER	4	T	Global	No	3	1	T4:0
COUNTER	5	C	Global	No	6	2	C5:1
CONTROL	6	R	Global	No	3	1	R6:0
INTEGER	7	N	Global	No	7	7	N7:6
FLOAT	8	F	Global	No	2	1	F8:0
BNRYSEARCH	253	N	Global	No	72	72	N253:71
LINRSEARCH	254	N	Global	No	72	72	N254:71
SORTEDLIST	255	N	Global	No	8	8	N255:7

Cf. https://reprog.wordpress.com/2010/04/19/are-you-one-of-the-10-percent/

Am I one of the 10% of programmers who can write a binary search?

Blind attempt at doing it in ladder logic on a Programmable Logic Controller (PLC)
- The PLC program is written in ladder logic, which is a digital form, and extension, of relay logic
- PLCs are roughly analogous to GAWK reading from a never-ending input stream.
  - Each pass, analogous to processing one GAWK input line, is called a scan

The sorted array has seven elements: 1; 2; 2; 3; 5; 6; 6; 6; the lengths tested run be from 0 to 8.
The range of sought values will be from 0 to 7.

This is the main program; on each PLC scan, this program will
- Select one problem from the set of 72 possibilities (eight lengths and seven sought values),
- Call the linear and binary searches
- Compare the results

0000

TEST_CASE_INDEX

LIM
Limit Test
Low Lim 72
72<
Test N7:2
0<
High Lim -1
-1<

TEST_CASE_INDEX

MOV
Move
Source 0
0<
Dest N7:2
0<

Convert the low three bits of TEST_CASE_INDEX to the sought value (T) in the range [0:7]

0001

SOUGHT_VALUE

AND
Bitwise AND
Source A N7:2
0000h<
Source B 7
7<
Dest N7:0
0000h<

Converts the next higher four bits of TEST_CASE_INDEX to a length, in the range [0:8], of the array in which T is sought

0002

ARRAY_LENGTH

SUB

Subtract

Source A N7:2 0<

Source B N7:0 0<

Dest N7:1 0<

ARRAY_LENGTH

DIV

Divide

Source A N7:1 0<

Source B 8 8<

Dest N7:1 0<

Call the linear search

0003

LINEAR_SEARCH

JSR

Jump To Subroutine

SBR File Number U:3

Optionally
- Call the binary search
- Check the result of the binary search against that of the linear search

0004

ENABLE_BINARY_SEARCH

B3:0

0

BINARY_SEARCH

JSR

Jump To Subroutine

SBR File Number U:4

Increment the test case to prepare for the next scan

0005

TEST_CASE_INDEX

ADD

Add	
Source A	N7:2 0<
Source B	1 1<
Dest	N7:2 0<

0006

END

Perform a linear search of a sought value in a sorted array of values

Inputs
- Sought value, T (0-7)
- Length of array (0-8)
- The sorted array, containing eight values: 1; 2; 2; 3; 5; 6; 6; 6.

Perform a linear search of a sought value in an array of values
Inputs
- Sought value, T (0-7)
- Length of array (0-8)
- A sorted array, containing up to eight values: 1; 2; 2; 3; 5; 6; 6; 6.

If the length of the array is less than 1, then the result is -98; don't perform a search

0000

ARRAY_LENGTH

LES
Less Than (A<B)
Source A N7:1
0<
Source B 1
1<

LINEAR_RESULT

MOV
Move
Source -98
-98<
Dest N254:[N7:2]
0<

RET
Return

If the sought value is less than the first value in the array, then the result is -99
- Do not perform a search

0001

SOUGHT_VALUE

LES
Less Than (A<B)
Source A N7:0
0<
Source B N255:0
1<

LINEAR_RESULT

MOV
Move
Source -99
-99<
Dest N254:[N7:2]
0<

RET
Return

Initialize high and low indices to zero

0002

#INDEX_LO

FLL
Fill File
Source 0
Dest #N7:4
Length 2

Loop over elements in array while the following conditions remains true:
- High index (offset from first element of array) is less than length of array, i.e. high index points to a value in the array,
AND
- Value at high index is less than or equal to sought value

On each pass through loop,
- MOVe value of high index to low index, and
- Increment high index

0003

Q3:0
LBL

INDEX_HI
LES
Less Than (A<B)
Source A N7:5 0<
Source B N7:1 0<

VALUE_HI
LEQ
Less Than or Eql (A<=B)
Source A N255:[N7:5] 1<
Source B N7:0 0<

INDEX_LO
MOV
Move
Source N7:5 0<
Dest N7:4 0<

INDEX_HI
ADD
Add
Source A N7:5 0<
Source B 1 1<
Dest N7:5 0<

Q3:0
JMP

If the sought value was found in the array before the length, then the result is the 0-based index of the last element with that value

0004

VALUE_LO

EQU
Equal
Source A N255:[N7:4]
1<
Source B N7:0
0<

LINEAR_RESULT

MOV
Move
Source N7:4
0<
Dest N254:[N7:2]
0<

RET
Return

If the sought value is greater than the last value in the array, then the result is -100 minus the length of the array.

0005

INDEX_HI

EQU
Equal
Source A N7:5
0<
Source B N7:1
0<

LINEAR_RESULT

SUB
Subtract
Source A -100
-100<
Source B N7:5
0<
Dest N254:[N7:2]
0<

RET
Return

If the sought value is greater than the value of the element offset by the low index value in the array, but not in the array,
then the result is -100 minus the offset the the last value in the array with a value less than the sought value

0006

VALUE_LO

LES
Less Than (A<B)
Source A N255:[N7:4]
1<
Source B N7:0
0<

LINEAR_RESULT

SUB
Subtract
Source A -100
-100<
Source B N7:4
0<
Dest N254:[N7:2]
0<

RET
Return

All cases should be handled by one of the above result-and-return rungs.
If not, return -9999 to indicate an error in logic

0007

LINEAR_RESULT

MOV
Move
Source -9999
-9999<
Dest N254:[N7:2]
0<

0008

END

Perform a binary search of a sought value in a sorted array of values

Inputs
- Sought value, T (0-7)
- Length of array (0-8)
- The sorted array, containing eight values: 1; 2; 2; 3; 5; 6; 6; 6.

Initialize the low and high indices values to 0 and the length of the array, respectively

0000

INDEX_LO

MOV

Move
Source 0
0<
Dest N7:4
0<

INDEX_HI

MOV

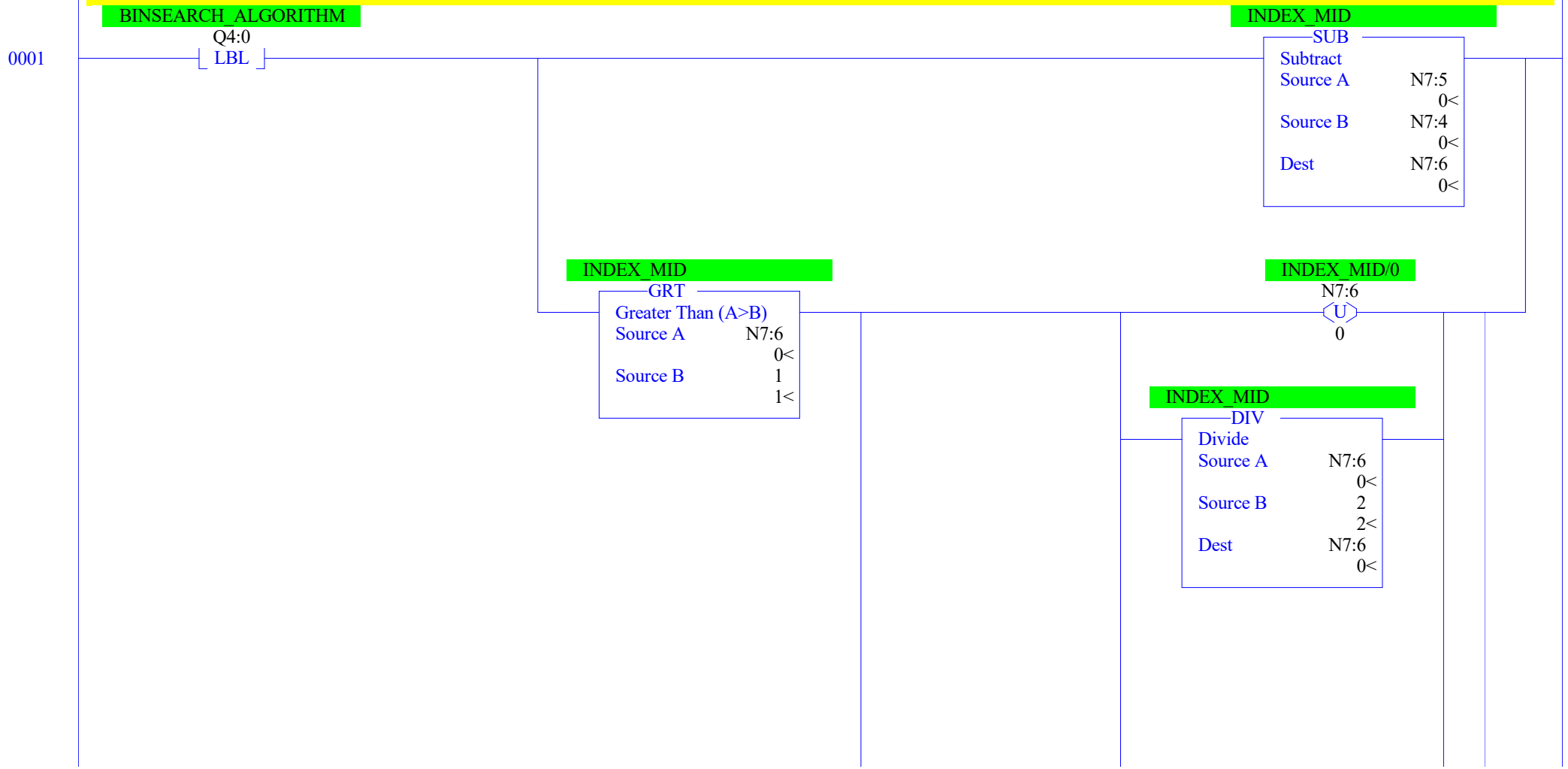
Move
Source N7:1
0<
Dest N7:5
0<

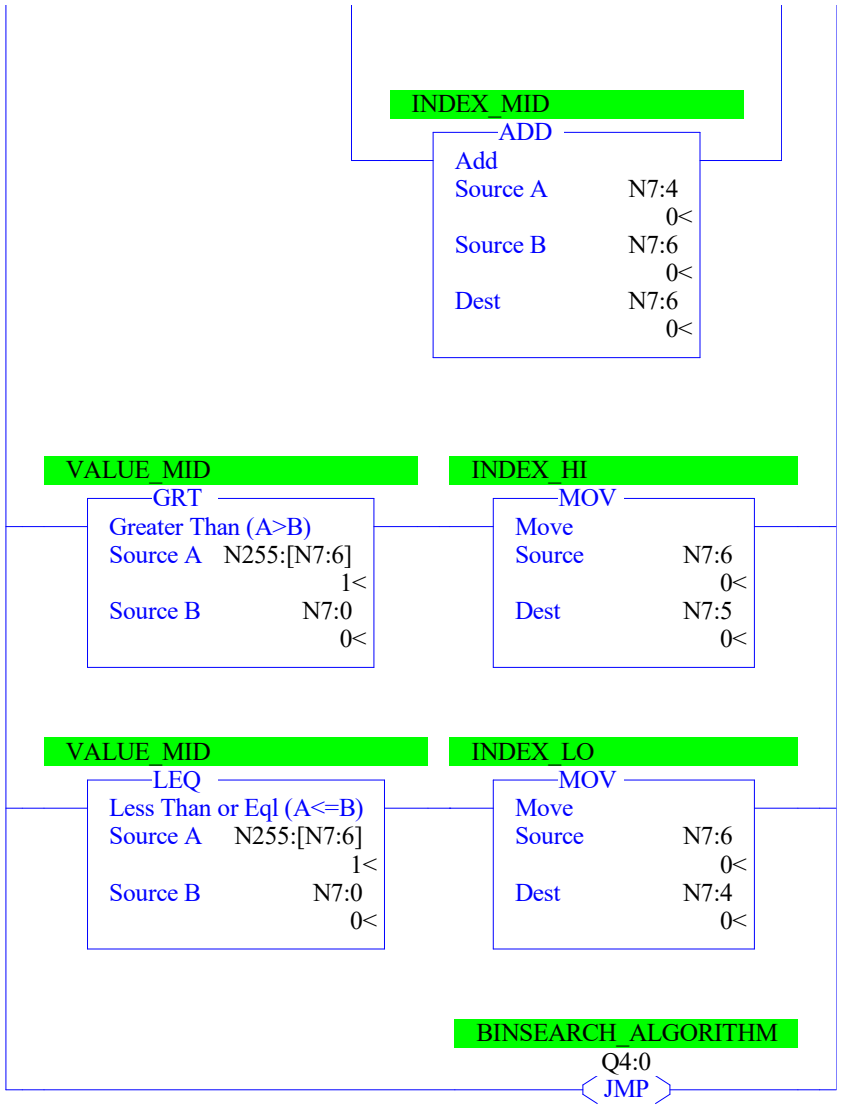
Loop while the following condition remains true:
- High index value is at least two greater than low index value

Maintain the following invariants:
- Array value at low index is less than or equal to sought value
AND
- EITHER high index is array length OR array value at high index is greater than or equal to sought value

N.B. Those invariants, if initially true, will eventually assign the low index to the offset of the last value in the array that is less than or equal to the sought value
N.B. Those invariants will be violated when the sought value is less than value of the first element in the array, however
in that case at the end the low and high index values will be 0 and 1, respectively.

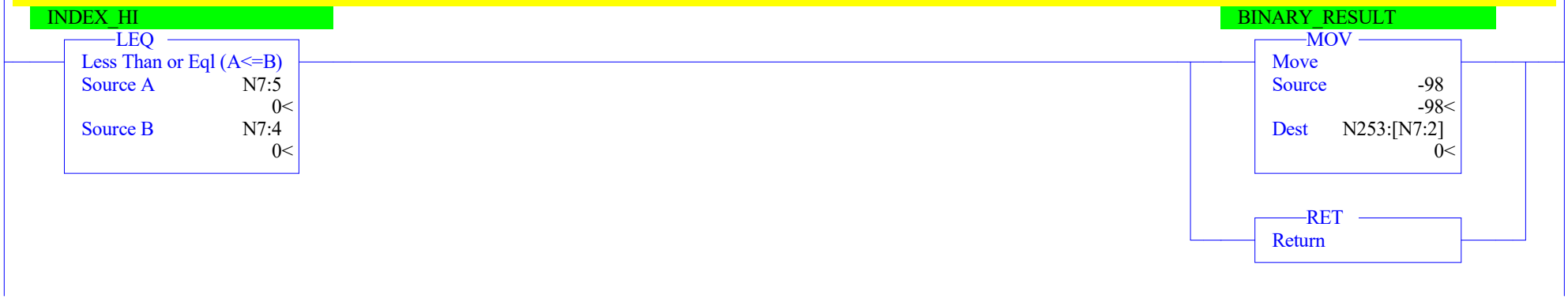
On each pass through loop,
- Calculate mid index value
- Shift either low or high index to mid index value to maintain the invariants above




INDEX_MID
ADD
Add
Source A N7:4
0<
Source B N7:6
0<
Dest N7:6
0<
VALUE_MID
GRT
Greater Than (A>B)
Source A N255:[N7:6]
1<
Source B N7:0
0<
INDEX_HI
MOV
Move
Source N7:6
0<
Dest N7:5
0<
VALUE_MID
LEQ
Less Than or Eql (A<=B)
Source A N255:[N7:6]
1<
Source B N7:0
0<
INDEX_LO
MOV
Move
Source N7:6
0<
Dest N7:4
0<
BINSEARCH_ALGORITHM
Q4:0
JMP

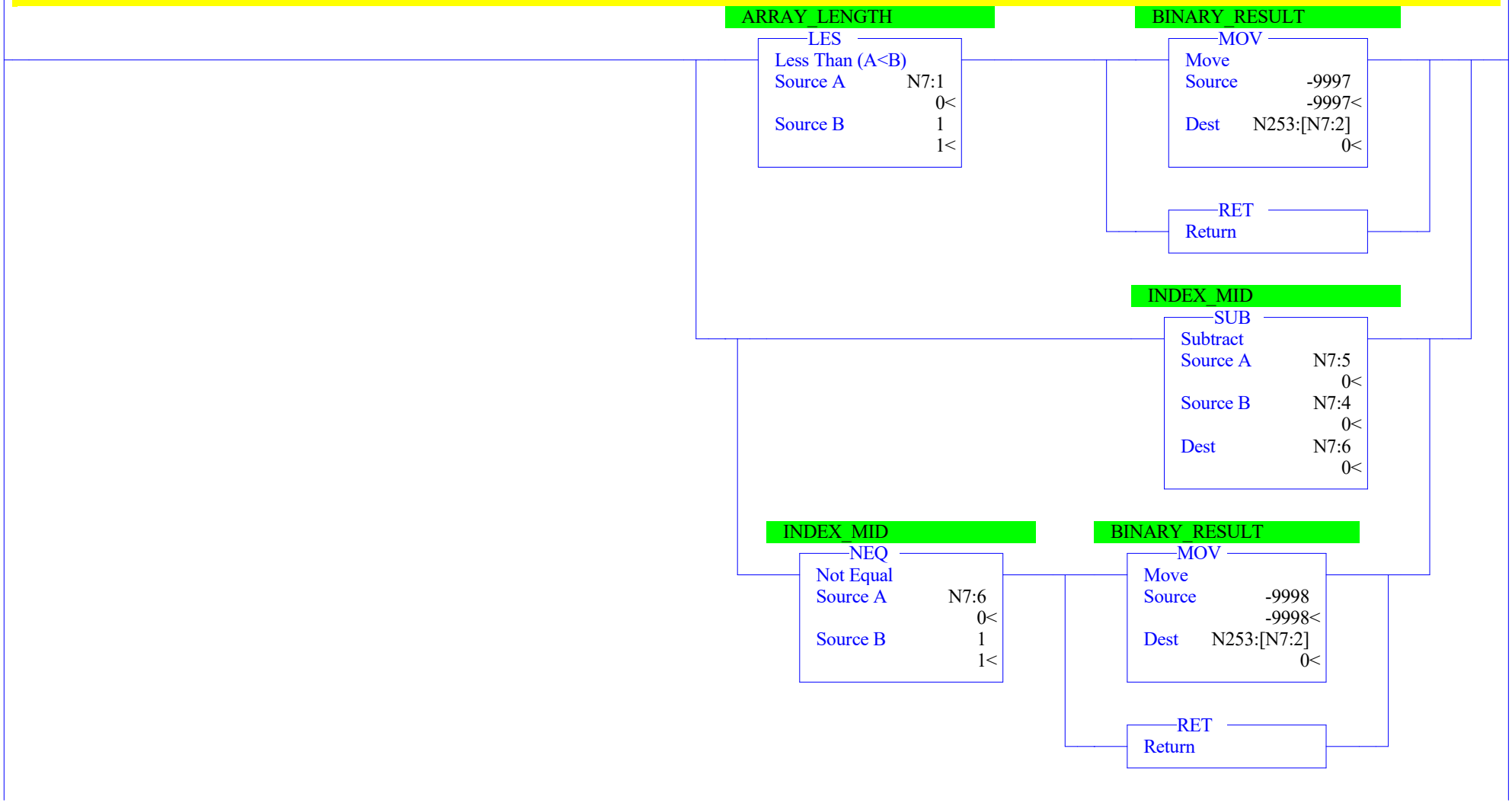
If high index value is less than or equal to low index values are equal, then array length must be zero or less: assign -98 as result and return

0002



Validate logic expressed in comment of previous rung:
- If array length is non-positive, assign -9997 as result to indicate error in logic and return
- If difference from low to high index is not +1, assign -9998 as result to indicate error in logic and return

0003



To here, array length must be positive and difference between low and high indices must be 1
- If value at low index is equal to sought value, then assign low index as result and return

0004

VALUE_LO

EQU
Equal
Source A N255:[N7:4] 1<
Source B N7:0 0<

BINARY_RESULT

MOV
Move
Source N7:4 0<
Dest N253:[N7:2] 0<

RET
Return

To here, sought value cannot equal any value in array
- If low index value is 0 and sought value is less than first element in array, then assign -99 as result and return
- If high index value is equal to array length, then sought value is greater than any value in array, so assign -100 minus array length as result and return

0005

INDEX_LO

EQU
Equal
Source A N7:4 0<
Source B 0 0<

SOUGHT_VALUE

LES
Less Than (A<B)
Source A N7:0 0<
Source B N255:[N7:4] 1<

BINARY_RESULT

MOV
Move
Source -99 -99<
Dest N253:[N7:2] 0<

RET
Return

INDEX_HI

EQU
Equal
Source A N7:5 0<
Source B N7:1 0<

BINARY_RESULT

SUB
Subtract
Source A -100 -100<
Source B N7:5 0<
Dest N253:[N7:2] 0<

RET
Return

If the sought value is greater than the first value in the array, but not in the array,
then the result is -100 minus the offset the the last value in the array with a value less than the sought value

0006

VALUE_LO

LES
Less Than (A<B)
Source A N255:[N7:4]
1<
Source B N7:0
0<

BINARY_RESULT

SUB
Subtract
Source A -100
-100<
Source B N7:4
0<
Dest N253:[N7:2]
0<

RET
Return

All cases should be handled by one of the above result-and-return rungs.
If not, return -9999 to indicate an error in logic

0007

BINARY_RESULT

MOV
Move
Source -9999
-9999<
Dest N253:[N7:2]
0<

0008

END

Data File O0 (bin) -- OUTPUT

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
O:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
O:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
O:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
O:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A

Data File I1 (bin) -- INPUT

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
I:0.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
I:0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
I:0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
I:0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bul.1763	MicroLogix 1100 Series A
I:0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	Bul.1763	MicroLogix 1100 Series A-Analog Inp 0
I:0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	Bul.1763	MicroLogix 1100 Series A-Analog Inp 1

Data File S2 (hex) -- STATUS

Main

Processor Mode S:1/0 - S:1/4 = Remote Run
On Power up Go To Run (Mode Behavior) S:1/12 = 0
First Pass S:1/15 = No
Free Running Clock S:4 = 1101-0100-0011-0111

Proc

OS Catalog Number S:57 = 1100
OS Series S:58 = B
OS FRS S:59 =
Processor Catalog Number S:60 =
Processor Series S:61 = A
Processor FRN S:62 =
User Program Type S:63 = 8001h
Compiler Revision Number S:64 =

Scan Times

Maximum (x10 ms) S:22 = 25
Watchdog (x10 ms) S:3 (high byte) = 10
Last 100 uSec Scan Time S:35 = 6
Scan Toggle Bit S:33/9 = 0

Math

Math Overflow Selected S:2/14 = 1
Overflow Trap S:5/0 = 0
Carry S:0/0 = 0
Overflow S:0/1 = 0
Zero Bit S:0/2 = 0
Sign Bit S:0/3 = 1
Math Register (lo word) S:13 = 0
Math Register (high word) S:14-S:13 = 0
Math Register (32 Bit) S:14-S:13 = 0

Chan 0

Processor Mode S:1/0- S:1/4 = Remote Run
Node Address S:15 (low byte) = 0
Baud Rate S:15 (high byte) = ?
Channel Mode S:33/3 = 0
Comms Active S:33/4 = 0
Incoming Cmd Pending S:33/0 = 0
Msg Reply Pending S:33/1 = 0
Outgoing Msg Cmd Pending S:33/2 = 0

Debug

Suspend Code S:7 = 0
Suspend File S:8 = 0

Errors

Fault Override At Power Up S:1/8 = 0
Startup Protection Fault S:1/9 = 0
Major Error Halt S:1/13 = 0
Overflow Trap S:5/0 = 0
Control Register Error S:5/2 = 0
Major Error Executing User Fault Rtn. S:5/3 = 0
Battery Low S:5/11 = 0
Input Filter Selection Modified S:5/13 = 0
ASCII String Manipulation error S:5/15 = 0

Fault Routine S:29 = 0
Major Error S:6 = 0h
Error Description:

Protection

Deny Future Access S:1/14 = No
Data File Overwrite Protection Lost S:36/10 = True

Mem Module

Memory Module Loaded On Boot S:5/8 = 0
Password Mismatch S:5/9 = 0
Load Memory Module On Memory Error S:1/10 = 0
Load Memory Module Always S:1/11 = 0
On Power up Go To Run (Mode Behavior) S:1/12 = 0
Program Compare S:2/9 = 0
Data File Overwrite Protection Lost S:36/10 = 1

Forces

Forces Enabled S:1/5 = Yes
Forces Installed S:1/6 = No

Data File B3 (bin) -- BINARY

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	(Symbol) Description
B3:0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	

Data File T4 -- TIMER

Offset	EN	TT	DN	BASE	PRE	ACC	(Symbol)	Description
T4:0	1	1	0	.001 sec	1000	777		

Data File C5 -- COUNTER

Offset	CU	CD	DN	OV	UN	UA	PRE	ACC	(Symbol)	Description
C5:0	0	0	1	1	0	0	32767	-32359		
C5:1	0	0	0	0	0	0	32767	24453		

Data File R6 -- CONTROL

Offset	EN	EU	DN	EM	ER	UL	IN	FD	LEN	POS	(Symbol)	Description
R6:0	0	0	0	1	0	0	0	0	4	1		

Data File N7 (dec) -- INTEGER

Offset	0	1	2	3	4	5	6	7	8	9
N7:0	0	0	0	0	0	0	0			

Data File F8 -- FLOAT

Offset	0	1	2	3	4
F8:0	0				

Data File N253 (dec) -- BNRYSEARCH

Offset	0	1	2	3	4	5	6	7	8	9
N253:0	0	0	0	0	0	0	0	0	0	0
N253:10	0	0	0	0	0	0	0	0	0	0
N253:20	0	0	0	0	0	0	0	0	0	0
N253:30	0	0	0	0	0	0	0	0	0	0
N253:40	0	0	0	0	0	0	0	0	0	0
N253:50	0	0	0	0	0	0	0	0	0	0
N253:60	0	0	0	0	0	0	0	0	0	0
N253:70	0	0								

Data File N254 (dec) -- LINRSEARCH

Offset	0	1	2	3	4	5	6	7	8	9
N254:0	0	0	0	0	0	0	0	0	0	0
N254:10	0	0	0	0	0	0	0	0	0	0
N254:20	0	0	0	0	0	0	0	0	0	0
N254:30	0	0	0	0	0	0	0	0	0	0
N254:40	0	0	0	0	0	0	0	0	0	0
N254:50	0	0	0	0	0	0	0	0	0	0
N254:60	0	0	0	0	0	0	0	0	0	0
N254:70	0	0								

Data File N255 (dec) -- SORTEDLIST

Offset	0	1	2	3	4	5	6	7	8	9
N255:0	1	2	2	3	5	6	6	6		

Address/Symbol Database

Address	Symbol	Scope	Description	Sym Group	Dev. Code	ABV	BLW
B3:0/0	ENABLE_BINARY_SEARCH	Global					
N7:0	SOUGHT_VALUE	Global					
N7:1	ARRAY_LENGTH	Global					
N7:2	TEST_CASE_INDEX	Global					
N7:3	ERROR_COUNT	Global					
N7:4	INDEX_LO	Global					
N7:5	INDEX_HI	Global					
N7:6	INDEX_MID	Global					
N253:[N7:2]	BINARY_RESULT	Global					
N254:[N7:2]	LINEAR_RESULT	Global					
N255:[N7:4]	VALUE_LO	Global					
N255:[N7:5]	VALUE_HI	Global					
N255:[N7:6]	VALUE_MID	Global					
Q4:0	BINSEARCH_ALGORITHM	Global					
S:0			Arithmetic Flags				
S:0/0			Processor Arithmetic Carry Flag				
S:0/1			Processor Arithmetic Underflow/ Overflow Flag				
S:0/2			Processor Arithmetic Zero Flag				
S:0/3			Processor Arithmetic Sign Flag				
S:1			Processor Mode Status/ Control				
S:1/0			Processor Mode Bit 0				
S:1/1			Processor Mode Bit 1				
S:1/2			Processor Mode Bit 2				
S:1/3			Processor Mode Bit 3				
S:1/4			Processor Mode Bit 4				
S:1/5			Forces Enabled				
S:1/6			Forces Present				
S:1/7			Comms Active				
S:1/8			Fault Override at Powerup				
S:1/9			Startup Protection Fault				
S:1/10			Load Memory Module on Memory Error				
S:1/11			Load Memory Module Always				
S:1/12			Load Memory Module and RUN				
S:1/13			Major Error Halted				
S:1/14			Access Denied				
S:1/15			First Pass				
S:2/0			STI Pending				
S:2/1			STI Enabled				
S:2/2			STI Executing				
S:2/3			Index Addressing File Range				
S:2/4			Saved with Debug Single Step				
S:2/5			DH-485 Incoming Command Pending				
S:2/6			DH-485 Message Reply Pending				
S:2/7			DH-485 Outgoing Message Command Pending				
S:2/15			Comms Servicing Selection				
S:3			Current Scan Time/ Watchdog Scan Time				
S:4			Time Base				
S:5/0			Overflow Trap				
S:5/2			Control Register Error				
S:5/3			Major Err Detected Executing UserFault Routine				
S:5/4			M0-M1 Referenced on Disabled Slot				
S:5/8			Memory Module Boot				
S:5/9			Memory Module Password Mismatch				
S:5/10			STI Overflow				
S:5/11			Battery Low				
S:6			Major Error Fault Code				
S:7			Suspend Code				
S:8			Suspend File				
S:9			Active Nodes				
S:10			Active Nodes				
S:11			I/O Slot Enables				
S:12			I/O Slot Enables				
S:13			Math Register				

Address/Symbol Database

Address	Symbol	Scope	Description	Sym Group	Dev. Code	ABV	BLW
S:14			Math Register				
S:15			Node Address/ Baud Rate				
S:16			Debug Single Step Rung				
S:17			Debug Single Step File				
S:18			Debug Single Step Breakpoint Rung				
S:19			Debug Single Step Breakpoint File				
S:20			Debug Fault/ Powerdown Rung				
S:21			Debug Fault/ Powerdown File				
S:22			Maximum Observed Scan Time				
S:23			Average Scan Time				
S:24			Index Register				
S:25			I/O Interrupt Pending				
S:26			I/O Interrupt Pending				
S:27			I/O Interrupt Enabled				
S:28			I/O Interrupt Enabled				
S:29			User Fault Routine File Number				
S:30			STI Setpoint				
S:31			STI File Number				
S:32			I/O Interrupt Executing				
S:33			Extended Proc Status Control Word				
S:33/0			Incoming Command Pending				
S:33/1			Message Reply Pending				
S:33/2			Outgoing Message Command Pending				
S:33/3			Selection Status User/DF1				
S:33/4			Communicat Active				
S:33/5			Communicat Servicing Selection				
S:33/6			Message Servicing Selection Channel 0				
S:33/7			Message Servicing Selection Channel 1				
S:33/8			Interrupt Latency Control Flag				
S:33/9			Scan Toggle Flag				
S:33/10			Discrete Input Interrupt Reconfigur Flag				
S:33/11			Online Edit Status				
S:33/12			Online Edit Status				
S:33/13			Scan Time Timebase Selection				
S:33/14			DTR Control Bit				
S:33/15			DTR Force Bit				
S:34			Pass-thru Disabled				
S:34/0			Pass-Thru Disabled Flag				
S:34/1			DH+ Active Node Table Enable Flag				
S:34/2			Floating Point Math Flag Disable,Fl				
S:35			Last 1 ms Scan Time				
S:36			Extended Minor Error Bits				
S:36/8			DII Lost				
S:36/9			STI Lost				
S:36/10			Memory Module Data File Overwrite Protection				
S:37			Clock Calendar Year				
S:38			Clock Calendar Month				
S:39			Clock Calendar Day				
S:40			Clock Calendar Hours				
S:41			Clock Calendar Minutes				
S:42			Clock Calendar Seconds				
S:43			STI Interrupt Time				
S:44			I/O Event Interrupt Time				
S:45			DII Interrupt Time				
S:46			Discrete Input Interrupt- File Number				
S:47			Discrete Input Interrupt- Slot Number				
S:48			Discrete Input Interrupt- Bit Mask				
S:49			Discrete Input Interrupt- Compare Value				
S:50			Processor Catalog Number				
S:51			Discrete Input Interrupt- Return Number				
S:52			Discrete Input Interrupt- Accumulat				
S:53			Reserved/ Clock Calendar Day of the Week				
S:55			Last DII Scan Time				

Address/Symbol Database

Address	Symbol	Scope	Description	Sym Group	Dev. Code	ABV	BLW
S:56			Maximum Observed DII Scan Time				
S:57			Operating System Catalog Number				
S:58			Operating System Series				
S:59			Operating System FRN				
S:61			Processor Series				
S:62			Processor Revision				
S:63			User Program Type				
S:64			User Program Functional Index				
S:65			User RAM Size				
S:66			Flash EEPROM Size				
S:67			Channel 0 Active Nodes				
S:68			Channel 0 Active Nodes				
S:69			Channel 0 Active Nodes				
S:70			Channel 0 Active Nodes				
S:71			Channel 0 Active Nodes				
S:72			Channel 0 Active Nodes				
S:73			Channel 0 Active Nodes				
S:74			Channel 0 Active Nodes				
S:75			Channel 0 Active Nodes				
S:76			Channel 0 Active Nodes				
S:77			Channel 0 Active Nodes				
S:78			Channel 0 Active Nodes				
S:79			Channel 0 Active Nodes				
S:80			Channel 0 Active Nodes				
S:81			Channel 0 Active Nodes				
S:82			Channel 0 Active Nodes				
S:83			DH+ Active Nodes				
S:84			DH+ Active Nodes				
S:85			DH+ Active Nodes				
S:86			DH+ Active Nodes				
U:3	LINEAR_SEARCH	Global					
U:4	BINARY_SEARCH	Global					

Instruction Comment Database

Address	Instruction	Description

Symbol Group Database

Group_Name Description