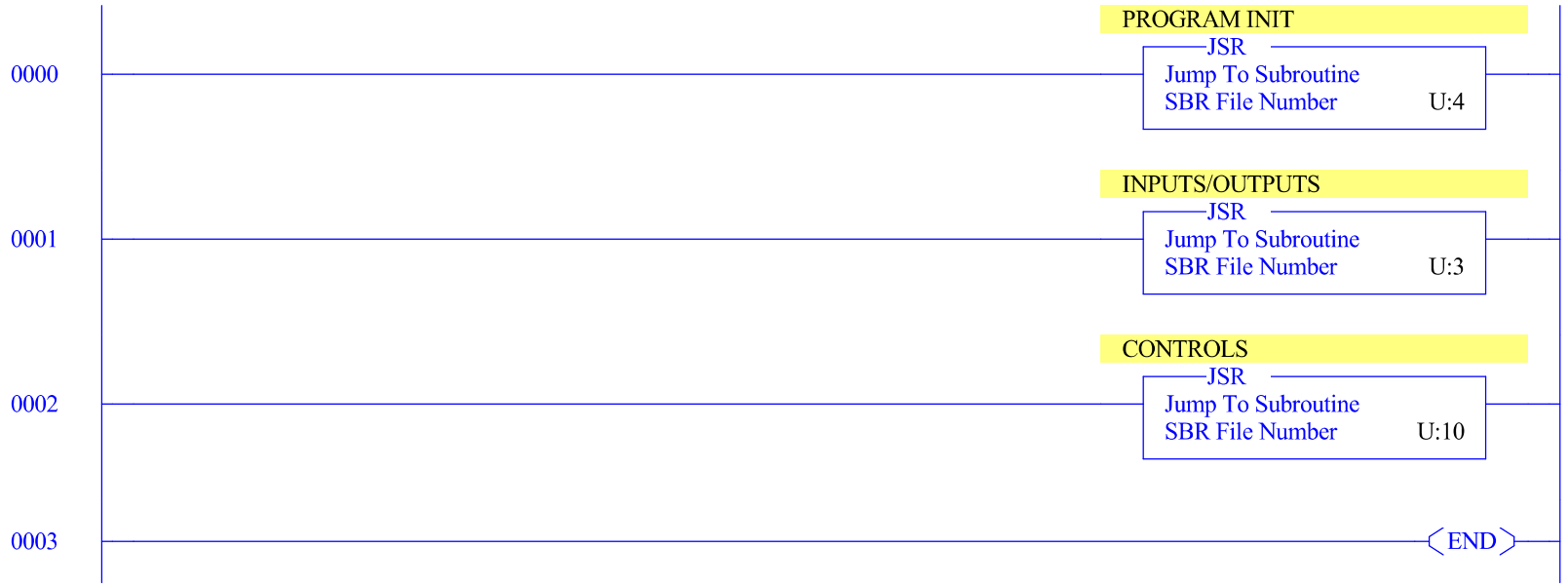
LAD 2 - MAIN --- Total Rungs in File = 4



NO ACTIVE INPUTS/OUTPUTS *CAN BE REMOVED*

0000 END

LAD 4 - INIT - PROGRAM INIT --- Total Rungs in File = 1

NO ACTIVE CONDITIONS NEEDING INITIALIZATION *CAN BE REMOVED*

0000 END

IF BAD PART NUMBER, RETURN "ERROR: PART NUMBER DOES NOT EXIST (11-1)" TO THE BARCODE STRING FILE

0000

STRING INPUTTED B3:1 0 — PT NUM CHECK OK B3:1 1 — ERROR OUTPUT

COP
Copy File
Source #ST11:1
Dest #ST9:8
Length 1

IF REQ SUBTRACTION IS GRT INVETORY, RETURN "ERROR: CANNOT WITHDRAW NEGATIVE INVENTORY (11-2)" TO THE BARCODE STRING FILE

0001

STRING INPUTTED B3:1 0 — INV SUBTRACT ERROR B3:1 3 — ERROR OUTPUT

COP
Copy File
Source #ST11:2
Dest #ST9:8
Length 1

IF NO ERRORS, RETURN "GOOD SCAN" TO THE BARCODE STRING FILE

0002

STRING INPUTTED B3:1 0 — NO ERRORS B3:1 15 — ERROR OUTPUT

COP
Copy File
Source #ST11:0
Dest #ST9:8
Length 1

0003

END

Checks if there is a new string that has been inputted via BCR string

0000

BARCODE INPUT

NEQ
Not Equal
Source A ST9:0.LEN
0<
Source B 0
0<

ONS
B3:0
ONS
0

STRING INPUTTED
B3:1
0

COPIES THE INPUTTED STRING TO A HOLDING MEMORY LOCATION

0001

STRING INPUTTED
B3:1
0

COP
Copy File
Source #ST9:0
Dest #ST9:1
Length 1

CLEARS THE STRING FROM THE BCR WITH A BLANK STRING TO REFRESH INPUT

0002

STRING INPUTTED
B3:1
0

COP
Copy File
Source #ST9:2
Dest #ST9:0
Length 1

JUMPS TO THE STRING PARSE FILE (ST_PARSE), LOCATES THE DELIMITERS (:) AND (-) AND PULLS PART NUMBER, QTN AND DIRECTION FROM THE INPUT STRING

0003

PERFORM STRING LOCATING AND PARSING

JSR
Jump To Subroutine
SBR File Number U:11

JUMPS TO THE ERROR CHECKING FILE (ERROR CHK) TO CHECK THE INPUTTED INFORMATION. WILL CHECK IF THE PART NUMBER EXISTS AND IF THE QTY OF THE INPUT CAN BE PULLED OUT FROM THE INVENTORY SYSTEM

0004

PERFORM ERROR CHECKING

JSR
Jump To Subroutine
SBR File Number U:12

IF NO ERRORS AND IS AN ADD COMMAND (DIRECTION = 1), THEN PERFORM ADDITION COMMAND OF INV TO THE SYSTEM

0005

NO ERRORS
B3:1
15

INV SUBTRACT COND
B3:1
2

ONS
B3:0
ONS
3

ADD CONDITION
B3:1
4

IF NO ERRORS AND IS AN SUBTRACT COMMAND (DIRECTION = 1), THEN PERFORM SUBTRACTION COMMAND FROM THE SYSTEM

0006

NO ERRORS
B3:1
15

INV SUBTRACT COND
B3:1
2

ONS
B3:0
ONS
3

SUB CONDITION
B3:1
5

USING THE ADD OR SUB CONDITIONS, WILL JUMP TO THE LOGIC FOR ADDING AND SUBTRACTING

0007

ADD_SUB

JSR
Jump To Subroutine
SBR File Number U:13

JUMP TO SECTION FOR INVENTORY ALARMS. WILL PROVIDE FEEDBACK BASED ON CONDIITONS TO BE RETURNED TO THE BARCODE READER FILE FOR OUT OF BOUNDS TRANSACTIONS, BAD PART NUMBER AND A GOOD SCAN

0008

ALARMS

JSR
Jump To Subroutine
SBR File Number U:7

0009

END

GETS THE INDEX OF THE FIRST DELIMITER (-) AND RETURNS AS N7:4. REMOVES 1 FROM THE INTEGER TO GIVE TRUE INDEX FOR PT NUMBER

0000

STRING INPUTTED

B3:1

0

SPECIAL STRING (-)

ASC

String Search

Source ST9:3

Index 1

String Search ST9:1

Result N7:4

0<

RETURNS THE LOCATION OF THE COLON STRING (:)

0001

STRING INPUTTED

B3:1

0

SPECIAL STRING (:)

ASC

String Search

Source ST9:4

Index 1

String Search ST9:1

Result N7:5

0<

PERFORMS CALCULATIONS TO LOCATE DATA CHUNKS
PART NUMBER CHARS - QTY CHARS ; DIRECTION CHAR

0002

STRING INPUTTED

B3:1
0

STR LEN

MOV
Move
Source ST9:1.LEN
2<
Dest N7:6
0<

DIST (-) - 1

SUB
Subtract
Source A N7:4
0<
Source B 1
1<
Dest N7:10
0<

DIST (-) + 1

ADD
Add
Source A N7:4
0<
Source B 1
1<
Dest N7:11
0<

DIST (:) - (DIST (-)+1)

SUB
Subtract
Source A N7:5
0<
Source B N7:11
0<
Dest N7:12
0<

DIST (:)+1

ADD
Add
Source A N7:5
0<
Source B 1
1<
Dest N7:13
0<

LEN - DIST (:)

SUB
Subtract
Source A N7:6
0<
Source B N7:5
0<
Dest N7:14
0<

USES MATH ABOVE TO COLLECT THE CHARS TO PARED OUTPUT STRINGS

0003

STRING INPUTTED
B3:1
0

STORED INPUT
AEX
String Extract
Source ST9:1
Index 1
Number N7:10
Dest ST9:5

STORED INPUT
AEX
String Extract
Source ST9:1
Index N7:11
Number N7:12
Dest ST9:6

STORED INPUT
AEX
String Extract
Source ST9:1
Index N7:13
Number N7:14
Dest ST9:7

WILL INTAKE THE PART NUMBER / QTY / DIRECTION TO BE STORED AS AN INTEGER FOR LATER

0004

STRING INPUTTED
B3:1
0

PARSED INPUT STRING
ACI
String to Integer
Source ST9:5
Dest N7:16
0<

PARSED QNT STRING
ACI
String to Integer
Source ST9:6
Dest N7:17
0<

PARSED DIR STRING
ACI
String to Integer
Source ST9:7
Dest N7:18
0<

0005

END

IF A STRING IS INPUTTED, AND INPUT STRING MATCHES WITH A PART NUMBER FROM THE PT NUMBER FILE (HARDCODED), THEN RETURNS A BIT TO TELL US THE PART NUMBER IS A GOOD PART NUMBER

0000

STRING INPUTTED
B3:1
0

PARSED INPUT STRING
ASR
ASCII String Compare
Source A ST9:5
Source B ST10:0

PARSED INPUT STRING
ASR
ASCII String Compare
Source A ST9:5
Source B ST10:1

PARSED INPUT STRING
ASR
ASCII String Compare
Source A ST9:5
Source B ST10:2

B3:0
ONS
1

PT NUM CHECK OK
B3:1
1

CHECKS THE INPUTED DIRECTION IF IT IS A 2 TO DENOTE A SUBTRACTION FROM THE INVENTORY

0001

PT NUM CHECK OK
B3:1
1

EQU
Equal
Source A N7:18
0<
Source B 2
2<

ONS
B3:0
ONS
2

INV SUBTRACT COND
B3:1
2

ST10:0, 1, 2 ARE OUR LIVE PART NUMBERS. IF WE ARE SUBTRACTING FROM THE INVENTORY, MATCH THE INPUTTED PART NUMBER WITH THE LIBRARY OF PT NUMBERS AND ATTEMPT TO SUBTRACT THE NUMBER USING A WORKING INT. ROOM FOR IMPROVEMEMT IS TO SWITCH TO A GREATER THAN CHECK (IF INPUTTED QTY > STORED QTY, THEN ERROR)

0002

INV SUBTRACT COND
B3:1
2

PARSED INPUT STRING
ASR
ASCII String Compare
Source A ST9:5
Source B ST10:0

WORKING INT
SUB
Subtract
Source A N7:0
0<
Source B N7:17
0<
Dest N7:20
0<

ST10:0, 1, 2 ARE OUR LIVE PART NUMBERS. IF WE ARE SUBTRACTING FROM THE INVENTORY, MATCH THE INPUTTED PART NUMBER WITH THE LIBRARY OF PT NUMBERS AND ATTEMPT TO SUBTRACT THE NUMBER USING A WORKING INT. ROOM FOR IMPROVEMEMT IS TO SWITCH TO A GREATER THAN CHECK (IF INPUTTED QTY > STORED QTY, THEN ERROR)

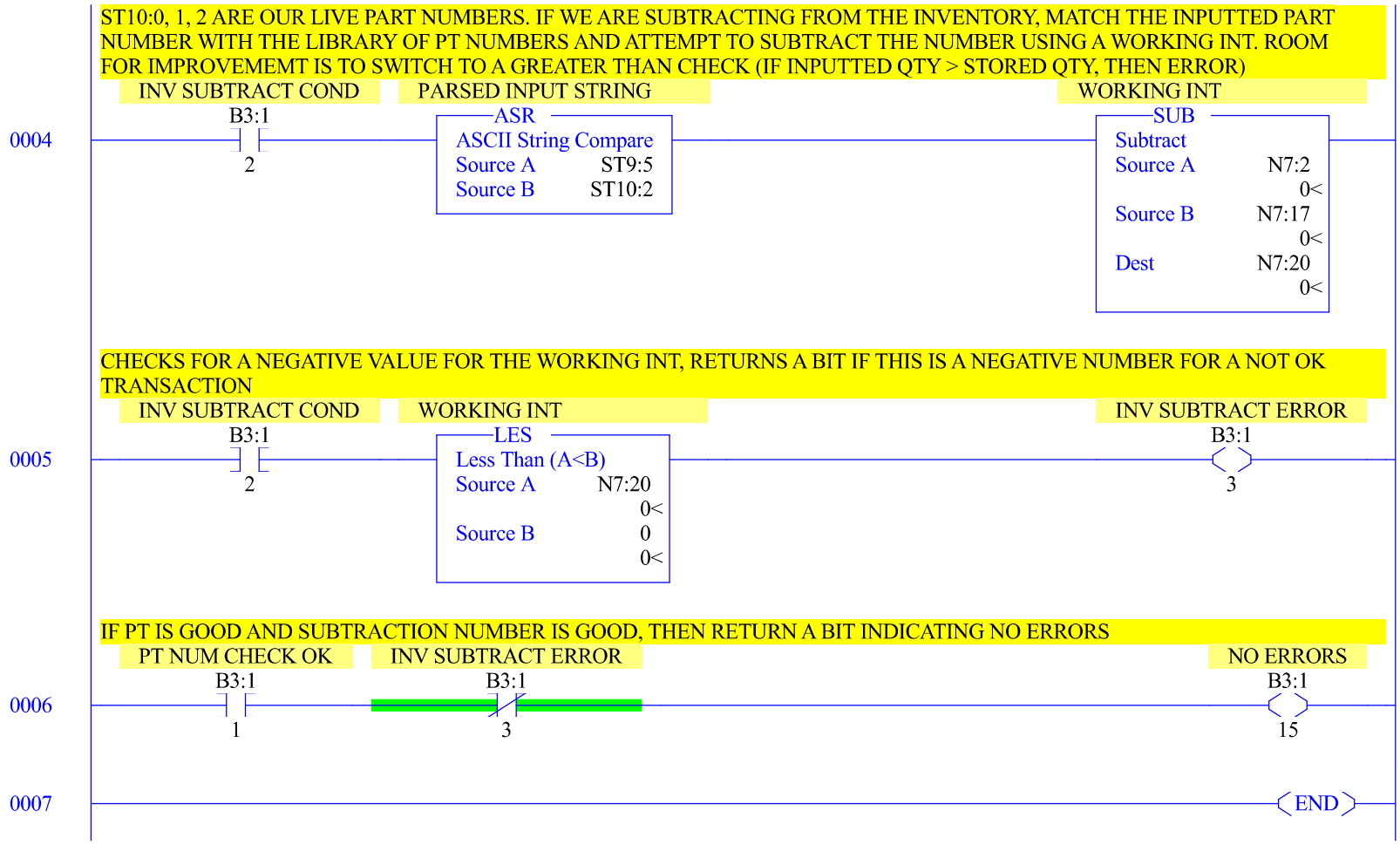
0003

INV SUBTRACT COND
B3:1
2

PARSED INPUT STRING
ASR
ASCII String Compare
Source A ST9:5
Source B ST10:1

WORKING INT
SUB
Subtract
Source A N7:1
0<
Source B N7:17
0<
Dest N7:20
0<

PT NUM CHECK OK
B3:1
1


ST10:0, 1, 2 ARE OUR LIVE PART NUMBERS. IF WE ARE SUBTRACTING FROM THE INVENTORY, MATCH THE INPUTTED PART NUMBER WITH THE LIBRARY OF PT NUMBERS AND ATTEMPT TO SUBTRACT THE NUMBER USING A WORKING INT. ROOM FOR IMPROVEMEMT IS TO SWITCH TO A GREATER THAN CHECK (IF INPUTTED QTY > STORED QTY, THEN ERROR)
INV SUBTRACT COND
PARSED INPUT STRING
WORKING INT
0004
B3:1
2
ASR
ASCII String Compare
Source A ST9:5
Source B ST10:2
SUB
Subtract
Source A N7:2
0<
Source B N7:17
0<
Dest N7:20
0<
CHECKS FOR A NEGATIVE VALUE FOR THE WORKING INT, RETURNS A BIT IF THIS IS A NEGATIVE NUMBER FOR A NOT OK TRANSACTION
INV SUBTRACT COND
WORKING INT
INV SUBTRACT ERROR
0005
B3:1
2
LES
Less Than (A<B)
Source A N7:20
0<
Source B 0
0<
B3:1
3
IF PT IS GOOD AND SUBTRACTION NUMBER IS GOOD, THEN RETURN A BIT INDICATING NO ERRORS
PT NUM CHECK OK
INV SUBTRACT ERROR
NO ERRORS
0006
B3:1
1
B3:1
3
B3:1
15
0007
END

PERFORMS ADD OR SUBTRACT OF INPUTTED PT NUMBER WITH THE ACTIVE INVENTORY

0000

ADD CONDITION

B3:1
4

PARSED INPUT STRING

ASR
ASCII String Compare
Source A ST9:5
Source B ST10:0

P#: 123 COUNT

ADD
Add
Source A N7:0
0<
Source B N7:17
0<
Dest N7:0
0<

0001

SUB CONDITION

B3:1
5

PARSED INPUT STRING

ASR
ASCII String Compare
Source A ST9:5
Source B ST10:0

P#: 123 COUNT

SUB
Subtract
Source A N7:0
0<
Source B N7:17
0<
Dest N7:0
0<

0002

ADD CONDITION

B3:1
4

PARSED INPUT STRING

ASR
ASCII String Compare
Source A ST9:5
Source B ST10:1

P#: 456 COUNT

ADD
Add
Source A N7:1
0<
Source B N7:17
0<
Dest N7:1
0<

0003

SUB CONDITION

B3:1
5

PARSED INPUT STRING

ASR
ASCII String Compare
Source A ST9:5
Source B ST10:1

P#: 456 COUNT

SUB
Subtract
Source A N7:1
0<
Source B N7:17
0<
Dest N7:1
0<

0004

ADD CONDITION

B3:1
4

PARSED INPUT STRING

ASR
ASCII String Compare
Source A ST9:5
Source B ST10:2

P#: 789 COUNT

ADD
Add
Source A N7:2
0<
Source B N7:17
0<
Dest N7:2
0<

0005

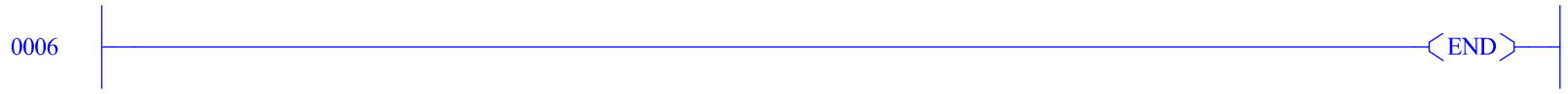
SUB CONDITION

B3:1
5

PARSED INPUT STRING

ASR
ASCII String Compare
Source A ST9:5
Source B ST10:2

P#: 789 COUNT

SUB
Subtract
Source A N7:2
0<
Source B N7:17
0<
Dest N7:2
0<

LAD 13 - ADD_SUB - ADD AND SUBTRACT LOGIC --- Total Rungs in File = 7