



# DECUS

## PROGRAM LIBRARY

DECUS NO.	8-273
TITLE	ALGONQUIN ASSEMBLER
AUTHOR	John Kiss
COMPANY	Algonquin College Ottawa, Ontario, Canada
DATE	May 12, 1970
SOURCE LANGUAGE	PAL III

DECLASSIFIED  
FEDERAL BUREAU OF INVESTIGATION



CONFIDENTIAL

1. [Illegible]

2. [Illegible]

3. [Illegible]

4. [Illegible]

5. [Illegible]

6. [Illegible]

7. [Illegible]

8. [Illegible]

9. [Illegible]

10. [Illegible]

11. [Illegible]

12. [Illegible]

13. [Illegible]

14. [Illegible]

15. [Illegible]

16. [Illegible]

17. [Illegible]

18. [Illegible]

19. [Illegible]

20. [Illegible]

## ALGONQUIN ASSEMBLER

DECUS Program Library Write-up

DECUS No. 8-273

### PURPOSE

The purpose of this is to modify the Phoenix Assembler so that it

1. will accept symbolic programs from the card reader;
2. will do all three passes automatically;
3. will load automatically after assembly;
4. will execute automatically provided no error is found in assembly.  
No binary tape is punched.

In general the system is supposed to operate without human intervention.

There are two patches assembled separately:

### PATCH 1

A number of pseudoinstructions were added to facilitate I/O instructions.

1. RDO - Read octal. Four digit octal numbers are accepted from the keyboard into the AC. If digits 8 or 9 or more than four digits are entered, the error message "DE" (or data error) is typed and the monitor gets ready for the next job. Every octal number must be terminated by a comma. Spaces, line feeds and carriage returns are ignored. Negative signs are allowed.
2. RDA - Reads the ASCII code of an alphanumeric character into the AC. Blanks and Rubouts are ignored. No terminating character is required.

These two pseudoinstructions occupy locations 6000 - 6077 and 2157 - 2175. The latter patch overlays the PAL III pseudoinstruction "FIELD."

On this patch there is also a routine that allows stopping a program in midstream and return to the monitor for the next job by 200 LOAD ADDRESS - START. This is important as the users can notice errors early and thus they do not want all three passes concluded to save time.

### PATCH 2

The first part of this patch is the CARD read unit which reads and loads the symbolic program on the card. Note here that the symbols for CR and LF are interchanged. This part takes up locations 6100 - 6141.

The second part is the program for the pseudo TYB - or type the contents of the AC as a 12 bit binary number. The AC is cleared after execution of TYB. This program takes up 6142 - 6163.

The most important part of this whole patch is the monitor. This monitor affects the automatic transition from pass to pass. It also checks for errors. If there are any, it will inhibit execution, typing the message "NO EXECUTION." It will also type the heading for a new job "ALGON-QUIN ASSEMBLER." It will initiate execution of the loaded program. It will return to pick up a new job after the termination of the last one (pseudo STP will do it). The monitor occupies 6200 - 6260.

Locations 6261 - 6307 are for typing headings for the program listing.

The rest of this patch is a collection of patches onto the Phoenix Assembler. The remarks to these programs are self explanatory.

This patch also provides a number of pseudoinstructions which are to facilitate I/O.

TYO - Types contents of AC as a four bit octal number. AC = 0 after execution. No CR or LF given.

TYA - Types the character whose ASCII code is in bits 4 - 11 in the AC. The AC = 0 after execution. Again no CR or LF or space is given.

STP - This is equivalent to HLT. STP however does not stop the computer but returns the monitor to pick up the next job.

TYB - Types the contents of AC as a 12 bit binary number. AC = 0 after execution.

In the permanent symbol table five instruction symbols were replaced with the new pseudoinstructions.

PCF → RDO	2452
PLS → RDA	2462
PPC → TYA	2456

PSF → TYO	2446
MMMM → STP	2662
MMML → TYB	2672

The octal codes for these pseudos are:

RDO = 4524
RDA = 4427
TYA = 4422
TYO = 4431
TYB = 4452
STP = 5451

## Memory Map for the PHOENIX ASSEMBLER and the two patches

0	2735	3215	6000	6377	7400
Phoenix Assembler	User's Symbol Table	User's Text Buffer for Symbolic Programs	Monitor Programs Patch 1 Patch 2	User's Object Program	Loaders

### Note

1. The optical mark reader used with this system is the Hewlett-Packard 2761A.

A technical write-up on the interface and operating instructions of this reader is included.

2. This system was used for a whole year in teaching technicians computer fundamentals. We used it basically as a batch-processing system. Classes up to 50 students could write and run programs at the same time. Programs are easy to debug by changing incorrect cards and by rerunning the program. The cost of the reader is inexpensive (\$3,500) and the interface is easy.

3. A sample program is attached as well as the listings of the two patches.

4. In the complete ALGONQUIN ASSEMBLER binary tape provided, the first long part is the PHOENIX ASSEMBLER and the last two smaller parts are Patch 1 and Patch 2 referred to above.

## INTRODUCTION AND DESCRIPTION

### Introduction

This document provides interconnection and operating instructions for the DS-370 Marked Card Reader Interface.

### References

Hewlett-Packard - Manual for Model HP 2761A Optical Mark Readers.

Digital Systems Associates Limited - LD-370-F003.

### Description

The DS-370 Interface is an interface to the HP 2761A Optical Mark Reader for a PDP-8/S computer - the Reader which reads at a rate of 105 characters per second. It uses the existing interface of the ASR-33 with only a higher frequency clock and signal conditioning gates added.

A switch on the console selects either the Card Reader or ASR-33 input to the PDP-8/S, while output remains on the ASR-33.

## HARDWARE

### Connections

Input - Input to the system is the HP 2761A Optical Mark Reader and an ASR-33 Teletypewriter. The rotary switch on the console selects the appropriate clock and input signal to the existing interface.

Output - Output is accomplished on the ASR-33 in the form of punched paper tape and printed output.

### Operator Controls and Action

Since the computer has no direct control over the card reader, all cards in the hopper will be read by the reader. The operator must turn the card reader on and off manually. An example of a logical sequence of events is given below.

- a) Load card hopper with the number of cards to provide correct amount of data to be accepted by the PDP-8/S.
- b) Turn console switch to position marked CARD.
- c) Start program. Program will proceed and wait for flag from card readers.
- d) Push START button on reader.
- e) The cards will be read until the hopper is empty at which time push the STOP button. This will stop the reader and the program will continue until a halt is encountered.

**CAUTION:** STARTING THE READER GIVES CODE "0005." OPERATOR'S PROGRAM SHOULD TAKE CARE OF IT.

### Command Set

The card reader uses the same commands as the ASR-33 teletype reader.

### Performance Specifications

HP 2761A Optical Mark Reader  
Option 04 - 40 column @ 105 character/sec.  
Accessory 12841 - Hollerith Code, 64 characters.

The leading edge of the card produces an automatic CR, while the trailing edge produces an automatic LF.

DOCTAL READ ROUTINE

\*6000

6000	4000	RDT,	0	
6001	7200		CLA	
6002	1266		IAD M5	
6003	3273		DCA NDIG	
6004	3275		DCA TEMP	
6005	7040		CMA	
6006	3274	BAK,	DCA NEG	
6007	4427	GET,	JMS I 27	
6010	1267		IAD M240	
6011	7650		SNA CLA	/NO SP
6012	5207		JMP GET	
6013	1111		IAD STO	
6014	1106		IAD M212	
6015	7450		SNA	/NO LF
6016	5207		JMP GET	
6017	1074		IAD M3	
6020	7650		SNA CLA	
6021	5207		JMP GET	
6022	1111		IAD STO	
6023	1270		IAD M255	
6024	7450		SNA	/NEG. NO.
6025	5206		JMP BAK	
6026	7001		IAC	
6027	7650		SNA CLA	/COMMA ENDS NO.
6030	5254		JMP END	
6031	1111		IAD STO	/DIGITS 0-7 ALLOWED
6032	1271		IAD M260	
6033	7510		SPA	
6034	5260		JMP ERR	
6035	1341		IAD M10	
6036	7700		SMA CLA	
6037	5260		JMP ERR	
6040	2273		ISZ NDIG	
6041	7410		SKP	
6042	5260		JMP ERR	/ TOO MANY DIGITS
6043	1111		IAD STO	
6044	1271		IAD M260	/ STRIP OFF 260
6045	3111		DCA STO	/ PACK IN TEMP
6046	1275		IAD TEMP	
6047	7106		CLL RTL	
6050	7004		KAL	
6051	1111		IAD STO	
6052	3275		DCA TEMP	
6053	5207		JMP GET	/ NEXT DIGIT
6054	1275	END,	IAD TEMP	
6055	2274		ISZ NEG	
6056	7041		CIA	/NEGATIVE
6057	5600		JMP I RDO	
6060	7200	ERR,	CLA	
6061	1117		IAD CRLF	
6062	4430		JMS I EEE	
6063	1276		IAD DE	
6064	4430		JMS I EEE	/TYPE ERROR CODE
6065	5677		JMP I REI	/RETURN TO MONITOR

```

        //
6066  7773  M5,      -5
6067  7540  M240,    -240
6070  7523  M255,    -255
6071  7520  M260,    -260
6072  7401  M377,    -377
6073  0000  NDIG,    0
6074  0000  NEG,     0
6075  0000  TEMP,    0
6076  0405  DE,      0405
6077  6212  RET,      6212
        STO=111
        //
*2157                                /OVER FIELD
2157  0000  LISN,    0                                /RDA ROUT
2160  6032  KCC
2161  6031  REJ,    KSF
2162  5361  JMP     .-1
2163  6036  KRB
2164  7450  SNA
2165  5361  JMP REJ   /NO BLANK L/T
2166  3111  DCA STJ
2167  1111  TAD STJ
2170  1375  TAD N377
2171  7650  SNA CLA
2172  5361  JMP REJ   /NO RUBOUT
2173  1111  TAD STO
2174  5757  JMP I LISN
2175  7401  N377,    -377
        //
/ MANUAL INTERRUPT
*200
0200  5601  JMP I  .+1
0201  2202
        //
*2202                                /OVER FIX
2202  6046  TLS
2203  6032  KCC
2204  3611  DCA I STX
2205  7040  CMA
2206  3134  DCA 134
2207  5610  JMP I  .+1
2210  0202  202
2211  6253  STX,    6253
        //
M212=106
M3=74
EEE=30
CRLF=117
MIO=6141
        //
/ PUT RDO IN SYM. TABLE

```



\* 2452  
 2452 2204 2204  
 2453 1700 1700  
 2454 0000 0000  
 2455 4524 JMS I 124

//////  
 / PUT RDA IN SYM. TABLE

\* 2462  
 2462 2204 2204  
 2463 0100 0100  
 2464 0000 0000  
 2465 4427 JMS I 27

\* 27  
 0027 2157 LISN  
 BAK 6006  
 CKLF 0117  
 DE 6076  
 EEE 0030  
 END 6054  
 ERK 6060  
 GET 6007  
 LISN 2157  
 M10 6141  
 M212 0106  
 M240 6067  
 M255 6070  
 M260 6071  
 M3 0074  
 M377 6072  
 M5 6066  
 VD16 6073  
 NEG 6074  
 V377 2175  
 RDD 6000  
 REJ 2161  
 RET 6077  
 SI7 0111  
 STX 2211  
 TEMP 6075

END OF PATCH 1.

/CARD READER PATCH TO ALGONQUIN PAL.

\*6100

6100	6036	CARD,	KRB
6101	0333		AND P177
6102	1331		TAD M46
6103	7450		SNA
6104	5326		JMP D2
6105	1332		TAD P34
6106	7450		SNA
6107	5321		JMP D1
6110	2337		ISZ OFF
6111	7410		SKP
6112	5326		JMP D2
6113	1335		TAD P5
6114	7450		SNA
6115	5734		JMP I TINT
6116	1341		TAD M10
6117	7450		SNA
6120	5734		JMP I TINT
6121	2337	D1,	ISZ OFF
6122	7000		NOP
6123	1340		TAD P215
6124	1101		TAD 101
6125	5736		JMP I BAK
6126	7240	D2,	CLA CMA
6127	3337		DCA OFF
6130	5734		JMP I TINT
6131	7732	M46,	7732
6132	0034	P34,	34
6133	0177	P177,	0177
6134	1465	TINT,	1465
6135	0005	P5,	5
6136	1471	BAK,	1471
6137	0000	OFF,	0
6140	0215	P215,	215
6141	7770	M10,	7770
		/////	
		/BINARY OUTPUT ROUTINE.	
6142	0000	BIOUT,	0
6143	3362		DCA SAVE
6144	1360		TAD M12
6145	3363		DCA KUNT
6146	1362	RJT,	TAD SAVE
6147	7104		CLL RAL
6150	3362		DCA SAVE
6151	7430		SZL
6152	7001		IAC
6153	1361		TAD C260
6154	4422		JMS I 22
6155	2363		ISZ KUNT
6156	5346		JMP RJT
6157	5742		JMP I BIOUT

```

6160 7764 M12, -14
6161 0260 C260, 260
6162 0000 SAVE, 0
6163 0000 KUNT, 0
/////
/PATCH TO THE START OF PAL.
/IF CHECKS FOR PASS 0.
6164 7200 PASS, CLA
6165 1771 TAD I XST
6166 0124 AND 124
6167 5770 JMP I .+1
6170 0225 225
6171 6253 XST, ST
/////
/PASS MONITOR ROUTINE.
/ALSO SPACES BETWEEN JOBS AND PRINTS HEADINGS.
*6200
6200 7300 MON, CLA CLL
6201 1253 TAD ST
6202 1036 TAD C2000
6203 3253 DCA ST
6204 7430 SZL
6205 5207 JMP TRY
6206 5652 JMP I C204
6207 1253 TRY, TAD ST
6210 7650 SNA CLA
6211 5221 JMP DOIT
6212 1655 TAD I STP
6213 3654 DCA I ORG
6214 1117 TAD CRLF
6215 4430 JMS I EEE
6216 1256 TAD NO
6217 4430 JMS I EEE
6220 5224 JMP .+4
6221 1062 DOIT, TAD C4
6222 4300 JMS LF
6223 4430 JMS I EEE
6224 1064 TAD C6
6225 4261 JMS TEXT
6226 6344 EXEC
6227 1117 TAD CRLF
6230 4430 JMS I EEE
6231 5654 JMP I ORG
6232 7300 RETURN, CLA CLL
6233 3253 DCA ST
6234 1062 TAD C4
6235 4261 JMS TEXT
6236 6352 EXIT
6237 1257 TAD C20
6240 4300 JMS LF
6241 5465 JMP I C200
6242 7200 HEAD, CLA /TYPE THE HEADING.
6243 1260 TAD C13
6244 4261 JMS TEXT
6245 6356 ALCON
6246 1117 TAD CRLF
6247 4430 JMS I EEE
6250 5651 JMP I .+1

```

6251	0242		242
6252	0204	C204,	204
6253	0000	ST,	0
6254	0000	ORG,	0
6255	2665	STP,	2665
6256	1617	NO,	1617
6257	0020	C20,	20
6260	0013	C13,	13

/ROUTINE TYPES A HEADING.ON ENTRANCE THE AC CONTAINS  
 /THE NUMBER OF WORDS TO BE TYPED, AND THE LOC. AFTER THE  
 /CALL CONTAINS THE ADDRESS OF THE FIRST LETTER.

6261	0000	TEXT,	0
6262	7041		CIA
6263	3276		DCA TALY
6264	1661		TAD I TEXT
6265	3277		DCA FIRST
6266	1677		TAD I FIRST
6267	4430		JMS I EEE
6270	2277		ISZ FIRST
6271	2276		ISZ TALY
6272	5266		JMP .-4
6273	7300		CLA CLL
6274	2261		ISZ TEXT
6275	5661		JMP I TEXT
6276	0000	TALY,	0
6277	0000	FIRST,	0

//////////

/ROUTINE TYPES AS MANY CRLF'S AS THE NUMBER FOUND  
 /IN THE AC. AC IS CLEARED ON EXIT.

6300	0000	LF,	0
6301	7041		CIA
6302	3276		DCA TALY
6303	1117		TAD CRLF
6304	4430		JMS I EEE
6305	2276		ISZ TALY
6306	5303		JMP .-3
6307	5700		JMP I LF

//////////

/ROUTINE LOADS PROGRAM IN MEMORY. IT ALSO CHECKS THAT  
 /THE ADDRESSES USED ARE IN THE RANGE 6400-7400. IF NOT  
 /THE JOB IS TERMINATED.

6310	1142	ADRCH,	TAD AACT
6311	1336		TAD LOW
6312	7420		SNL
6313	5327		JMP QUIT
6314	7300		CLA CLL
6315	1142		TAD AACT
6316	1065		TAD C200
6317	1065		TAD C200
6320	7430		SZL
6321	5327		JMP QUIT
6322	7200		CLA
6323	1140		TAD BINW
6324	3542		DCA I AACT
6325	5726		JMP I .+1
6326	0335		335
6327	7300	QUIT,	CLA CLL

```

6330 1142 TAD AACT
6331 4431 JMS I 31
6332 1062 TAD C4
6333 4261 JMS TEXT
6334 6371 LOC
6335 5232 JMP RETURN
6336 1400 LOW, 1400
//////////
/THIS ROUTINE ADDS 1 TO THE PASS COUNTER WORD ST
/IF THERE HAS BEEN AN ERROR. THIS WILL PREVENT
/EXECUTION OF THE JOB.
/THIS IS PATCHED TO THE ERROR HANDLER ROUTINE OF PAL.
6337 0000 ERR, 0
6340 4430 JMS I EEE
6341 2253 ISZ ST
6342 7200 CLA
6343 5737 JMP I ERR
//////////
/CODES FOR EXECUTION
6344 0005 EXEC, 0005
6345 3005 3005
6346 0325 0325
6347 2411 2411
6350 1716 1716
6351 4543 4543
/CODES FOR EXIT
6352 4543 EXIT, 4543
6353 4300 4300
6354 0530 0530
6355 1124 1124
/CODES FOR ALGONQUIN ASSEMBLER
6356 4500 ALGON, 4500
6357 0114 0114
6360 0717 0717
6361 1621 1621
6362 2511 2511
6363 1655 1655
6364 0123 0123
6365 2305 2305
6366 1502 1502
6367 1405 1405
6370 2256 2256
/CODES FOR BAD ADDRESS.
6371 0002 LOC, 0002
6372 0104 0104
6373 0014 0014
6374 1703 1703
/MISCELLANEOUS PATCHES.
/DO NOT PUNCH THE ORIGIN BUT SET ORG IN PASS 2.
*2011
2011 1142 TAD AACT
2012 3614 DCA I .+2
2013 5446 POPJ
2014 6254 ORG

```

```

//////////
/NO LEADER TRAILER BEFORE PASS 2.

```

```

*252
0252 7000      NOP
////////
/LOAD PROGRAM IN STORAGE AND OMIT BINARY PUNCH IN PASS 2.
*333
0333 5734      JMP I .+1
0334 6310      ADRCH
////////
/INDICATE ERROR
*1647
1647 4777      JMS I POINT
/POINTER TO ERR.
*1777
1777 6337      POINT, ERR
////////
/ENTER MONITOR
*230
0230 5631      JMP I .+1
0231 6200      MON
////////
/PATCH TO BEGINNING OF PAL.
*223
0223 5624      JMP I .+1
0224 6164      PASS
////////
/NO SYMBOL TABLE LIST IN PASS 3.
*601
0601 7640      SZA CLA
////////
/INSTEAD OF PUNCHING CHECKSUM AND L/T AFTER PASS 2.
/GIVE A CRLF.
*654
0654 1117      TAD 117
0655 4430      JMS I EEE
0656 7000      NOP
////////
/PATCH TO READ-IN ROUTINE FOR CARD INPUT
*1467
1467 5670      JMP I .+1
1470 6100      CARD
////////
/PSEUDO OP CODE FOR STP.
*2662
2662 2324      2324
2663 2000      2000
2664 0000      0000
2665 5451      JMP I 51
////////
/PSEUDO OP TYO
*2446
2446 2431      2431
2447 1700      1700
2450 0000      0000
2451 4431      JMS I 31
////////
/PSEUDO OP CODE FOR TYA

```

```

*2456
2456 2431 2431
2457 0100 0100
2460 0000 0000
2461 4422 JMS I 22
/////
/PSEUDO OP CODE FOR TYB
*2672
2672 2431 2431
2673 0200 0200
2674 0000 0000
2675 4452 JMS I 52
/////
/POINTER TO TYPE HEADING
*202
0202 7410 SKP
0203 6242 HEAD
/////
/TYPE HEADING BEFORE PASS 1.
*234
0234 5603 JMP I 203
/////
/POINTER FOR STP.
*51
0051 6232 RETURN
/////
/POINTER FOR BIN TYPE.
*52
0052 6142 BIOUT
C6=64
C4=62
C200=65
C2000=36
EEE=30
CRLF=117
AACT=142
BINW=140
POPJ=5446

```

AACT	0142
ADRCH	6310
ALGN	6356
BAK	6136
BINW	0140
BIDUT	6142
CARD	6100
CRLF	0117
C13	6260
C20	6257
C200	0065
C2000	0036
C204	6252
C260	6161
C4	0062
C6	0064
DOIT	6221
D1	6121
D2	6126
EEE	0030
ERR	6337
EXEC	6344
EXIT	6352
FIRST	6277
HEAD	6242
KUNT	6163
LF	6300
LOC	6371
LOW	6336
MON	6200
M10	6141
M12	6160
M46	6131
NO	6256
OFF	6137
ORG	6254
PASS	6164
POINT	1777
POPJ	5446
P177	6133
P215	6140
P34	6132
P5	6135
QUIT	6327
RETURN	6232
ROT	6146
SAVE	6162
ST	6253
STP	6255
TALY	6276
TEXT	6261
TINT	6134
TRY	6207
XST	6171

END OF PATCH NO. 2.



ALGONQUIN-ASSEMBLER.

COUNT 7023  
 CR 7027  
 LF 7030  
 SAVE 7025  
 SP 7026  
 TALLY 7024  
 JP 7003

/EXAMPLE OF A PROGRAM DONE USING THE  
 /ALGONQUIN-ASSEMBLER

/////

/THE PROGRAM DUMPS ITSELF BOTH IN OCTAL  
 /AND IN BINARY

\*7000

7000	7200		CLA
7001	1223		TAD COUNT
7002	3224		DCA TALLY
7003	1200	UP,	TAD 7000
7004	3225		DCA SAVE
7005	1225		TAD SAVE
7006	4431		TYJ
7007	1226		TAD SP
7010	4422		TYA
7011	1225		TAD SAVE
7012	4452		TYB
7013	1227		TAD CR
7014	4422		TYA
7015	1230		TAD LF
7016	4422		TYA
7017	2203		ISZ UP
7020	2224		ISZ TALLY
7021	5203		JMP UP
7022	5451		STP
7023	7747	COUNT,	-31
7024	0000	TALLY,	0
7025	0000	SAVE,	0
7026	0240	SP,	240
7027	0215	CR,	215
7030	0212	LF,	212

/SYMBOL TABLE IS OMITTED AT THE END  
 /OF PASS III.

EXECUTION

7200 111010000000  
1223 001010010011  
3224 011010010100  
1203 001010000011  
3225 011010010101  
1225 001010010101  
4431 100100011001  
1226 001010010110  
4422 100100010010  
1225 001010010101  
4452 100100101010  
1227 001010010111  
4422 100100010010  
1230 001010011000  
4422 100100010010  
2203 010010000011  
2224 010010010100  
5203 101010000011  
5451 101100101001  
7747 111111100111  
7773 111111110111  
7773 111111110111  
0240 000010100000  
0215 000010001101  
0212 000010001010

EXIT