MORSE CODE

DECUS Program Library Write-up DECUS NO. 8-424

ABSTRACT

This program accepts Morse code via a logic sense line in real-time and outputs the decoded message on the teleprinter. The pattern recognition algorithm in the program automatically adapts to the sending rate with the maximum reception rate of the computer being limited by the teleprinter to about 100 words per minute. The program classifies a key down condition as either a dot or a dash. The key up conditions are classified either as a space in a character, a space between characters, or a space between words. These pattern classifications permit each character to be decoded via a table look up.

INTRODUCTION

This Morse Code Translator was written to satisfy a project requirement in a 3 credit junior level course entitled "Real-Time Computation". This course is taught by Dr. T. L. Drake, Associate Professor, Department of Electrical and Computer Engineering, Clemson University.

This program has been written to run on a 4K PDP-8 with a logic sense line and a real-time clock. The real-time clock commands are:

6141	Skip if clock flag = 1
6142	Clear clock flag. Discount the flag from the interrupt.
6144	Connect the flag to the interrupt.

The logic sense lines are read in parallel with a 6074 to provide the program with key up and key down information. The accumulator should be cleared prior to this transfer. The switch register at the start of the program determines a mask to mask the bit of interest.

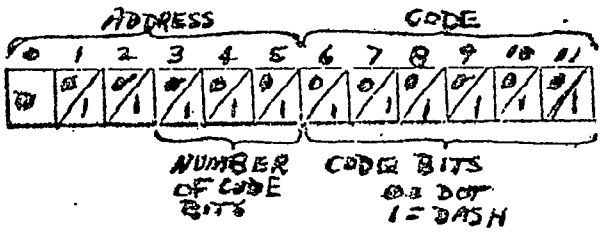
DESCRIPTION

MORSE is an interrupt controlled Morse code translator written for the PDP-8 and compiled through PAL-III. The program will accept code through the logic input lines on the TR-48 analog computer. It reads a "1" as a key up, and a "0" as a key down.

The program will adjust to the sending rate by comparing all information received to the length of the last dash received. The program can become confused if the rate is increased suddenly, and would read everything as a dot. To cause this, the sender would have to suddenly reduce his dash length by more than 50%.

The starting address is 0200_8. When the program is started, it is necessary to set the switch register to the logic trunk mask because the program reads the switch register in the initialization routine and deposits the result in the logic mask. We have found that a clock rate of about 400 Hz is sufficient for sampling the logic inputs. If the user wishes to restart the program to re-initialize either the logic mask, or the dash length, and keyboard interrupt will send the program back to location 0200_8.

The decoded dots and dashs upon reception are assembled into a code word. Upon detecting an end of character, the program references this word indirectly to find the ASCII code for the letter sent. The code word is divided into two sections, as shown on the following page: the address part (bits 0 through 5), and the code part (bits 6 through 11).



As each dot or dash is received, the proper number (0 for a dot; 1 for a dash) is placed in the link and then rotated into bit 11 of the code part. 0100_8 is added to the address part, which increments bits 3 - 5. The two parts are combined, with proper masking to prevent overflow, into the complete code word. The code word can uniquely identify any combination of dots and dashes up to a total of six bits. More than a total of six causes the code part to overflow and the first bits sent are lost. If the code sent contains, for example, 8 dots (the code for "mistake") the first two dots will be lost. However, the address part will still be incremented for each dot, and the number 2000_8 will result. This is the only code in the data list that has an overflow of the code part. Some examples are shown below to illustrate the code word.

LETTER	MORSE CODE	CODE WORD
E	.	1100
T	-	1101
C	-.-.	1412
F	..-.	1402
S	...	1300
8	---..	1534
?	..--..	1614

When the program is started, it initializes the dash length to zero, so the first bit received will be interpreted as a dash. It clears the software printer flag, sets the logic mask by reading the switch register and initializes the code word and its parts. The interrupt is activated and the clock is connected to the interrupt bus.

The program then enters the background program in a wait loop to wait for data to be printed. As soon as data is placed in the buffer, the background program loads the character, prints it, and tests the buffer count for more data. If there is no more data, the buffer address is initialized and the program branches back to the original wait loop. If there is more data in the buffer, the program enters a wait loop to wait for the software printer flag. When the printer is ready, the program loads and prints the data as before.

When an interrupt is received, the program is branched to the interrupt executive (location 0400_8). The machine status is saved. The executive then checks for three sources of interrupts: 1) the clock, 2) the printer, and 3) the keyboard. When a printer interrupt occurs, the software flag is set and the program exits back to the background. When a keyboard interrupt occurs, the program is sent back to the initialization routine.

When a clock interrupt occurs, the last key status (up or down) is placed in the link. The logic trunks are read in and all unused trunks are masked out. The present key status (from the logic trunk) is compared to the past key status (in the link). If they are the same, the count is incremented (routine C01, location 0050_8) and the program exits to the background. If the status changed, the program jumps to the STORE routine (location 0070_8) to decide what was sent. If the past key status was down, the decision is between a dot and a dash. If the past status was up, the decision is either a) the space between a dot and a dash, or b) the space letters, or c) the space between words. All decisions are based on the length of the last dash received.

If the past status was down, the routine compares the count to ½ (dash length). If the count is less than ½ of the dash length, it decides that a dot was sent, places a 0 in the link and jumps to RCWIC (location 0600_8) which places the dot in the code word. When the count is greater than ½ of the past dash length, the dash length is updated to the value of the count, and a 1 is placed in the link. The program then branches to the RCWIC routine.

If the past status was up, again decisions are based on dash length as shown below:

a) count < 3/4 (dash length) = space between dot and dash

b) 3/4 (dash length) < count < 3 (dash length) = space between letters

c) count > 3 (dash length) = space between words

If a) above is true, the count is initialized and the program jumps to the COI routine and then exits. If b) is true, the program is sent to the Letter routine (location 0551_8) which places the letter to be printed into the buffer. If c) is true, the program goes to the SPACE routine (location 0500_8), which does one of two things: 1) deposits the letter to be printed, and a space in the printer buffer, or 2) if the line of print contains more than 50_8 character, it deposits the letter to be printed, a carriage return, and a line feed in the buffer. The program is then sent to the COI routine.

Since the program does not print the letter which was received until the following key up has ended (this is a result of the fact that Morse code is not uniquely decodable), the last letter of a message would never be printed. To remedy this, when the count overflows (after about 10 seconds at 400 Hz) the program executes a JMS to the LETR routine (location 0622_8), which deposits the last letter, two line feeds, and a carriage return in the buffer. The LETR routine also clears a flag (POINT) that prevents LETR from being executed again until more code is received The flag is set in the STORE routine.

```
                *1
0001  5402    JMP I C400
0002  0400    C400,400
                *20
0020  0000    DALTH,0
0021  0000    COUNT,0
0022  0200    MASK,0200
0023  0000    SAVE,0
0024  0000    LINK,0
0025  0000    UPDNS,0
0026  7716    LINCNT,7716
0027  0170    BFFR,170
0030  0000    BFCNT,0
0031  0215    C215,215
0032  0212    C212,212
0033  7716    CM50,7716
0034  0240    C240,240
0035  0170    C170,170
0036  1000    C1000,1000
0037  0000    TPFLG,0
0040  7777    M1,7777
0041  0170    BFF1,170
0042  7777    POINT,7777
                *50
0050  2021    COI,      ISZ COUNT       /COUNT INC. PROGRAM
0051  5062              JMP EXIT
0052  7200              CLA
0053  1042              TAD POINT
0054  7440              SZA
0055  4461              JMS I LETT      /DEPOSITS LAST LETTER WHEN OVERFLOWS
0056  7240              CLA CMA
0057  3021              DCA COUNT
0060  5062              JMP EXIT
0061  0622    LETT,LETR

0062  7300    EXIT,     CLA CLL
0063  1024              TAD LINK
0064  7004              RAL
0065  1023              TAD SAVE
0066  6001              ION
0067  5400              JMP I 0

0070  7200    STORE,    CLA             /STORE ROUTINE
0071  7430              SZL             /IS PAST STATUS UP OR DN?
0072  5074              JMP CPARDN      /DN; GO TO DOWN COMPARING
0073  5113              JMP CPARUP      /UP; GO TO UP COMPARING
0074  3025    CPARDN,   DCA UPDNS       /COMPARE DOWN COUNT TO DASH LENGTH
0075  1020              TAD DALTH       /LOAD DASH LENGTH
0076  7100              CLL
0077  7010              RAR             /MUL. DASH LENGTH BY 1/2
0100  7120              CLL CML         /CONVERT TO 13 BIT ARITHMETIC
0101  7041              CMA IAC         /HAVE -1/2 X DASH LENGTH IN AC & L
```

```
0102  1021          TAD COUNT       /COMPARE LAST COUNT TO DASH LENGTH
0103  7630          SZL CLA         /WAS LAST COUNT A DOT OR DASH?
0104  5111          JMP NEG         /DOT.
0105  1021          TAD COUNT       /DASH, UPDATE DALTH
0106  3020          DCA DALTH
0107  7120          CLL CML         /PUT 1 IN LINK
0110  5551          JMP I RCWIC     /PUT DATA INTO CODE WORD
0111  7100  NEG,     CLL             /PUT 0 IN LINK
0112  5551          JMP I RCWIC     /PUT DATA INTO CODE WORD
0113  7140  CPARUP,  CMA CLL
0114  3025          DCA UPDNS       /SET LAST UP-DN STATUS TO DN (1)
0115  1040          TAD M1
0116  3042          DCA POINT
0117  1020          TAD DALTH       /WANT TO COMPARE TO DASH LENGTH
0120  7004          RAL             /MUL. AC BY 2
0121  1020          TAD DALTH       /HAVE 3X DALTH IN AC
0122  7120          CLL CML
0123  7041          CMA IAC         /L = 1; AC = -3(DALTH)
0124  1021          TAD COUNT
0125  7430          SZL             /IS 3X DALTH > COUNT?
0126  5130          JMP TEST        /YES -> TEST AGAIN
0127  5552          JMP I SPAC      /NO -> L = 0 -> PRINT SPACE
0130  7300  TEST,    CLA CLL         /BEGIN TO TEST AGAIN
0131  1020          TAD DALTH
0132  7004          RAL             /2X DALTH
0133  1020          TAD DALTH       /3X DALTH
0134  7100          CLL
0135  7010          RAR
0136  7100          CLL
0137  7010          RAR             /DIVIDE 3X DALTH BY 4
0140  7120          CLL CML         /CONVERT TO 13 BIT ARITHMETIC
0141  7041          CMA IAC         /L = 1; AC = -3/4(DALTH)
0142  1021          TAD COUNT
0143  7430          SZL             /IS 3/4 X DALTH > COUNT?
0144  5146          JMP .+2         /YES -> SPACE BETWEEN DIT & DA
0145  5553          JMP I LET       /NO -> L = 0 -> PRINT LETTER
0146  7200          CLA
0147  3021          DCA COUNT       /SET COUNT TO 0
0150  5050          JMP COI
0151  0600  RCWIC,  600
0152  0500  SPAC,SPACE
0153  0551  LET,LETTER
```

```
                *600
0600    7200            CLA             /RCWIC: ROTATES DATA FROM LINK
0601    3021            DCA COUNT       /INTO CODE WORD;  CLEAR COUNT
0602    1215            TAD CW          /LOAD PREVIOUS CODE PORTION
0603    7004            RAL             /ROTATE STATUS INTO CODE WORD
0604    3215            DCA CW          /DEPOSIT IN CODE PORTION OF WORD
0605    1217            TAD CWA         /LOAD ADDRESS PORTION (AC0-5)
0606    1216            TAD CDCNT       /UPDATE AC0-5
0607    3217            DCA CWA
0610    1215            TAD CW          /LOAD CODE (AC6-11)
0611    0221            AND CWMSK       /MASK OUT AC0-5
0612    1217            TAD CWA         /ADD AC0-5 TO CODE
0613    3220            DCA CWP         /DEPOSIT UPDATED CODE WORD
0614    5050            JMP COI         /INCREMENT COUNT
0615    0000    CW,0
0616    0100    CDCNT,0100
0617    1000    CWA,1000
0620    0000    CWP,0
0621    0077    CWMSK,0077

0622    0000    LETR,   0               /PLACES LAST LETTER SENT,
0623    7200            CLA             /CARR. RETURN, & LINE FEED
0624    1220            TAD CWP         /IN TTY BFFR;LOAD CODE WORD
0625    3256            DCA SCW         /DUMP IN TEMP. LOCATION
0626    1656            TAD I SCW       /LOAD ASCII FOR CODE
0627    3427            DCA I BFFR      /DUMP IN TTY BUFFER
0630    2027            ISZ BFFR
0631    2030            ISZ BFCNT
0632    1036            TAD C1000
0633    3217            DCA CWA         /INITIALIZE CODE WORD ADDRESS
0634    3220            DCA CWP         /INITIALIZE CODE WORD
0635    3215            DCA CW          /INITIALIZE CODE PORTION
0636    3042            DCA POINT       /SET POINTER
0637    1032            TAD C212        /LOAD LINE FEED
0640    3427            DCA I BFFR      /DUMP IN TTY BUFFER
0641    2027            ISZ BFFR
0642    2030            ISZ BFCNT
0643    1031            TAD C215
0644    3427            DCA I BFFR      /CARRIAGE RETURN IN BUFFER
0645    2027            ISZ BFFR
0646    2030            ISZ BFCNT
0647    1032            TAD C212
0650    3427            DCA I BFFR      /LINE FEED IN BUFFER
0651    2027            ISZ BFFR
0652    2030            ISZ BFCNT
0653    1033            TAD CM50
0654    3026            DCA LINCNT      /INITIALIZE LINE COUNT
0655    5622            JMP I LETR      /EXIT
0656    0000    SCW,0
```

```
                *200
0200   7300  START,  CLA CLL              /INITIALIZING PROGRAM
0201   3020          DCA DALTH            /SET DASH LENGTH = 0
0202   3037          DCA TPFLG
0203   7404          OSR
0204   3022          DCA MASK             /SET LOGIC MASK FROM SR
0205   3653          DCA I BCW
0206   3654          DCA I BCWP
0207   1036          TAD C1000
0210   3655          DCA I BCWA
0211   6001          ION
0212   6144          6144                 /CONNECT CLOCK FLAG TO INTER.

0213   7200          CLA                  /BACKGROUND PROGRAM
0214   1030  LOOP,   TAD BFCNT
0215   7550          SPA SNA              /SKIP IF AC > 0
0216   5214          JMP .-2              /WAIT FOR CHAR. IN BUFFER
0217   7200          CLA                  /PRINT OUT FOLLOWS
0220   1035          TAD C170
0221   3041          DCA BFF1             /RESET BUFFER ADDRESS TO 170
0222   1441  ANT,    TAD I BFF1           /LOAD CHARACTER
0223   6046          TLS                  /PRINT CHARACTER
0224   7200          CLA
0225   6002          IOF
0226   1030          TAD BFCNT
0227   1040          TAD M1
0230   3030          DCA BFCNT            /DECREMENT BUFFER COUNT
0231   1030          TAD BFCNT
0232   6001          ION
0233   7440          SZA                  /FINISHED?
0234   5237          JMP CONT             /NO
0235   6002          IOF
0236   5246          JMP INIT             /YES
0237   2041  CONT,   ISZ BFF1            /INC. BUFFER ADDRESS
0240   7200          CLA
0241   3037          DCA TPFLG            /CLEAR TPFLG
0242   1037          TAD TPFLG
0243   7650          SNA CLA
0244   5242          JMP .-2              /WAIT FOR PRINTER FLAG
0245   5222          JMP ANT              /PRINT NEXT CHARACTER
0246   7200  INIT,   CLA
0247   1035          TAD C170
0250   3027          DCA BFFR
0251   6001          ION
0252   5214          JMP LOOP
0253   0615  BCW,CW
0254   0620  BCWP,CWP
0255   0617  BCWA,CWA
```

```
                *400                    /INTERRUPT EXECUTIVE
0400    3023            DCA SAVE        /SAVE MACHINE STATUS
0401    7010            RAR
0402    3024            DCA LINK
0403    6141            6141            /IS IT CLOCK?
0404    7410            SKP             /NO
0405    5217            JMP CLOCK       /YES
0406    6041            6041            /IS IT PRINTER?
0407    7410            SKP             /NO
0410    5236            JMP PRSER       /YES
0411    6031            6031            /IS IT KEYBOARD?
0412    7410            SKP             /NO
0413    5215            JMP KYBS        /YES
0414    7402            HLT             /UNWANTED INTER.
0415    6032    KYBS,   KCC
0416    5641            JMP I RSTART    /KEYBOARD RESTART
0417    6146    CLOCK,  6146            /CLOCK SERVICE
0420    7200            CLA
0421    1025            TAD UPDNS       /PLACE LAST UP-DN STATUS INTO AC
0422    7004            RAL             /ROTATE UP-DN STATUS INTO LINK
0423    7200            CLA
0424    6074            6074            /READ LOGIC TRUNKS
0425    0022            AND MASK        /MASK OUT ALL UNUSED BITS
0426    7430            SZL             /IF LINK=0 -> UP; 1 -> DN
0427    5233            JMP NZL         /LINK = 1
0430    7450            SNA
0431    5070            JMP STORE       /STATUS CHANGED - STORE UP COUNT
0432    5050            JMP COI         /STATUS SAME - INC. COUNT
0433    7450    NZL,    SNA
0434    5050            JMP COI         /STATUS SAME - INC. COUNT
0435    5070            JMP STORE       /STATUS CHANGED - STORE DN COUNT
0436    6042    PRSER,  6042            /PRINTER SERVICE
0437    2037            ISZ TPFLG       /SET PRINTER FLAG
0440    5042            JMP EXIT
0441    0200    RSTART, START
```

```
                *500
0500   2026  SPACE,  ISZ LINCNT        /SPACE & CARRIAGE RETURN DEPOSIT
0501   7000          NOP
0502   7200          CLA               /DEPOSITS LAST LETTER SENT IN
0503   1746          TAD I DCWP        /TTY BFFR BEFORE DEPOSITING
0504   3350          DCA TMCW          /THE SPACE OR THE CARRIAGE
0505   1750          TAD I TMCW        /RETURN & LINE FEED
0506   3427          DCA I BFFR
0507   2027          ISZ BFFR
0510   2030          ISZ BFCNT
0511   1036          TAD C1000
0512   3745          DCA I DCWA
0513   3746          DCA I DCWP
0514   3747          DCA I DCW
0515   1026          TAD LINCNT
0516   7510          SPA
0517   5335          JMP PRSPA
0520   7200          CLA               /DEPOSIT OF THE CARRIAGE RETURN
0521   1031          TAD C215          /& LINE FEED IN TTY BFFR
0522   3427          DCA I BFFR
0523   2027          ISZ BFFR
0524   2030          ISZ BFCNT
0525   1032          TAD C212
0526   3427          DCA I BFFR
0527   2027          ISZ BFFR
0530   2030          ISZ BFCNT
0531   1033          TAD CM50
0532   3026          DCA LINCNT
0533   3021          DCA COUNT
0534   5050          JMP COI
0535   7200  PRSPA,  CLA               /DEPOSIT SPACE IN TTY BFFR
0536   1034          TAD C240
0537   3427          DCA I BFFR
0540   2027          ISZ BFFR
0541   2030          ISZ BFCNT
0542   2026          ISZ LINCNT
0543   3021          DCA COUNT
0544   5050          JMP COI
0545   0617  DCWA,CWA
0546   0620  DCWP,CWP
0547   0615  DCW,CW
0550   0000  TMCW,0

0551   2026  LETTER, ISZ LINCNT        /PLACES LETTER SENT INTO
0552   7000          NOP               /TTY BUFFER
0553   7200          CLA
0554   1771          TAD I ACWP
0555   3373          DCA TCW
0556   1773          TAD I TCW
0557   3427          DCA I BFFR
0560   2027          ISZ BFFR
0561   2030          ISZ BFCNT
```

```
0562  1036          TAD C1000
0563  3770          DCA I ACWA
0564  3771          DCA I ACWP
0565  3772          DCA I ACW
0566  3021          DCA COUNT
0567  5050          JMP COI
0570  0617  ACWA,CWA
0571  0620  ACWP,CWP
0572  0615  ACW,CW
0573  0000  TCW,0

              *1100                    /ASCII CODE LIST
1100  0305  305
1101  0324  324
              *1200
1200  0311  311
1201  0301  301
1202  0316  316
1203  0315  315
              *1300
1300  0323  323
1301  0325  325
1302  0322  322
1303  0327  327
1304  0304  304
1305  0313  313
1306  0307  307
1307  0317  317
              *1400
1400  0310  310
1401  0326  326
1402  0306  306
1403  0000  000
1404  0314  314
1405  0000  000
1406  0320  320
1407  0312  312
1410  0302  302
1411  0330  330
1412  0303  303
1413  0331  331
1414  0332  332
1415  0321  321
              *1500
1500  0265  265
1501  0264  264
1502  0000  000
1503  0263  263
1504  0000  000
1505  0000  000
1506  0000  000
1507  0262  262
1510  0243  243
```

```
1511  0243  243
1512  0336  336
            *1517
1517  0261  261
1520  0266  266
1521  0337  337
1522  0257  257
1523  0243  243
1524  0243  243
1525  0243  243
1526  0246  246
1527  0243  243
            *1530
1530  0267  267
1531  0000  000
1532  0000  000
1533  0000  000
1534  0270  270
1535  0000  000
1536  0271  271
1537  0260  260
            *1600
1600  0252  252
            *1605
1605  0244  244
            *1614
1614  0277  277
            *1625
1625  0256  256
            *1663
1663  0254  254
            *1700
1700  0252  252
            *2000
2000  0252  252
```

ACW	0572
ACWA	0570
ACWP	0571
ANT	0222
BCW	0253
BCWA	0255
BCWP	0254
BFCNT	0030
BFFR	0027
BFF1	0041
CDCNT	0616
CLOCK	0417
CM50	0033
COI	0050
CONT	0237
COUNT	0021
CPARDN	0074
CPARUP	0113
CW	0615
CWA	0617
CWMSK	0621
CWP	0620
C1000	0036
C170	0035
C212	0032
C215	0031
C240	0034
C400	0002
DALTH	0020
DCW	0547
DCWA	0545
DCWP	0546
EXIT	0062
INIT	0246
KYBS	0415
LET	0153
LETR	0622
LETT	0061
LETTER	0551
LINCNT	0026
LINK	0024
LOOP	0214
MASK	0022
M1	0040
NEG	0111
NZL	0433
POINT	0042
PRSER	0436
PRSPA	0535
RCWIC	0151
RSTART	0441
SAVE	0023
SCW	0656
SPAC	0152
SPACE	0500
START	0200
STORE	0070
TCW	0573
TEST	0130
TMCW	0550
TPFLG	0037
UPDNS	0025