
SPOOL
RANDOM.LST
05/04/82
15:53:39

SERIES-III 8086/8087/8088 MACRO ASSEMBLER V1.1 ASSEMBLY OF MODULE RANDOM
 OBJECT MODULE PLACED IN :F1:RANDOM.OBJ
 ASSEMBLER INVOKED BY: ASM86.86 :F1:RANDOM.A86 PRINT(:F1:RANDOM.LST) DEBUG

```

LOC  OBJ          LINE      SOURCE
                                1 +1  $TITLE ('CQ$Random Subroutine for 550          11/20/81 15:00')
                                2      NAME      RANDOM
                                3 +1  $include (:f1:cpyrt.dca)
=1      4      ;
=1      5      ;          /* Intel Corporation Proprietary Information.
=1      6      ;          This listing is supplied under the terms of a
=1      7      ;          license agreement with Intel Corporaton and
=1      8      ;          may not be copied nor disclosed except in
=1      9      ;          accordance with the terms of that agreement. */
=1     10      ;
                                11      ;
                                12      ; This routine is a duplicate of the "Random" call in KAOS, provided
                                13      ; here because there's no room left in ROM for a "cq" version.
                                14
                                15      DGROUP      GROUP      DATA
                                16      DATA      SEGMENT PUBLIC 'DATA'
                                17      DATA      ENDS
                                18
                                19      CGROUP      GROUP      CODE
                                20      CODE      SEGMENT PUBLIC 'CODE'
                                21      ASSUME      CS:CGROUP,DS:DGROUP
                                22
0000  ----          R      23      DGRP      DW      DGROUP
                                24      ;
                                25      ;          Define Post addresses for iSBC 550
                                26      ;
                                27 +1  $INCLUDE (:f1:PORTS.INC)
=1     28
=1     29      ;          CONTOL PORTS
=1     30
0000          =1     31      SET_TXSRT      EQU      000H
0001          =1     32      RESET_TXSRT      EQU      001H
0002          =1     33      SET_RXAV1      EQU      002H
0003          =1     34      SET_RXAV2      EQU      003H
0004          =1     35      SET_RXAV3      EQU      004H
0005          =1     36      RESET_ERROR      EQU      005H
0006          =1     37      RESET_CHANNEL_COUNTER      EQU      006H
00A0          =1     38      SET_SYS      EQU      0A0H
00B0          =1     39      SET_LOC      EQU      0B0H
=1     40
=1     41
=1     42      ;          PIT VALUES
=1     43
00D3          =1     44      PITCMD      EQU      0D3H          ;PIT COMMAND PORT
00D2          =1     45      PIT_BCK      EQU      0D2H          ;PIT BACKOFF TIMER
00D1          =1     46      PIT_RTC      EQU      0D1H          ;PIT REAL-TIME CLOCK PORT
00D0          =1     47      PIT_ALC      EQU      0D0H          ;PIT ALARM-CLOCK PORT
=1     48
0000          =1     49      LATCH_RTC      EQU      00H          ;LATCH REAL-TIME CLOCK VALUE
0040          =1     50      LATCH_ALC      EQU      40H          ;LATCH ALARM-CLOCK VALUE

```

| LOC | OBJ | LINE | SOURCE |
|------|-----|------|---|
| | | =1 | 51 |
| | | =1 | 52 |
| | | =1 | 53 ; DMA VALUES |
| | | =1 | 54 |
| 00C8 | | =1 | 55 DMACMD EQU 0C8H ;DMA COMMAND AND STATUS PORT |
| 00C9 | | =1 | 56 DMAREQ EQU 0C9H ;DMA REQUEST PORT |
| 00CA | | =1 | 57 DMAMSKB EQU 0CAH ;DMA MASK BIT PORT |
| 00CB | | =1 | 58 DMAMODE EQU 0CBH ;DMA MODE PORT |
| 00CC | | =1 | 59 DMABPTR EQU 0CCH ;DMA BYTE POINTER PORT |
| 00CD | | =1 | 60 DMACLR EQU 0CDH ;? (TMP & CLEAR??) |
| 00CF | | =1 | 61 DMAMASK EQU 0CFH ;DMA MASK PORT |
| | | =1 | 62 |
| 00C0 | | =1 | 63 CH0ADDR EQU 0C0H ;CHANNEL 0 ADDRESS PORT |
| 00C1 | | =1 | 64 CH0WC EQU 0C1H ;CHANNEL 0 WORD COUNT PORT |
| 00C2 | | =1 | 65 CH1ADDR EQU 0C2H ;CHANNEL 1 ADDRESS PORT |
| 00C3 | | =1 | 66 CH1WC EQU 0C3H ;CHANNEL 1 WORD COUNT PORT |
| 00C4 | | =1 | 67 CH2ADDR EQU 0C4H ;CHANNEL 2 ADDRESS PORT |
| 00C5 | | =1 | 68 CH2WC EQU 0C5H ;CHANNEL 2 WORD COUNT PORT |
| 00C6 | | =1 | 69 CH3ADDR EQU 0C6H ;CHANNEL 3 ADDRESS PORT |
| 00C7 | | =1 | 70 CH3WC EQU 0C7H ;CHANNEL 3 WORD COUNT PORT |
| | | =1 | 71 |
| | | =1 | 72 |
| | | =1 | 73 ; PIO VALUES |
| | | =1 | 74 |
| 00E3 | | =1 | 75 PIOCMD EQU 0E3H ;PIO COMMAND PORT (CHECK: F3?) |
| 00E0 | | =1 | 76 PIOA EQU 0E0H ;PIO PORT A |
| 00E1 | | =1 | 77 PIOB EQU 0E1H ;PIO PORT B |
| 00E2 | | =1 | 78 PIOC EQU 0E2H ;PIO PORT C |
| | | =1 | 79 |
| 00F3 | | =1 | 80 PIOCCLR EQU 11110011B ;CLEAR SERDES |
| 002A | | =1 | 81 PIOSEN EQU 00101010B ;SERIAL ENABLE |
| 0022 | | =1 | 82 PIOSENP EQU 00100010B ;PROMISCUOUS SERIAL ENABLE |
| 00E0 | | =1 | 83 PIOREAD EQU 11100000B ;READ ADDRESS COMMAND |
| | | =1 | 84 |
| | | =1 | 85 |
| | | =1 | 86 ; PIC VALUES |
| | | =1 | 87 |
| 00F0 | | =1 | 88 PICCMD EQU 0F0H ;PIC COMMAND PORT |
| 00F0 | | =1 | 89 PICDATA EQU 0F0H ;PIC DATA PORT |
| 00F1 | | =1 | 90 PICMASK EQU 0F1H ;PIC MASK PORT |
| | | =1 | 91 |
| 0020 | | =1 | 92 EOI_PIC EQU 20H ;PIC END-OF-INTERRUPT COMMAND |
| 0060 | | =1 | 93 SEOI_PIC EQU 60H ;PIC SELECTIVE EOI COMMAND |
| 000C | | =1 | 94 POLL_PIC EQU 0CH ;POLL PIC COMMAND |
| 000A | | =1 | 95 READ_IRR EQU 0AH ;PIC READ-IRR COMMAND |
| 0040 | | =1 | 96 RTC_DONE EQU 40H ;MASK FOR REAL-TIME CLOCK INTERRUPT |
| 0066 | | =1 | 97 RTC_INT_SEOI EQU 60H+6 ;COMMAND TO EOI RTC INTERRUPT |
| 0086 | | =1 | 98 RTC_INT EQU 80H+6 ;POLL COMMAND RETURN IF INTERRUPT |
| | | =1 | 99 |
| 0001 | | =1 | 100 CH1_DONE EQU 01H ;RX CHANNEL 1 DONE |
| 0002 | | =1 | 101 CH2_DONE EQU 02H ;RX CHANNEL 2 DONE |
| 0004 | | =1 | 102 CH3_DONE EQU 04H ;RX CHANNEL 3 DONE |
| | | =1 | 103 |
| | | =1 | 104 |
| | | =1 | 105 ; MISC. DEFINITIONS |

```

LOC  OBJ          LINE      SOURCE
                                =1  106
0001          =1  107      MCFLAG          EQU      01H          ;MULTICAST BIT
0007          =1  108      JUNK_BYTES      EQU      7           ;NUMBER OF RECIEVE JUNK BYTES
                                109
                                110      ;;;      RANDOM - RETURN "RANDOM" NUMBER.
                                111      ;
                                112      ;           THIS ROUTINE RETURNS THE VALUE OF THE REAL-TIME
                                113      ;           CLOCK TIMER IN THE 8253
                                114      ;
                                115      ;           RETURNS:
                                116      ;           AX = TIMER VALUE.
                                117
                                118
                                119      PUBLIC  RANDOM
0002  9C          120      RANDOM:  PUSHF          ;SAVE INTERRUPT FLAG
0003  FA          121          CLI           ;CLEAR INTERRUPTS
0004  B000        122          MOV          AL,LATCH_RTC
0006  E6D3        123          OUT          PITCMD,AL      ;LATCH UP COUNT
0008  E4D1        124          IN           AL,PIT_RTC      ;READ REAL-TIME CLOCK
000A  8AE0        125          MOV          AH,AL
000C  E4D1        126          IN           AL,PIT_RTC
000E  86E0        127          XCHG         AH,AL
0010  9D          128          POPF
0011  C3          129          RET
                                130
-----          131      CODE   ENDS
                                132      END

```

ASSEMBLY COMPLETE, NO ERRORS FOUND