

The following program waits until a SW connected to PT0 goes low, and then it flashes an LED connected to PT7 at a 1 second rate. The number of flashes is maintained in 16-bit RAM location "FlashCount".

```

5408      7                                ORG $800
5409      8  a000800      FlashCount DS.W 1      ; Allocate 2 bytes
                                           ; for 16-bit variable

5410      9                                ORG $4000
5411     10                                Entry:
5412     11  a004000 CF10 00                LDS $1000 ; initialize stack pointer
5413     12  a004003 1C02 4280              BSET DDRT,%10000000 ; Make PT7 output
5414     13  a004007 1D02 4201              BCLR DDRT, 1 ; Make PT0 an input
5415     14  a00400B CC00 00                LDD #0
5416     15  a00400E 7C08 00                STD FlashCount ; Zero 16-bit counter
5417     16  a004011 B602 40      WT_PT0_LOW: LDAA PTT ; Wait for SW on PT0
5418     17  a004014 8401 F9             ANDA #00000001 ; to be pressed
5419     18  a004016 26 F9             BNE WT_PT0_LOW
5420     19                                FLASH_AGAIN:
5421     20  a004018 1D02 4080 4018      BCLR PTT, %10000000 PT7
5422     21  a00401C 1640 2F 4018      JSR DELAY_1SEC ; Wait about 1 sec
5423     22  a00401F 1C02 4080 4018      BSET PTT,%10000000 ; Raise PT7
5424     23  a004023 1640 2F 402F      JSR DELAY_1SEC ; Wait about 1 sec
5425     24  a004026 FE08 00                LDX FlashCount
5426     25  a004029 08 FFE9             INX
5427     26  a00402A 7E08 00 STX FlashCount ; increment 16-bit cntnr
5428     27  a00402D 20 E9             BRA FLASH_AGAIN
5429     28
5430     29  a00402F 34      DELAY_1SEC: PSHX
5431     30  a004030 35                PSHY
5432     31  a004031 CE00 20                LDX #$20
5433     32  a004034 CDFE FF      OUTERLOOP: LDY #FFFF
5434     33  a004037 03 Fd      INNERLOOP: DEY
5435     34  a004038 26 Fd      BNE INNERLOOP
5436     35  a00403A 09 F7      DEX
5437     36  a00403B 26 F7      BNE OUTERLOOP
5438     37  a00403D 31                PULY
5439     38  a00403E 30                RTS
5440     39  a00403F 3D                RTS
    
```

- Fill in the EIGHT missing blanks in the assembly code so that this program works as described above.
- Fill in the FOUR 8-bit relative displacement blanks in the conditional branch instructions.
- What is the contents of stack pointer register "S"...

- (a) Just BEFORE the PSHX instruction at Line 5430 executes? S = \$ OFFE
- (b) Just BEFORE the PULY instruction at Line 5438 executes? S = \$ OFFA
- (c) Just BEFORE the RTS at line 5440 executes? S = \$ OFFE
- (d) Just AFTER the RTS instruction at line 5440 executes? S = \$ 1000
- (e) How many bytes are on the stack when it is at its largest? 6 Bytes

(f) Assuming that the JSR instruction at line 5424 has just executed, fill in the contents of the following RAM locations

\$0FFF = #26      \$0FFE = \$40 ← "Always 'Big Endian' high byte goes into lower addr"