

MOP Demo 2

Processbyte och assemblerprogrammering

14.5 (Blå Bok) Exekvera två program "samtidigt".

11.28 Skriv en subrutin som omvandlar en NBCD-kodad siffra till motsvarande "segmentmönster" för en 7-segments display enligt följande:

0 FCH 2 DAH 4 66H 6 BEH 8 FEH 1 60H 3 F2H 5 B6H 7 E0H 9 F6H

Segmentmönstren skall finnas i en tabell som börjar på adress SEGTAB.

Indata: A = NBCD-siffra Utdata: A = segmentmönster

X.8

Skriv en subrutin HEXASC i assemblerspråk som översätter det binära talet (b3b2b1b0)2 i register A till motsvarande 7-bitars ASCII-tecken. Innehållet i b7b6b5b4 i register A är okänt. Vid återhopp skall ASCII-tecknet finnas i register A med b7 nollställd. Endast register A och register CC får vara förändrade vid återhopp från subrutinen.

Lösning: X.8

HEXASC PSHX

ANDA #\$0F Nollställ bit 4-7

LDX #ASCTAB Sätt pekare till tabell med alla 16 ASCII-tecken 0-F

LDA A,X Hämta ASCII-tecken från tabell

PULX

RTS

*

ASCTAB FCB \$30,\$31,\$32,\$33,\$34,\$35,\$36,\$37 ASCII för 0-7

FCB \$38,\$39,\$41,\$42,\$43,\$44,\$45,\$46 ASCII för 8-F

File: test 1128.lst

0000 0400	1. inport equ	\$400
0000 0401	2. outport equ	\$401
0000 2B00	3. BOS equ	\$2b00
001000	4. org	\$1000
001000 CF 2B 00	5. lds	#BOS
001003	6.	
001003 B6 04 00	7. loop	ldaa inport
001006 16 10 0E	8. jsr	Hex2Asc
001009 7A 04 01	9. staa	outport
00100C 20 F5	10. bra	loop
	11.	
00100E 34	12. Hex2Asc	pshx
00100F 84 0F	13. anda	#\$0f nollställ B7-B4, indata i B3-B0
001011 CE 10 18	14.	ldx #AsciiTab
001014	15.	ldaa a,x
001016 30	16.	pulx
001017 3D	17.	rts
001018	18.	
001018 31 32 33 34 35 36	19. AsciiTab	fcs "123456789ABCDEF"
37 38 39 41 42 43		
44 45 46		

X.11

I ett datorsystem med processorn CPU12 behövs en subrutin, BLKASC, som omvandlar ett block med 8-bitars dataord i minnet till en nollterminerad (= som avslutas med ett byte med värdet 0) sträng med ASCII-tecknen för hexadecimala siffror. Omvandlingsprincipen framgår av följande exempel:

Minnesblocket 10110101, 01111000, 01100010, 10100011.

ger ASCII-strängen 42H, 35H, 37H, 38H, 36H, 32H, 41H, 33H, 0

Skriv subrutinen BLKASC. Vid anrop av subrutinen skall startadressen till minnesblocket resp. ASCII-strängen finnas i X-registret resp. Y-registret och antalet minnesord i A-registret. Endast flaggregistret får vara förändrat vid återhopp.

Veckans problem:

Bygg en realtidsklocka med en upplösning av 100 ms, använd en "avbrottsklocka" på ca 400 Hz som finns på labbsystemet. Ersätt "Delay"-rutinen med en "Hold"-rutin som har samma anropssätt som Delay. I Hold läses starttiden från realtidsklockan, beräkna stopptiden och vänta tills stopptiden är uppnådd, i stället för att vänta, gör ett "processbyte" mha SWI!

Prova "klockan" i simulator och hårdvaran i labbet!