B. DELLSTRAND

TENTAMEN

Datorteknik

D1/E1/Mek1/Ö1

10-05-28
0900 - 1300

Hjälpmedel: Häfte "ARM-instruktioner", A4-format, 17 sidor

Maxpoäng:	**60p**
Betyg 3	**24p**
Betyg 4	**36p**
Betyg 5	**48p**

Frågor under tentamen:

Börje Dellstrand **tel. 16 71 22**
alt. 0702-98 63 58

Bilaga (som skall återlämnas): **Port0 control register**
Interruptregister

1. **(2p+2p+2p)**

a) Flyttalet X lagras som 1 1000 0100 000 1101 0000 0000 0000 0000.
Bit 31 är längst till vänster. Ange värdet av X i decimal form.

b) Hur lagras talet Y = 17,625 i flyttalsformatet enligt ovan?

Ledning: Flyttalet är kodat i IEEE-standarden för enkel precision. För denna gäller

$(-1)^S \cdot 1.M \cdot 2^{E-127}$ där bitarna fördelar sig enligt nedan.

S	E	M
1 bit	8 bitar	23 bitar

c) Hur lagras talen +50 och -50 som 8-bitars heltal?
2-komplementsrepresentation används för negativa tal.

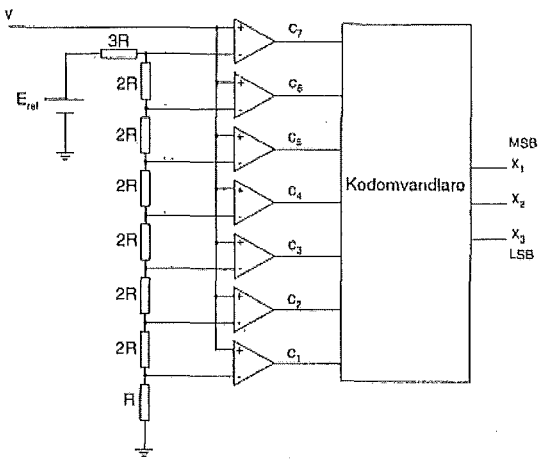
2. **(5p)**

(0,5p/rätt, felaktigt alternativ = -0,5p. Dock lägst noll poäng. Max 5p)
Kombinera en term i den vänstra kolumnen med tillhörande fras(er) i den högra kolumnen.

1. Assemblator	A. asynkron seriell överföring
2. RISC	B. synkroniseringsmetod
3. Refresh	C. datariktningsregister
4. Baud	D. översättningsprogram
5. Random accessminne	E. alfanumerisk kod
6. Programräknare	F. förenklad instruktionsuppsättning
7. PROM	G. alltid samma åtkomsttid
8. Paritetsbit	H. krävs för dynamiska minnen
9. ASCII	I. enhet för signaleringshastighet
10. PCON	J. generella register
	K. pekar på nästa instruktion
	L. kan ej raderas
	M. här sker beräkningar
	N. 4 bitar

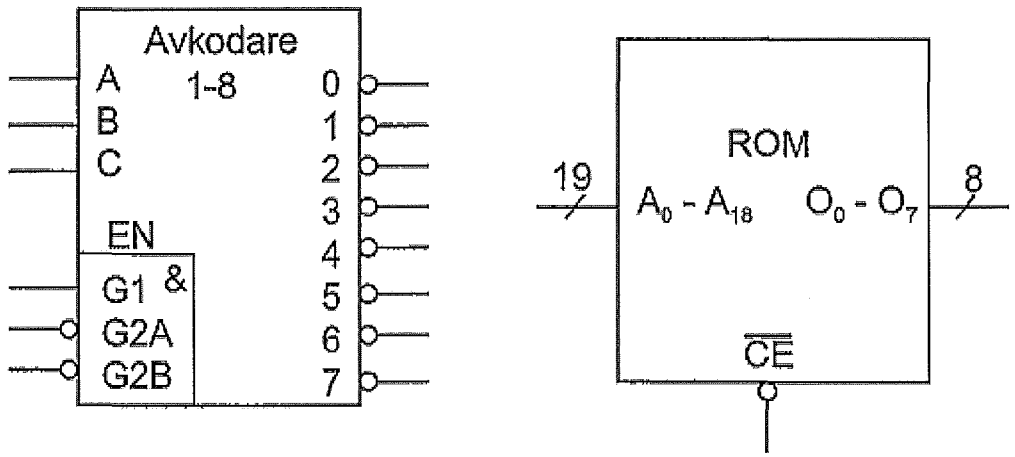
3. **(4p)**

Figuren visar en A/D-omvandlare.
Ange namn, fördelar och nackdelar.
Beskriv också utgående från figuren hur omvandling sker.



4. **(4p)**

I ett minnessystem ingår läsminne och läs/skrivminne. Läsminnet ska byggas upp med minneskapslar 4M (512k x 8), se symbol nedan. Minnessystemet adresseras med 24 adressbitar A0 – A23. Läsminnet ska ha kapaciteten 1 Mord x 24 bitar och vara placerat i adressområdet 0x600000 till 0x6FFFFF. Chipselect till läsminnet ska genereras med en AVK 8-1 (74LS138), till vilken ska anslutas adressbitarna A19-A23 så att den kan generera chip-select i hela adressområdet 0x400000-0x7FFFFF. Rita schema.



5. **(3p)**

7-bitars ASCII-kod används för att sända data via ett RS-232 gränssnitt (seriell kommunikation). Följande gäller: 19200 bit/s, 7 databitar, 1 startbit, 1 stoppbit och en paritetsbit (udda paritet).

a) Hur många ASCII-tecken kan överföras per sekund?

b) Beskriv hur överföringen ser ut bitvis. Använd tecknet ”a”, ASCII-kod 0x61, i din beskrivning. Rita ett tidsdiagram och förklara.

6. **(6p)**

Skriv ett väl kommenterat program i ARM-assembler som ger rinnande ljus på Port0 (lysdioderna ska kopplas till Port0, 8 bitar) enl. ex. nedan. Programmet ska vara komplett med portinitiering. En subrutin DELAY med lämplig tidsfördröjning finns tillgänglig (behöver ej skrivas).

Ex. Rinnande ljus med 8 bitar: 00000000, 10000000, 11000000, 11100000, 11110000, 11111000, 11111100, 1111110, 11111111, 01111111, 00111111, 00011111,00001111, 00000111, 00000011, 00000001, 00000000, . . .

7. **(5p)**

Anta att vi har ett ARM-kort där Port0 är ansluten till 8 insignaler (knappar där en nedtryckt knapp ger en etta). Förutom Port0 har vi en minnescell VISA (8 bitar).

```
MAIN    LDR     R0, =rPDAT0
        LDRb    R1, [R0]
        ANDS    R1, R1, #0X70
        EORS    R1, R1, #0X70
        BEQ     ON
        LDR     R1, =0X0
        B       SLUT
ON      MOV     R1, #0X20
SLUT    STRb    R1, VISA
        B       MAIN

VISA    DS8     1
```

Anta att ovanstående program körs om och om igen.
Här kommer fem påståenden om vad som händer då. Ange för varje påstående, vilket som är sant och vilket som är falskt!
(1p/rätt, felaktigt alternativ = -1p. Dock lägst noll poäng. Max 5p)

1. Om alla knappar trycks ner, så kommer en bit i VISA att vara 1-ställd.
2. Om man trycker ner flera än tre knappar, så kommer alltid en bit i VISA att vara 1-ställd.
3. Man kan aldrig få mer än en bit i VISA att vara 1-ställd.
4. Det finns inget sätt att 1-ställa fler än två bitar i VISA.
5. Om man inte trycker på någon knapp, så 1-ställs ändå en bit i VISA.

8. **(2p)**

a) Hur många byte programminne tas i anspråk av programmet i uppgift 7? Motivera!
b) På vilken adress (hexadecimal form) hamnar resultatet i uppgift 7 om programmets startadress är 0x00001000? Motivera!

9. **(4p)**

Skriv en väl kommenterad avbrottshanterare (handler) för ARM, som tar hand om avbrott från EINT0 (knappavbrott, bit0) samt från Timer0 (bit8). Anta att avbrottskoden för knapp-avbrottet kallas ALARM och den för Timer0 kallas TIMER (koden för subrutinerna ALARM och TIMER behöver ej skrivas). Lokala variabler sparas undan.
Avbrottshanteraren ska:

- Läsa av vilket avbrott som skett samt anropa rätt avbrottskod
- Kvittera rätt avbrott
- Återgå från avbrottet

10. **(4p)**

Följande program ska summera 10 siffror lagrade i en sträng TEXT samt lagra summan på platsen SVAR. Koden innehåller ett antal fel. Ange var i koden felet är och korrigera alternativt lägg till saknade delar. **Ledning**: Se bilaga ASCII-tabell.

```
Rad1:   RAM_START EQU   0X00001000

Rad2:             ORG   RAM_START
Rad3:             LDR   R2, TEXT
Rad4:             MOV   R1, #10

Rad5:   LOOP      SUB   R1, R1, #1
Rad6:             CMP   R1, #0
Rad7:             BEQ   FINISH
Rad8:             LDR   R0, [R2]
Rad9:             ADD   R4, R4, R0
Rad10:            ADD   R2, R2, #1
Rad11:            B     LOOP

Rad12:  FINISH    STRh  R4, [R3]
Rad13:  QUIT      B     QUIT

Rad14:  TEXT      DC8   "13128571983758134612846182746"
Rad15:  SVAR      DS16  1
```

11. **(3p+4p+3p)**

I denna uppgift ska du skriva välkommenterad assemblerkod för ARM.
Lokala variabler sparas undan.

a) Skriv en subrutin bitset, som ettställer en angiven bit i en operand. Vid anrop av subrutinen finns inparametrar i R4 och R5. Register R4 innehåller den operand som ska bearbetas, och register R5 innehåller index (0-31) för den bit som ska ettställas. Den minst signifikanta biten har index noll. Det nya värdet ska vid returen från subrutinen återfinnas i register R2.

b) Skriv i välkommenterad assemblerkod subrutin rightmostzero, som undersöker en angiven operand. Vid anrop av subrutinen finns operanden i register R4. Vid returen ska register R2 innehålla index för den minst signifikanta nollan i operanden. Den allra minst signifikanta biten har index noll.
Om alla bitar i operanden är ettställda ska värdet 0XFFFF FFFF returneras i register R2.

c) Skriv i välkommenterad assemblerkod en subrutin lsb_set, som ettställer den minst signifikanta nollan i det 32-bitars tal som finns i den minnescell vars adress finns i register R4. Du **måste** använda subrutinerna ovan.
Vid retur ska register R2 innehålla index för den nolla som har ettställts, eller värdet 0XFFFF FFFF om ingen nolla fanns att nollställa.

12. **(7p)**

Pelle behöver en sorteringsmaskin för LEGO-bitar. Du ska i den här uppgiften programmera den ingående styrenheten, som är baserad på ARM-processorn.
LEGO-bitarna kommer på ett transportband, som passerar förbi tre fotoceller P1, P2 och P3 placerade med 5 cm mellanrum (se fig.).
Tre olika typer av LEGO-bitar förekommer: Små bitar med längden 4 cm, mellanstora bitar med längden 8 cm samt stora med längden 12 cm.
De små bitarna ska knuffas av bandet och ner i en låda, så fort de har passerat P2. Detta sker genom att styrsignalen KNUFF1 aktiveras.
De mellanstora bitarna ska på samma sätt knuffas av bandet, när de har passerat P3, (styrsignal KNUFF2). De stora bitarna ska fortsätta framåt på bandet.
Avståndet mellan LEGO-bitarna är alltid minst 10 cm.
Insignaler till styrenheten: P1 (P0:2), P2 (P0:1) och P3 (P0:0) (aktivt höga).
Utsignaler från styrenheten: KNUFF1 (P0:7) och KNUFF2 (P0:6) (aktivt höga)
Uppgift: Skriv ett komplett, väl kommenterat, ARM-program för styrningen.