

Uppgift D1 **10 poäng**

D1.1 En godtycklig boolesk funktion kan alltid realiseras med (1p)

- a) enbart ELLER-grindar
- b) enbart NOR-grindar
- c) enbart XOR-grindar

D1.2 Funktionen $f = x' \cdot y' + x \cdot z$ kan skrivas som (1p)

- a) $(x + y') \cdot (x' + z)$
- b) $x \cdot z + y' \cdot x' + y \cdot z'$
- c) $x' \cdot y + x \cdot z'$
- d) något annat

D1.3 SP-formen för en boolesk funktion kan i komprimerad form (1p)

skrivas som $A = \Sigma(6, 7, 14, 15)$. De ingående variablerna kallas x y z w. x är mest signifikant, w minst signifikant. Funktionen kan skrivas som:

- a) $x' \cdot z$
- b) $y \cdot z$
- c) $x \cdot z$
- d) $y' \cdot w'$

D1.4 Man använder 5 bitar för att bilda ett binärt 2-komplementtal. (1p)

Talet betecknas med R. Talområdet blir då . . .

- a) $0 \leq R \leq 32$
- b) $-16 \leq R \leq 15$
- c) $-17 \leq R \leq 16$
- d) $-32 \leq R \leq 0$

D1.5 Vilket av de hexadecimala talen nedan är den negativa mot- (1p)

svarigheten till det positiva talet 77_{16} ?

- a) 88
- b) 89
- c) 87
- d) 8A

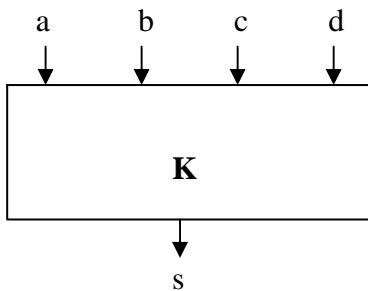
D1.6 Realisera med NAND-grindar booleska funktionen (5p)

$$f(w, x, y, z) = \Sigma(3, 5, 7, 11) + d(4, 6, 15)$$

Insignalernas inverser antas tillgängliga.

Uppgift D2

10 poäng



| a | b | c | d | s |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Ett kombinatoriskt nät K har fyra ingångar <a, b, c, d> och en utgång s.
s = 1 om i mönstret <a b c d> finns minst två ettor intill varandra.

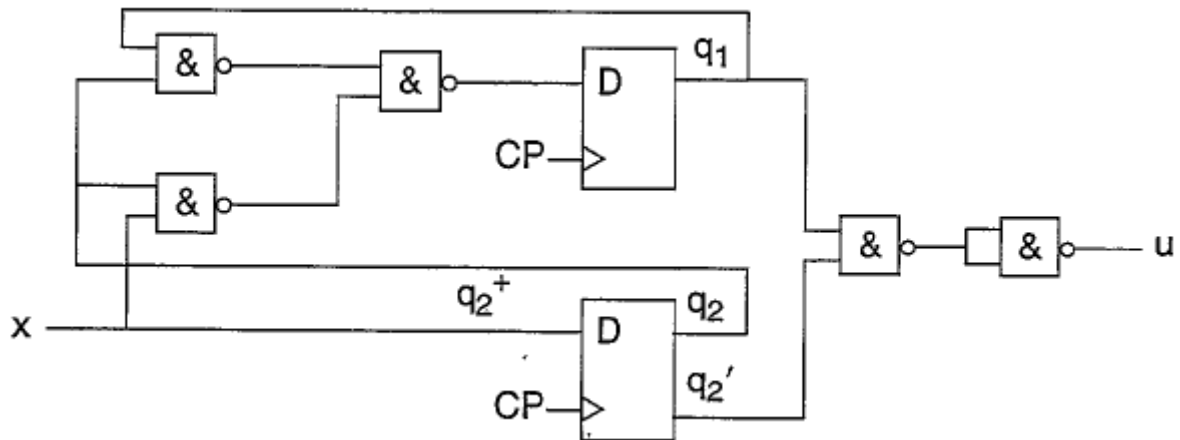
- Realisera med a) NAND-grindar
b) en 8/1 MUX plus ev. en inv.
c) 2 st 3/8 AVK (74LS138) plus en valfri grind.

Insignalernas inverser antas tillgängliga.

Uppgift D3

5 poäng

Betendet hos nedanstående synkrona sekvensnät skall bestämmas. Ta fram tillståndstabell och rita tillståndsgraf. Är sekvensnätet av typen Mealy eller Moore? q₁ mest signifikant.



Uppgift D4

10 poäng

Konstruera en synkron sekvenskrets som svarar mot följande beskrivning:
Kretsen skall ha **en** ingång x och **en** utgång u. Insignalen består av en serie ettor och nollor i godtycklig ordning. Utsignalen u skall vara 1 om de senaste tre mottagna värdena på insignalen varit ettor. I övriga fall skall den vara 0.

Ex. x 0 1 1 0 0 1 0 1 1 1 0 0 1 1 1 1 0 1 0 ...
 u 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0 0 ...

Använd JK-vippor och valfria grindar

Uppgift D5 **5 poäng**

Konstruera en dekadräknare (räknaren ska räkna 0-9) med hjälp av en räknare 74LS163 och en valfri grind. Räknaren ska kunna startas/stoppas med en insignal x. Rita ett fullständigt schema!

Uppgift M6 **5 poäng**

För varje begrepp i den vänstra kolumnen, ange det begrepp eller påstående i den högra, som det passar bäst ihop med.

- | | |
|---------------------|---|
| 1. Stack | A. Von Neuman |
| 2. RAS | B. Här sker beräkningar |
| 3. Assembler | C. Elektriskt raderbart |
| 4. Carry-flaggan, C | D. Översättningsprogram |
| 5. CISC | E. Pekar på nästa instruktion |
| 6. Startbit | F. Påverkas vid ett subrutinanrop |
| 7. TRIS | G. Dynamiskt RAM |
| 8. EEPROM | H. Klarsignal från minnet |
| 9. Programräknare | I. Ger snabb åtkomsttid |
| 10. ALU | J. Harward |
| | K. Påverkas ofta av aritmetiska operationer |
| | L. Datariktningregister |
| | M. Ej skrivbart |
| | N. Asynkron seriell överföring |

Uppgift M7 **3 poäng**

En microcontroller kan byggas upp på olika sätt.

Vad innebär det att en microcontroller är uppbyggd enligt Harward-arkitektur resp. von Neuman-arkitektur ?

Förklara kortfattat skillnaden.

Uppgift M8 **3 poäng**

Förklara begreppen (gärna med figurer):

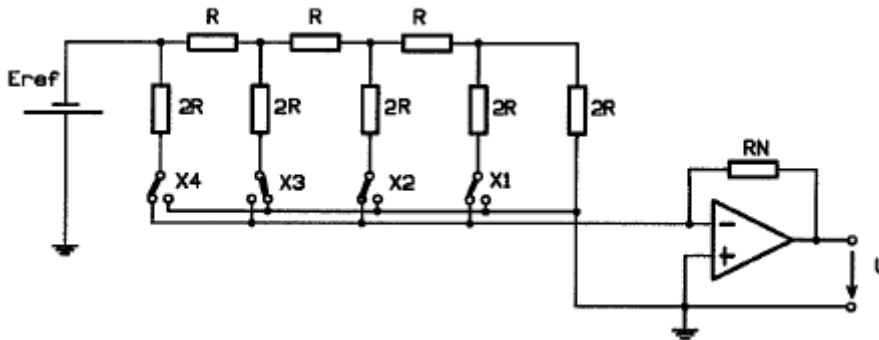
- stack
- PUSH
- POP

Vilken funktion har stacken vid subrutinanrop (CALL) respektive subrutinåterhopp (RETURN) ?

Uppgift M9 2 poäng

I kopplingen i figuren nedan har $R=10\text{ kohm}$ och $R_N=20\text{ kohm}$. $E_{ref}=5\text{ V}$.

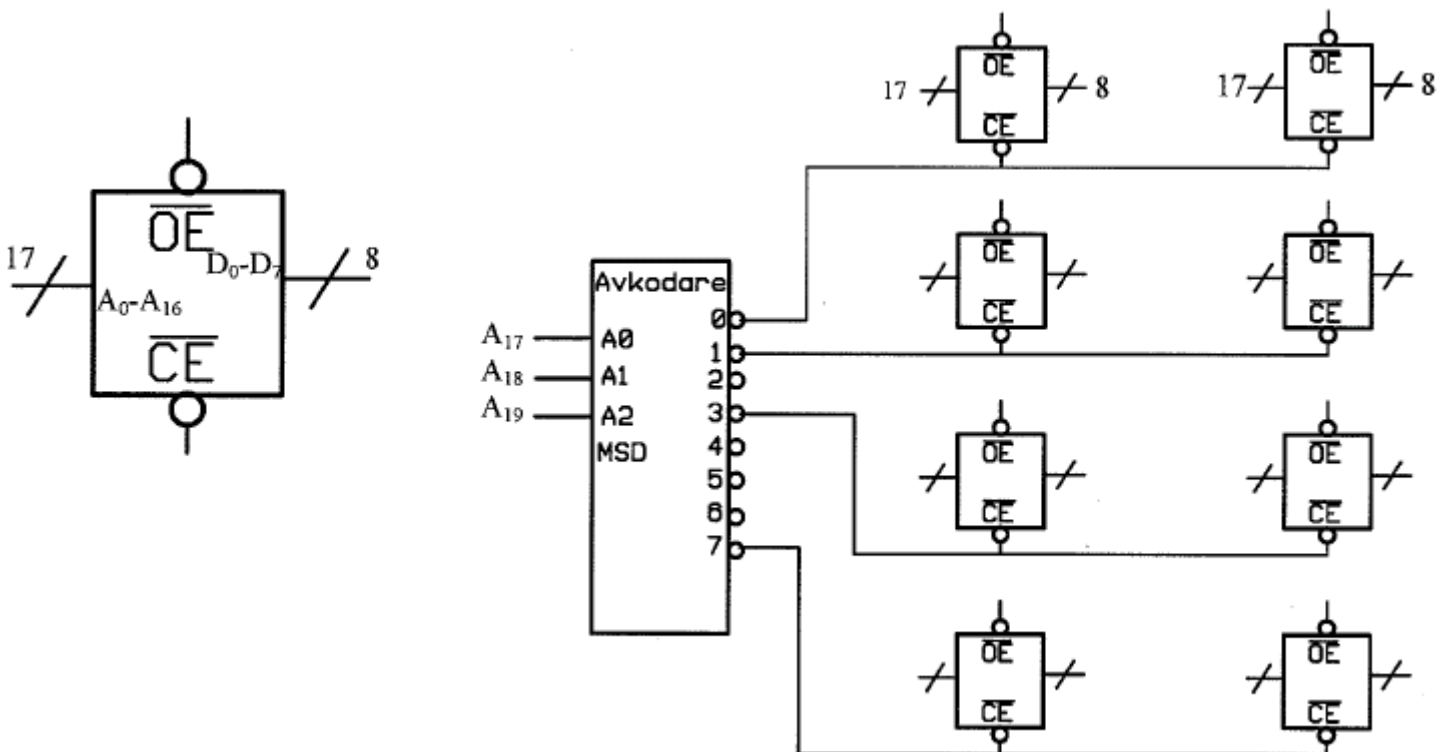
- a) Vad används denna koppling till?
- b) Ange upplösningen (i V). Det räcker att sätta upp sambandet



Uppgift M10 5 poäng

I minnessystemet nedan till höger utgörs minneskapslarna av läsminnen enligt figuren till vänster. Varje minneskapsel har 1M uppbyggt som $128k \cdot 8\text{ bitar}$. Adressledningarna A_0-A_{16} går till varje minneskapsel. Varje kapsel har 8 dataledningar

- a) Ange minnessystemets storlek, det vill säga, ordlängd och antal ord. (2p)
- b) Ange adressområdet i hexkod för varje minneskapsel i minneskonstruktionen nedan till höger. (3p)



Uppgift M11 **3 poäng**

Vilka väsentliga delar innehåller en dators centralenhet, CPU:n? Beskriv de ingående delarnas funktion och samverkan.

Uppgift M12 **4 poäng**

Nedan visas en subrutin skriven för PIC 16F877A. Analysera rutinen och fyll i tabellen. Flaggornas status och arbetsregistrets innehåll skall anges efter att återhopp utförts. (Se bilaga!)

```
PICSUB    CLRF    TEMP
          MOVLW   3
          ADDLW   2
          ADDWF   TEMP,1
LOOP      DECFSZ  TEMP,1
          GOTO    LOOP
          BSF     TEMP,0
          DECF    TEMP,1
          RETURN
```

| Efterfrågad uppgift | PIC |
|---------------------------|------------------------------|
| Programminne som åtgår | Word |
| Exekveringstid | Cykler |
| Arbetsregistrets innehåll | W = |
| Flaggors status | Z = C = |

Uppgift M13 **9 poäng**

Skriv en instruktionssekvens i PIC-assembler som:

- Anropar subrutinen OUTX 8 gånger. Använd en räknare för att hålla reda på antalet gånger (filregister ANTAL på adress 0x21).
- Dividerar talet som finns på adress 0x20 med 4 och skriver tillbaka resultatet till samma adress. Talet är på binär form.
- Undersöker det 2-komplementtal (teckensatt) som finns på adress 0x30 och om talet är negativt så skall talet 2-komplementeras och läggas tillbaka på adress 0x30.

Uppgift M14 **6 poäng**

Ett övergångsställe är försett med ljussignaler. Normalläget är gult blinkande ljus, kopplat till RB6 (PortB bit 6). För att få fast grönt ljus, kopplat till RB7, påverkas en switch, kopplad till RB0. Skriv ett **komplett** PIC-program, väl kommenterat, som använder **RB0-interrupt** för grönt ljus. Två fördröjningsrutiner finns givna (behöver ej skrivas), WAITSHORT, för det gula blinkande ljuset och WAITLONG, för det fasta gröna ljuset.

Följande delar ska ingå:

- Initiering
- Huvudprogram
- Avbrottshanterare (handler)

TABLE 13-2: PIC16F87X INSTRUCTION SET

| Mnemonic, Operands | Description | Cycles | 14-Bit Opcode | | Status Affected | Notes |
|---|------------------------------|--------|---------------|----------------|--------------------|-------|
| | | | MSb | LSb | | |
| BYTE-ORIENTED FILE REGISTER OPERATIONS | | | | | | |
| ADDWF | Add W and f | 1 | 00 | 0111 dfff ffff | C,DC,Z | 1,2 |
| ANDWF | AND W with f | 1 | 00 | 0101 dfff ffff | Z | 1,2 |
| CLRF | Clear f | 1 | 00 | 0001 1fff ffff | Z | 2 |
| CLRWF | Clear W | 1 | 00 | 0001 0xxx xxxx | Z | |
| COMF | Complement f | 1 | 00 | 1001 dfff ffff | Z | 1,2 |
| DECF | Decrement f | 1 | 00 | 0011 dfff ffff | Z | 1,2 |
| DECFSZ | Decrement f, Skip if 0 | 1(2) | 00 | 1011 dfff ffff | | 1,2,3 |
| INCF | Increment f | 1 | 00 | 1010 dfff ffff | Z | 1,2 |
| INCFSZ | Increment f, Skip if 0 | 1(2) | 00 | 1111 dfff ffff | | 1,2,3 |
| IORWF | Inclusive OR W with f | 1 | 00 | 0100 dfff ffff | Z | 1,2 |
| MOVF | Move f | 1 | 00 | 1000 dfff ffff | Z | 1,2 |
| MOVWF | Move W to f | 1 | 00 | 0000 1fff ffff | | |
| NOP | No Operation | 1 | 00 | 0000 0xxx 0000 | | |
| RLF | Rotate Left f through Carry | 1 | 00 | 1101 dfff ffff | C | 1,2 |
| RRF | Rotate Right f through Carry | 1 | 00 | 1100 dfff ffff | C | 1,2 |
| SUBWF | Subtract W from f | 1 | 00 | 0010 dfff ffff | C,DC,Z | 1,2 |
| SWAPF | Swap nibbles in f | 1 | 00 | 1110 dfff ffff | | 1,2 |
| XORWF | Exclusive OR W with f | 1 | 00 | 0110 dfff ffff | Z | 1,2 |
| BIT-ORIENTED FILE REGISTER OPERATIONS | | | | | | |
| BCF | Bit Clear f | 1 | 01 | 00bb bfff ffff | | 1,2 |
| BSF | Bit Set f | 1 | 01 | 01bb bfff ffff | | 1,2 |
| BTFSB | Bit Test f, Skip if Clear | 1 (2) | 01 | 10bb bfff ffff | | 3 |
| BTFSB | Bit Test f, Skip if Set | 1 (2) | 01 | 11bb bfff ffff | | 3 |
| LITERAL AND CONTROL OPERATIONS | | | | | | |
| ADDLW | Add literal and W | 1 | 11 | 111x kkkk kkkk | C,DC,Z | |
| ANDLW | AND literal with W | 1 | 11 | 1001 kkkk kkkk | Z | |
| CALL | Call subroutine | 2 | 10 | 0kkk kkkk kkkk | | |
| CLRWDT | Clear Watchdog Timer | 1 | 00 | 0000 0110 0100 | | TO,PS |
| GOTO | Go to address | 2 | 10 | 1kkk kkkk kkkk | | |
| IORLW | Inclusive OR literal with W | 1 | 11 | 1000 kkkk kkkk | Z | |
| MOVLW | Move literal to W | 1 | 11 | 00xx kkkk kkkk | | |
| RETFIE | Return from interrupt | 2 | 00 | 0000 0000 1001 | | |
| RETLW | Return with literal in W | 2 | 11 | 01xx kkkk kkkk | | |
| RETURN | Return from Subroutine | 2 | 00 | 0000 0000 1000 | | |
| SLEEP | Go into standby mode | 1 | 00 | 0000 0110 0011 | | TO,PS |
| SUBLW | Subtract W from literal | 1 | 11 | 110x kkkk kkkk | C,DC,Z | |
| XORLW | Exclusive OR literal with W | 1 | 11 | 1010 kkkk kkkk | Z | |

REGISTER 2-3: INTCON REGISTER (ADDRESS 0Bh, 8Bh, 10Bh, 18Bh)

| | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-x |
| GIE | PEIE | TOIE | INTE | RBIE | TOIF | INTF | RBIF |
| bit 7 | | | | | | | bit 0 |

- bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts
- bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
- bit 5 **TOIE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt
- bit 4 **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt
- bit 3 **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt
- bit 2 **TOIF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow
- bit 1 **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur
- bit 0 **RBIF:** RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state; a mismatch condition will continue to set the bit. Reading PORTB will end the mismatch condition and allow the bit to be cleared (must be cleared in software).
0 = None of the RB7:RB4 pins have changed state

Status registret (adress 03h, 83h)

| | | | | | | | |
|------|-------|-------|-------|-------|-------|-------|-------|
| bit7 | bit 6 | bit 5 | bit 4 | bit 3 | bit 2 | bit 1 | bit 0 |
| IRP | RP1 | RP0 | TO' | PD' | Z | DC | C |

- Bit7: IRP: Register Bank Select bit (används för indirekt adressering)
0=Bank 0, 1 (00h-FFh)
1=Bank 2, 3 (100h-1FFh)
- Bit5-6 IRP biten används inte på PIC16F8X och bör vara satt till noll för att garantera framåtkompatibilitet med större PIC'ar.
RP1:RP0: Register Bank Select bit (används för direkt adressering).
00=Bank 0 (00j-7Fh)
01=Bank 1 (80h-FFh)
10=Bank 2 (100h-17Fh)
11=Bank 3 (180h-1FFh)
Varje bank innehåller 128 bytes. Endast RP0 används på PIC16F8). RP1 bör vara satt till noll för att garantera framåtkompatibilitet med större PIC'ar.
- Bit4: TO': Time out bit
1=Efter påslag av kretsen, efter en CLRWDT instruktion eller SLEEP instruktion.
0=En Watch Dog Timer time-out har inträffat.
- Bit3: PD': Power-down bit
1=Efter en power-up eller då instruktionen CLWDWT har utförts.
0=Då en SLEEP instruktion har utförts
- Bit2: Z: Zero bit
1=Resultatet av en räkneoperation eller en logisk operation har blivit noll.
0=Resultatet av en räkneoperation eller en logisk operation inte har blivit noll.
- Bit1: DC: Digit carry/borrow' bit (för ADDWF och ADDLW instruktioner)(För borrow' är det tvärt om)
1=En carry bit från fjärde biten från msb har inträffat.
0=ingen carry bit från fjärde biten från msb har inträffat.
- Bit0: C: Carry/borrow bit (för ADDWF och ADDLW instruktioner)
1=En carry bit har ramlat ut från msb av resultatbyten
0=ingen carry bit har ramlat ut från msb av resultatbyten
OBS när det gäller borrow(lånebit) är förhållandena omkastade