

# HÅRDVARUNÄRA PROGRAMMERING I C

ÖZGUN MIRTCEV

## INNEHÅLL

Anvisningar för svar i XCEL-dokument	2
1. Tid för en beräkning	2
2. Vilken datatyp är att föredra?	2
3. Pekararitmetik	2
4. Little Endian	3
5. Register för parameter 2 i en funktion	3
6. Returnering av ett värde av typen int	3
7. Makrokonstant och tolkning av denna	4
8. IO-register på 16 bitar	4
9. Bitmaskning	5
10. Pekare (Little endian)	5
11. Skift	6
12. Vänsterskift	6
Bilaga A. Svarstyper	7
A.1. Decimalpunkt	8
A.2. Enkla matematiskt uttryck	8
A.3. Enkla differentialekvationsuttryck	8

### Anvisningar för svar i XCEL-dokument

. Följande bör du tänka på då svaren skall anges i ett XCEL-dokument:

- (1) Ändra ALDRIG format (.xls) när du sparar XCEL-dokumentet till något annat (.xlsx, etc.).
- (2) Problem med inmatning av svar till vissa svarsceller där tecken tas bort av kalkylprogrammet.
  - (a) Formatera svarscellerna genom att markera dessa och ändra formatet till TEXT.
  - (b) Ett enkelt och generellt råd är att markera alla celler i svarskolumnen och ändra deras format till TEXT.
  - (c) I OpenOffice använder du höger musknapp för att via kontextmenyn (Fomat Cells) formatera svarscellerna till formatet TEXT.

### 1. TID FÖR EN BERÄKNING

*c\_computation\_time\_a.aw2*

Olika beräkningsoperationer tar olika lång tid att utföra i en processor. Vilken av följande beräkningar tar minst att utföra?

- a) `float = float * int;`
- b) `float = float * float;`
- c) `int = int * int;`
- d) `long = long * int;`

### 2. VILKEN DATATYP ÄR ATT FÖREDRA?

*avr\_c\_types\_a.aw2*

Vilken datatyp är att föredra ur effektivitetssynvinkel på en AVR-processor för en heltalsvariabels som kommer att anta värden i intervallet -28 till +28?

- a) `float`
- b) `int`
- c) `unsigned char`
- d) `long`
- e) `signed char`

### 3. PEKARARITMETIK

*c\_hw\_pointer\_b.aw2*

Antag att följande pekardeklaration gäller:

**double** \*p=0X0176;

Vilken adress i minnet accessas vid följande operation:

`*(p+5)`

Antag att int är på 2 bytes, long och float på 4 bytes och double på 8 bytes. Svara med det hexadecimala värdet med fyra siffror.

#### 4. LITTLE ENDIAN

*c\_hw\_pointer\_c.aw2*

I ett C-program finns följande kodfragment:

```
long x;  
unsigned char b;  
  
x = 0xE3DFD6CE;  
b = *((unsigned char *) &x)+1);
```

Vilket värde får variabeln b? Antag att little-endian lagring av data. Svaret ges i hexadecimala talsystemet.

#### 5. REGISTER FÖR PARAMETER 2 I EN FUNKTION

*c\_avr\_func\_subr\_a.aw2*

En funktion har följande prototyp:

```
int Func(int x, int y );
```

Till vilket register skall parametern y överföras?

- a) R19:R18
- b) R21:R20
- c) R23:R22
- d) R25:R24
- e) R27:R26

#### 6. RETURNERING AV ETT VÄRDE AV TYPEN INT

*c\_avr\_func\_subr\_b.aw2*

En funktion har följande prototyp:

```
int Func ( );
```

I vilka register skall funktionen returnera beräknat resultat?

- a) R27:R26
- b) R21:R20
- c) R23:R22
- d) R25:R24
- e) R29:R28

## 7. MAKROKONSTANT OCH TOLKNING AV DENNA

*c\_hw\_ports\_b.aw2*

Vilken skillnad föreligger mellan följande två deklARATIONER:

Fall 1:

```
#define TEST (*(unsigned char *) 0x08)
```

Fall 2:

```
#define TEST ((unsigned char) 0x08)
```

- a) Det är en stor skillnad mellan Fall 1 och 2.
- b) Ingen större skillnad föreligger.
- c) Båda varianterna betyder att TEST är ett symboliskt namn för minnesutrymme på adress 8 i dataminnet.

## 8. IO-REGISTER PÅ 16 BITAR

*c\_hw\_ports\_a.aw2*

Ett I/O-register på 16 bitar för en A/D-omvandlare finns i minnesmappen på adress 0x0024 respektive 0x0025. Deklarera en makrokonstant så att du programmässigt kan hantera detta som en vanlig variabel av datatypen unsigned int.

a) `#define AD (*(volatile unsigned int *) 0x24)`

b) `#define AD (*(volatile int ) 0x24)`

c) `#define AD (*(unsigned int *) 0x24)`

d) `#define AD (*(volatile long *) 0x24)`

e) `#define AD ((volatile unsigned int *) 0x24)`

## 9. BITMASKNING

*c\_hw\_bit\_manip\_a.aw2*

En strömställare (switch) är ansluten till PORTA (PA7-PA0) och pinnen PA3. Vilket är korrekta sättet att avläsa switchen till en heltalsvariabel *v*, som skall spegla switchens logiska värde (0 eller 1).

- a) *v* = (PIN<sub>A</sub> & 0x08) != 0;
- b) *v* = (DDRA & 0x08) == 0;
- c) *v* = PIN<sub>A</sub> & 0x08;
- d) *v* = (PIN<sub>A</sub> & 0x08) == 0;
- e) *v* = PORT<sub>A</sub> & 0x08;

## 10. PEKARE (LITTLE ENDIAN)

*c\_hw\_pointer\_a.aw2*

Antag att följande kod har exekverats:

```
int a;
char c;
int b;
int *p;
```

```
a=5193;
b=2040;
c=30
```

```
p = (int *) &c;
```

&a ⇔	0199	01001001
	019A	00010100
&c ⇔	019B	00011110
	019C	11111000
&b ⇔	019D	00000111
	019E	10011011
&p ⇔	019F	00000001
	01A0	00110110
	01A1	11001100

```
*p = 8887;
```

Vilket decimalt värde har variabeln b?

### 11. SKIFT

*c\_hw\_shift\_b.aw2*

Antag att följande deklaration har gjorts

```
int i = 231;
```

Vilket värde antar i efter följande tilldelningsats:

```
i = i << 6;
```

Ange svaret som ett decimalt tal.

### 12. VÄNSTERSKIFT

*c\_hw\_shift\_a.aw2*

Om följande deklaration gäller

```
#define RXC 4
```

vad beräknas då följande till

```
(1 << RXC)
```

Ange svaret som ett decimalt tal.

## BILAGA A. SVARSTYPER

Olika typer av frågor förekommer såsom multipelt val, numeriskt värde, etc., varje typ av fråga har sin typ av svarsalternativ.

- Ordagrann jämförelse
  - Jämförelse görs bokstav för bokstav.
- Multipelt val (multiple choice)
  - Svarsalternativen är en bokstav A, B, C, etc. Rättningen är okänslig för om svaret anges med VERSALER eller gemener.
- Sant/falskt, rätt/fel, och ja/nej
  - Frågor av typen sant/falskt, rätt/fel och a/nej. anges med svarsvärden för det som är RÄTT/SANT/TRUE/YES med bokstäverna R, S, T, och Y och för det som är FEL/FALSE/NO med bokstäverna F och N. Svarsalternativet fungerar både med VERSALER och gemener. För sant kan även siffran 1 användas och för falskt siffran 0.
- Kommaseparerad lista av ord
- Numeriskt värde med storhet
  - Numeriska svarsvärden är heltal eller reella tal och anges med DECIMALPUNKT istället för decimalkomma.
  - Exempel 1: 125.3
- Numeriskt värde med storhet och enhet
  - Enheter som understöds:
    - \* Alla enheter som representeras med 1 tecken understöds, t.ex. *s* för sekunder, *m* för meter, etc.
    - \* ohm - anges som ohm.
    - \* Ett blanktecken skall finnas mellan det numeriska värdet och enheten.
  - Prefix som understöds t.ex. n för ns (nanosekunder):
    - \* m - milli
    - \* u - mikro
    - \* n - nano
    - \* p - pico
    - \* k - kilo
    - \* M - Mega
    - \* G - Giga
    - \* T - Terra
- Enkelt matematiskt uttryck
  - Exempel 1:  $y=x^{**2}+3*x+4$
  - Exempel 2:  $24*12+3$
- Hexadecimal sträng
  - Exempel 1: bb45
  - Exempel 2: 0x00A6

## – Exempel 3: 0XFFE3

Svaren är ej känsliga för extra vita-tecken (blanktecken, etc.) då t.ex tre blanktecken efter varann ersätts med ett blanktecken.

**A.1. Decimalpunkt.** I alla svar som kräver ett reellt tal som svar används decimalpunkt istället för decimalkomma.

**A.2. Enkla matematiskt uttryck.** Ett enkelt matematiskt uttryck har en syntax som motsvaras av det man finner i programmeringsspråk som C, MatLab, Ruby, etc. Då små detaljer kan skilja olika programmeringsspråk åt.

Observera att skillnad föreligger mellan HELTALSDIVISION och division av reella all så kallad FLYTTALSDIVISION. Ex 1:  $10 / 6$  ger som resultat 1, däremot så ger  $10.0 / 6$  resultat 1.666667. Den enkla regeln är att om en av operanderna vid division är ett reellt tal fås en flyttalsdivision, dvs en helt normalt resultat med decimaldelar. Om båda operanderna är ett heltal erhålls alltid ett heltalsresultat av division dvs decimaldelarna kastas.

I exemplena som följer i tabellen har variablerna a och b heltalsvärdena 10 respektive 20 och c och d de reella värdena 10.0 respektive 20.0.

Operator	Beskrivning	Exempel
+	Additionsoperatör	a+b ger som resultat 30
-	Subtraktionsoperatör	a-b ger som resultat -10
*	Multiplikationsoperatör	a*b ger som resultat 200
/	Divisionsoperatör	a/b ger som resultat 0 (heltalsdivision) c/d ger som resultat 0.5 (flyttalsdivision, reell division)
%	Modulusoperatör	
**	Exponent	a ** b ger a upphöjt i b dvs 10 upphöjt i 20 b ** 2 ger som resultat 400

Följande matematiska funktioner kan användas:

Funktion	Beskrivning	Exempel
acos	Arcuscos	
asin	Arcussinus	
atan	Arcustangens	
cos	Cosinus	
exp	Exponentialfunktionen	
log	e-logaritmen (ln)	
log10	10-logaritmen	log10(100) ger värdet 2
log2	2-logaritmen	
sin	Sinus	sin(PI/2) ger värdet 1.0
sqrt	Kvadratroten	sqrt(4) ger värdet 2
tan	Tangens	

**A.3. Enkla differentialekvationsuttryck.** En differentialekvation som t. ex.

$$\frac{d^2y}{dt^2} + 3\frac{dy}{dt} + 2y = 3\frac{du}{dt} + 2u$$



anges som ett enkelt differentialekvationsuttryck anges på följande form:

$$y'' + 3*y' + 2*y = 3*u' + 2*u$$

Den enkla regeln är att varje derivering anges med en apostrof, dvs för andraderivatan blir det två apostrofer efter varann.