

# INNEHÅLL

7	DATORER	3
7.1	Allmänt	3
7.2	Digitala datorer	4
7.2.1	Organisation	4
7.2.2	Ord m m	4
7.2.3	Minnesenheten	5
7.2.4	Aritmetiska enheten	6
7.2.5	Styrenheten	6
7.2.6	In/utenheten	6
7.2.7	Avbrott	7
7.2.8	Språk och programmering	7
7.3	Analoga datorer	8

## 7 DATORER

### 7.1 ALLMÄNT

Den ökande komplexiteten hos moderna avioniksystem och det likaledes ökande behovet av noggrann information har lett till att *datorer* av olika slag numera installeras i de flesta militära och i många civila flygplan. Med dator skall vi här förstå en databehandlare som utan mänskligt ingripande kan utföra omfattande beräkningar med stort antal aritmetiska eller logiska operationer. En förteckning över de åtgärder som datorn vidtar kallas *program*.

Med hänsyn till den tidpunkt vid vilka data bearbetas skiljer man på *realtidsbearbetning* och *satsvis bearbetning*. Vid realtidsbearbetning rapporteras och behandlas omedelbart varje inträffad händelse (t ex mätning eller en knapptryckning), varefter bearbetade data genast sänds vidare till en lämplig abonnent (t ex ett presentationsorgan). Vid satsvis bearbetning registreras en mängd vid olika tidpunkter inträffade händelser, varefter bearbetning sker periodiskt (t ex en gång/dag). Navigeringsbearbetningar görs praktiskt taget alltid i real tid.

Data kan representeras på olika sätt. Huvudformerna är *analog* och *diskret* representation. Analoga data representeras av kontinuerligt variabla fysikaliska storheter, t ex längder, vinklar, spänningar eller frekvenser. Diskreta data representeras av tecken, varvid varje möjligt datumvärde motsvaras av ett bestämt tecken eller av en bestämd teckengrupp. En räknesticka eller en kvicksilvertermometer ger analoga data; från en kulram, räknesnurra eller trippmätare i en bil fås däremot diskreta data. Om tecknen i den diskreta representationen utgörs av siffror kallas representationen *digital*. Med hänsyn till representationen av data i datorn talar man om *analogdatorer* (analog representation), *digitaldatorer* (diskret representation) *hybriddatorer* (både analog och diskret representation). I digitaldatorerna används vanligen en representation om två tecken (oftast benämnda noll och ett). Dessa kan representeras av t ex höga respektive låga spänningar eller de två magnetiseringsriktningarna i en järnkärna.

Digitala data måste ofta omvandlas till analoga eller tvärtom. Detta görs i *D/A-omvandlare* och *A/D-omvandlare*.

De första flygburna datorerna var analoga, decentraliserade enheter som var specialtillverkade för att bearbeta data från en eller flera sensorer och sedan sända dessa vidare för presentation.

På 60-talet kom centraldatorbegreppet att dominera. En enda dator skulle sköta all automatisk databehandling i flygplanet. Det ökade bearbetningsbehovet har emellertid lett till att man på senare år åter börjat installera med sensorer och instrument hårt integrerade specialdatorer som för- eller efterbearbetar data till och från centraldatorn.

Exempel på olika användningsområden för flygburna datorer är

- Luftdataberäkningar. Datorn beräknar ur primära mätningar av tryck, temperatur m m flygplanets hastighet och höjd
- Navigeringsberäkningar. Här ingår positionsberäkningar ur primärdata från RN-system, automatisk dödräkning, samt styrorderberäkningar vid brytpunkts- och landningsnavigering
- Beräkning av bränslemarginaler
- Komplementär- och Kalmanfiltrering av primärdata (avsn 14)

- Förinställning och precedering av attityd- och TN-plattformar
- Siktes- och vapenberäkningar
- Beräkning av data för presentationsutrustningar
- Funktionskontroll
- Dataregistrering

## 7.2 DIGITALA DATORER

### 7.2.1 Organisation

En typisk digital dator består i allmänhet av fyra huvudenheter: *styrenhet*, *minne*, *in/utenhet* och *aritmetisk enhet* (bild 7.1). Minnesenheten används för att lagra programinstruktioner och andra data. Den aritmetiska enheten utför aritmetiska operationer och andra manipulationer på data. In/utenheten (ofta flera enheter) handhar datorns förbindelser med omvärlden (instrument, presentationsorgan, reglage etc). Styrenheten samordnar och administrerar de övriga enheternas verksamhet i enlighet med det i minnet lagrade programmet. Det är inte alltid möjligt att fysiskt särskilja styrenhet och aritmetisk enhet eftersom de ofta är starkt integrerade. De brukar ofta sammanfattas under benämningen *centralenhet*.

### 7.2.2 Ord m m

Den minsta informationsbärande enheten är i de flesta datorer en *binär siffra* (*bit*, plur bitar) dvs en etta eller nolla. Bitarna samordnas till *ord* om 8–64 bitar som representerar data.

Överföring av ett ord från en del av datorn till en annan kan ske på två sätt: i serie eller parallellt. Vid serieöverföring behandlas en bit i taget, medan man i parallellvarianten överför flera bitar (t ex ett helt eller ett halvt ord) på en gång. Den senare metoden är snabbast; den förra kräver mindre komplicerad kretsteknik och är därför billigare.

I minnena hos flygburna datorer består de data som ej är instruktioner vanligen av tal. Dessa kan representeras på två sätt: *fasttalsrepresentation* och *flyttalsrepresentation*.

Vid fasttalsrepresentation är decimalkommats placering underförstådd och konstant. Vid flyttalsrepresentation anges även en exponent som utmärker decimalkommats plats. Man har då större möjlighet att låta talet variera mellan vitt skilda storleksordningar.

Är ordlängden liten representeras talen vanligen av två ord (dubbelord). För att ange en decimal siffra krävs i genomsnitt ca 3.3 bitar. Desutom måste en (fasttalsrepr) eller två (flyttalsrepr) bitar reserveras för tecken.

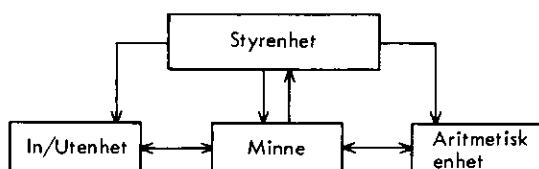


Bild 7.1 Digitaldator, blockschema

Ord kan även innehålla *programinstruktioner*. Dessa består av en *operationsdel* och en *adressdel*. Operationsdelen anger vad som skall göras (t ex en addition) medan adressdelen anger den *lokation* (plats) i minnet eller de register i aritmetiska enheten som berörs av operationen. Med en lokation förstås här en *adresserbar* enhet i minnet eller aritmetiska enheten. Normalt rymmer en lokation ett eller ett halvt ord, men det finns också datorer där man kan adressera enstaka bitar.

En bit i varje ord brukar användas som *paritetsbit*. den väljs så att totala antalet ettor i ordet alltid är t ex udda. Härigenom kan datorn utföra en automatisk felkontroll.

### 7.2.3 Minnesenheten

Minnen hos flygburna datorer brukar f n (1972) ha en utrymmeskapacitet om 2000 – 64 000 ord.

I flygburna datorer används vanligen *direktminne*, dvs minnen där varje ord kan nås på en tid som är oberoende av ordets läge i minnet. (cykeltiden, dvs den tid det tar att läsa och åter lagra ett ord i ett direktminne rör sig f n hos snabbare datorer om  $< 1 \mu\text{s}$ . *Sekvensminnen* å andra sidan måste alltid genom-sökas i viss följd tills man finner vad man söker.

Många datorsystem har flera minnen: ett *primärminne*, som har direktförbindelse med centralenheten, där programinstruktioner och aktuella data lagras samt ett eller flera *sekundärminnen* där stora mängder mindre aktuella data lagras. Sekundärminnet kan endast utbyta data med primärminnet, ej med någon annan enhet.

Med ett *flyktigt minne* förstås ett minne vars innehåll förstörs vid strömavbrott. Vid flygburna tillämpningar där datorn ofta arbetar i en påfrestande miljö är det önskvärt med icke-flyktiga minnen.

*Fäst minne (skrivskyddat minne)* kallas ett minne där det krävs speciella ingrepp för att lagra information. För att öka tillförlitligheten är i flygburna datorer ofta den del av minnet där programinstruktionerna och vissa konstanter är lagrade skrivskyddad.

Det för närvarande vanligaste primärminnet är *kärnminnet* som är ett icke-flyktigt minne. Det består av ett antal (en för varje bit) ferritringar, vilkas magnetiseringsriktning avgör om respektive bit är en etta eller nolla. Lagring och läsning sker genom att sända elektriska pulser genom trådar som genomlöper ringarna i ett visst mönster. Läsning av information leder i konventionella typer av kärnminnen vanligen till att informationen förstörs och därför omedelbart måste återlagras. Speciella typer av kärnminnen har utvecklats där denna nackdel eliminerats, varigenom tillförlitligheten kan ökas. Nackdelarna är främst att kostnaderna ökar och snabbheten minskar.

Hos s k *magnetminnen* lagras informationen i ett tunt magnetiserbart skikt som kan vara placerat på ett band (*bandminne*), en roterande cylinder (*trumminne*) eller en roterande skiva (*skivminne*). Lagring och läsning sker genom att man låter skiktet passera och växelverka med ett magnethuvud. Denna typ av minnen används som sekundärminnen, och har hittills rönt föga användning för flygbruk.

I början av 70-talet kommer emellertid s k automatiska integrerade datasystem (dataloggar) att tas i bruk för flygplan. Systemen registrerar på bandspelare under flygningen data som efteråt kan (satsvis) bearbetas för bl a utbildnings-, underhålls- och flygsäkerhetsändamål. Spaningstillämpningar är också tänkbara.

Även trumminnen och skivminnen (som i motsats till bandminnet är direktminnen) väntas få viss användning i flygplan. Den stora kapaciteten gör att de lämpar sig för lagring av icke regelbundet använda program (t ex vissa testprogram eller program avsedda att användas vid speciella uppdrag). När programmet skall användas flyttas det över till kärnminnet.

Minnen baserade på halvledarteknik har också på senare år börjat användas i flygplan. Halvledarminnen är (1972) antingen flyktiga och då mindre lämpliga för flygbruk eller också fasta. I motsats till ferritminnen där skrivskyddet åstadkoms med kretstekniska medel härrör »fastheten» hos halvledarminnen från deras fysikaliska egenskaper, vilket gör en omprogrammering relativt besvärlig.

#### 7.2.4 Aritmetiska enheten

Aritmetiska enheten består av ett eller flera register (lokationer avsedda för speciella ändamål) samt logiska och aritmetiska kretsar som utför de aritmetiska operationerna. I registren lagras de ord som skall behandlas av enheten eller är resultatet av behandlingen. De aritmetiska kretsarna består i sitt enklaste utseende endast av en adderare. Samtliga elementära räkneoperationer kan nämligen genom vissa manipulationer brytas ned till additioner. I allmänhet finns dock särskilda faciliteter för multiplikation och division inbyggda. Typiska operationstider för flygburna högprestandadatorer är (1972):

addition, subtraktion 3  $\mu$ s      multiplikation 6  $\mu$ s      division 10  $\mu$ s

#### 7.2.5 Styrenheten

Styrenheten innehåller bl a *avkodarkretsar*, *klocka* och diverse register. Avkodaren översätter de från minnet hämtade instruktionerna till signaler som genererar respektive operation i de övriga enheterna. Klockan utgörs vanligen av en pulsgenerator med stabil pulsfrekvens. I *synkrona* datorer kan de olika operationerna endast påbörjas då generatoren ger ifrån sig en puls. I *asynkrona* datorer påbörjas istället varje operation när den föregående avslutas. Den senare typen kan i princip göras snabbare, men kräver mer komplicerad elektronik. Klockan spelar också stor roll för *avbrottsystemet* (avsn 7.2.7).

*Indexregistren* möjliggör flexibilitet i adresseringen genom att man adderar innehållet i ett av dessa till en del av adressdelen av instruktionen. Härigenom kan adressen variera med indexregistrets innehåll.

I *instruktionsadressregistret* lagras adressen till den instruktion som är under utförande.

*Indikatorregister* (som i allmänhet är små, med plats för en eller ett par bitar) används för ett flertal uppgifter. De kan t ex indikera paritetsfel, spill (en aritmetisk operation ger ett resultat som ej får rum i ett ord), otillåten operation (t ex division med noll) eller tecknet på resultatet av en aritmetisk operation.

#### 7.2.6 In/utenheten

In/utenheten sköter datorns förbindelser med omvärlden, t ex givare, presentationsorgan (avsn 11) och manöverorgan m m. I många system har man skilda enheter för in- och utmatning och man kan också ha flera enheter av vardera slaget. Kontakterna med omvärlden kan ha både digital och analog form. I det senare fallet måste in/utenheten vara försedd med D/A- och A/D-omvandlare. De olika informationskällorna kan ha mycket olika intensitet. En kurs- eller fartgivare kan ge praktiskt taget kontinuerlig information, medan t ex DME-fixar endast kommer in med (relativt) stora och oregelbundna mellanrum.

Av dessa skäl måste information av den senare typen ges *prioritet* för att inte gå förlorad. Alternativt kan in/utenheten vara försedd med *buffertregister* där informationen kan lagras tills minnet kan ta emot den.

### 7.2.7 Avbrott

Vissa regelbundet eller oregelbundet inträffade händelser gör det ibland nödvändigt att avbryta körningen av det pågående programmet för att hoppa in i ett annat program. När det nya programmet fullföljt sin uppgift kan återgång till det gamla eller in hopp i ett tredje program ske. De avbrytande programmen kan ges olika *prioritet* så att lågprioriterade program ej kan avbryta ett högre prioriterat program utan måste vänta tills det förra *kvitterat*, dvs deklarerat att det fullföljt sin uppgift. Nedan följer några exempel på typiska avbrott i en flygburen dator (i tänkbar prioritetsordning).

*Från- och tillslag av spänning.* Datorn kan i allmänhet fortsätta att arbeta ett antal  $\mu$ s efter det att matningsspänningen börjat sjunka. Under denna tid genomlöps ett program som förbereder datorn för förnyat arbete om spänningen åter slås till. Om detta inträffar, sker in hopp till ett program som återställer datorn i det tillstånd som rådde före frånslaget.

*Programfel,* t ex spill, paritetsfel, otillåten operation, skrivorder till skrivskyddad lokation. Avbrott av denna typ leder till in hopp i lämpligt diagnosprogram som söker detektera felet.

*In/utmatning* kan leda till avbrott med regelbundna eller oregelbundna mellanrum. In/utkanalerna kan sinsemellan vara prioriterade.

*Klockavbrott* inträffar med regelbundna mellanrum. Huvudprogrammet genomlöps normalt en gång per *samplingsintervall* (typisk längd: ca 100 ms). Därefter finns i allmänhet tid över innan samplingsintervallet är slut. Denna tid kan användas för ett lägre prioriterat program, t ex funktionskontroll. När samplingsintervallet är slut sker klockavbrott och återin hopp i huvudprogrammet. Ibland måste vissa beräkningar utföras flera gånger per samplingsintervall. I så fall måste ytterligare klockavbrott införas.

### 7.2.8 Språk och programmering

Ett program skrivet på en för datorn direkt begriplig form (på *maskinspråk*) utgörs väsentligen av ett antal *instruktioner* skrivna med ettor och nollor. Att programmera i ett sådant språk är utomordentligt besvärligt och tidsödande. Proceduren underlättas väsentligt om man i stället använder ett så kallat *assembleringspråk* som skrivs med bokstäver och decimala siffror och där varje sats svarar mot en viss instruktion i maskinspråket och omvänt. Översättning sker med ett speciellt program som kallas *assembleringsprogram*. Högre *problemorienterade* språk som t ex ALGOL eller FORTRAN tillåter programmeraren att använda komplicerade aritmetiska uttryck som vart och ett motsvarar en stor mängd satser i maskin- eller assemblerspråk. Översättning till maskinspråk sker här med ett program som kallas *kompilator*. Program skrivna i dessa språk kräver i allmänhet större minnesutrymme och längre beräkningstid än assembleringsprogram. I gengäld går programmeringen långsammare med de senare. Vid programmering av flygburna datorer används i allmänhet assembleringspråk.

När programmet assemblerats (kompilerats; dvs översatts), lagras det i minnet som en följd av *instruktioner*. En instruktion upptar vanligen en eller två hela minneslokationer och består som förut framgånget av två delar, operationsdel och adressdel. Operationsdelen anger vad som skall uträttas och adressdelen anger vilken minneslokation som berörs. Instruktionerna utförs efter inlas-

ning till styrenheten normalt i ordning efter de lokationer i vilka de är lagrade. Med s k *hoppinstruktioner* kan dock denna ordning brytas.

Vi skall visa ett prov på hur en programsekvens i assembleringsspråk för en (mycket enkel) dator kan se ut. Programmet läser från in/utenhetens inkanal nr 6 aktuell kurs som sedan lagras i datorns minne och jämförs med i datorn lagrad kommenderad kurs. Skillnaden sänds i form av styrorder vidare till t ex siktlinjesindikatorn genom in/utenhetens kanal 8.

<u>Cell nr</u>	<u>Operationsdel</u>	<u>Adressdel</u>
0031	IN	6
0032	SKR	KURS
0033	SUB	KOMKURS
0034	UT	8

Instruktionssekvensen är lagrad i minnescellerna 31–34. Instruktionen i cell 31 läser av in/utenhetens inkanal nr 6 och placerar informationen i ett register i aritmetiska enheten. Utan att registerinnehållet förstörs skrivs detta sedan med instruktionen i cell 32 in i en minneslokation som i programmet har namnet KURS. (Numret på denna lokation behöver i allmänhet ej specificeras av programmeraren – det sköter i stället assemblern om). Instruktionen i cell 33 subtraherar sedan innehållet i en lokation med namnet KOMKURS från innehållet i registret, som sedan innehåller skillnaden (lokation KOMKURS ändras ej). Slutligen utmatas med instruktionen i cell 34 registerinnehållet till indikatorn, varvid registret förblir oförändrat. Motsvarande programsekvens skulle i maskinspråk bestå av fyra (t ex) 24-siffriga binära tal.

I ALGOL skulle sekvensen kunna se ut så här:

Read (6) kurs; styrorder: = kurs–komkurs; write (8) styrorder;

(Läsaren bör observera att i levande livet är beräkning av styrorder en långt mer komplicerad procedur än vad som här har antytts).

Vissa typer av beräkningar (t ex trigonometriska funktioner och rotutdragningar) behöver ofta genomföras flera gånger i ett program. De placeras då i specialprogram, s k *subrutiner* vilka anropas av huvudprogrammet så snart en dylik beräkning måste göras.

### 7.3 ANALOGA DATORER

Analoga datorer använder kontinuerliga storheter, t ex spänningsnivåer, vridningsvinklar eller frekvenser för att representera olika parametrar. De består av elektriska, mekaniska och hydrauliska komponenter som ofta är starkt integrerade med de instrument, manöver- och presentationsorgan med vilka de samarbetar.

Analogdatorer kan utföra elementära aritmetiska operationer, beräkna trigonometriska funktioner, utföra koordinattransformationer derivera och integrera, samt lösa enkla differentialekvationer. De är lämpliga att använda när uppgiften är att snabbt och med måttlig noggrannhet lösa ett fåtal differential- eller algebraiska ekvationer. De passar däremot inte för noggranna beräkningar och stora datamängder. Något liknande de digitala datorernas program finns ej, utan »programmeringen» består i att på lämpligt sätt sammankoppla de olika komponenterna.

Utvecklingstendenserna tycks för närvarande vara inriktade på olika typer av *hybridisering*, dvs blandning av digital och analog teknik. Detta yttrar sig dels i att delar av analogdatorerna digitaliseras (vilket också innebär att elektroniska komponenter breder ut sig på de mekaniskas bekostnad), dels i att analogdatorer används för speciella ändamål i ett i övrigt digitalt system, t ex för förbearbetning av data från sensorer innan de sänds vidare till den digitala centraldatorn.