

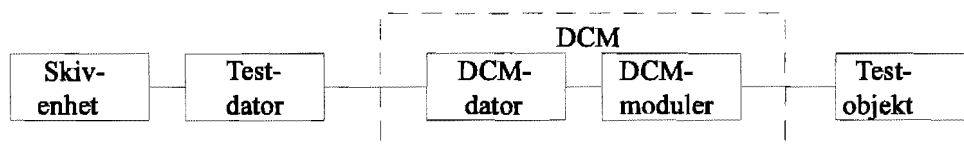
1. DCM-BESKRIVNING

1.1 SAMMANFATTNING

DCM (Digital Communication Module) är ett instrument som används vid test av JA elektronikenheter.

DCM:en är ett kombinerat mät/stimulinstrument, som simulerar den speciella parallella och seriebinära kommunikationen mellan olika objekt inom FPL-JA37.

DCM-systemet består hårdvarumässigt av en dator (DCM-datorn), samt speciellt framtagna moduler för olika typer av binära kommunikationer.



Figur 1. Blockschema DCM-systemet.

Mjukvarumässigt finns ett antal makroanrop för styrning av de olika modulerna. Anropen exekveras dels i testdatorn, dels i DCM-datorn. I testdatorn sker främst parameterkontroller och datakonverteringar samt telegramöverföringar till DCM-datorn. I DCM-datorn verkställs telegraminformationer ut mot DCM-modulerna.

1.2 UPPBYGGNAD DCM-MODULER

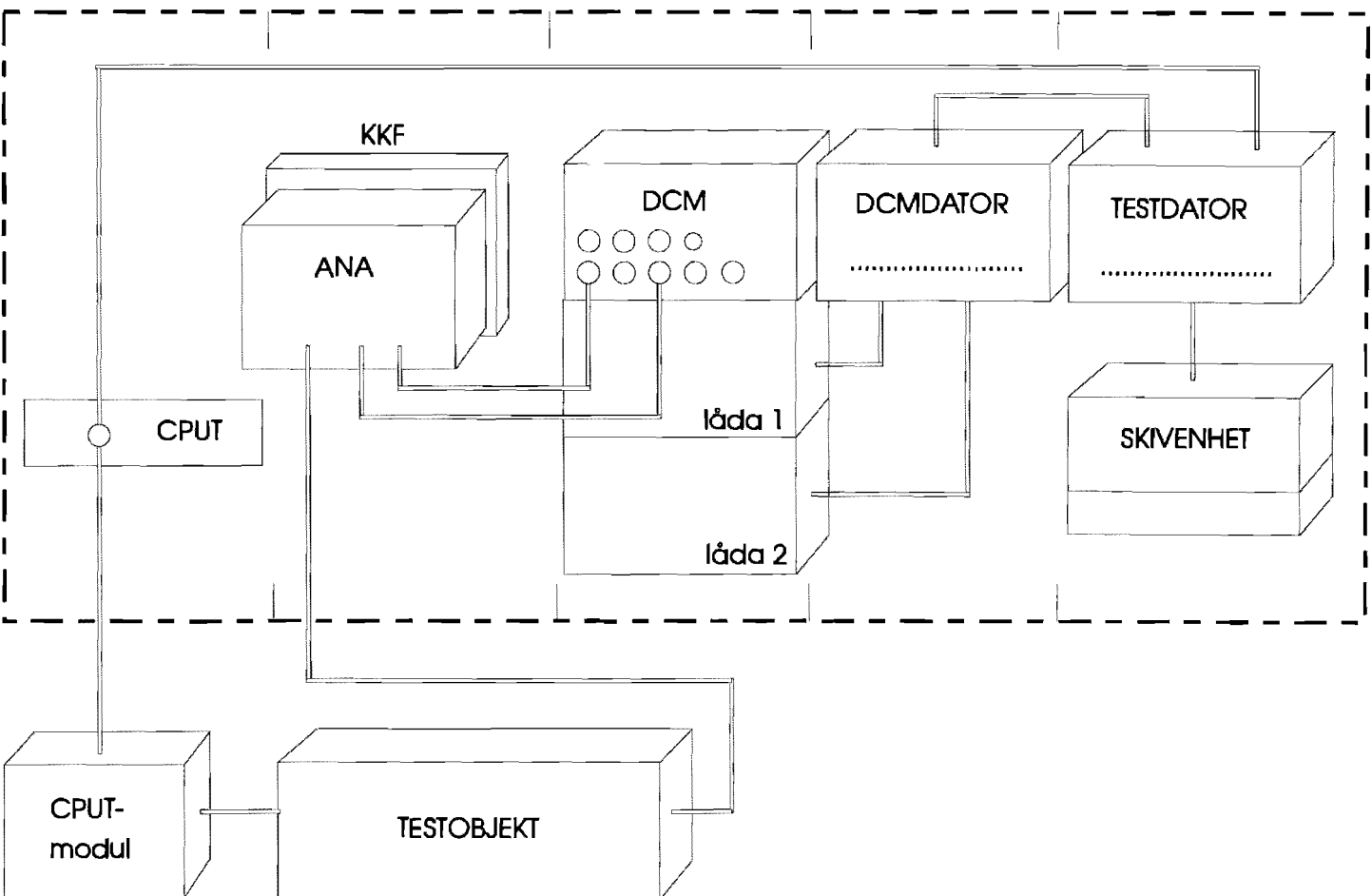
DCM-modulerna byggs upp i tre 19" lådor och placeras omedelbart till höger om KKF i ATSl0 (se Figure 2). Kretskorten är av europastandardformat (enkla och dubbla).

Balanserade anpassningskretsar för samtliga in- och utgångar är placerade på enkla europakort och tillsammans med en intern korskopplingslåda samlade i den övre lådan. På fronten finns även samtliga anslutningar till DCM:en (överföringspanel 1, ÖPI).

Kontakterna A1-A5 är avsedda för test av dator CD107, kontakt E för test av VXL29 och kontakterna B-D för test av övriga objekt.

Låda 1 (mellanlådan) innehåller moduler för parallellbinära kommunikationer samt seriebinär CD-kommunikation (CD-moduler).

Låda 2 (undre lådan) innehåller moduler för övriga typer av seriebinära kommunikationer (t.ex radarbuss, video, bifas).



Figur 2. Fysiskt blockschema DCM-systemet.

Figur 2 Fysiskt blockschema DCM-systemet

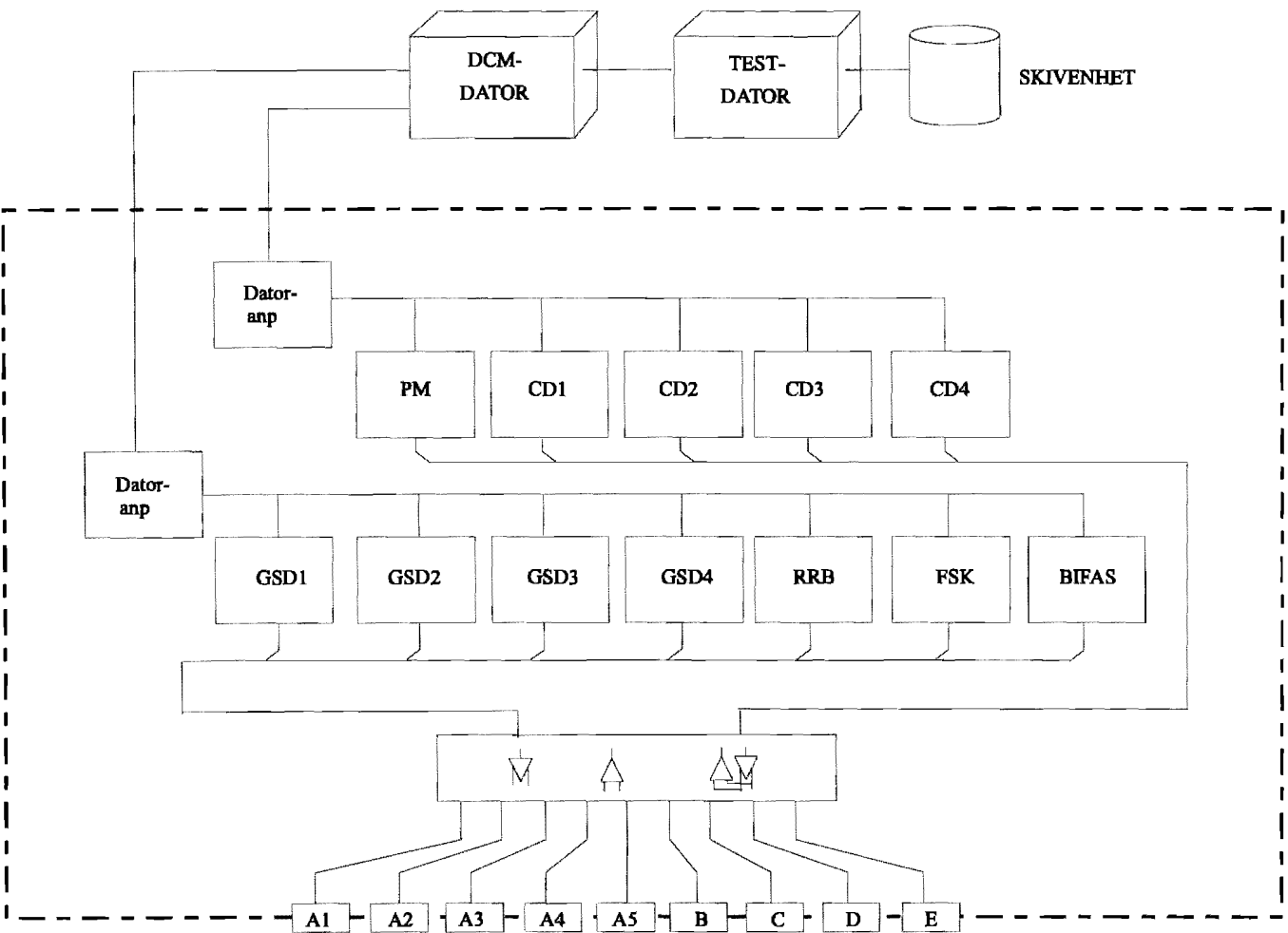
1.3 KORTFATTAD FUNKTIONSBESKRIVNING

1.3.1 IN/UT-system

DCM-datorn kommunicerar med testdatorn via en 16-bitars parallellkanal och med DCM-låda 1 resp 2 via en buss (K-bussen), som styrs med 2 st 16-bitars parallellkort. För att nå olika moduler på bussen används en moduladress, som också återkommer i de flesta DCM-makron. I testdatorn styr en standard-drivrutin in- och utmatningen och i DCM-datorn, som är avbrottsstyrd, finns ett styrprogram som administrerar alla inkommande avbrott.

1.3.2 Laddning av styrprogram

Innan DCM-funktionerna kan användas, måste DCM-datorns styrprogram laddas. Detta sker m.h.a makrot DCMOP i testdatorn och en "prom-loader" i DCM-datorn. Styrprogrammet förs då över från skivan, via testdatorn till DCM-datorn. Då överföringen är klar, sker en nollställning av DCM-modulerna innan styrprogrammet startas.



Figur 3. Blockschema DCM-moduler.

1.3.3 Parallellmodul

För test av parallellbinära signaler (CDOK etc) finns tillgång till 8 st in- och 8 st ut signaler i kontaktarna B och C. Signalernas nivåer läses av resp ändras genom makroanropet PARDAT. De 8 nivåerna behandlas som 8-bitars oktalt tal. För test av CD finns betydligt fler parallellsignaler tillgängliga i kontakt AI-2.

1.3.4 CD-moduler

För test av seriebinära CD-kanaler finns 4 st CD-moduler tillgängliga. Modulerna kan programmeras oberoende av varandra och användas parallellt, dvs upp till 4 CD-kanaler kan köras samtidigt. Den seriebinära kommunikationen kan även pågå samtidigt som testdatorn utför övriga stim/mätfunktioner. I överföringen kan också viss dynamik ingå, t.ex ett ord på en kanal som ändrar värde varje överföringscykel enligt en viss funktion. Ett visst värde på inkommande data kan även upptäckas, varvid hela överföringsbufferten "fryses" och finns tillgänglig För utvärdering. Modulerna "körs" med makrot DCMOD.

CD-modulerna kan programmeras att fungera på fyra olika sätt:

Synk ut - data in/ut,

Synk in - data in/ut

Antal ord per överföringscykel (även kallad programcykel, PC), fördröjning inom programcykel och "cycle-reset"-puls är också valbart. Makroanropen som utför initieringen heter INSYNK och EXSYNK.

Data i den seriebinära överföringen manipuleras med hjälp av BASIC via dataareor på skivan. Varje modul har en sådan area tilldelad, och med inakrona SOUA (skriv ord i utarea) och LOIA (läs ord i inarea) kan dessa areor fyllas med data respektive forts läsas in till testprogrammet. Makrona SUUD (sänd ut utdata) och TIID (ta in indata) överför innehållet i dataareorna till/från DCM-modulerna.

Genom att generera avsiktliga fel på överföringen på de seriebinära kanalerna kan testobjektets felupptäckande kretsar kontrolleras.

Följande fel typer kan programmeras (makrot FELGEN):

- Paritetsfel (fel paritet på data till testobjekt).
- Formatfel (felaktiga synkpulser på synk till testobjekt).
- Fokbitfel (felaktig fokbit på data till testobjekt).

Dessutom övervakas överföringen kontinuerligt av DCM-datorn och vid fel larmas testdatorn.

1.3.5 GSD-moduler

För test av vissa övriga seriebinära överföringar finns 4 st GSDmoduler tillgängliga (GSD står för generell seriedata - modul). Modulerna kan programmeras oberoende av varandra och användas parallellt. Den seriebinära kommunikationen pågår också oberoende av testdatorn. Modulerna initieras med makrona SYIN resp SYUT. Här väljs synkriktning, datariktning och antal ord. Med en funktionsparameter programmeras modulernas olika buffertar upp enligt den begärda funktionen. Typiska funktioner är radar, EP, video och flygradaröverföring. Data i den seriebinära överföringen manipuleras från BASIC på samma sätt som För CD-modulerna, dvs med makrona SOUA, LOIA, SUUD och TIID.

1.3.6 RRB-Modul

För test av objekt, som utnyttjar radarbusskommunikation, finns en radarbussmodul i DCM:en. Modulen kan programmeras och användas oberoende av övriga moduler. Med makrot RRBADR definieras de adresser som skall användas på bussen under en viss testsekvens. Oataareor För olika överföringscyklar byggs upp och sparas på skivan. Makrot RRBINI Initlerar hårdvaran i radarbussmodulen att fungera på visst sätt, samt För över information som gäller övervakning av radarbusskommunikationen. Data som hör till olika radarbussadresser manpuleras från BASIC m.h.a makrona SOUA, LOIA, SUUD och TIID. Ordnnummer betyder dock i detta fall radarbussadress. Modul en "körs" med makrot DCMOD.

1.3.7 FSK-modul

För att simulera styrdatakod till FR29 finns en styrdatamodul i DCM:en. Styrdatamodulen omvandlar inkommande seriebinärt data till en analog frekvensskiftad signal. Med makrot FSKINI kan utsignalen styras och varieras på olika sätt (t.ex nivå, frekvenser, brus). Det seriebinära indata hämtas från en GSD-modul som måste initieras och fyllas med aktuellt data.

1.3.8 BIFAS-Modul

För att kontrollera bifaskod från olika objekt finns en bifasmodul i DCM:en. Modulen konverterar inkommande bifaskod till binärkod. DCM-datorn plockar sedan upp de konverterade orden och utvärderar efter hand mot rätt data.

1.4 SUBROUTINER

Följande anrop finns definierade för DCM-systemet:

1. Programladdning:
DCMOP - Laddar DCM-datorns operativsystem.
2. Datahantering:
PARDAT - Styrning av paralleldata.
SOUA - Skriv ord i utarea.
SBUA - Skriv bit i utarea.
LUIA - Läs ord i inarea.
LBI A - Läs bit i inarea.
SUUD - Sänd ut utdata.
TIID - Ta in indata.
3. Initiering:
INSYNK - Initiera CD-modul För intern synk (synk ut).
EXSYNK - Initiera CD-modul för extern synk (synk in).
SYUT - Initiera GSD-modul för synk ut.
SYIN - Initiera GSD-modul för synk in.
RRBADR - Definiera RRB-adresser.
RRBINI - Initiera RRB-modul.
FSKINI - Initiera FSK-modul.
BIFAS - Initiera BIFAS-modul.
4. Styrning:
DCMOD - Styrning av DCM-modulernas moder.
KOMP - Komparera inkommande data.
5. Felhantering:
FELGEN - Avsiktlig felgenerering pådataöverföring
FSTAT - Avläsning av felstatistik från DCM
FELINI - Initiera DCM-modul för fel larm
FKOD - Avläsning av larmande DCM-modul.
6. Speciella subrutiner för CD:
MR - Master reset
LOAD - Ladda helord DMA
LOADD - Ladda helord från BASIC
LOADE (alt. LOADF) - Ladda skivfil DMA (Assemblerad FOCAB-kod) 11-4

| | | |
|-------|---|---|
| FETCH | - | Läs DMA från CD minne Ladda helord från BASIC |
| LRJ | - | Läs register och jämför |
| LBJ | - | Läs bit och hoppa om # 1 |
| PSW | - | Programswitchar |
| SSKY | - | Skrivskydd |
| STEX | - | Starta exekvering |
| LREG | - | Ladda register |
| CCUT | - | Självtest och initiering |
| EXK | - | Exekvering |
| MEMCH | - | Jämför CD minnesinnehåll |
| MEML | - | Läs minne |

6. Speciella subrutiner för SA07:

| | | |
|----------|---|--|
| SA | - | Skriv i minnet |
| SAMC | - | Reset för SA07 elektronikenhet |
| SARS | - | Start/Stopp av CPU i SA07 elektronikenhet |
| SAWE | - | Skrivskydda minnet i SA07 elektronikenhet |
| SKX/ SKY | - | Diskreta insignaler till SA07 CPU |
| CSA | - | Jämför ATS10 skiva med SA07 minne |
| LSA | - | Läs från ATS10 skiva till SA07 minne |
| PSA/PSAO | - | Läs från SA07 minne till area i ATS10 |
| CPSAF | - | Jämför funktionsprogram och ISK/FK |
| LPSAF | - | Ladda funktionsprogram och ISK/FK |

Table I: Moduladresser i DCM

| Adress | Modul |
|--------|-------------------------------|
| 0 | Paralleldata - PM |
| 1 | CD-överföring 1 - CD1 |
| 2 | CD-överföring 2 - CD2 |
| 3 | CD-överföring 3 - CD3 |
| 4 | CD-överföring 4 - CD4 |
| 5 | Generell serie-data 1 - GSD1 |
| 6 | Generell serie-data 2 - GSD2 |
| 7 | Generell serie-data 3 - GSD3 |
| 8 | Generell serie-data 4 - GSD4 |
| 9 | Radarbussöverföring - RRB |
| 10 | Frekvensskiftöverföring - FSK |
| 11 | Bifasöverföring - BIFAS |
| 12 | Reserv |
| 13 | Reserv |
| 14 | Datoranpassning |
| 15 | Gemensamt block |

| Funkt | Överföringstyp |
|-------|---|
| -1 | Varierad PC-tid (ms) enligt följande: 21 -26 - 16 - 21 - 26 - 16 etc. |
| -2 | Växling av databuffert varannan PC21:a vid data ut. Krav att två ggr det antal ord som angivits i INSYNK skrivs i utarean för modulen och sänd med SUUD. |
| -3 | Trigg via paralleldata ord nr 1. Sätt det bitnummer som skall triggas på i PARDAT ord 1. Gäller både data in och data ut. |
| -4 | Sin och Cos för VG. |
| -5 | Bildcykler för VG. |
| -6 | <p>Jaktlänkfunktion (JL) SU-fdata för VX29. Hämtar varje PC21:a fyra bitar från ord lagrade i CD-modul 4 dataarea och maskar in i vald CD-moduls (1 - 3) dataarea utord nr 3 bit 1 - 4, samt sätter bit 0 (JL-flagga) (bit-nr refererar till HP-ord) under 52 PC21:or, sedan skickas det 52:a PC21-meddelandet ut men med bit 0 i ord 3 nollställt. Funktionen kräver att CD-modul 4 satts upp med EXSYNK kanal 23 för att ej få paritetsgenerering, samt att 14 st ord (2*104 JL-bitar) sänts till modul 4. I ord 0 - 6 läggs JL-bit 0 - 103 och i ord 7 - 13 läggs JL-bit 104 - 208. Dessutom skall 7 ord sändas till vald modul (1 - 3) med första fyra JL-bitar samt JL-flagga satta till 0 i ord 3. Nedan visas hur man skall tolka bitarna i de ord som sänds till modul 4. Till CD sänds CD-bitar med CD-bit 13 först och sedan bit 12, 11, 10, dvs JL-bit 3, 2, 1, 0 men i VX29 vänds ordningen så att ut från VX29 kommer JL-bitarna i följande ordning: 0, 1, 2, 3.</p> <ol style="list-style-type: none"> 1. JL-ord nr 1 2. HP-bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 3. JL-bit 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 4. CD-bit 10 11 12 13 10 11 12 13 10 11 12 13 10 11 12 13 5. JL-ord nr 2 6. HP-bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 7. JL-bit 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 8. CD-bit 10 11 12 13 10 11 12 13 10 11 12 13 10 11 12 13 9. Osv. |
| -7 | JL-funktionen SI-data för VX29. Inläsning av två på varandra följande PC21:or samt frysning av buffert då kompareringsvillkoret uppfyllts. Data hämtas med anrop till program TIID och antalsordet satt till två ggr antalsordet i INSYNK.I inhämtad buffert |

Tabell 3: Funktionstabell för GSD-moduler

| Funktion | Överföringstyp |
|-----------------|--|
| 1 | RREPDATA |
| 2 | EPMÅLDATA |
| 3 | VGPPM |
| 4 | VIDEO |
| 5 | Specialfunktion för VG12 (används vid VISVEP) |
| 6 | Specialfunktion för VG12 (specialvariant av funktion 1) Funktionen innebär att RR-DATA ord 10 bit 9 växlar mellan 0 och 1. därmed erhålls en kontinuerlig utmatning på mätdatakanalen från VG12. |
| 7 | Specialfunktion för VG12. Används vid laddning av programminne MDE med makrot LDMDE. |
| 10 | KANALDATA |
| 11 | STYRDATA typ 1 |
| | |
| | |