

ZILOG
Z80-CPU
7738-15703



M-PIO

METRIC CARD



TKS
253200
1002552
81084

TKS
253200
78018



ZILOG
Z80-DTC
7718-08103

ZILOG
Z80-PIO
7802-804

TEKNISK MANUAL



M-PIO

METRIC CARD PIO

Parallellkort med 32 buffrade I/O ledningar



TEKNISK MANUAL
METRIC CARD PIO

INNEHÅLLSFÖRTECKNING

Avsnitt	Sida
1 Presentation	1
2 Tekniska data	2
3 Beskrivning	3
3.1 PIO	3
3.2 Buffer	3
3.3 Avbrott	3
3.4 Anslutning	3
4 Installation	4
5 Adressval, Avbrottsprioritet	4
6 Anpassning, Stiftsplacering	5
7 Programkontroll	6
7.1 Kontrollkommandon	6
7.2 Datakommandon	7
7.3 Programmering	7
8 Exempel parallell BCD från våg	8
8.1 Inkoppling av digital våg	8
8.2 Drivprogram digital våg	9
9 Exempel Z80-assembler larmcentral	10
10 Kontrollporten	11
10.1 Programmering av datariktning	11
10.2 Programmering av avbrott	11
10.3 Programexempel med avbrott	12, 13
11 I/O-adresser på Metric 85	14
12 Komponentplacering och komponentförteckning .	15
13 Kopplingschema	16

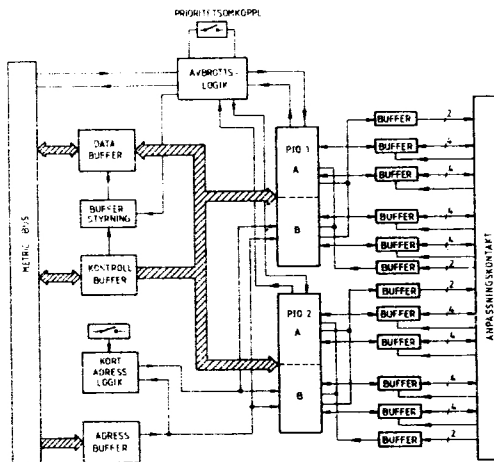
1. PRESENTATION AV METRIC CARD PIO

Metric Card PIO (M-PIO) är ett anpassningskort för 32 stycken parallella digitala signaler med TTL-nivå. Detta kort använder sig av en fullständig Z80-buss som finns i METRIC CARD-serien eller i Metric 85-ans externa buss. Detta ger möjlighet till bl.a. effektiv avbrottsshantering. M-PIO kan valfritt användas för ut- eller inmatning av TTL-signaler i grupper om fyra bitar. All datahantering sker i grupper om åtta bitar, men i programmet kan man hantera enskilda bitar.

----- ANVÄNDNINGSMÅRADE -----

M-PIO kan användas i bl.a. följande applikationer:

- * Digitala mätinstrument med parallella utsignaler (t.ex. BCD)
- * Skrivare med parallellsnitt
- * Statuskontroll av digitala signalnivåer, t.ex. lägesbrytare, larmsignaler
- * Styrning av digital utrustning, t.ex. multiplexer, stegmotorer, reläer
- * Hantering av externa A/D- och D/A-omvandlare



----- UPPBYGGNAD -----

M-PIO är uppbyggt kring två parallellkretsar av typ Z80 PIO som har 2x8 bitar vardera. Dubbelriktade buffertkretsar skyddar PIO-kretsarna och driver TTL ut till omgivningen. Buffertriktningen styrs med TTL-ingångar i anslutningskontakten. Vidare ingår logik för adressavkodning och den avbrottsshantering.

PIO:n initieras från program med avseende på riktning in eller ut och villkor för avbrott. Därefter kan man sända och ta emot data till och från omgivningen.

2. TEKNISKA DATA METRIC CARD PIO

2

==ALLMÄNT==

Spänningsmatning : +5 V
 Strömförbrukning : Min 500 mA (vid alla buffrar inåt).
 Strömmen ökar beronde på extern last.
 Storlek : Enkelt Europakort (100*160 mm)
 Bussanslutning : M-BUS eller extern METRIC 85-buss med
 busskontakt enligt DIN 41612-96/64
 Anpassningskontakt : 64-pin Europadon, DIN 41612-64/64
 Kortadresser : 5 omkopplare ger 32 möjliga adresser, 0 - 248.

==IN-/UTGÅNGAR==

Portar : 4 stycken om 8 bitar benämnda A, B, C och D
 Riktning : Grupper om 4 bitar väljs ut eller in
 Signalnivåer : TTL-nivåer
 Logisk nolla "0" : < 0.8 Volt
 Logisk etta "1" : > 2.4 Volt
 Ström ut
 till signaljord : Max 15 mA
 till + 5 Volt : Max 24 mA
 Extern belastning : Min 200 ohm, resistiv last
 Ström in : 200 uA vid "0", 20 uA vid "1"
 Buffer in-ut : Bidirektionell buffer av typ 74LS243

==PORTAR, PROGRAMMERING==

PORT	FUNKTION
UT, IN A+0	: Dataport A
UT A+1	: Kontrollport A
UT, IN A+2	: Dataport B
UT A+3	: Kontrollport B
UT, IN A+4	: Dataport C
UT A+5	: Kontrollport C
UT, IN A+6	: Dataport D
UT A+7	: Kontrollport D

: A = Kortadress, 0 - 248 i steg om 8
 : Dataport = port för data TTL ut och in
 : Kontrollport = port för initiering av arbetsmode
 på PIO, såsom riktning och avbrottsvillkor

BIT	7	6	5	4	3	2	1	0
DATAPORT	D7	D6	D5	D4	D3	D2	D1	D0
KONTROLLPORT								
Set Mode	M1	M0	-	-	1	1	1	1
Set Int. Enable	INTEN	-	-	-	0	0	1	1
Set Int. Mode 3	INTEN	ANDOR	HILOW	MASKF	0	1	1	1
Set Int.-vector	V7	V6	V5	V4	V3	V2	V1	0

Datariktning

DO - D7 = Data in ut, bit 0 - 7.

MO - M1 = Arbetsmode,

0 = Ut 8 bitar

1 = In 8 bitar

2 = Bidirektionell 8 bitar

3 = Bit, följt av kontrollord för val av In/Ut-förhållande

Avbrottskontroll

INTEN = Interrupt enable

ANDOR = Villkor för avbrott i Mode 3, en eller alla villkorsbitar
 0 = OR avbrott på en bit aktiv.
 1 = AND avbrott på alla bitar aktiva.

HILOW = Aktiv nivå för avbrott.

MASKF = Mask follows, 1 = Mask följer i nästa kontrollord vid
 Mode 3 med låga bitar för aktiva villkorsbitar till avbrott.

V1 - V7 = Avbrottsvektor.

3. BESKRIVNING AV METRIC CARD PIO

Metric Card PIO, som även benämns M-PIO, anpassar TTL-kompatibel utrustning till mikrodatoren.

3.1 ==PIO==

Kortet är uppbyggt kring två Zilog Z80 PIO-kretsar om totalt 32 bitar. Dessa är grupperade på fyra portar om åtta bitar vardera, benämnda port A, B, C och D.

3.2 ==BUFFRAR==

Två dubbelriktade buffrar finns mellan varje port på PIO:n och anslutningskontakten. Varje buffer kan ställas i valfri riktning in eller ut genom en TTL-ingång på anslutningskontakten. Denna signal kan byglas till "0" eller "1" eller styras från annat håll. Bufferriktning och portriktning på PIO-kretsen måste överensstämma med varandra.

3.3 ==AVBROTT==

Det finns en pulsutgång för varje port. Denna kan användas för att klocka in data till den anslutna utrustningen. M-PIO kan även generera avbrott från ansluten utrustning till mikrodatoren. Man väljer med en omkopplare på M-PIO-kortet samt i programmet av PIO-kretsen hur avbrottsfunktionen skall fungera. Se vidare avsnitt 10.

3.4 ==ANSLUTNING==

M-PIO kan användas för många olika typer av digital anpassning. Anpassningen utföres i två steg:

- * Anslutning av TTL-signaler från digital utrustning till M-PIO.
- * Programmering av drivrutin för denna anslutning.

Anslutningen utföres så att man lätt kan ta emot eller sända data. Om man har åtta binära dataledningar från något instrument eller till en parallell skrivare kopplas dessa till samma port t.ex. port A, på anslutningskontakten. Kontrollsignaler kan då kopplas till nästa port.

I följande text beskrivs de fyra lägre bitarna av en buffer med portbokstav och Låg samt de fyra högre bitarna med portbokstav och hög (AHög=port A, bit fyra till sju).

Om man har ett instrument med sex parallella BCD-siffror och några in- och utgående kontrollsignaler, kopplas dessa lämpligen på följande sätt:

- * Port AHög : Kontrollsignaler IN
- * Port ALåg : Första siffra (MSD)
- * Port BHög till CLåg : Siffra två till fem
- * Port DHög : Siffra sex (LSD)
- * Port DLåg : Kontrollsignaler UT

Alla portar utom DLåg byglas för in-riktning på anpassningskontakten. Signalerna för utpulser samt strobingångarna byglas. RDY A byglas till STB A osv. De inåtriktade dataledningar och kontrolledningar som inte används, bör kopplas till jord.

4. INSTALLATION AV METRIC CARD PIO

1. Mikrodatorn och samtliga enheter som skall anslutas samman stängs av.
2. M-PIO-kortet ställs in på den adress som skall användas.
Se appendix 2.
3. M-PIO-kortet placeras i en ledig kortplats i METRIC-CARD-bussen eller i expansionslådan M85-EX85.
4. Anslut kabeln mellan M-PIO-kortet och utrustningen.
5. Slå på strömmen till samtliga ingående enheter.
6. Prova M-PIO-s funktion med ex. nedanstående program i CBASIC:
PIOUTBIT sätter ett allt högre värde ut på alla portar. Den lägsta biten byter nivå oftast, medan allt högre bitar halverar nivåbytet tidsmässigt.

```

10 ***** PIOUTBIT *****HMSCM 810731*****
20 FOR A%=193% TO 199% STEP 2%'..... Loop för alla portar utåt
30 OUT A%,15%'..... Sätt en port ut
40 NEXT A%
50 FOR A%=0% TO 255%'..... Alla data ut
60 OUT 192%,A% : OUT 194%,A% : OUT 196%,A% : OUT 198%,A%
70 NEXT A% : GOTO 50'..... Nästa data och repetera

```

PIOINBIT visar bitmönstret på alla portar, när dessa är inåtriktade:

```

10 ***** PIOINBIT *****HMSCM 810731*****
20 ? CHR$(12);"BITTEST PÅ M-PIO INGANGAR"
40 ? CUR(2,0);" PORT A PORT B PORT C PORT D"
50 ? CUR(4,0);"76543210 76543210 76543210 76543210"
53 FOR I%=193% TO 199% STEP 2% : OUT I%,79% : NEXT I%'.Sätt alla in
60 ? CUR(6,0);'..... Position av bitmönster
70 FOR P%=192% TO 198% STEP 2%'..... Läs alla kanaler in
80 A%=INP(P%)'..... Läs en port
90 J%=128%
100 FOR I%=7% TO 0% STEP -1%'..... Loop för bitmönster
110 IF A% AND J% THEN ? "1"; ELSE ? "0";'..... Skriv bitmönster
120 J%=J%/2%
130 NEXT I%
140 ? " ";
150 NEXT P% : GOTO 60

```

5. ADRESSVAL, AVBROTTPRIORITET PÅ METRIC CARD PIO

Adressval utförs med 5 omkopplare i position SW1 på M-PIO-kortet. Adressen rekommenderas till att i första hand vara mellan 192 till 240. Omkopplarna är märkta 1 till 6, där de motsvarar en vikt enligt vidstående tabell. Genom att addera ihop vikterna på omkopplare som står i läge OPEN, får man adressen. När omkopplare 1 och 2 står i läge OPEN, får man adress 192. Adressvalet benämns i denna manual med bokstaven A. Se även avsnitt 11 för adresser till Metric 85.

SW1 NR	VIKT
1	128D, 80H
2	64D, 40H
3	32D, 20H
4	16D, 10H
5	8D, 08H
6	
7	
8	INT.PR.

Med omkopplare 8 (INT.PR.) väljer man prioritet mellan PIO-na vid avbrott. I läge OPEN har Port AB högre prioritet än Port CD.

6. ANPASSNING, STIFTSPLACERING PÅ METRIC CARD PIO

Nedanstående tabell visar hur signalerna är placerade på stiften i anslutningskontakten på M-PIO. Två lika grupper finns, en till varje PIO, som innehåller samtliga signaler. Anslutningskontakten på M-PIO är ett Europadon 64/64-stifts kretskortsdon.

Data A0-D7 (I/O) Data in/ut-signal för respektive port.

DIRAL - DIRDH (I) Riktningväljare (DIR) för respektive buffer för in- eller utgående data. Buffern är utåtriktad när denna signal är "1" eller fri och inåtriktad om den är "0", alltså ansluten till signaljord. "L" gäller bit 0 till 3, och "H" bit 4 till 7 för respektive port.

RDY A-D (O) READY ut från respektive port vid arbetsmode 0 till 2, markerar att PIO-n har laddat eller hämtat data från denna port.

STB A-D (I) STROBE-puls in till respektive port, för att klocka in eller ut data till PIO-n, vid arbetsmode 0 till 2. Används oftast tillsammans med avbrott.

OBS. RDY - STB När RDY och STB ej används, skall dessa sammankopplas för respektive port. Vid arbetsmode 0 och 1, ges då en puls ut på 0.3 usek. för varje gång som man läser eller skriver till denna port.

SIGNAL	STIFT	STIFT	SIGNAL
Data A1	A 01	B 01	Data A0
Data A3	A 02	B 02	Data A2
Data A5	A 03	B 03	Data A4
Data A7	A 04	B 04	Data A6
Data B1	A 05	B 05	Data B0
Data B3	A 06	B 06	Data B2
Data B5	A 07	B 07	Data B4
Data B7	A 08	B 08	Data B6
Signaljord	A 09	B 09	Signaljord
Signaljord	A 10	B 10	Signaljord
DIRAL	A 11	B 11	DIRAH
DIREL	A 12	B 12	DIRBH
RDY A	A 13	B 13	RDY B
STB A	A 14	B 14	STB B
+ 5 Volt	A 15	B 15	+ 5 Volt
Ledig	A 16	B 16	Ledig
Data C1	A 17	B 17	Data C0
Data C3	A 18	B 18	Data C2
Data C5	A 19	B 19	Data C4
Data C7	A 20	B 20	Data C6
Data D1	A 21	B 21	Data D0
Data D3	A 22	B 22	Data D2
Data D5	A 23	B 23	Data D4
Data D7	A 24	B 24	Data D6
Signaljord	A 25	B 25	Signaljord
Signaljord	A 26	B 26	Signaljord
DIRCL	A 27	B 27	DIRCH
DIRDL	A 28	B 28	DIRDH
RDY C	A 29	B 29	RDY D
STB C	A 30	B 30	STB D
+ 5 Volt	A 31	B 31	+ 5 Volt
Ledig	A 32	B 32	Ledig

7. PROGRAMKONTROLL AV METRIC CARD PIO

Kortets arbetssätt bestäms i två steg:

1. Byglingar i anslutningskontakten styr t.ex. val av IN/UT-gång
2. Programmering av PIO-kretsarna som styr datariktning, arbetsmod och avbrott.

Från mikrodatorn initierar man kortet, styr, läser och skriver med speciella kommandon. Man kan dela upp dessa i två grupper:

1. Kontrollkommandon för styrning av M-PIO-kortet.
2. Datakommandon för dataöverföring via M-PIO-kortet.

7.1 ==KONTROLLKOMMANDON==

Man börjar alltid med att via kontrollkommandon initiera PIO-kretsarna i samma datariktning som buffrarna är byglade till. Detta avsnitt behandlar den enklare typen av TTL-styrning av PIO:n. Mer utförlig beskrivning av kontrollkommandona redovisas i avsnitt 10, t.ex. de som behövs för att definiera avbrotts hanteringen.

PORT	KONTROLLKOMMANDO
A	UT A+1,C
B	UT A+3,C
C	UT A+5,C
D	UT A+7,C

A markerar den kortadress i decimalform som valts på kortet.
C markerar i vilken mode som porten skall arbeta. I nedanstående kommandospalt syftar "KPORT" på ett av ovanstående portnummer, alltså A+1, A+3, A+5 eller A+7.

RIKTNING	MODE	KOMMANDO C	BYGLING TILL JORD
UT	UT	UT KPORT, 15	
IN	IN	UT KPORT, 79	DIR-KPORT-L, DIR-KPORT-H
Hög UT Låg IN	BIT	UT KPORT, 207 UT KPORT, 15	DIR-KPORT-L
Hög IN Låg UT	BIT	UT KPORT, 207 UT KPORT, 240	DIR-KPORT-H

==EXEMPEL==

I nedanstående exempel visas hur kommandot ser ut i CBASIC.
Antag att adressen är byglad till 192 decimalt eller C0 Hex.

- A. Ställ port C i utmode: OUT 192+5,15
Bygla ej stift A28 och B28 till jord.
- B. Ställ port A i inmode: OUT 192+1,79
Bygla stift A11 och B11 till jord (A10).
- C. Ställ port D i in/utmode dvs.
DLåg ut och DHög in : OUT 192+7,207
OUT 192+7,240
Bygla stift B29 till jord (B26).

7.2 ==DATAKOMMANDON==

Med datakommandona överför man data till och från den anslutna utrustningen. Före dessa kommandon skall man dessförinnan ha byglat för bufferriktning och anslutit utrustningen till anslutningskontakten, samt initierat M-PIO med kontrollkommandon enligt föregående sida. För att detektera en enskild bit, måste man skala respektive bit med hjälp av logiska operationer och bitens vikt eller använda bittestfunktioner i assembler.

I nedanstående tabell är A kortadressen.

PORT	UTKOMMANDO	INKKOMMANDO
A	UT A	IN (A)
B	UT A+2	IN (A+2)
C	UT A+4	IN (A+4)
D	UT A+6	IN (A+6)

==UT/INDATA==

Om man läser av en utåtriktad port med IN, får man samma data som vid föregående skrivning med UT till denna port. Även vid blandad ut/in-riktning av en port används ovanstående kommandon. De bitar som skrivs ut till en inåtriktad port har ingen funktion, medan de bitar som hämtas in från en utåtriktad port, är samma som vid föregående skrivning dit.

==EXEMPEL==

Nedanstående exempel visar hur datakommando ser ut i CBASIC. Det förutsätts att exempel i avsnitt 7.1 är utförda.

- Ettställ enbart bit ett och sju till port C
Kommando blir: OUT A+4,130
- Läs in data från port A till variabeln D
Kommando blir: D=INP(A)
- Läs av ett BCD-värde på port DHög till variabeln D
Kommando blir: D=(INP(A+6) AND 240)/16

7.3 ==PROGRAMMERING==

Här presenteras hur kommandona kan hanteras på de olika språken. I assembler finns det fler möjligheter än nedanstående kommandon.

PORT	Z80-ASM	M85-ASM	CBASIC
Data ut A	OUT (A),r	OUT A	OUT A,D
Data in A	IN r,(A)	INP A	D=INP(A)
Kontroll A	OUT (A+1),r	OUT A+1	OUT A+1,D
Data ut B	OUT (A+2),r	OUT A+2	OUT A+2,D
Data in B	IN r,(A+2)	INP A+2	D=INP(A+2)
Kontroll B	OUT (A+3),r	OUT A+3	OUT A+3,D
Data ut C	OUT (A+4),r	OUT A+4	OUT A+4,D
Data in C	IN r,(A+4)	INP A+4	D=INP(A+4)
Kontroll C	OUT (A+5),r	OUT A+5	OUT A+5,D
Data ut D	OUT (A+6),r	OUT A+6	OUT A+6,D
Data in D	IN r,(A+6)	INP A+6	D=INP(A+6)
Kontroll D	OUT (A+7),r	OUT A+7	OUT A+7,D

A = Kortadress

B = Data in eller ut

r = CPU-reg A då porten (A+X) är en konstant eller

r = CPU-reg A, B, C, D, E, H el. L då porten = (CPU-reg. C).

8. INKOPPLING AV PARALLELL BCD, EN DIGITAL VÅG TILL METRIC CARD PIO

Nedanstående exempel redovisar dels hur man ansluter en digital våg med 5 parallella BCD-siffror till M-PIO samt hur ett program i CBASIC kan se ut för att mäta värdet.

8.1 == KOPPLINGSBESKRIVNING ==

Inkopplingen består av 5 BCD-siffror med exponent och tecken samt en del kontrollfunktioner. Ingångar till M-PIO som inte används, skall kopplas till signaljord GND.

VÅGSIDAN		M - P I O - S I D A N			
Anm.	Stift	Stift	Funktion	Anmärkning	Kommando
READY	47 --	A4	A7	READY	} Status Port A Data : A Kontr: A+1
ERROR	45 --	B4	A6	ERROR	
SIGN	43 --	A3 B3-A10	A5 A4	SIGN FlagA	
8*10E4	25 --	A2	A3	8 Digit 1	(MSD)
4*10E4	24 --	B2	A2	4	
2*10E4	23 --	A1	A1	2	
1*10E4	22 --	B1	A0	1	
8*10E3	20 --	A8	B7	8 Digit 2	
4*10E3	19 --	B8	B6	4	Port B Data : A+2 Kontr: A+3
2*10E3	18 --	A7	B5	2	
1*10E3	17 --	B7	B4	1	
8*10E2	15 --	A6	B3	8 Digit 3	
4*10E2	14 --	B6	B2	4	
2*10E2	13 --	A5	B1	2	
1*10E2	12 --	B5	B0	1	
8*10E1	10 --	A20	C7	8 Digit 4	Port C Data : A+4 Kontr: A+5
4*10E1	9 --	B20	C6	4	
2*10E1	8 --	A19	C5	2	
1*10E1	7 --	B19	C4	1	
8*10E0	5 --	A18	C3	8 Digit 5	
4*10E0	4 --	B18	C2	4	
2*10E0	3 --	A17	C1	2	
1*10E0	2 --	B17	C0	1	
DP3	11 --	A24-A31 B24-B25	D7	Exp +/-	
DP2	6 --	A23	D6	Exp 4	
REQ.IN	48 --	B23	D5	Exp 2	
		A22	D4	Exp 1	
		B22	D3	UT REQ. }	
		A21	D2	UT 4 }	
		B21	D1	UT 2 }	
GND	50 --	A10-B10	D0	UT 1 }	ut-signal
		B11-B10	DIRAH	IN Port AHög	
		A11-A10	DIRAL	IN Port ALåg	
		B12-B10	DIRBH	IN Port BHög	
		A12-A10	DIRBL	IN Port BLåg	
		B28-B27	DIRCH	IN Port CHög	
		A28-A27	DIRCL	IN Port CLåg	
		B29-B27	DIRDH	IN Port DHög	
		A29	DIRDL	UT Port DLåg, ingen bygling	

REQ.IN = Handskakning sätts hög före avläsning, för att låsa mätvärde.

READY = Klarsignal ut inväntas aktiv hög.

ERROR = Överlastmarkering är aktiv låg vid överbelastning av vågen.

* FLAG A = Extern flaggmarkör för att avläsa någon typ av status.

Exp = Exponent bildas av decimalsättningen DP2 och DP3, samt i detta fall fast byglat minustecken.

* UT1, UT2, UT4 = Binära utgångar för annan styrning än request. Man kan använda den till extern tarering av våg, styrning av provväxlare, kanalval och dylikt.

* markerar funktioner som ej används i denna uppkoppling.

8.2 ==DRIVPROGRAM DIGITAL VAG==

Här nedan följer ett program för avläsning av vågen i föregående inkoppling. Först initieras M-PIO och sedan utförs handskakning med UTREQ och READY. Därefter avläses alla BCD-siffror till en ASCII-sträng, avskalas och lämnar resultat till huvudrutinen.

```

10 ! ***** VIKTBCD *****
20 ! 810604 HMSCM
30 GOSUB 1000 : ! Initiering
40 GOSUB 1090 : ! Ta in ett mätvärde
50 ON ERROR 90
60 V=VAL(V$) : ! Omvandla sträng till värde
70 PRINT V
80 GOTO 40
90 PRINT "ÖVERLAST"
100 GOTO 40

1000 ! ===== VÄGINITIERING =====
1010 ! UTFÖRS EN GANG FÖRE LÄSVAG
1020 ! =====
1030 OUT 193%,79% : ! Inmode PORT A
1040 OUT 195%,79% : ! Inmode PORT B
1050 OUT 197%,79% : ! Inmode PORT C
1060 OUT 199%,207%,199%,240% : ! Inmode DLåg, Utmode DHög
1070 RETURN
1080 ! ===== LÄSVAG =====
1090 ! AVLÄS BCD-VÄG PÅ M-PIO-KORT
1100 ! UTVARIABEL V$=VÄGVÄRDE
1110 ! OM V$=" " +värde ÄR VÄGEN ÖVERLASTAD
1120 ! =====
1130 OUT 198%,8% : ! Request UT på bit D3
1140 V1%=INP(192%) : ! Läs status
1150 IF (V1% AND 128%)=0% THEN 1150 : ! Vänta Ready
1160 V$=""
1170 FOR V2%=192% TO 198% STEP 2% : ! Läsloop fyra portar
1180 V%=INP(V2%) : ! Läs en port
1190 V$=V$+CHR$( (V% AND 240%)/16%+48%) : ! Bilda ASCII av BCD-Hög
1200 V$=V$+CHR$( (V% AND 15%)+48%) : ! Bilda ASCII av BCD-Låg
1210 NEXT V2%
1220 OUT 198%,0% : ! Ingen mer REQUEST UT
1230 V$=MID$(V$,2%,5%) : ! Ta ut data
1240 IF (V1% AND 32%)=0% THEN V$=" "+V$ ELSE V$="-"+V$ : ! tecken
1250 IF (V1% AND 64%)=0% THEN V$=" "+V$ ELSE V$=" "+V$ : ! errormärke
1260 V$=V$+"E"+CHR$( (V%/16% AND 7%)+48%) : ! V%= Port D, exponentvärde
1270 RETURN

```

Den utgående strängen V\$ ser ut på följande sätt:

Pos.	1	2	3	4	5	6	7	8	9	10
Data	Err	+/-	BCD1	BCD2	BCD3	BCD4	BCD5	"E"	"-"	EXP

+/- = Tecken på data

Err= Felmärke " " om ERROR är aktiv låg eller " " om ERROR är hög. BCD1 - BCD5 är siffrorna.

"E","-" = Exponentmärke och tecken

9. ENKELT ALARMSYSTEM MED 24 INGANGAR MED METRIC CARD PIO

Nedanstående exempel visar hur ett enkelt alarmsystem är programmerat i Z80-assembler. Larm kommer in på port A, B och C i form av en TTL-signal hög. Numret på kanalen presenteras då på en två-siffrig display, som styrs med BCD-kod ut, samt att alarm-signalen blir aktiv.

	PORT A								PORT B								PORT C								PORT D								
Bit	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	
Kanal	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	BCD-	BCD-	Alarm						
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	Ental	Total	ut						

--ASSEMBLERLISTNING==

LOC	OBJCODE	M	STMT	SOURCE	STATEMENT	PAGE	1
					M85PIOZ		ASM 5.8
			1		;***** M85PIOZ *****		
			2		;ALARMSYSTEM FÖR 24 KANALER MED		
			3		;PRESENTATION AV ALARMNUMMER PÅ BCD		
			4		;-----		
			5	PIOADR	EQU 0COH		;M-PIO kortadress
			6	INMODE	EQU 04FH		;PIO-inmode-kod
			7	OUTMODE	EQU 0FH		;PIO-utmode-kod
			8		INIT		
0000	3E4F		9		LD A,INMODE		;Sätt port A,B&C in
0002	D3C1		10		OUT (PIOADR+1),A		;IN port A
0004	D3C3		11		OUT (PIOADR+3),A		;IN port B
0006	D3C5		12		OUT (PIOADR+5),A		;IN port C
0008	3E0F		13		LD A,OUTMODE		;Sätt Port D ut
000A	ED79		15		OUT (PIOADR+7),A		;UT port D
			16		HLOOP		
000C	0EC0		17		LD C,PIOADR		;Data port A
000E	ED50		18	HLOOP1	IN D,(C)		;Läs data från port (C)
0010	0601		19		LD B,1		;BCD-siffror 1
0012	CB02		20	HLOOP2	RLC D		;Roterar data
0014	3015		21		JR NC,HLOOP4		;Hopp om ej alarm
0016	79		22		LD A,C		;A=aktuell PIOADR
0017	D6C0		23		SUB PIOADR		;A=0,2,4
0019	07		24		RLCA		;Roterar till BCD10
001A	07		25		RLCA		;
001B	07		26		RLCA		;
001C	F680		27		OR 080H		;Sätt alarm
001E	E6F0		28		AND 0F0H		;Nollställ BCD1
0020	B0		29		OR B		;Lägg på BCD1
0021	D3C6		30		OUT (PIOADR+6),A		;Styr BCD + alarm
0023	210000		31		LD HL,0		;Delay ca 1 sek.
0026	2B		32	HLOOP3	DEC HL		;
0027	7C		33		LD A,H		;
0028	B5		34		OR L		;
0029	20FB		35		JR NZ,HLOOP3		;
002B	AF		36	HLOOP4	XOR A		;A=0
002C	D3C6		37		OUT (PIOADR+6),A		;Nollställ port
002E	78		38		LD A,B		;Test för max siffror
002F	FE09		39		CP 8+1		;
0031	20DF		40		JR NZ,HLOOP2		;Nästa siffror
0032	0C		41		INC C		;Peka nästa dataport
0033	0C		42		INC C		;
0034	79		43		LD A,C		;
0035	FEC6		44		CP PIOADR+6		;Port C + 2
0037	20D4		45		JR NZ,HLOOP1		;Till nästa port
0039	18D0		46		JR HLOOP		;Till port A

10. KONTROLLPORTEN PÅ METRIC CARD PIO

Med kontrollporten definieras hur PIO-kretsen skall arbeta. Här följer en kortfattad beskrivning dess funktioner.

BIT	7	6	5	4	3	2	1	0
Set Mode	M1	M0	-	-	1	1	1	1
Set Int.-vector	V7	V6	V5	V4	V3	V2	V1	0
Set Int. Enable	INTEN	-	-	-	0	0	1	1
Set Int. Mode 3	INTEN	ANDOR	HILOW	MASKF	0	1	1	1

10.1 ==PROGRAMMERING AV DATARIKTNING==

"Set Mode" bestämmer riktningen av data på PIO-n med bitarna M0 och M1, vilket ger fyra möjliga kombinationer:

M1, M0

0 0 Utmode 0 överför data från CPU-n till dataporten, alla 8 bitar, vid skrivning till denna dataport. Utskriven data kan även avläsas med läsning av samma dataport.

0 1 Inmode 1 överför data från dataporten till CPU-n, alla 8 bitar, vid avläsning från denna dataport. Data som skrivs till porten har ingen funktion.

1 0 Bidirektionell mode 2 överför data mellan dataporten och CPU-n med hjälp av alla RDY och STB-ledningar på den PIO-n. Därför kan enbart den första porten (port A och C) ställas i denna mode, medan den andra porten rekommenderas till bitmode 3.

1 1 Bitmode 3 ger möjlighet till valfri riktning på varje bit på dataporten. Riktningen på varje bit bestäms av nästa skrivning till kontrollporten, där de 8 bitarna definierar riktningen.

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Riktning	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0

När en IO-bit är 0 ställs riktning ut och vid 1 riktning in.

10.2 ==PROGRAMMERING AV AVBROTT==

Vid programmering av avbrott skall man ta hänsyn till fler funktioner i PIO-n.

"Set Int.-vector" initierar PIO-ns avbrottsvektor. Vektorn används tillsammans med CPU-registret I, för att kunna peka ut var i minnet pekaren till avbrottsrutinen finns vid begärt avbrott, enligt det vektoriserade avbrottsystem som Z80 CPU har.

"Set Int. Enable" används för att reglera möjligheten att denna port skall kunna ge avbrott (INTEN), där "1" möjliggör avbrott. Avbrott genereras i mode 0 - 2 av STB-ingången.

"Set Int. Mode 3" används i mode 3 för att definiera villkor som skall gälla för att avbrott skall kunna ske från någon eller några bitar i denna port.

ANDOR anger vilken logisk operation för avbrott. "0" ger avbrott då en bit är aktiv (OR) och "1" då alla bitar aktiva (AND).

HILOW anger vilken aktiv nivå som gäller för avbrott.

MASKF är "1" när mask följer i nästa kontrollord som anger vilka bitar (MB) som skall kunna generera avbrott.

Bit	D7	D6	D5	D4	D3	D2	D1	D0
Maskebit	MB7	MB6	MB5	MB4	MB3	MB2	MB1	MB0

När en bit är 0 blir denna portbit aktiv för avbrott.

För mer detaljerad beskrivning av PIO-ns programmering se:

- * Zilog Z80 PIO Technical Manual
- * Applications notes: Z80 family program interrupt structure
- * Advanced programming manual for the COMPUCORP 600 series

10.3 ==PROGRAMEXEMPEL MED AVBROTT==

Avbrottsrutinen som tar vara på avbrottsbegäran, måste vara i assembler, medan initiering och annan hantering till följd av avbrottet, kan vara i något högnivåspråk.

I CBASIC på Metric 85 finns satsen "TRAP EXT TO", som utför GOSUB på t.ex. avbrott från M-PIO. En avbrottsrutin i assembler tar hand om avbrottet från M-PIO och sätter en flagga på adress 1104Hex. Flaggan avbryter exekveringen i det ordinarie BASIC-programmet och istället exekverar subrutinen i BASIC på radnummer som angavs efter sista "TRAP EXT TO".

Detta exempel visar hur TRAP EXT TO kan användas med M-PIO. Medan ett BASIC-program håller på med utskrift av vissa beräkningar, kommer BASIC-avbrottsrutinen att läsa in 100 stycken 2-siffriga BCD-värden från port A. När 100 mätvärden lästs in, kommer dessa att skrivas ut. Ett M-PIO-kort med adress 128 används för att generera avbrott varje gång bit noll går hög på port B.

==BASIC-rutin==

```

10 ! ***** INTBCDIN *****810605 hmsem
20 DIM A%(100)..... Matris för 100 mätvärden
30 TRAP EXT TO 500..... Vid ett avbrott skall GOSUB 500 utföras
35 K%=131..... Adress till kontrollport B
40 OUT 129%,&8F..... Port A Inmode
50 OUT K%,&CF..... Port B Bitmode
60 OUT K%,&FF..... Port B Alla bitar in
70 OUT K%,&37..... Int.mode 3: No INTEN, OR, Hög aktiv, MASK
80 OUT K%,&FE..... Mask, aktiv bit 0
90 F%=1% : CALLP "PIOOBJ",F%,K%..Sätt på avbrott
100 ` Utför här annat medan mätningen pågår
110 E%=E%+1%
120 PRINT E%,SQR(E%),SIN(2*PI/100*E%)
130 IF I% <100% THEN 110.. Invänta alla mätningar klara
140 F%=0% : CALLP "PIOOBJ",F%,K%..Stäng av avbrott före programavslut
150 PRINT "MÄTVÄRDEN"
160 FOR I=0 TO 99..... Skriv samtliga mätvärden
170 PRINT I,A%(I)
180 NEXT I
190 END
500 ` *****SUBRUTIN FÖR AVBROTTSHANTERING*****
510 X%=INP(128)..... Läs BCD-data
520 A%(I%)=(X% AND 240%)/16%*10%+(X% AND 15%)... Gör BCD-värde
530 I%=I%+1%..... Öka antal med ett
540 Z%=1% : CALLP "PIOOBJ",Z%,K%.... Sätt på avbrott igen
550 TRAP EXT TO 500
560 RETURN ..... Fortsätt exekvera på ordinarie program före avbrott

```

==ASSEMBLER-RUTIN==

Programmet PIOAVBR i assembler på Metric 85 används för att initiera och ta hand om avbrott från PIO. Denna rutin anropas från CBASIC med CALLP "PIOAVBR",F%,K%, där F%=Avbrottsfunktion (0=Av 1=På) och K%=Adress på PIO-ns kontrollport. Rutinen använder sig av LION (Logical I/O Nucleus) i operativsystemet, för att med hjälp av detta reservera och returnera avbrottsvektor. Mer information om detta finns i boken "Advanced programming manual for the COMPU CORP 600 series".

METRIC 85 RELOCATING ASSEMBLER V6.02
 PIOOBJ - PIOAVBR -0000

PAGE 1

STMT 1

ERROR CHAIN 0

```

;*****;
;PIOAVBR för avbrottsshantering på M-PIO till TRAP EXT ;
; F%=1 aktiverar avbrott och vektor på port K%. ;
; F%=0 passiverar avbrott och vektor på port K%. ;
; För CBASIC 7.03 och senare 821002 hmscm;

```

```

0000' 22 006D'      SHLD  STACK  ;Spara returadress
0003' 79           MOV   A,C    ;
0004' FE02        CPI   2      ;Kontroll om två argument
0006' C2 005A'    JNZ  EXIT   ;Hopp om ej två arg.
0009' E1         POP   H       ;HL pekar K%
000A' C1         POP   B       ;BC = arg.typ
000B' 79         MOV   A,C    ;
000C' FE02        CPI   2      ;Typ 2 = heltal ?
000E' C2' 005A'   JNZ  EXIT   ;Hopp om ej heltal
0011' 7E         MOV   A,M     ;Hämta K%
0012' 32 0070'   STA  P1OKADR ;Lagra Kontrolladress
0015' E1         POP   H       ;HL pekar F%
0016' C1         POP   B       ;BC = arg.typ
0017' 79         MOV   A,C    ;
0018' FE02        CPI   2      ;Heltal ?
001A' C2 005A'   JNZ  EXIT   ;Hopp om ej heltal
001D' 7E         MOV   A,M     ;Hämta F%
001E' A7         ANA  A        ;Testa om F%=0
001F' CA 0047'   JZ   ..AV    ;Hopp om avbrott av
0022' 3A 006F'   LDA  VEKTOR  ;Påslag avbrott
0025' A7         ANA  M        ;Finns redan en vektor ?
0026' C2 003A'   JNZ  ..EJV   ;Hopp om vektor finns
0029' 01 0089'   LXI  B,89H   ;LION-funktion = "ALOCIV"
002C' 3EFF       MVI  A,OFFH  ;A = valfri vektor
002E' 11 0072'   LXI  D,AVDTA ;DE = adress till avbrottsrutin
0031' 21 005F'   LXI  H,AVBR  ;HL = adress till avbrottsrutin
0034' CD 10BD   CALL LION    ;LION-funktion reservera ledig vektor
0037' 32 006F'   STA  VEKTOR  ;Lagra reserverad vektor
003A' ED4B 0070' ..EJV: LBCD  P1OKADR ;C = P1OKADR
003E' ED79       OUTP A        ;Vektor till PIO-port
0040' 3E83       MVI  A,83H   ;
0042' ED79       OUTP A        ;Påslag avbrott
0044' C3 005A'   JMP  EXIT   ;-----
0047' ED4B 0070' ..AV:  LBCD  P1OKADR ;Avstängning av avbrott
004B' ED41       OUTP B        ;Ge PIO kommando att stänga av
004D' 3A 006F'   LDA  VEKTOR  ;
0050' 01 008A   LXI  B,8AH   ;Funktion = "DEALOCIV"
0053' CD 10BD   CALL LION    ;LION-FUNKTION Återlämna vektor
0056' AF        RA  A         ;
0057' 32 006F'   STA  VEKTOR  ;Nollställ egen vektor
005A' ED7B 006D' EXIT:  LSPD  STACK ;
005E' C9        ET             ;-----
005F' ED4B 0070' AVBR:  LBCD  P1OKADR ;Avbrottsrutin, hämta portadress
0063' ED41       OUTP B        ;Stäng avbrott på PIO-n
0065' 3EFF       MVI  A,OFFH  ;Sätt flagga
0067' 32 10A4   STA  10A4H   ; till TRAP EXT i BASIC
006A' FB        EI             ;
006B' ED4D       RETI          ;Retur efter avbrott
006D' 0000      STACK: .WORD 0 ;
006F' 00       VEKTOR: .BYTE 0 ;
0070' 00       P1OKADR: .BYTE 0 ;
0071' 07       .BYTE 7      ;Kod för avstängning av avbrott
0072' 00       AVDTA:  .BYTE 0 ;Avbrottsdata, används ej
10BD          LION      ==    10BDH ;LION Entry point

.END

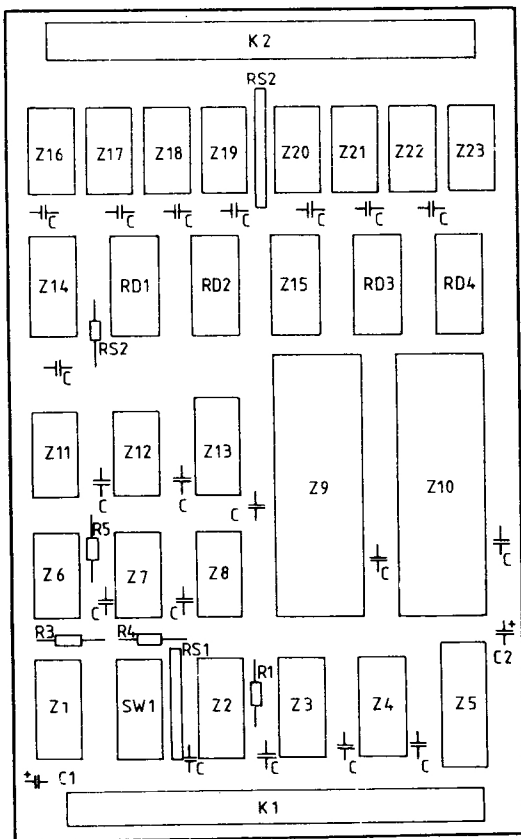
```

11. I/O-ADRESSER PÅ METRIC 85

Nedanstående tabell redovisar hur de olika enheterna i Metric 85 är anslutna till mikrodatorn. De adresser som är utmärkta med flera bindestreck kan användas till M-PIO och andra separata anpassningskort till Metric 85.

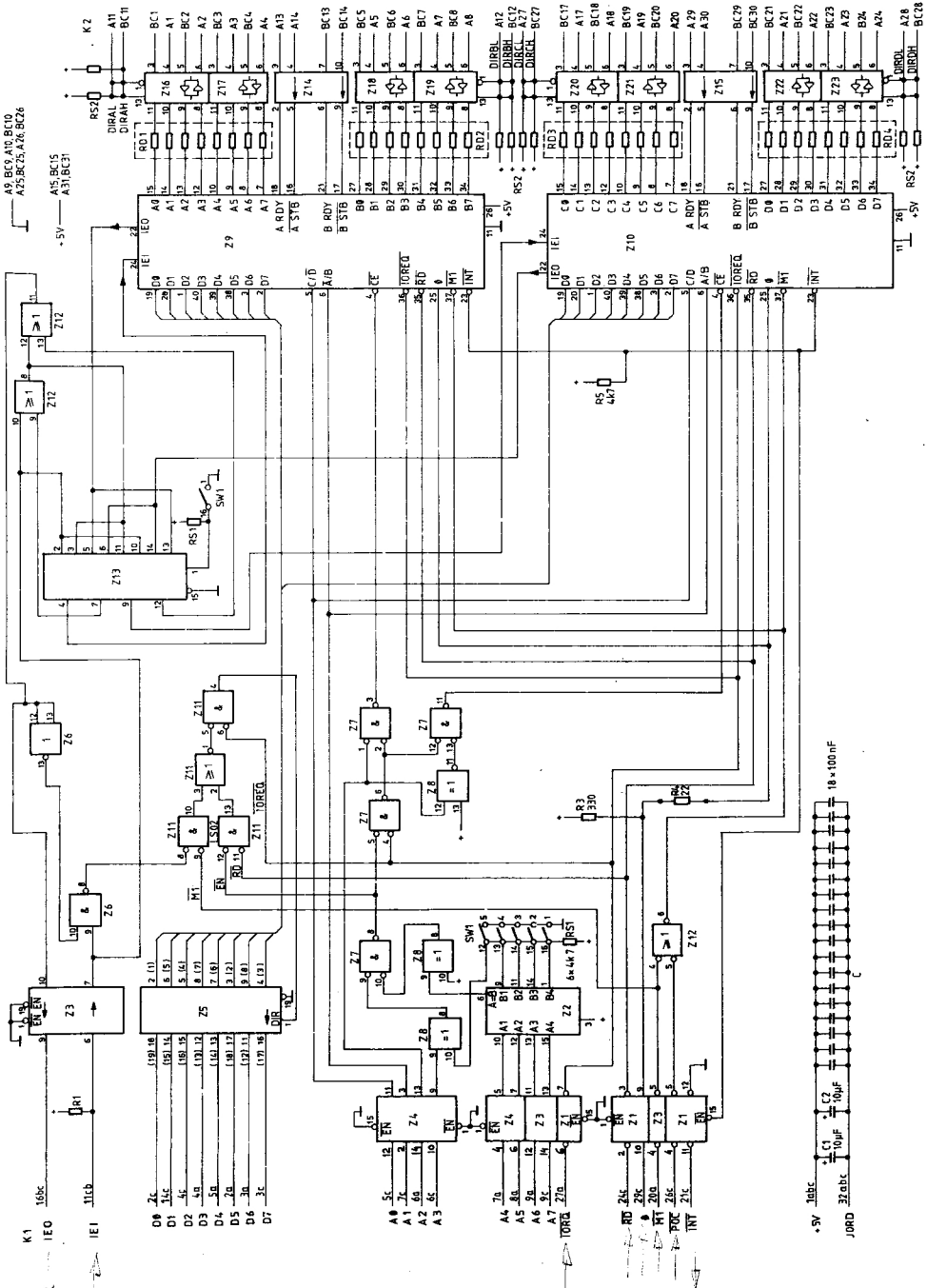
Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x	EIGHT INCH FLOPPY DISK CONTROLLER -675															
1x	691-RIGID DISK				RAM				EXTERN USER				IEE-488 INTERFACE			
2x	Z80-SIO				Z80-CTC				Z80-PIO				KEYBD		DMA	
3x	RESERVED FOR I/O-EXPANSION															
4x	PARALLEL 0 (2 ports)								(CENTRONICS) PARALLEL 1 (2p)							
5x	SIO-0 (2 channels)								SIO-1 (2 channels)							
6x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7x	60 line w.p.ctrl				24 line w.p.				12" CRT CONTROL							
8x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Ax	-	-	60-LINE DISPLAY REFRESH MEMORY OR EXTERNAL USER										-	-		
Bx	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cx	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Dx	-	-	-	-	-	-	EXTERNAL USER				-	-	-	-	-	-
Ex	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Fx	-	-	-	-	-	-	-	-	12" CRT REFRESH							

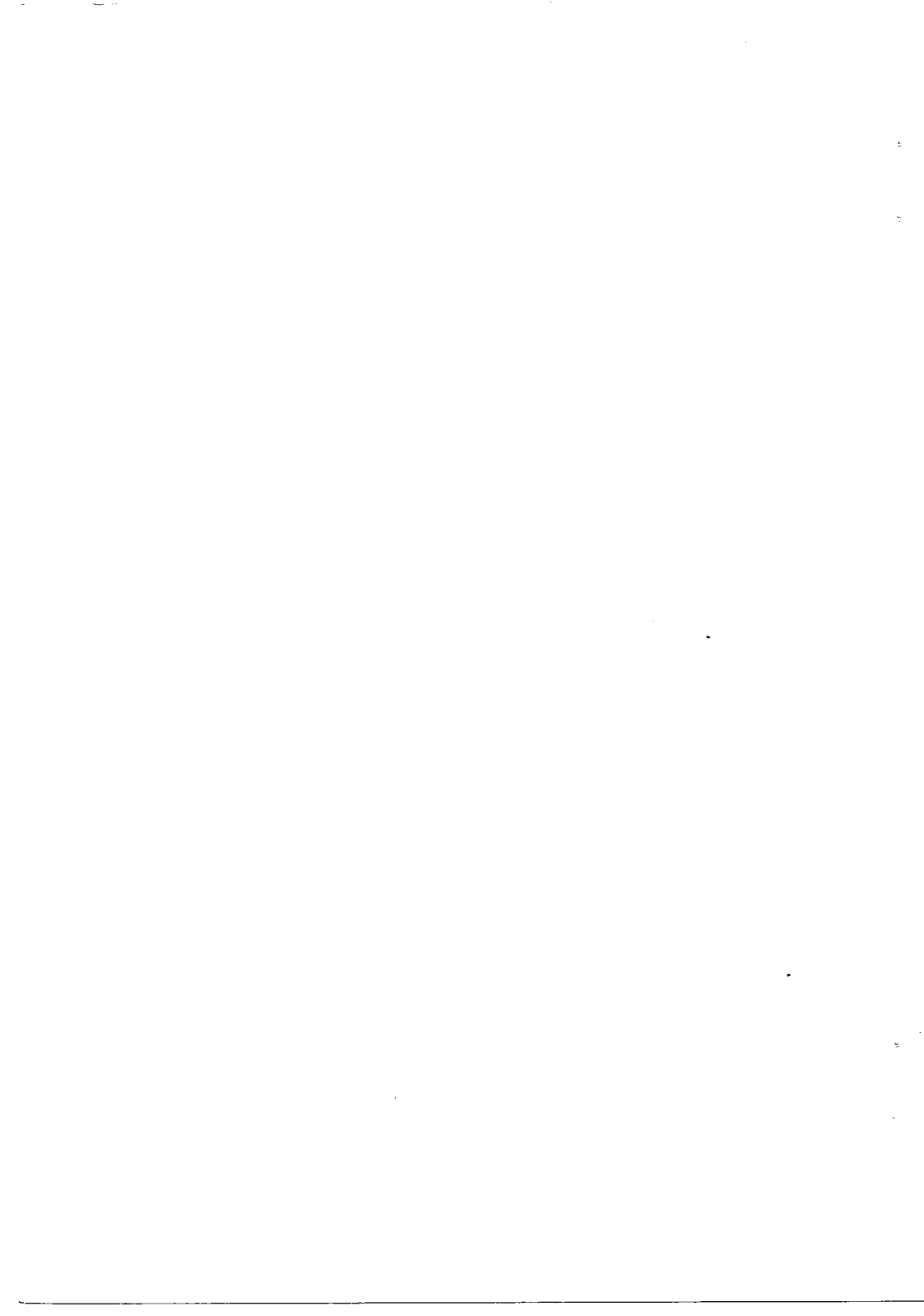
13. PLACERINGSSCHEMA OCH KOMPLEMENTFÖRTECKNING PÅ METRIC CARD PIO



Z1, Z3, Z4	74LS367
Z2	74LS85
Z5	74LS245
Z6	74LS00
Z7	74LS32
Z8	74LS86
Z9, Z10	ZILOG Z80A-PIO
Z11	74LS02
Z12	74LS08
Z13	74LS157
Z14, Z15	74LS367
Z16 - Z23	74LS243

RS1, RS2	SIL 4,7 Kohm
RD1 - RD4	DIL 1 Kohm
R1, R5	4,7 Kohm
R3	330 Ohm
R4	22 Ohm
C1, C2	10 uF Tantal
C	100 nF Ker.
K1	Europadon DIN 41612-96/64
K2	Europadon DIN 41612-64/64





LÄSARENS KOMMENTARER

Dina synpunkter på denna dokumentation är viktig för oss; Det hjälper oss till förbättringar och kommer dig till gagn i framtiden. Var vänlig att besvara frågorna i formuläret och sänd det till oss.

Tack.

Ditt namn:

Företagets namn:

Adress:

Postnr och postadress:

Dokumentets namn:

Hårdvarukonfiguration:

Aktuell programvara:

Uppfyller dokumentationen dina behov ? Ja Nej

Om Nej, varför inte?

.....

.....

Hur använder du denna dokumentation ?

... Som introduktion till ämnet.

... Som referens-handbok.

... Som handledning vid utbildning.

Kvalitet på dokumentationen:		Brå	Varken/eller	Dålig
Tekniskt
Uppbyggnad
Fullständighet

Vad skulle förbättra materialet ?

.....

.....

Övriga kommentarer eller förslags

.....

.....

Felaktigheter i dokumentationen:

.....

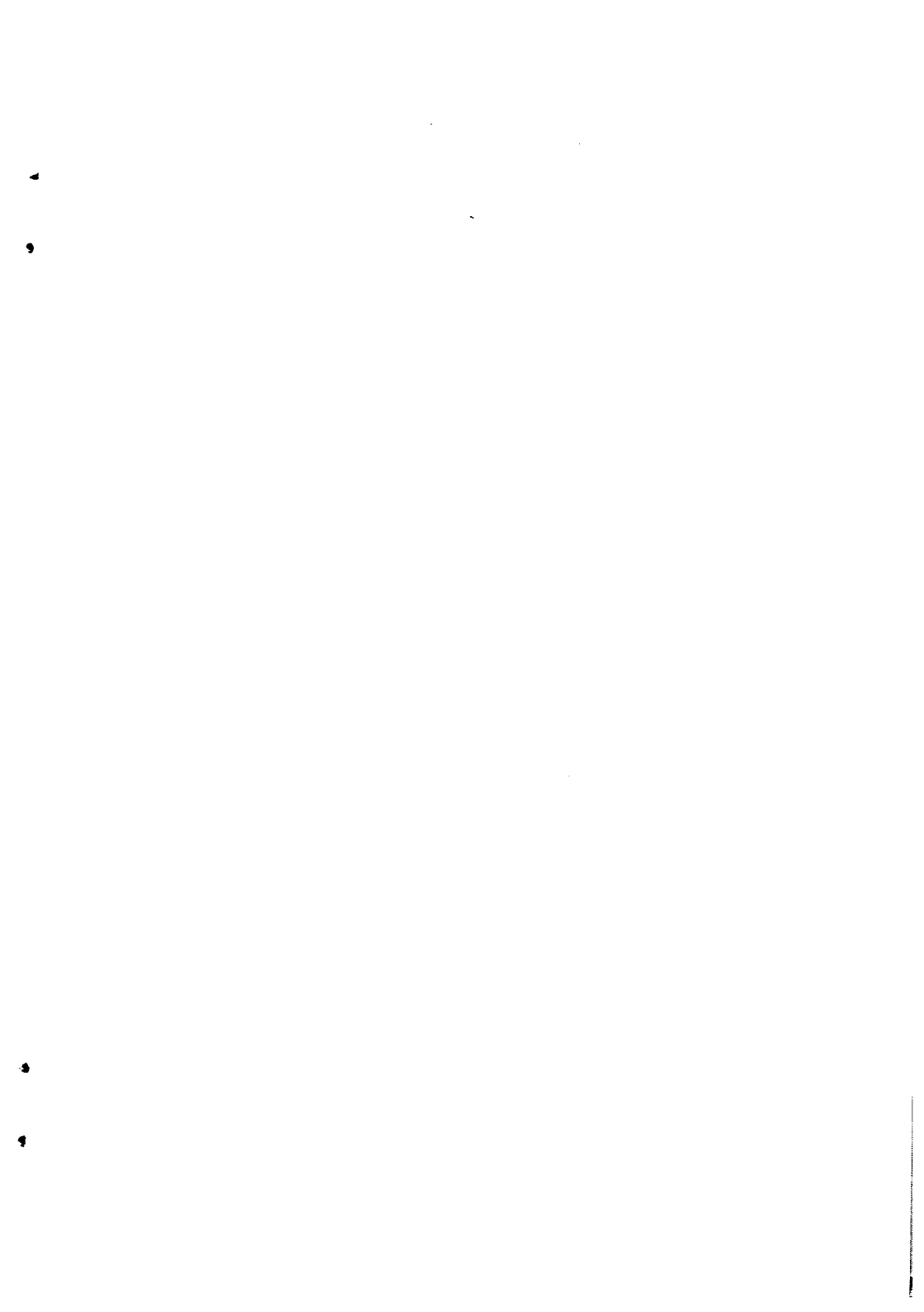
.....

Frankeras ej.
Mottogaren
betalar
portot.

SCANDIA METRIC AB

Mät och Industrisystem

Svarspost
Kundnummer 290 280 16
171 25 SOLNA



SCANDIA **METRIC** AB
INCENTIVE-GRUPPEN

BANVAKTSVÄGEN 20, BOX 1307, 171 25 SOLNA, TEL 08/82 04 00

REGIONSKONTOR: ÅBÄCKSGATAN 6, 431 37 MÖLNDAL, tel 031/20 06 50