
RCSL Nr.: 99 0 00865

Udgave: 1985.08 .13

Forfatter: Knud Erik Hansen

Titel:

BRUGERVEJLEDNING
til
MF915/919
BBC User Porte til
PICCOLINE

Nøgleord: RC759, MF915, MF919, Brugervejledning

Resumé: Denne vejledning beskriver den logiske struktur af PICCOLINE BBC user portene, og hvordan man selv programmerer til disse porte.

(34 trykte sider)

INDHOLDSFORTEGNELSE

SIDE

1.	INDLEDNING	1
	FUNKTIONSBESKRIVELSE	2
2.1	I/O device numre	2
2.2	R6522 Interface	6
2.2.1	Register adresse	8
2.2.2	Port B registre	8
2.2.3	TIMER1	9
2.2.3.1	Skriv i TIMER 1 registre	10
2.2.3.2	Læs TIMER 1 registre	10
2.2.3.3	Timer 1 modes	11
2.2.3.4	One-shot mode	12
2.2.3.5	Free-running mode	12
2.2.4	TIMER 2	14
2.2.4.1	TIMER 2 registre	14
2.2.4.2	Timer 2 modes	14
2.2.4.3	One shot mode	15
2.2.4.4	Taller mode	15
2.2.5	Skifteregister (SR)	16
2.2.6	Interrupt Flag Register (IFR)	18
2.2.7	Interrupt Enable Register (IER)	20
2.2.8	Ekstern Kontrol Register (ACR)	20
2.2.8.1	PB Latch (ACR 1)	21
2.2.9	Perifert kontrol register (PCR)	21
2.2.9.1	CB1 Kontrol (PCR4)	21
2.2.9.2	CB2 Kontrol (PCR7-5)	22
2.2.10	Handshake Kontrol	23
2.2.10.1	Læs hand-shake	23
2.2.10.2	Skriv Hand-shake	24
3.	ELEKTRISK INTERFACE BESKRIVELSE	25
4.	Konnektor beskrivelse	26

BILAG:

A.	REGISTRE	27
A.1	Oversigt over figurer	27



INDLEDNING

1.

Dette skrift beskriver MF915/MF919, der udvider Piccolinen med 2 parallelle interface porte, der er elektrisk/mekaniske kompatible med user porten på BBC Computeren.

BBC interfacen består af 2 enheder:

- 1. BBC701.(MF915), der installeres internt i Piccoline centralenheden i iSBX konnektoren, som vist på figur 1. BBC701 indeholder de 2 interface kredsløb.
- 2. BBC702 (MF919), Ekstern kabel adapter, der splitter multikablet fra BBC701 op i 2 stk. fladkabel konnektorer, der er BBC-kompatible.

2. FUNKTIONSBESKRIVELSE

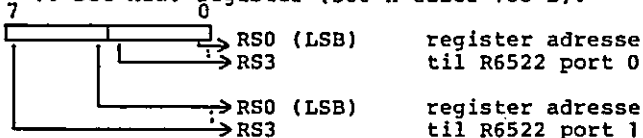
På figur 2 er vist den logiske struktur af BBC701. De to interface porte er implementeret med LSI-kredstypen R6522. R6522 ses fra programmørens synspunkt som 16 8-bits registre, der kan accesses med programmerbar input/ output instruktioner.

Først sættes et adresse register, hvis indhold udpeger et register i hver R6522 kontroller. En efterfølgende Input/output instruktion accesser de udpegede R6522 registre.

2.1 I/O device numre

BBC701 benytter følgende I/O device numre:

1. Set Add. register (300 H eller 768 D).



Eksempel 1

Udpeg register 2 for port 0 og 11 for port 1.

```
290 pout (768, 11 x 16 +2) ; pout se
                          ; Appendix B
```

2. Skriv/læs port 0 (312 H eller 786 D)
Input eller output til dette I/O device nummer læser fra / skriver i det under 1. udpegede register for port 0.

Eksempel 2

```
290 pout (768, reg0_add) ; udpeg register
300 pout (786, out_val) ; skriv værdien
                          ; out val ind i
                          ; register
310 pin (786, in_val) ; inval = indhold
```

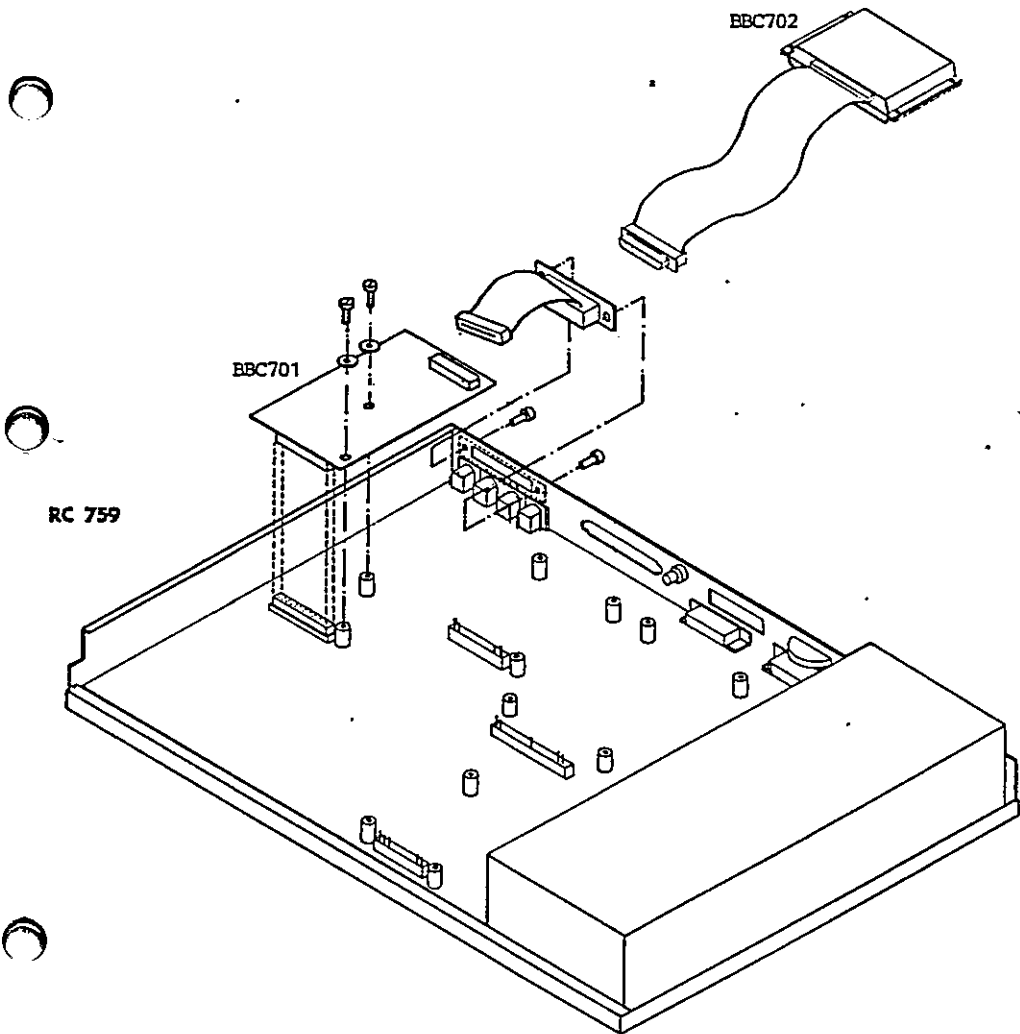


Fig. 1. BBC701/702 Installation i RC759

1985.04.05 XNEH

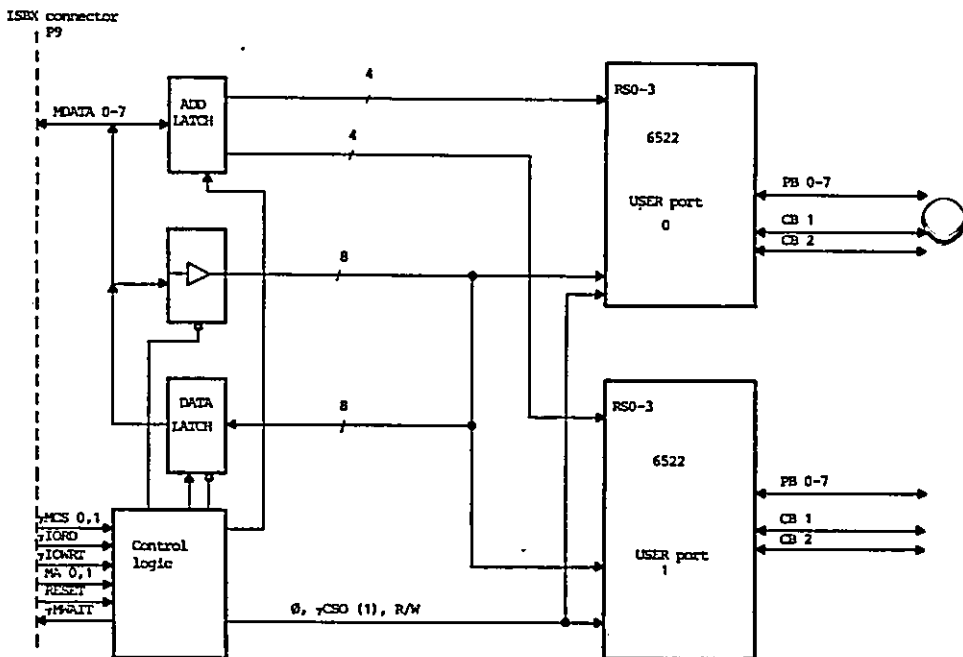


Fig. 2. BBC701 Blokdiagram

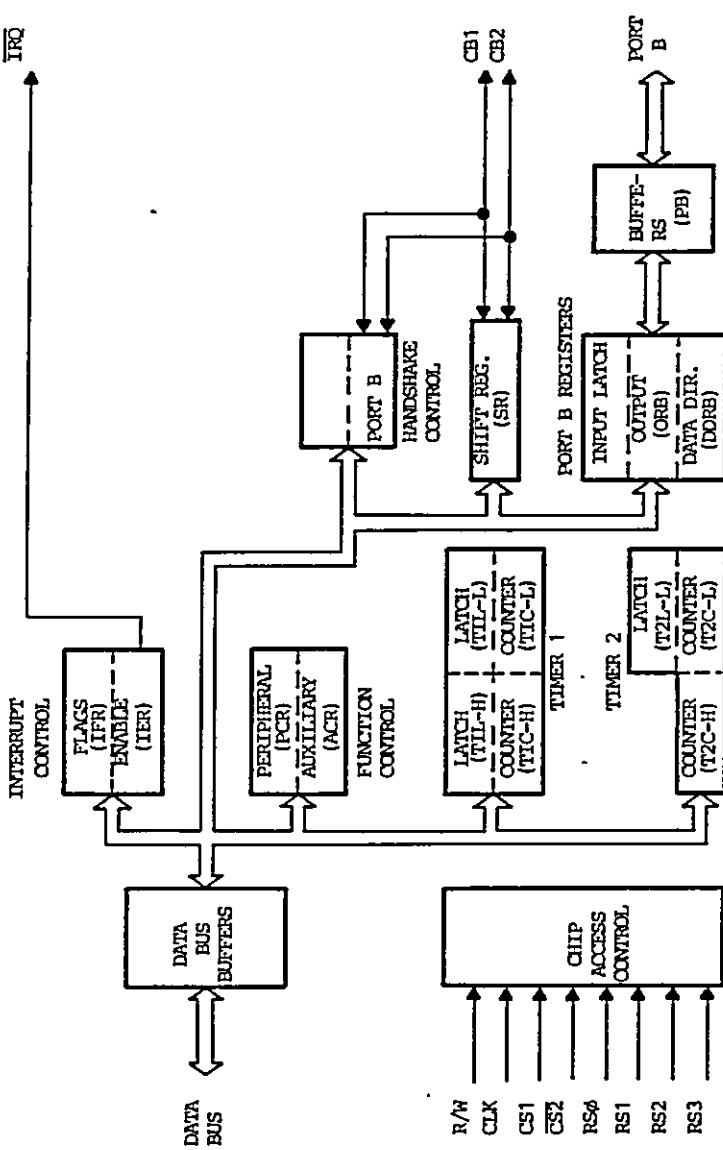


Fig. 3 RC522 Blockdiagram

3. Skriv/læs port 1 (310H eller 784D).
 Input eller output til dette I/O device nummer
 læser fra/skriver i det under 1 udpegede re-
 gister for port 1.

Eksempel 3

```
290 pout (768, regl_add * 16) ; udpeg regl_add.
300 pout (784, out_val) ; skriv
310 pin (784, in_val) ; læs
```

2.2 R6522 Interface

2.

Dette afsnit beskriver i detaljer (og understøttet med Comal-80 eksempler) funktionen af R6522 interface kredsløbet. R6522 giver mulighed for at styre eksternt udstyr via en 8 bits (PB 0-7) bus. Disse data signaler kan programmeres til enten at være input eller output. Desuden kan eksterne signaler forbindes til interne tællere med det formål f.eks. at generere klok pulser til eksternt udstyr eller at tælle eksternt genereret pulser. På fig. 3 er vist et blok diagram af den interne struktur for R6522.

RS				Register navn	Bemærkninger
3	2	1	0		
0	0	0	0	ORB/ IRB	Output register/Input register
0	0	0	1	-	
0	0	1	0	DDRB	Retningsregister for ORB/IRB. 1=Output; 0=Input
0	0	1	1	-	
0	1	0	0	T1L-L T1C-L	Skriv latch bit 0-7 for Timer 1 Læs tæller bit 0-7 for Timer 1
0	1	0	1	T1L-H T1C-H	Skriv i latch/tæller bit 0-7 Overfør T1L-L til T1C-L. Reset T1 inter- rupt flag
0	1	1	0	T1L-L	Skriv/læs T1 Latch bit 0-7
0	1	1	1	T1L-H	Skriv/læs T1 Latch bit 8-15. Reset T1 Interrupt flag.
1	0	0	0	T2L-L T2C-L	Skriv i Latch T2 bit 0-7 Læs fra tæller T2 bit 0-7, reset T2 in- terrupt flag.
1	0	0	1	T2C-H	Skriv/læs T2 tæller bit 8-15. Ved skriv overføres desuden T2L-L til T2C-L, og T2 interrupt flag 0-stilles.
1	0	1	0	SR	Skifregister
1	0	1	1	ACR	Ekstern kontrol register.
1	1	0	0	PCR	Perifert kontrol register.
1	1	0	1	IFR	Interrupt Flag Register.
1	1	1	0	IER	Interrupt Enable Register
1	1	1	1	-	

- ikke brugte registre

Fig. 4. Register struktur

2.2.1 Register adresse

2.2.1

Ved hjælp af RS 3-0 vælges et af de ovenfor viste registre (se SETADD afsnit 2.1).

I de følgende afsnit beskrives betydningen af disse registre.

2.2.2 Port B registre

2.2.2

Parallel-bussen, PB 0-7, styres ved hjælp af 3 registre ORB/IRB (Output Register/Input Register) og DDRB (Data Direction Register). Indholdet af DDRB bestemmer hvilke af PC0-7, der er output, og hvilke der er input. Et "0" i en bit i DDRB beskriver, at den tilsvarende bit af PB0-7 bliver input. Et "1" bevirker en output.

Hvert eksternt signal, PB0-7, er også styret af en bit i output register (ORB) og en i Input Register (IRB). Når et signal er programmeret som output, styres niveauet af signalet med den tilhørende bit i ORB. En "1" i ORB bevirker at signalet går høj, en "0" bevirker at signalet går lav. Data skrevet på bit-positioner af ORB, der er programmeret som input, har ingen virkning.

Ved input læses enten PB0-7 direkte, eller også "latched" PB0-7. Hvis latch mode (afsnit 2.2.8.1) er valgt er det "handshake" signalet CB1, der "latches" input data. Ved indlæsning af signaler, der er programmeret som input, fås indholdet af ORB.

Eks. 4 Læs data fra f.eks. 8 kontakter tilsluttet PB0-7 og udskriv værdien.

```
290 pout (768,2)      ; udpeg DDRB port 0
300 pout (786,0)     ; alle bits = input mode
310 pout (768,0)     ; udpeg IRB
320 pin (786, switch); læs stilling af kontakter
330 print "kontakt stilling ="; switch
340 End
```

Eks. 5 Send data til simpel output port.
F.eks. styring af 8 relæer. En "1" trækker det pågældende relæ.

```
290 pout (768,2) ; udpeg DDRB port 0
300 pout (786, 255); alle bits = output mode
310 pout (768,0) ; udpeg ORB
320 pout (786, relæ_val); set relæ'er.
330 End
```

Eks. 6 Dette eksempel er en kombination af Eks. 4 og Eks. 5. Der er 4 kontakter tilsluttet PB 3-0; og 4 relæer til PB 7-4. Relæerne på PB 7-4 skal følge kontakterne PB 3-0, således at hvis f.eks. kontakt PB3 er sluttet (PB 3=0) skal relæ PB7 trække (PB7=1)

Sluttet kontakt giver "0".

```
290 proc INIT_PB
300 pout (768,2) ; udpeg DDRB
310 pout (786, 128+64+32+16); PB 7-4 = output.
; PB 3-0 = input

320 pout (768, 0) ; udpeg ORB/IRB
330 endproc INIT_PB
340 //MAIN PROGRAM
350 INIT_PB
360 While 1 > 0
370 Relæ : = 0
380 K_stilling : = 0
390 pin (786, K_stilling) ; Læs kontakt stilling
400 If (K_stilling and 1) = 0 then relæ: relæ + 16
410 If (K_stilling and 2) = 0 then relæ: relæ + 32
420 If (K_stilling and 4) = 0 then relæ: = relæ + 64
430 If (K_stilling and 8) = 0 then relæ: = relæ + 128
440 pout (786, relæ); træk relæer
450 endwhile
```

2.2.3 TIMER1

2.2.3

Timer 1 består af 2 8-bit latches og en 16-bit tæller. Latchene benyttes til at gemme data som senere skal loades ind i tælleren. Efter at være loaded, tæller tælleren ned med frekvensen 2 Mhz. Når tælleren når 0 sættes T1 interrupt flag. (afsnit 2.2.6). Timeren vil da stoppe yderlig interrupts, eller automatisk overføre indholdet fra latchene til tælleren og fortsætte nedtælling. Desuden kan timeren

programmeres til at invertere et output signal, hvergang tælleren når 0. Hver af disse modes beskrives yderlig i de følgende afsnit.

2.2.3.1 Skriv i TIMER 1 registre

2.2.3.1

RS3	RS2	RS1	RS0	Skriv operation
0	1	0	0	Skriv i latch bit 0-7 (T1L-L)
0	1	0	1	Skriv i latch bit 8-15 (T1L-H). Skriv i tæller bit 8-15 (T1C-H). Overfør latch bit 0-7 til tæller bit 0-7. Reset T1 interrupt flag.
0	1	1	0	Skriv i latch bit 0-7.
0	1	1	1	Skriv i latch bit 8-15. Reset T1 interrupt flag.

Fig. 5. Timer 1 skriv

Vær opmærksom på, at man ikke skriver direkte i tælleren bit 0-7 (T1C-L). Derimod, overføres T1L-L automatisk til T1C-L, når der skrives i tælleren bit 8-15 (T1C-H).

Det andet set adresser (6 og 7) giver mulighed for at skrive i latch registre uden at påvirke den pågældende nedtælling.

2.2.3.2 Læs TIMER 1 registre

2.2.3.2

RS3	RS2	RS1	RS0	Læs operation
0	1	0	0	Læs T1 tæller bit 0-7 (T1C-L). Reset T1 interrupt flag.
0	1	0	1	Læs T1 tæller bit 8-15 (T1C-H).
0	1	1	0	Læs T1 latch bit 0-7 (T1L-L).
0	1	1	1	Læs T1 latch bit 8-15 (T1L-H).

Fig. 6. TIMER 1 læs

2.2.3.3 Timer 1 modes

2.2.3.3

To bits i ACR-registeret (fig. 4) bestemmer, hvilken mode T1 skal have. Disse 2 bits samt de tilhørende modes er følgende:

ACR7	ACR6	MODE
0	0	Genererer et enkelt time-out interrupt hvergang T1 loades. PB7 er disabled.
0	1	Lav continuerlig interrupts. PB7 disabled (FREE running).
1	0	Lav single interrupt. Hvergang T1 loades pulses PB7.
1	1	Lav continuerlig interrupts. Hvergang T1 timer ud skiftes niveau på PB7 (FREE running).

Fig. 7.

2.2.3.4 One-shot mode

2.2.3.4

One-shot mode giver mulighed for et enkelt T1 interrupt hvergang timer 1 loades. Tiden fra "skriv T1C-H" og til interruptet er en funktion af den værdi, der skrives ind i tælleren. Desuden kan TIMER1 programmeres til at give en enkel negativ puls på PB7. Med output enabled (ACR7=1), vil en "skriv T1C-H" bevirke, at PB7 går lav. PB7 vender tilbage til høj niveau.

Eks. 7 lav et delay på 1024 klok pulser (2MHz), ingen output på PB7.

```

290 DELAY: = 1024
300 pout (768, 11) ; udpeg ACR register
310 pout (786, 0 ) ; ACR7=0. ACR6=0
320 pout (768, 4) ; udpeg T1L=L
330 pout (786, DELAY)
340 pout (768, 5) ; udpeg T1L-H
350 pout (786, DELAY div 256)
360 END

```

Eks. 8 Samme som Eks. 7, blot med output på PB7.

```

290 DELAY: = 0
300 pout (768, 11) ; udpeg ACR register
310 pout (786, 128) ; ACR7=1. ACR6=0
320 pout (768, 4) ; udpeg T1L=L
330 pout (786, DELAY)
340 pout (768, 5) ; udpeg T1L-H
350 pout (786, DELAY Div 256)
360 END

```

2.2.3.5 Free-running mode

2.2.3.5

I denne mode, vil indholdet af latchene (T1L-L og T1L-H) automatisk loades ind i tælleren, hvergang den har talt ned til 0. Hvergang TIMER1 når 0, sættes T1 interrupt flag. Hvis output til PB7 er enabled, vil PB7 inverteres hvergang tælleren når 0.

Eks. 9 På PB7 genereres et puls signal, hvis negative periode og positive periode, hver har en længde på 1024 klokpulser (2MHz).

```

290 periode : = 1024
300 pout (768, 11) ; udpeg ACR register.

```



```

310  pout (786, 128 + 64) ; ACR 7 = ACR 6 = 1
320  pout (768, 4) ; udpeg T1L-L
330  pout (786, periode)
340  pout (768, 5) ; udpeg T1L-H
350  pout (786, periode div 256)
360  END

```

Eks. 10 I dette eksempel genereres et asymmetrisk klok signal på PB7, hvis negativ periode er 16384 klok signaler (2MHz) og hvis positive periode har en længde svarende til 32767 system klok signaler (2MHz). Hertil benyttes 2 registre, som endnu ikke er beskrevet: IFR (Interrupt Flag Register - afsnit 2.2.6) og IER (Interrupt Enable Register - afsnit 2.2.7)

```

290  proc set_reg (reg_add, værdi)
300  pout (768, reg_add) ; udpeg reg_add
310  pout (786, værdi)
320  endproc set_reg
330  proc read_reg (reg_add, ref INVAL)
340  pout (768, reg_add)
350  inval:=0
360  pin (786 INVAL)
370  endproc read_reg
380  low:=16384
390  HIGH:=32767
400  //MAIN PROGRAM
410  set_reg (14, 128 +64) ; Enable T1 interrupt
420  set_reg (4, LOW)
430  set_reg (5, low div 256)
440  WHILE 1>0 do
450    set_reg (6, high)
460    set_reg (7, high div 256)
470    INVAL:=0
480    WHILE (INVAL and 64)<> 0 do
490      read reg (13, INVAL) ; læs IFR register
500      endwhile
510      set_reg (6, low)
520      set_reg (7, low div 256)
530      INVAL:=0
540      WHILE (INVAL and 64) <> 0 do
550        read reg (13, inval )
560        endwhile
570      endwhile
580  END

```

2.2.4 TIMER 2

2.2.4

TIMER 2 kan arbejde i 2 modes, en "one-shot" mode som TIMER1 samt en mode hvor den tæller negative pulser på det eksterne signal PB6. En enkelt bit i ACR-registeret (ACR5) vælger TIMER 2 mode. TIMER 2 er en 16 bits tæller, der tæller ned med systemklokken (2MHz).

2.2.4.1 TIMER 2 registre

2.2.4.1

Følgende registre benyttes til at starte TIMER 2:

RS3	RS2	RS1	RS0	Skriv mode	Læs mode
1	0	0	0	Skriv i T2L-L	Læs fra T2C-L reset T2 interrupt flag.
1	0	0	1	Skriv i T2C-H Overfør T2L-L til T2C-L reset T2 interrupt flag	Læs fra T2C-H

Fig. 8. TIMER2 registre

TIMER2 latch bit 7-0 (T2L-L) kan der kun skrives i ("write-only").

TIMER2 tæller bit 7-0 (T2C-L) kan der kun læses fra ("read-only").

TIMER 2 tæller bit 15-8 (T2C-H) kan der både skrives i og læses fra.

2.2.4.2 Timer 2 modes

2.2.4.2

TIMER2 kan som ovenfor nævnt virke i enten "one-shot" mode eller i en tæller mode, hvor negative pulser på PB6 tælles.

2.2.4.3 One shot mode

2.2.4.3

I denne mode vil TIMER2 tælle til 0, og derefter sætte T2 interrupt flag. T2C-H skal genskrives for at enable interrupt igen.

Eks. 11 Lav et delay på 1024 system klok pulser (2MHZ). Samme som eks. 7 for timer 1.

```

290 DELAY: = 1024
300 pout (768, 11) ; udpeg ACR-register
310 pout (786, 32) ; ACR 5 = 1
320 pout (768, 8) ; udpeg T2L-L
330 pout (786, delay)
340 pout (768, 9) ; udpeg T2C-H
350 pout (786 Delay div 256)
360 End

```

2.2.4.4 Tæller mode

2.2.4.4

I denne mode tælles negative pulser på det eksterne signal PB6. Bemærk at denne funktion af PB6 har højere prioritet end parallel bus funktionen (afsnit 2.2.2). En værdi loades ind i TIMER 2. Når der skrives ind i T2C-H resættes T2 interrupt flag og Timer 2 vil derefter tælle ned hvergang en puls modtages på PB6. Når Timer 2 når til 0, sættes interrupt flag T2. Timer 2 vil derefter fortsætte nedtællingen. Det er dog nødvendigt at genskrive T2C-H for at tillade at interrupt flaget T2 kan sættes igen.

Eks. 12 I dette eksempel benyttes Timer 2 tæller mode til at lave en omdrejningstæller (antal omdr./ min). Det kan f.eks. være en fjernvarmemåler, der giver en elektrisk puls på PB6, hvergang dens skive har taget et omløb.

Forudsætning:

1. Måleren tager mindst 1 omløb pr. minut.
2. Antal omløb i 10 minutter skal kunne være i TIMER2's 16 bits.

```

290 proc set_reg (reg_add, værdi)
300   pout (768, reg_add)
310   pout (786, værdi)
320 endproc set_reg

```

```

330 proc read_reg (reg_add, ref inval)
340     pout (768, reg_add)      ;udpeg reg_add
350     INVAL:=0
360     pin (786, INVAL)
370 endproc read_reg
380 //main program 385 set_reg (11,0); ACR5=0
385 set_reg (11,0); ACR5=0
390 set_reg (14, 128 + 32)      ; Enable T2 interrupt
400 while l>0 do
410     start_klok := SYS (3)
420     set_reg (8, 255)
430     set_reg (9, 255)        ; Timer 2 sættes til max.
440     while SYS (3) <(start_klok + 30000) do
450         endwhile
460         low:= 0
470         high:= 0
480         pin (9, high)
490         pin (8, low)
500         opm:= (high * 256 + low) div 10; omdr/min
510         Print "Omdrejning/min = " ; opm
520     endwhile

```

2.2.5 Skifteregister (SR)

2.2.5

Skifteregisteret (SR) kan skifte seriel data ind eller ud af det eksterne kontrol signal CB2 under kontrol af en intern modulo-8 tæller. Skiftepulser kan enten fra en ekstern kilde forsynes på CB1 eller skiftepulser genereret internt kan leveres ud på CB1 og benyttes af externt udstyr.

Skifte mode vælges med bits i ACR registeret (ACR4 - ACR2).

Skift ind foregår fra SR0 mod SR7. Skift ud foregår fra SR7, SR7 skiftes tilbage på SR0.

Skifteregisteret har et interrupt flag (SR-flag), der sættes hvergang 8 skift har fundet sted. SR flaget sættes tilbage igen ved enten at læse fra eller skrive i SR-registeret. SR-registeret kan arbejde i følgende modes:

ACR			SR-FUNKTION
4	3	2	
0	0	0	Ingen skiftefunktion
0	0	1	Skift ind under kontrol af Timer 2. T2C-L virker som 8-bits neddeler, der laver skiftepulser, som sendes ud på CB1.
0	1	0	Skift ind med systemklokken (2MHz). Systemklokken sendes ud på CB1. Efter 8 skift sættes SR-Interrupt og CB1 pulser stopper.
0	1	1	Skift ind under kontrol af eksternt klok signal leveret ind på CB1. Efter 8 skift sættes interrupt flag.
1	0	0	Skift ud under kontrol af Timer 2 (T2C-L). Skifte pulser er leveret ud på CB1. SR7 er recirkuleret tilbage på SR0. Efter 8 skift stoppes ikke.
1	0	1	Som 4, blot stoppes efter 8 skift og SR interrupt flaget sættes, CB2 går til en hviletilstand bestemt af PCR5. PCR7-5 benyttes til at sætte CB2 til manuel output (enten høj eller lav).
1	1	0	Skift ud med system klok (2MHz) CB1 følger system klokken.
1	1	1	Skift ud under kontrol af eksternt klok leveret ind på CB1. Interrupt flaget sættes efter 8 skift, men skiftefunktionen fortsætter. Interrupt flaget resættes, når der skrives i SR_registeret.

Fig. 9. Skifteregister modes

Eks. 13 Procedure til at skifte 8 bits data ud på CB2 styret af et kloksignal leveret fra eksternt udstyr på CB1.

```

290 proc set_reg (reg_add, værdi)
300     pout (768, reg_add)
310     pout (786, værdi)
320 endproc set_reg
330 proc read_reg (reg_add, ref inval)

```

```

340     pout (768, reg_add)
350     inval: = 0
360     pin (768, inval)
370 endproc read_reg
380 proc skift_ud (output)
390 set_reg (11, 16 + 8 +4) ; ACR4=ACR3=ACR2=0
400 set_reg (14, 128 +4) ; sæt Enable SR intr.
410 set_reg (10, output) ; skriv i SR register
420 read_val: = 0
430 while (read_val and 4) = 0 do
440     pin (13, read_val)
450 endwhile
460 endproc skift_ud

```

Eks. 14 Procedure til at skifte 8 bit ind fra eksternt udstyr som også leverer skifteklod på CBI.

```

380 proc skift_ind (ref_input)
390 set_reg (11, 8 +4) ; ACR 4 = 0. ACR3 = ACR2 = 1
400 set_reg (14, 128 + 4) ; sæt enable SR intr.
410 read_val:=0
420 while (read_val and 4) = 0 do
430     pin (13, read_val)
440 endwhile
450 read_reg (10, input)
460 endproc skift_ind

```

2.2.6 Interrupt Flag Register (IFR)

2.2.6

7	6	5	4	3	2	1	0	
IRQ	T1	T2	CBI	CB2	SR	0	0	

Fig. 10. IFR Register

IFR-registeret er et læs bit/reset register, forstået på den måde at en "1" bit resættes efter læsning af registeret.

IRQ er sand, hvis en af bit'ene 6-0 er sat. De to BBC porte har hver deres interrupt til Piccoline CPU'en.

Port 0 giver interrupt vector 13..

Port 1 giver interrupt vector 15.

Bit 6-2 sættes og resættes af følgende funktioner:

BIT	SET	RESET
6	Timeout af Timer1	Læs fra T1C-L og skriv i T1L-H
5	Timeout af Timer2	Læs fra T2C-L og skriv i T2C-H
4	Aktiv skift på signalet CB1	Læs fra IRB eller skriv i ORB
3	Aktiv skift på signalet CB2	Læs fra IRB eller skriv i ORB
2	Afslutning på 8 skift i SR	Læs fra/ eller skriv i skifteregisteret SR.

Fig. 11. SET/RESET af IFR-flag

Foruden de ovenfor nævnte muligheder kan flagene simpelthen 0-stilles ved at skrive direkte i IFR-registeret. En "1" er skrevet i en bit-position vil 0-stille det pågældende flag. Et "0" lader det pågældende flag være uforandret. Flagene kan ikke sættes ved skrivning i IFR-registeret, men kun ved de i figur 11 nævnte metoder. Disse set-funktioner har kun virkning, såfremt de tilsvarende bit er sat i IER-registeret (Interrupt Enable Register).

Eks. 15 Nulstil alle de bits, der er "1" i IFR-registeret.

```

290 pout (768, 13) ; udpeg IFR register
300 IFR_FLAG = 0
310 pin (786, IFR_FLAG) ; læs flag
320 pout (786, IFR_FLAG) ; reset flag, der er "1"
330 END

```

2.2.7 Interrupt Enable Register (IER)

2.2.7

7	6	5	4	3	2	1	0
SET/ RESET	T1	T2	CB1	CB2	SR	0	0

Fig. 12. IER Register

Hvert flag i IFR registeret har en tilsvarende bit i IER registeret. Bit'en i IER registeret skal være "1" for at det tilsvarende flag i IFR kan sættes. Ved at skrive i IER kan de enkelte bit sættes eller resættes. Hvis bit 7 =1 vil de øvrige positioner, hvori der skrives en "1"er, "1" - stilles; øvrige bits forbliver uændret. Hvis bit 7=0, nulstilles de positioner, hvori der skrives en "1"er, øvrige bits uændret. Desuden kan der læses fra IER registert. Under læsning er bit 7 altid "0".

Eks. 16 Sæt bit 4 i IER-registeret.

```
290  pout (768, 14)  ; udpeg IER
300  pout (786, 128 + 16)
310  END
```

Eks. 17 Reset bit 4 i IER-registeret.

```
290  pout (768, 14)  ; udpeg IER
300  pout (786, 16)
310  END
```

Bemærk at bit 0 og 1 altid skal være "0".

2.2.8 Ekstern Kontrol Register (ACR)

2.2.8

7	6	5	4	3	2	1	0
T1 Kon- trol		T2 Kon- trol	Skifteregister Kontrol			PB Latch	0

Fig. 13. ACR-register

ACR 7-6 styrer Timer 1. Funktionen af disse bit er beskrevet i afsnit 2.2.3.3.

ACR 5 styrer TIMER2. Funktionen af TIMER2 er beskrevet i afsnit 2.2.4.3 og i afsnit 2.2.4.4.

ACR 4-2 styrer funktionen af skifteregisteret SR, beskrevet i afsnit 2.2.5.

2.2.8.1 PB Latch (ACR 1)

2.2.8.1

Hvis ACR 1=1 vil PB0-7 input signaler blive latched, når CB1 interrupt flaget sættes. Hvis nogle af PB0-7 er defineret som output (afsnit 2.2.2), latches de tilsvarende bits fra ORB.

2.2.9 Perifert kontrol register (PCR)

2.2.9

7	6	5	4	3	2	1	0
CB2 Kontrol			CB1 Kontrol	0	0	0	0

Fig. 14. PCR-register

Hver af disse funktioner er yderligere beskrevet i de følgende afsnit.

2.2.9.1 CBI Kontrol (PCR4)

2.2.9.1

Hvis PCR4=1, sættes CB1 interrupt flaget (afsnit 2.2.6) ved et negativ skift på CB1 signalet.

Hvis PCR4=0, sættes CB1 interrupt flaget ved et positiv skift på CB1 signalet.

2.2.9.2 CB2 Kontrol (PCR7-5)

2.2.9.2

Hvis skifteregister-funktionen (afsnit 2.2.5) ikke benyttes, styres funktionen af signalet CB2 ved hjælp af PCR7-5.

PCR			FUNKTION
7	6	5	
0	0	0	Interrupt Input mode. Sæt CB2 interrupt flag ved en negativ skift på CB2 input signalet. Reset IRF3 ved læs fra IRB eller skriv i ORB.
0	0	1	Sæt CB2 interrupt flag (IFR3) ved et negativ skift på input signalet CB2. Læs fra IRB eller skriv i ORB har ingen indflydelse på IFR3.
0	1	0	Input mode. Sæt CB2 interrupt flag ved positiv skift på CB2 input signalet. Læs/skriv operation på IRB/ORB resætter IFR3.
0	1	1	Sæt IFR3 ved positiv skift på CB2 signalet. Læs/skriv på IRB/ORB har ingen indflydelse på IFR3.
1	0	0	"Handshake mode". CB2 output signalet sættes lav ved skriv i ORB. CB2 sættes igen høj ved aktivt skift på CB1 input signal.
1	0	1	Puls output mode. CB2 sættes lav i en klokperiode af systemklokken (2MHz) efter en skriv operation i ORB.
1	1	0	CB2 output signalet holdes konstant lav i denne mode.
1	1	1	CB2 output signalet holdes konstant høj i denne mode.

Fig. 15. CB2 Kontrol

2.10 Handshake Kontrol

2.2.10

Ved at benytte CB1, CB2 kan der overføres data mellem R6522 porten og eksternt udstyr asynkront; idet CB1 og CB2 programmeres som "handshake" kontrol signaler.

2.2.10.1 Læs hand-shake

2.2.10.1

Eksternt udstyr signaler på CB1, at data er parat. Processoren signalerer tilbage på CB2, når data er læst fra IRB.

Eks. 18 I dette eksempel anskueliggøres læs handshake mode ved hjælp af 2 procedurer; en der sætter porten i denne mode og en der indlæser bytes. Det forudsættes:

1. negativ skift på CB1 input betyder at data er klar.
2. porten sender ACK tilbage ved en negativ puls på CB2.

```

290  proc set_reg (reg_add, værdi)
300      pout (768, reg_add)
310      pout (786, værdi)
320  endproc set_reg
330  proc read_reg (reg_add, ref inval)
340      pout (768, reg_add)
350      inval: = 0
360      pin (786, inval)
370  endproc read_reg
380  //procedure til at sætte handshake mode
390  proc init
400      set_reg (2, 0) ; PB0-7 sættes til input
410      set_reg (12, 128 + 32 + 16) ; PCR (7:5)={101} PCR4=1
420      set_reg (11, 0) ; ACR 0-7 = 0
430      set_reg (14, 128 + 16); IER4 (CB1) = 1
440      read_reg (13, inval)
450      set_reg (13, inval) ; reset aktive flags
460  endproc init
470  proc read_byte (ref inbyte)
480      read_flag : = 0
490      while (read_flag and 16) = 0 do
500          read_reg (13, read_flag)
510      endwhile
520      read_reg (0, inbyte) ; læs fra IFR
530      set_reg (13, read_flag); reset Cb1 flag.
540      set_reg (0,0) ; puls CB2 ved skriv

```

```

550 endproc read_byte ; i ORB
560 END

```

2.2.10.2 Skriv Hand-shake

2.2.10.2

Ved skriv handshake genererer porten et "Data Ready" signal på CB2, som besvares fra det eksterne udstyr med et "Data Taken" signal på CB2.

Eks. 19 Eksempel på skriv handshake data overførsel. Fra Eks. 18 genbruges procedurerne set_reg, og read_reg.

Eks. 19 indeholder en procedure, der starter porten op i skriv mode samt en byte skriv procedure.

Porten sender negativ skift på CB2, som tegn på at data er parat. Det eksterne udstyr svarer med negativ skift på CB1, som tegn på at data er taget. Derefter sættes porten CB2 høj igen, og CB1 sættes høj af det eksterne udstyr.

```

380 //procedure til at sætte porten i skriv mode
390 proc init
400 set_reg (2, 255) ; PB0-7 sættes til output
410 set_reg (12, 128) ; PCR(7:5)=(100).PCR4=0
420 set_reg (11, 0) ; ACRO-7=0
430 set_reg (14, 128 +16) ; IER4 (CB1 enable)=1
440 read_reg (13, inval);
450 set_reg (13, inval); reset aktive IFR flag
460 endproc init
470 proc skriv_byte (outbyte)
480 set_reg (0, outbyte); skriv i ORB
490 read_flag: =0
500 while (read_flag and 16) = 0 do
510 read_reg (13, read_flag)
520 endwhile
530 set_reg (13, read_flag); reset CB1 flag
540 endproc skriv_byte
550 END

```

ELEKTRISK INTERFACE BESKRIVELSE

3.

Dette afsnit beskriver de elektriske parametre for input/output signalerne i BBC portene.

PB0-7, CB1-2input mode

1 standard TTL load.
 $V_{input} = 0.4V \rightarrow I_{input} = -1.6mA$
 $v_{input} = 2.4V \rightarrow I_{input} = 40 \text{ micro A}$

output mode

Kan drive 1 standard TTL load. Specifikationer som ovenfor.

.+5V forsyning

Forsyning til udstyr tilsluttet BBC portene. Max. forbrug for begge porte tilsammen er 1.0A.

4. Konnektor beskrivelse

Dette afsnit indeholder en beskrivelse af de konnek-
torer hvorigennem BBC portene er tilgængelige.

BBC KONNEKTOR		PORTO/PORT1	
PIN		PIN	
1	+5V	2	CB1
3	+5V	4	CB2
5	} 0V	6	PB0
7		8	PB1
9		10	PB2
11		12	PB3
13		14	PB4
15		16	PB5
17		18	PB6
19		20	PB7

Fig. 16. BBC User port

Konnektor set forfra (ind mod pins).

Konnektortype: 20 pins flad-kabel connector (pins)

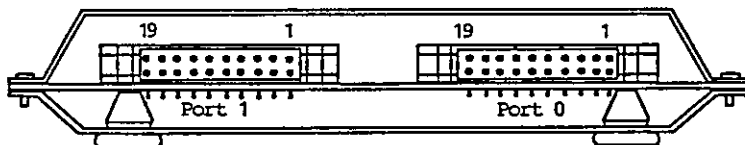


Fig. 17. Konnektor type

På figur 17 er vist hvordan nummerering af pins
foregår i de to user port konnektoer.

Som modpart skal benyttes en 20 pin flad-kabel kon-
nektor af socket typen.

A. REGISTRE

A.

A.1 Oversigt over figurer

A.1

1. BBC701/702 Installation i RC759	3
2. BBC701 Blokdiagram	4
3. R6522 Blokdiagram	5
4. Register struktur	7
5. Timer 1 skriv	10
6. TIMER 1 læs	11
7.	11
8. TIMER2 registre	14
9. Skifteregister modes	17
10. IFR Register	18
11. SET/RESET af IFR-flag	19
12. IER Register	20
13. ACR-register	20
14. PCR-register	21
15. CB2 Kontrol	22
16. BBC User port	26
17. Konnektor type	26

```
0010 // APPENDIX B 85.04.08 kneh
0020 // GENERELLE INPUT/OUTPUT PROCEDURE BENYTTET
0030 // I SKRIFTET OM BBC701
0040 PROC pin(port,REF d)
0050 OPEN FILE 3,"/"+STR$(port)+"/port". READ
0060 d:= ORD(GET$(3,1))
0070 CLOSE FILE 3
0080 ENDPROC pin
0090 PROC pout(port,d)
0100 OPEN FILE 3,"/"+STR$(port)+"/port". WRITE
0110 marg:= SYS(6)
0120 MARGIN 0
0130 PRINT FILE 3: CHR$(d);
0140 CLOSE FILE 3
0150 MARGIN marg
0160 ENDPROC pout
0170 FUNC and_func(in_val,mask)
0180 i:= 128
0190 buf_val:= in_val
0200 x_val:= in_val
0210 WHILE i<>0 DO
0220 x_val:= x_val-i*(buf_val>i-1)*(mask<i)
0230 mask:= mask-i*(mask>i-1)
0240 buf_val:= buf_val-i*(buf_val>i-1)
0250 i:= i DIV 2
0260 ENDWHILE
0270 RETURN x_val
0280 ENDFUNC and_func
```


LÆSERBEMÆRKNINGER

Titel: Brugervejledning til MF915/919 RCSL Nr.: 99 0 00865
BBC User porte til PICCOLINE

A/S Regnecentralen af 1979 bestræber sig på at forbedre kvalitet og brugbarhed af sine publikationer. For at opnå dette ønskes læserens kritiske vurdering af denne publikation.

Kommenter venligst manualens fuldstændighed, nøjagtighed, disposition, anvendelighed og læsbarhed:

Angiv fundne fejl (reference til sidenummer):

Hvordan kan manualen forbedres:

Andre kommentarer:

Navn: _____ Stilling: _____

Firma: _____

Adresse: _____

Dato: _____

På forhånd tak!

..... Fold her

..... Riv ikke - Fold her og hæft

Frankeres
som
brev

 **REGNECENTRALEN**
af 1979

Informationsafdelingen
Lautrupbjerg 1
2750 Ballerup