```
*
*
*  HC12 i/o register locations (9s12dp256)
*
*
porta:          equ 0   ;port a = address lines a8 - a15
portb:          equ 1   ;port b = address lines a0 - a7
ddra:           equ 2   ;port a direction register
ddrb:           equ 3   ;port a direction register

porte:          equ 8   ;port e = mode,irqandcontrolsignals
ddre:           equ 9   ;port e direction register
pear:           equ $a  ;port e assignments
mode:           equ $b  ;mode register
pucr:           equ $c  ;port pull-up control register
rdriv:          equ $d  ;port reduced drive control register
ebictl:         equ $e  ;e stretch control

initrm:         equ $10 ;ram location register
initrg:         equ $11 ;register location register
initee:         equ $12 ;eeprom location register
misc:           equ $13 ;miscellaneous mapping control
mtst0:          equ $14 ; reserved
itcr:           equ $15 ;interrupt test control register
itest:          equ $16 ;interrupt test register
mtst1:          equ $17 ; reserved

partidh:        equ $1a ;part id high
partidl:        equ $1b ;part id low
memsiz0:        equ $1c ;memory size
memsiz1:        equ $1d ;memory size
intcr:          equ $1e ;interrupt control register
hprio:          equ $1f ;high priority reg

bkpct0:         equ $28 ;break control register
bkpct1:         equ $29 ;break control register
bkp0x:          equ $2a ; break 0 index register
bkp0h:          equ $2b ; break 0 pointer high
brp0l:          equ $2c ; break 0 pointer low
bkp1x:          equ $2d ; break 1 index register
bkp1h:          equ $2e ; break 1 pointer high
brp1l:          equ $2f ; break 1 pointer low
ppage:          equ $30 ;program page register

portk:          equ $32 ;port k data
ddrk:           equ $33 ;port k direction
synr:           equ $34 ; synthesizer / multiplier register
refdv:          equ $35 ; reference divider register
ctflg:          equ $36 ; reserved
crgflg:         equ $37 ; pll flags register
crgint:         equ $38 ; pll interrupt register
clksel:         equ $39 ; clock select register
pllctl:         equ $3a ; pll control register
rtictl:         equ $3b ;real time interrupt control
```

```
copctl:         equ $3c ;watchdog control
forbyp:         equ $3d ;
ctctl:          equ $3e ;
armcop:         equ $3f ;cop reset register

tios:           equ $40 ;timer input/output select
cforc:          equ $41 ;timer compare force
oc7m:           equ $42 ;timer output compare 7 mask
oc7d:           equ $43 ;timer output compare 7 data
tcnt:           equ $44 ;timer counter register hi
*tcnt:          equ $45 ;timer counter register lo
tscr:           equ $46 ;timer system control register
ttov:           equ $47 ;reserved
tctl1:          equ $48 ;timer control register 1
tctl2:          equ $49 ;timer control register 2
tctl3:          equ $4a ;timer control register 3
tctl4:          equ $4b ;timer control register 4
tmsk1:          equ $4c ;timer interrupt mask 1
tmsk2:          equ $4d ;timer interrupt mask 2
tflg1:          equ $4e ;timer flags 1
tflg2:          equ $4f ;timer flags 2
tc0:            equ $50 ;timer capture/compare register 0
tc1:            equ $52 ;timer capture/compare register 1
tc2:            equ $54 ;timer capture/compare register 2
tc3:            equ $56 ;timer capture/compare register 3
tc4:            equ $58 ;timer capture/compare register 4
tc5:            equ $5a ;timer capture/compare register 5
tc6:            equ $5c ;timer capture/compare register 6
tc7:            equ $5e ;timer capture/compare register 7
pactl:          equ $60 ;pulse accumulator controls
paflg:          equ $61 ;pulse accumulator flags
pacn3:          equ $62 ;pulse accumulator counter 3
pacn2:          equ $63 ;pulse accumulator counter 2
pacn1:          equ $64 ;pulse accumulator counter 1
pacn0:          equ $65 ;pulse accumulator counter 0
mcctl:          equ $66 ;modulus down conunter control
mcflg:          equ $67 ;down counter flags
icpar:          equ $68 ;input pulse accumulator control
dlyct:          equ $69 ;delay count to down counter
icovw:          equ $6a ;input control overwrite register
icsys:          equ $6b ;input control system control

timtst:         equ $6d ;timer test register

pbctl:          equ $70 ; pulse accumulator b control
pbflg:          equ $71 ; pulse accumulator b flags
pa3h:           equ $72 ; pulse accumulator holding register 3
pa2h:           equ $73 ; pulse accumulator holding register 2
pa1h:           equ $74 ; pulse accumulator holding register 1
pa0h:           equ $75 ; pulse accumulator holding register 0
mccnt:          equ $76 ; modulus down counter register
*mccntl:        equ $77 ; low byte
tcoh:           equ $78 ; capture 0 holding register
tc1h:           equ $7a ; capture 1 holding register
```

```
tc2h:           equ $7c ; capture 2 holding register
tc3h:           equ $7e ; capture 3 holding register

atd0ctl0:       equ $80 ;adc control 0 (reserved)
atd0ctl1:       equ $81 ;adc control 1 (reserved)
atd0ctl2:       equ $82 ;adc control 2
atd0ctl3:       equ $83 ;adc control 3
atd0ctl4:       equ $84 ;adc control 4
atd0ctl5:       equ $85 ;adc control 5
atd0stat:       equ $86 ;adc status register hi
*atd0stat       equ $87 ;adc status register lo
atd0test:       equ $88 ;adc test (reserved)
*atd0test       equ $89 ;

atd0dien:       equ $8d ;

portad:         equ $8f ;port adc = input only
adr00h:         equ $90 ;adc result 0 register
adr01h:         equ $92 ;adc result 1 register
adr02h:         equ $94 ;adc result 2 register
adr03h:         equ $96 ;adc result 3 register
adr04h:         equ $98 ;adc result 4 register
adr05h:         equ $9a ;adc result 5 register
adr06h:         equ $9c ;adc result 6 register
adr07h:         equ $9e ;adc result 7 register

pwme:           equ $a0 ;pwm enable
pwmpol:         equ $a1 ;pwm polarity
pwmclk:         equ $a2 ;pwm clock select register
pwmprclk:       equ $a3 ;pwm prescale clock select register
pwmcae:         equ $a4 ;pwm center align select register
pwmctl:         equ $a5 ;pwm control register
pwmtst:         equ $a6 ;reserved
pwmprsc:        equ $a7 ;reserved
pwmscla:        equ $a8 ;pwm scale a
pwmsclb:        equ $a9 ;pwm scale b
pwmscnta:       equ $aa ;reserved
pwmscntb:       equ $ab ;reserved
pwmcnt0:        equ $ac ;pwm channel 0 counter
pwmcnt1:        equ $ad ;pwm channel 1 counter
pwmcnt2:        equ $ae ;pwm channel 2 counter
pwmcnt3:        equ $af ;pwm channel 3 counter
pwmcnt4:        equ $b0 ;pwm channel 4 counter
pwmcnt5:        equ $b1 ;pwm channel 5 counter
pwmcnt6:        equ $b2 ;pwm channel 6 counter
pwmcnt7:        equ $b3 ;pwm channel 7 counter
pwmper0:        equ $b4 ;pwm channel 0 period
pwmper1:        equ $b5 ;pwm channel 1 period
pwmper2:        equ $b6 ;pwm channel 2 period
pwmper3:        equ $b7 ;pwm channel 3 period
pwmper4:        equ $b8 ;pwm channel 4 period
pwmper5:        equ $b9 ;pwm channel 5 period
pwmper6:        equ $ba ;pwm channel 6 period
pwmper7:        equ $bb ;pwm channel 7 period
```

```
pwmdty0:        equ $bc ;pwm channel 0 duty cycle
pwmdty1:        equ $bd ;pwm channel 1 duty cycle
pwmdty2:        equ $be ;pwm channel 2 duty cycle
pwmdty3:        equ $bf ;pwm channel 3 duty cycle
pwmdty4:        equ $c0 ;pwm channel 0 duty cycle
pwmdty5:        equ $c1 ;pwm channel 1 duty cycle
pwmdty6:        equ $c2 ;pwm channel 2 duty cycle
pwmdty7:        equ $c3 ;pwm channel 3 duty cycle
pwmsdn:         equ $c4 ;pwm shutdown register

sc0bdh:         equ $c8 ;sci 0 baud reg hi byte
sc0bdl:         equ $c9 ;sci 0 baud reg lo byte
sc0cr1:         equ $ca ;sci 0 control1 reg
sc0cr2:         equ $cb ;sci 0 control2 reg
sc0sr1:         equ $cc ;sci 0 status reg 1
sc0sr2:         equ $cd ;sci 0 status reg 2
sc0drh:         equ $ce ;sci 0 data reg hi
sc0drl:         equ $cf ;sci 0 data reg lo
sc1bdh:         equ $d0 ;sci 1 baud reg hi byte
sc1bdl:         equ $d1 ;sci 1 baud reg lo byte
sc1cr1:         equ $d2 ;sci 1 control1 reg
sc1cr2:         equ $d3 ;sci 1 control2 reg
sc1sr1:         equ $d4 ;sci 1 status reg 1
sc1sr2:         equ $d5 ;sci 1 status reg 2
sc1drh:         equ $d6 ;sci 1 data reg hi
sc1drl:         equ $d7 ;sci 1 data reg lo
spi0cr1:        equ $d8 ;spi 0 control1 reg
spi0cr2:        equ $d9 ;spi 0 control2 reg
spi0br:         equ $da ;spi 0 baud reg
spi0sr:         equ $db ;spi 0 status reg hi

sp0dr:          equ $dd ;spi 0 data reg

ibad:           equ $e0 ;i2c bus address register
ibfd:           equ $e1 ;i2c bus frequency divider
ibcr:           equ $e2 ;i2c bus control register
ibsr:           equ $e3 ;i2c bus status register
ibdr:           equ $e4 ;i2c bus message data register

dlcbcr1:        equ $e8 ;bdlc control regsiter 1
dlcbsvr:        equ $e9 ;bdlc state vector register
dlcbcr2:        equ $ea ;bdlc control register 2
dlcbdr:         equ $eb ;bdlc data register
dlcbard:        equ $ec ;bdlc analog delay register
dlcbrsr:        equ $ed ;bdlc rate select register
dlcscr:         equ $ee ;bdlc control register
dlcbstat:       equ $ef ;bdlc status register
spi1cr1:        equ $f0 ;spi 1 control1 reg
spi1cr2:        equ $f1 ;spi 1 control2 reg
spi1br:         equ $f2 ;spi 1 baud reg
spi1sr:         equ $f3 ;spi 1 status reg hi

sp1dr:          equ $f5 ;spi 1 data reg
```

```
spi2cr1:        equ $f8 ;spi 2 control1 reg
spi2cr2:        equ $f9 ;spi 2 control2 reg
spi2br:         equ $fa ;spi 2 baud reg
spi2sr:         equ $fb ;spi 2 status reg hi

sp2dr:          equ $fd ;spi 2 data reg

fclkdiv:        equ $100 ;flash clock divider
fsec:           equ $101 ;flash security register

fcnfg:          equ $103 ;flash configuration register
fprot:          equ $104 ;flash protection register
fstat:          equ $105 ;flash status register
fcmd:           equ $106 ;flash command register

eclkdiv:        equ $110 ;eeprom clock divider

ecnfg:          equ $113 ;eeprom configuration register
eprot:          equ $114 ;eeprom protection register
estat:          equ $115 ;eeprom status register
ecmd:           equ $116 ;eeprom command register

atd1ctl0:       equ $120 ;adc1 control 0 (reserved)
atd1ctl1:       equ $121 ;adc1 control 1 (reserved)
atd1ctl2:       equ $122 ;adc1 control 2
atd1ctl3:       equ $123 ;adc1 control 3
atd1ctl4:       equ $124 ;adc1 control 4
atd1ctl5:       equ $125 ;adc1 control 5
atd1stat:       equ $126 ;adc1 status register hi
*atd1stat       equ $127 ;adc1 status register lo
atd1test:       equ $128 ;adc1 test (reserved)
*atd1test       equ $129 ;

atddien:        equ $12d ;adc1 input enable register

portad1:        equ $12f ;port adc1 = input only
adr10h:         equ $130 ;adc1 result 0 register
adr11h:         equ $132 ;adc1 result 1 register
adr12h:         equ $134 ;adc1 result 2 register
adr13h:         equ $136 ;adc1 result 3 register
adr14h:         equ $138 ;adc1 result 4 register
adr15h:         equ $13a ;adc1 result 5 register
adr16h:         equ $13c ;adc1 result 6 register
adr17h:         equ $13e ;adc1 result 7 register
can0ctl0:       equ $140 ;can0 control register 0
can0ctl1:       equ $141 ;can0 control register 1
can0btr0:       equ $142 ;can0 bus timing register 0
can0btr1:       equ $143 ;can0 bus timing register 1
can0rflg:       equ $144 ;can0 receiver flags
can0rier:       equ $145 ;can0 receiver interrupt enables
can0tflg:       equ $146 ;can0 transmit flags
can0tier:       equ $147 ;can0 transmit interrupt enables
can0tarq:       equ $148 ;can0 transmit message abort control
can0taak:       equ $149 ;can0 transmit message abort status
```

```
can0tbel:       equ $14a ;can0 transmit buffer select
can0idac:       equ $14b ;can0 identfier acceptance control

can0rerr:       equ $14e ;can0 receive error counter
can0terr:       equ $14f ;can0 transmit error counter
can0ida0:       equ $150 ;can0 identifier acceptance register 0
can0ida1:       equ $151 ;can0 identifier acceptance register 1
can0ida2:       equ $152 ;can0 identifier acceptance register 2
can0ida3:       equ $153 ;can0 identifier acceptance register 3
can0idm0:       equ $154 ;can0 identifier mask register 0
can0idm1:       equ $155 ;can0 identifier mask register 1
can0idm2:       equ $156 ;can0 identifier mask register 2
can0idm3:       equ $157 ;can0 identifier mask register 3
can0ida4:       equ $158 ;can0 identifier acceptance register 4
can0ida5:       equ $159 ;can0 identifier acceptance register 5
can0ida6:       equ $15a ;can0 identifier acceptance register 6
can0ida7:       equ $15b ;can0 identifier acceptance register 7
can0idm4:       equ $15c ;can0 identifier mask register 4
can0idm5:       equ $15d ;can0 identifier mask register 5
can0idm6:       equ $15e ;can0 identifier mask register 6
can0idm7:       equ $15f ;can0 identifier mask register 7
can0rxfg:       equ $160 ;can0 rx foreground buffer thru +$16f
can0txfg:       equ $170 ;can0 tx foreground buffer thru +$17f

can1ctl0:       equ $180 ;can1 control register 0
can1ctl1:       equ $181 ;can1 control register 1
can1btr0:       equ $182 ;can1 bus timing register 0
can1btr1:       equ $183 ;can1 bus timing register 1
can1rflg:       equ $184 ;can1 receiver flags
can1rier:       equ $185 ;can1 receiver interrupt enables
can1tflg:       equ $186 ;can1 transmit flags
can1tier:       equ $187 ;can1 transmit interrupt enables
can1tarq:       equ $188 ;can1 transmit message abort control
can1taak:       equ $189 ;can1 transmit message abort status
can1tbel:       equ $18a ;can1 transmit buffer select
can1idac:       equ $18b ;can1 identfier acceptance control

can1rerr:       equ $18e ;can1 receive error counter
can1terr:       equ $18f ;can1 transmit error counter
can1ida0:       equ $190 ;can1 identifier acceptance register 0
can1ida1:       equ $191 ;can1 identifier acceptance register 1
can1ida2:       equ $192 ;can1 identifier acceptance register 2
can1ida3:       equ $193 ;can1 identifier acceptance register 3
can1idm0:       equ $194 ;can1 identifier mask register 0
can1idm1:       equ $195 ;can1 identifier mask register 1
can1idm2:       equ $196 ;can1 identifier mask register 2
can1idm3:       equ $197 ;can1 identifier mask register 3
can1ida4:       equ $198 ;can1 identifier acceptance register 4
can1ida5:       equ $199 ;can1 identifier acceptance register 5
can1ida6:       equ $19a ;can1 identifier acceptance register 6
can1ida7:       equ $19b ;can1 identifier acceptance register 7
can1idm4:       equ $19c ;can1 identifier mask register 4
can1idm5:       equ $19d ;can1 identifier mask register 5
can1idm6:       equ $19e ;can1 identifier mask register 6
```

```
can1idm7:       equ $19f ;can1 identifier mask register 7
can1rxfg:       equ $1a0 ;can1 rx foreground buffer thru +$1af
can1txfg:       equ $1b0 ;can1 tx foreground buffer thru +$1bf

can2ctl0:       equ $1c0 ;can2 control register 0
can2ctl1:       equ $1c1 ;can2 control register 1
can2btr0:       equ $1c2 ;can2 bus timing register 0
can2btr1:       equ $1c3 ;can2 bus timing register 1
can2rflg:       equ $1c4 ;can2 receiver flags
can2rier:       equ $1c5 ;can2 receiver interrupt enables
can2tflg:       equ $1c6 ;can2 transmit flags
can2tier:       equ $1c7 ;can2 transmit interrupt enables
can2tarq:       equ $1c8 ;can2 transmit message abort control
can2taak:       equ $1c9 ;can2 transmit message abort status
can2tbel:       equ $1ca ;can2 transmit buffer select
can2idac:       equ $1cb ;can2 identfier acceptance control

can2rerr:       equ $1ce ;can2 receive error counter
can2terr:       equ $1cf ;can2 transmit error counter
can2ida0:       equ $1d0 ;can2 identifier acceptance register 0
can2ida1:       equ $1d1 ;can2 identifier acceptance register 1
can2ida2:       equ $1d2 ;can2 identifier acceptance register 2
can2ida3:       equ $1d3 ;can2 identifier acceptance register 3
can2idm0:       equ $1d4 ;can2 identifier mask register 0
can2idm1:       equ $1d5 ;can2 identifier mask register 1
can2idm2:       equ $1d6 ;can2 identifier mask register 2
can2idm3:       equ $1d7 ;can2 identifier mask register 3
can2ida4:       equ $1d8 ;can2 identifier acceptance register 4
can2ida5:       equ $1d9 ;can2 identifier acceptance register 5
can2ida6:       equ $1da ;can2 identifier acceptance register 6
can2ida7:       equ $1db ;can2 identifier acceptance register 7
can2idm4:       equ $1dc ;can2 identifier mask register 4
can2idm5:       equ $1dd ;can2 identifier mask register 5
can2idm6:       equ $1de ;can2 identifier mask register 6
can2idm7:       equ $1df ;can2 identifier mask register 7
can2rxfg:       equ $1e0 ;can2 rx foreground buffer thru +$1ef
can2txfg:       equ $1f0 ;can2 tx foreground buffer thru +$1ff

can3ctl0:       equ $200 ;can3 control register 0
can3ctl1:       equ $201 ;can3 control register 1
can3btr0:       equ $202 ;can3 bus timing register 0
can3btr1:       equ $203 ;can3 bus timing register 1
can3rflg:       equ $204 ;can3 receiver flags
can3rier:       equ $205 ;can3 receiver interrupt enables
can3tflg:       equ $206 ;can3 transmit flags
can3tier:       equ $207 ;can3 transmit interrupt enables
can3tarq:       equ $208 ;can3 transmit message abort control
can3taak:       equ $209 ;can3 transmit message abort status
can3tbel:       equ $20a ;can3 transmit buffer select
can3idac:       equ $20b ;can3 identfier acceptance control

can3rerr:       equ $20e ;can3 receive error counter
can3terr:       equ $20f ;can3 transmit error counter
can3ida0:       equ $210 ;can3 identifier acceptance register 0
```

```
can3ida1:       equ $211 ;can3 identifier acceptance register 1
can3ida2:       equ $212 ;can3 identifier acceptance register 2
can3ida3:       equ $213 ;can3 identifier acceptance register 3
can3idm0:       equ $214 ;can3 identifier mask register 0
can3idm1:       equ $215 ;can3 identifier mask register 1
can3idm2:       equ $216 ;can3 identifier mask register 2
can3idm3:       equ $217 ;can3 identifier mask register 3
can3ida4:       equ $218 ;can3 identifier acceptance register 4
can3ida5:       equ $219 ;can3 identifier acceptance register 5
can3ida6:       equ $21a ;can3 identifier acceptance register 6
can3ida7:       equ $21b ;can3 identifier acceptance register 7
can3idm4:       equ $21c ;can3 identifier mask register 4
can3idm5:       equ $21d ;can3 identifier mask register 5
can3idm6:       equ $21e ;can3 identifier mask register 6
can3idm7:       equ $21f ;can3 identifier mask register 7
can3rxfg:       equ $220 ;can3 rx foreground buffer thru +$22f
can3txfg:       equ $230 ;can3 tx foreground buffer thru +$23f

ptt:            equ $240 ;portt data register
ptit:           equ $241 ;portt input register
ddrt:           equ $242 ;portt direction register
rdrt:           equ $243 ;portt reduced drive register
pert:           equ $244 ;portt pull device enable
ppst:           equ $245 ;portt pull polarity select

pts:            equ $248 ;ports data register
ptis:           equ $249 ;ports input register
ddrs:           equ $24a ;ports direction register
rdrs:           equ $24b ;ports reduced drive register
pers:           equ $24c ;ports pull device enable
ppss:           equ $24d ;ports pull polarity select
woms:           equ $24e ;ports wired or mode register

ptm:            equ $250 ;portm data register
ptim:           equ $251 ;portm input register
ddrm:           equ $252 ;portm direction register
rdrm:           equ $253 ;portm reduced drive register
perm:           equ $254 ;portm pull device enable
ppsm:           equ $255 ;portm pull polarity select
womm:           equ $256 ;portm wired or mode register
modrr:          equ $257 ;portm module routing register
ptp:            equ $258 ;portp data register
ptip:           equ $259 ;portp input register
ddrp:           equ $25a ;portp direction register
rdrp:           equ $25b ;portp reduced drive register
perp:           equ $25c ;portp pull device enable
ppsp:           equ $25d ;portp pull polarity select
piep:           equ $25e ;portp interrupt enable register
pifp:           equ $25f ;portp interrupt flag register
pth:            equ $260 ;porth data register
ptih:           equ $261 ;porth input register
ddrh:           equ $262 ;porth direction register
rdrh:           equ $263 ;porth reduced drive register
perh:           equ $264 ;porth pull device enable
```

```
ppsh:           equ $265 ;porth pull polarity select
pieh:           equ $266 ;porth interrupt enable register
pifh:           equ $267 ;porth interrupt flag register
ptj:            equ $268 ;portp data register
ptij:           equ $269 ;portp input register
ddrj:           equ $26a ;portp direction register
rdrj:           equ $26b ;portp reduced drive register
perj:           equ $26c ;portp pull device enable
ppsj:           equ $26d ;portp pull polarity select
piej:           equ $26e ;portp interrupt enable register
pifj:           equ $26f ;portp interrupt flag register

can4ctl0:       equ $280 ;can4 control register 0
can4ctl1:       equ $281 ;can4 control register 1
can4btr0:       equ $282 ;can4 bus timing register 0
can4btr1:       equ $283 ;can4 bus timing register 1
can4rflg:       equ $284 ;can4 receiver flags
can4rier:       equ $285 ;can4 receiver interrupt enables
can4tflg:       equ $286 ;can4 transmit flags
can4tier:       equ $287 ;can4 transmit interrupt enables
can4tarq:       equ $288 ;can4 transmit message abort control
can4taak:       equ $289 ;can4 transmit message abort status
can4tbel:       equ $28a ;can4 transmit buffer select
can4idac:       equ $28b ;can4 identfier acceptance control

can4rerr:       equ $28e ;can4 receive error counter
can4terr:       equ $28f ;can4 transmit error counter
can4ida0:       equ $290 ;can4 identifier acceptance register 0
can4ida1:       equ $291 ;can4 identifier acceptance register 1
can4ida2:       equ $292 ;can4 identifier acceptance register 2
can4ida3:       equ $293 ;can4 identifier acceptance register 3
can4idm0:       equ $294 ;can4 identifier mask register 0
can4idm1:       equ $295 ;can4 identifier mask register 1
can4idm2:       equ $296 ;can4 identifier mask register 2
can4idm3:       equ $297 ;can4 identifier mask register 3
can4ida4:       equ $298 ;can4 identifier acceptance register 4
can4ida5:       equ $299 ;can4 identifier acceptance register 5
can4ida6:       equ $29a ;can4 identifier acceptance register 6
can4ida7:       equ $29b ;can4 identifier acceptance register 7
can4idm4:       equ $29c ;can4 identifier mask register 4
can4idm5:       equ $29d ;can4 identifier mask register 5
can4idm6:       equ $29e ;can4 identifier mask register 6
can4idm7:       equ $29f ;can4 identifier mask register 7
can4rxfg:       equ $2a0 ;can4 rx foreground buffer thru +$2af
can4txfg:       equ $2b0 ;can4 tx foreground buffer thru +$2bf

* end registers
```