- **The 9S12 Pulse Accumulator**
- ECT_16B8C Block User Guide
- Huang, Sections 8.8

Pulse Accumulator on the HCS12

• The pulse accumulator is the third timer function besides the input-capture and output-compare functions.

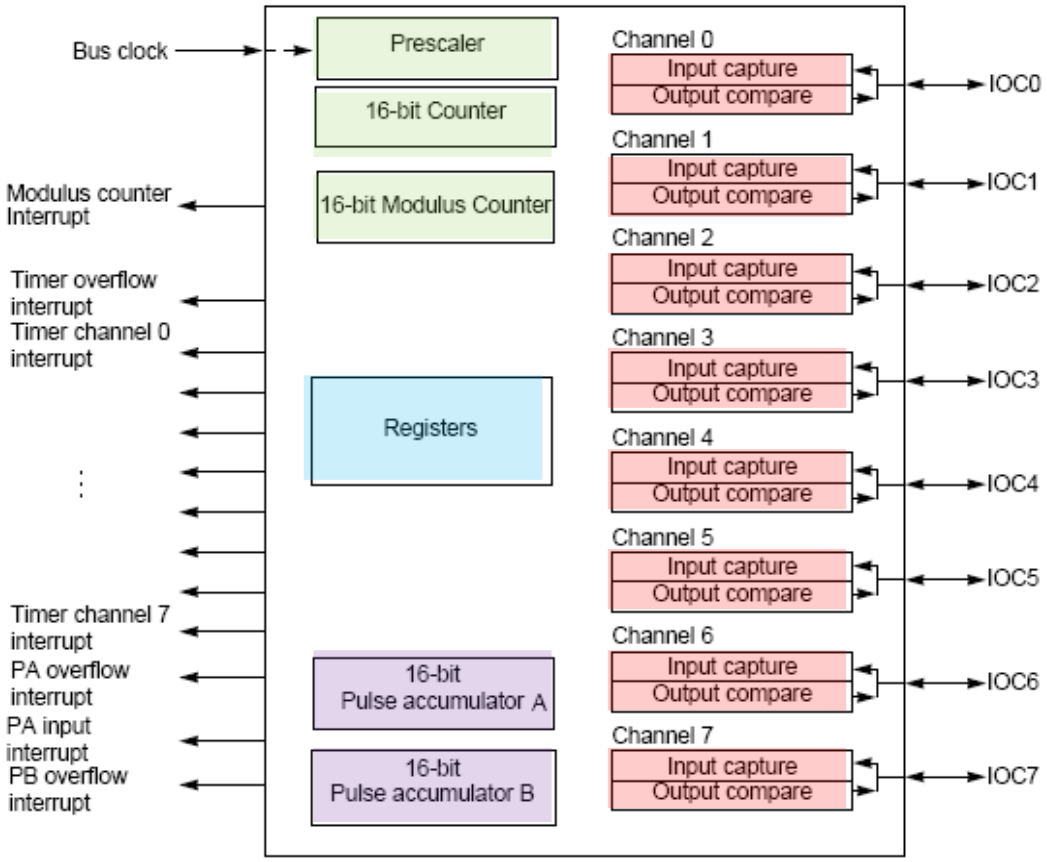


Figure 1-1 Timer Block Diagram

• A pulse accumulator counts the number of active edges at the input of its channel.

• The HCS12 has **four 8-bit pulse accumulators**, configurable as **two 16- bit pulse accumulators**. In the following we will discuss the 16-bit pulse accumulator A, PACA, made up of the two 8-bit pulse accumulators PACN3 and PACN2.

• To use the pulse accumulator connect an input to **Port T7** (PT7).

• The pulse accumulator operates in two modes:
1. Event-Count Mode
2. Gated Time Accumulation Mode

• **In Event-Count Mode**, the pulse accumulator counts the number of **rising** or **falling** edges on Port T7
– You can set up the pulse accumulator to select which edge to count
– The counts are held in the 16-bit PACA register
– On each selected edge the **PAIF** flag of the PAFLG register is set
– When PACA overflows from 0xFFFF to 0x0000, the **PAOVF** flag of the PAFLG register is set

• **In Gated Time Accumulation Mode** the pulse accumulator counts clock cycles while the input to Port T7 is high or low
– In Gated Time Accumulation Mode the pulse accumulator uses the Timer Clock. To use the pulse accumulator in Gated Time Accumulation Mode you must enable the Timer Clock by writing a 1 to the **TEN** bit of **TSCR1**
– You can set up the pulse accumulator to count while PT7 is high or to count while PT7 is low
– The clock for the pulse accumulator is **the bus clock divided by 64**
– With an 24 MHz bus clock, the clock frequency of the pulse accumulator is 375 kHz, for a period of 2.67 μs
– For example, if the pulse accumulator is set up to count while Port T7 is high, and it counts 729 clock pulses, then the input to Port T7 was high for 729 x 2.67 μs = 1.94 ms

The Pulse Accumulator

• The pulse accumulator uses PT7 as an input
– To use the pulse accumulator make sure bit 7 of TIOS is 0 (otherwise PT7 used as output compare pin)
– To use the pulse accumulator make sure bits 6 and 7 of TCTL3 are 0 (otherwise timer function connected to PT7)

• The pulse accumulator uses three registers: PACTL, PAFLG, PACA

• To use the pulse accumulator you have to program the PACTL register

• The PAFLG register has flags to indicate the status of the pulse accumulator
– You clear a flag bit by writing a 1 to that bit

• The count value is stored in the 16-bit PACA register

	BIT7	6	5	4	3	2	1	BIT0
R	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-15 16-Bit Pulse Accumulator Control Register (PACTL)

Read or write any time

All bits reset to zero.

PAEN: Pulse accumulator enable
1= Enable PA (PACN3/PAC3 and PACN2/PAC2 cannot be enabled)

PAMOD: Pulse accumulator mode
0=Event count mode
1=Gated time accumulator mode

PEDGE: Pulse edge control
For PAMOD = 0 (event counter mode)
0=Falling edge on PT7 pin
1=Rising edge on PT7 pin

For PAMOD = 1 (gated time accumulation mode)
0=PT7 input pin high enables bus clock divided by 64 to PAC and trailing falling edge on PT7 sets PAIF flag.
1=PT7 input pin low enables bus clock divided by 64 to PAC and trailing rising edge on PT7 sets PAIF flag.

PAMOD	PEDGE	Pin Action
0	0	Falling
0	1	Rising
1	0	Div by 64 clock with PT7 high
1	1	Div by 64 with PT7 low

PAOVI: Pulse accumulator overflow interrupt
0=disable
1=enable

PAI: Pulse accumulator input interrupt
0=disable
1=enable

	BIT7	6	5	4	3	2	1	BIT0
R	0	0	0	0	0	0	PAOVF	PAIF
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-16 Pulse Accumulator A Flag Register (PAFLG)

Read anytime.

Write used in the flag clearing mechanism. Writing a one to the flag clears the flag. Writing a zero will not affect the current status of the bit.

PAOVF: Pulse accumulator overflow flag
Set when PAC overflows from 0xFFFF to 0x0000 and can be cleared by writing a 1 to it.

PAIF: Pulse accumulator input flag
Set when selected edge is detected at the PT7 input pin.

The 16-bit PACA register is at address 0x0062

	BIT7	6	5	4	3	2	1	BIT0
R	pacnt7(15)	pacnt6(14)	pacnt5(13)	pacnt4(12)	pacnt3(11)	pacnt2(10)	pacnt1(9)	pacnt0(8)
W								
RESET:	0	0	0	0	0	0	0	0

Figure 3-17 Pulse Accumulators Count Register 3 (PACN3)

	BIT7	6	5	4	3	2	1	BIT0
R W	pacnt7	pacnt6	pacnt5	pacnt4	pacnt3	pacnt2	pacnt1	pacnt0
RESET:	0	0	0	0	0	0	0	0

Figure 3-18 Pulse Accumulators Count Register 2 (PACN2)

Read or write any time.

All bits reset to zero.

The 16-bit PACB register is at address 0x0064

	BIT7	6	5	4	3	2	1	BIT0
R W	pacnt7(15)	pacnt6(14)	pacnt5(13)	pacnt4(12)	pacnt3(11)	pacnt2(10)	pacnt1(9)	pacnt0(8)
RESET:	0	0	0	0	0	0	0	0

Figure 3-19 Pulse Accumulators Count Register 1 (PACN1)

	BIT7	6	5	4	3	2	1	BIT0
R W	pacnt7	pacnt6	pacnt5	pacnt4	pacnt3	pacnt2	pacnt1	pacnt0
RESET:	0	0	0	0	0	0	0	0

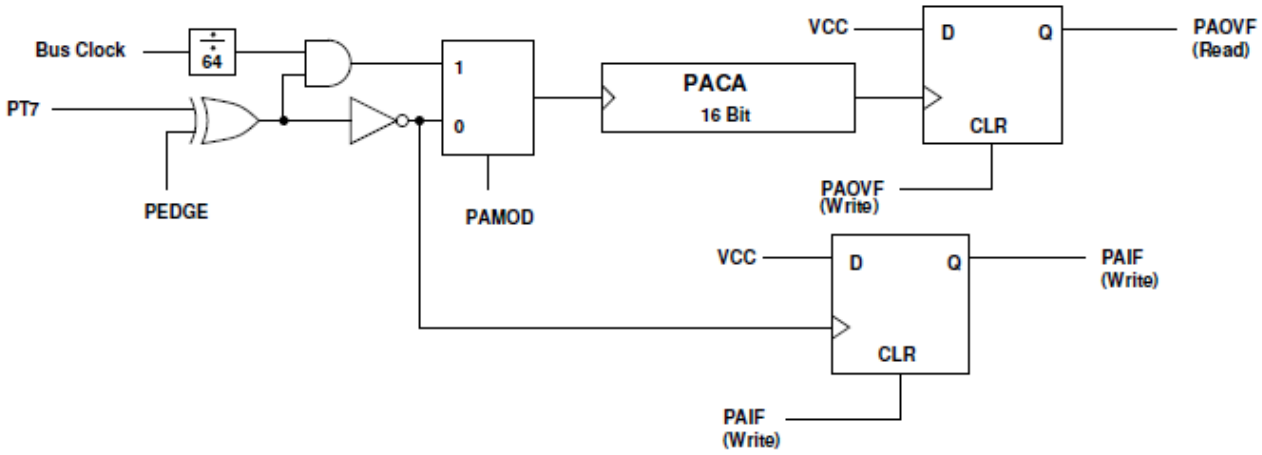
Figure 3-20 Pulse Accumulators Count Register 0 (PACN0)

Read or write any time.

All bits reset to zero.

The Pulse Accumulator

PULSE ACCUMULATOR LOGIC



PAMOD	PAEDGE	ACTION
0	0	Increment PACNT on falling edge of PAI
0	1	Increment PACNT on rising edge of PAI
1	0	Count E/64 if PT7 = 1
1	1	Count E/64 if PT7 = 0

The Pulse Accumulator PACA

• Here is a C program which counts the number of rising edges on PT7:

```
#include "hcs12.h"
#include "DBug12.h"

#define PACA *(unsigned int *) 0x62; /* pulse accumulator A3 count */

int start_count,end_count,total_count;
main()
{
        int i;
        TIOS = TIOS & ~0x80;        /* PT7 input */
        TCTL3 = TCTL3 & ~0xC0   /* Disconnect IC/OC logic from PT7 */
        PACTL = 0x50;         /* 0 1 0 1 0 0 0 0 */
                              /*   | | |       | |*/
                              /*   | | |       | \_ No interurrupt on edge */
                              /*   | | |       \___ No interurrupt on overflow */
                              /*   | | \___________ Rising Edge */
                              /*   | \_____________ Event Count Mode */
                              /*   \_______________ Enable PACA (16 bit mode) */
        start_count = PACA;
        for (i=0;i<10000;i++) ;         /* Software Delay */
        end_count = PACA;
        total_count = end_count - start_count;
        DB12FNP->printf("Total counts = %d\r\n",total_count);
}
```

The Pulse Accumulator PACA

• Here is a C program which determines how long the input on PA7 is high:

```
#include "hcs12.h"
#include "DBug12.h"

#define PACA *(unsigned int *) 0x62; /* pulse accumulator A3 count */

int start_count,end_count,total_count;
main()
{
        int i;
        TSCR = 0x80;                    /* Turn on timer clock */
        TIOS = TIOS & ~0x80;            /* PT7 input */
        TCTL3 = TCTL3 & ~0xC0           /* Disconnect IC/OC logic from PT7 */
        PACTL = 0x60;       /* 0 1 1 0 0 0 0 0 */
                            /*   |  | |       | | */
                            /*   |  | |       | \_ No interrupt on edge */
                            /*   |  | |       \___ No interrupt on overflow */
                            /*   |  | \___________ Count while input high */
                            /*   |  \_____________ Gated Counter Mode */
                            /*   \________________ Enable PACA (16 bit mode) */
        start_count = PACA;
        while ((PTT & 0x80) == 0) ;     /* Wait until input goes high */
        while ((PTT & 0x80) == 0x80) ;  /* Wait until input goes low */
        end_count = PACA;
        total_count = end_count - start_count;
        DB12FNP->printf("Total clock cycles = %d\r\n",total_count);
}
```