Timing & PWM System

1

Timing System

2

- Valvano’s chapter 6
- TIM Block User Guide, Chapter 15
- PWM Block User Guide, Chapter 12

Timing System components

3

- Counting mechanisms
- Input time capture mechanisms
- Time output compare mechanisms
- PWM mechanisms

Usage of Timing System

4

- Input Capture
  - Generate interrupts
  - Measure period
  - Measure pulse width
  - Count pulses
  - Decode remote control
- Output Compare
  - Generate periodic interrupts
  - Generate square waves
  - Generate pulse width encoded signals
- Both Together
  - Measure Frequency
Usage of PWM System

- Pulse Width Modulation
  - Programmable period
  - Programmable duty cycle

Pins
Counting system

- 16-bit unsigned counter (TCNT) incremented at a fixed rate determined by two programmable bits (PR2, 1 and 0). The counter cannot be stopped or reset.

<table>
<thead>
<tr>
<th>PR2</th>
<th>PR1</th>
<th>PR0</th>
<th>Timer Clock</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Bus Clock / 1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Bus Clock / 2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Bus Clock / 4</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Bus Clock / 8</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>Bus Clock / 16</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Bus Clock / 32</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Bus Clock / 64</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Bus Clock / 128</td>
</tr>
</tbody>
</table>

Input Capture System

- The 68HC12 has 8 input capture modules:
  - Each input Capture module has:
    - An external input pin ICn
    - A flag bit
    - Two edge control lines
    - An interrupt mask bit
    - A 16-bit input capture register

Input Capture basic structure

Actions occurring as a result of a capture event

1. The current TCNT value is copied into the input capture register (TCn) 16 bit
2. The input capture flag is set (CnF) (TFLG1 register)
3. If the mask bit is armed (CI_n set to 1) an interrupt is requested (TIE register)
Input Capture Output Compare Registers

- 8 bit Input-Output Select for each of 7 pins

Input Capture modes

- Edge
- TIE: interrupt enable (CI0..7)
- Prescalar

Input Capture Status

- Interrupt flag
- Event occurs and flag has not been cleared
  - Overflow:

Input Capture flags

- The flags are cleared by writing a 1 into the specific flag bit we wish to clear

Table 3-3: Edge Detector Circuit Configuration

<table>
<thead>
<tr>
<th>EDiGnB</th>
<th>EDiGnA</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Capture disabled</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Capture on rising edges only</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Capture on falling edges only</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Capture on any edge (rising or falling)</td>
</tr>
</tbody>
</table>

Example

```assembly
ldaa #$01
staa $1023
```

TFLG1 = 0x01

Do not use bset or TFLG1 |= 0x01 (no read-modify-write operations!)

If for example we had 0xFF in TFLG1 as a result of the OR we end up writing 0xFF in TFLG1 so we clear all the flags instead of the LSB only
**Pulse Accumulator**

- Pulse accumulator counter PACNT
- Enable
- Edge
  - Count on either rising and falling edges
- Overflow Flag
- Pin attached to IOC_7

**Response time to Input Capture**

- There is a latency between the rise edge of the interrupt and when the increment of TIME really occurs:
  1. finish the current instruction
  2. process the interrupt
  3. execute the interrupt handler (including changing TIME)

6811
- Longest instruction: 41 cycles
- Process the interrupt: 14 cycles
- Execute handler: 28 cycles (our code)
- Max latency: 83 cycles (=41.5 μs)

**Example Input Applications**

- Period
- Pulsewidth

**Period Measurement**

- **Resolution**
  smallest change in period that can reliably be measured
- **Precision**
  number of separate and distinguishable measurements that we can take (e.g. 65536 = 16 bits)
- **Range**
  The minimum and maximum values that can reliably be measured
16-bit period measurement with 500 ns resolution using input capture

- The period is calculated as the difference in TIC1 values from one rising edge to the other.
- This method does not operate properly if the period is larger than 65535 cycle.
- The shortest period that can be handled is given by:
  num. cycles to process the interrupt + num. cycles of the handler
- The resolution is 1/busclock frequency because the period must increase by at least this amount before the difference between TIC1 measurements can be appreciated.

Pulse-width measurements

- The basic idea is to cause an input capture event on both the rising and falling edges of an external signal.

Pulse width measurement using two input capture channels

- The lower bound on the range is determined by the software overhead to process the input capture interrupt.
- The rising edge time will be measured by IC2 without the need of an interrupt routine, and the falling edge interrupt will be handled by IC1.

Output compare system

- The 68HC12 has 8 output compare modules.
- Each output compare module has:
  - An external output pin OCn
  - A flag bit
  - A force output compare control bit FOCn
  - Two control bits (OMn, OLn)
  - An interrupt mask bit
  - A 16-bit output compare register.
Output compare basic structure

Figure 5.17: The basic components of output compare.

An output compare event occurs when either:
1. The 16-bit TCNT matches the 16-bit OC register
2. The software writes a 1 to the FOC bit

As a result of an output compare event:
1. The OCn output signal change
2. The output compare flag OCnF is set
3. If the mask bit OCnI is armed an interrupt is requested

Output compare modes and levels

The TCTL1,2 register determines what effect the output compare event will have (none, toggle, clear, set) on the output pins (OC0..7)

Output compare flags

- The software can determine if an output compare event occurred by reading the flag registers
- The flags are cleared by writing a 1 into the specific flag bit we wish to clear
- If the output compare flags are armed and TOI is set, then an interrupt will be requested as soon as the flag is set
Output compare applications

- create square waves
- create variable duty cycles waveforms
- generate pulses
- implement time delays
- execute periodic interrupts
- ...

Square wave generator

Fastest square waveform $\rightarrow 2 \times \text{Total execution time}$

Total execution time = time to process the interrupt + time to execute the handler

for the 68HC11 is 14 cycles

Frequency Measurement

- Basic idea:
  - count the number of input pulses occur for a fixed amount of time
- Use input capture to count the pulses
- Use output capture to create the fixed time interval

$$f = \frac{\text{counter}}{\text{fixed time}}$$

Frequency resolution = $\frac{1}{\text{fixed time}}$

PWM features

- PWM channels 0..5
- Programmable duty cycle and pulse width
  - free running 8 bit up/down counter 0..period reg
  - Period register
  - Duty register
- PWMPOL polarity (cycle starts low or high)
- Clock prescalar
  - 4 different clocks
  - A, SA (derived from A),
  - B, SB
- Double buffer (16bit) if 2 channels combined
Pulse Width Modulation

Average Current (= DC current)

- **small**
- **middle**
- **large**

Effective and popular mechanism for embedded systems to control external devices

Example pulse width modulator

duty cycle = high / (high + low)