Application Note

Buffer Operation of Timer Z

For H8/36077 series

Contact Info:

Brazen Tek, Inc.
20121 Ventura Blvd.
Suite 310
Woodland Hills, CA 91364
USA
Tel/Fax: (818) 710-9262
E-mail: info@brazentek.com

August, 2008

Introduction

Pulse widths are often measured in different circuits to evaluate their performance, control voltage effect and operation mode. The buffer operation of the input-capture function supported by Timer Z is used to measure both the high and low level widths of the pulse. This application gives a better understanding of the pulse position that varies with modulation signal.

Target Device

H8/36077 series

Contents

1. Specifications

i. Pin A0 of H8/36077, P60 (Pin 36) is used to measure the voltage of the pulse input.
ii. The 16-bit timer counter of channel 0 (TCNT_0) is to measure the time from rising edge to falling edge of the pulse so that high level width of the pulse is obtained.
iii. The 16-bit timer counter of channel 0 (TCNT_0) is to measure the time from falling edge to rising edge of the pulse so that high level width of the pulse is obtained.
iv. The maximum width of a pulse that can be measured is 32.768 ms with the accuracy of ±0.5us (note that the internal clock used is 32.768MHz).

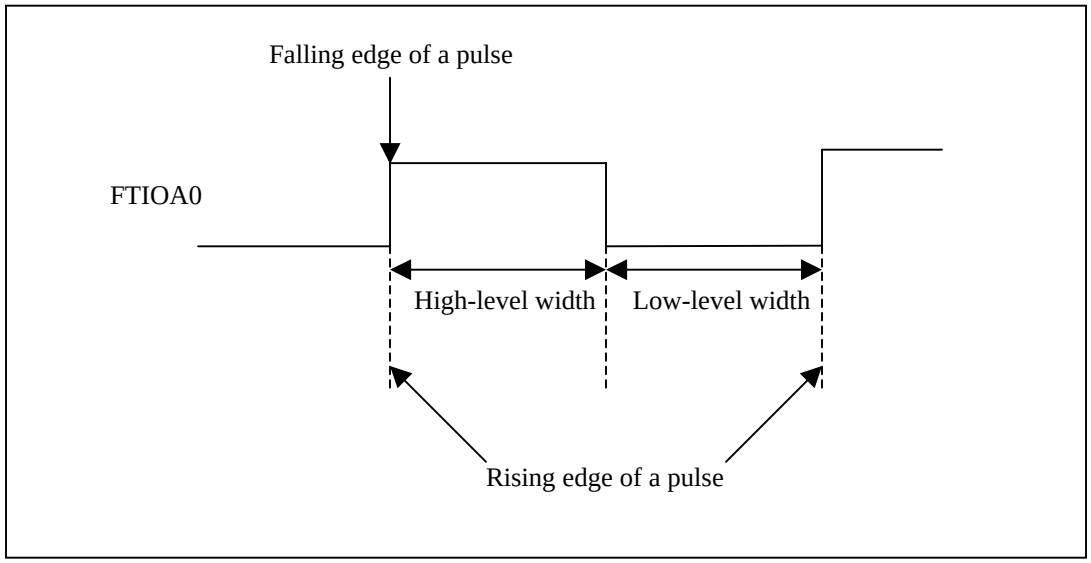


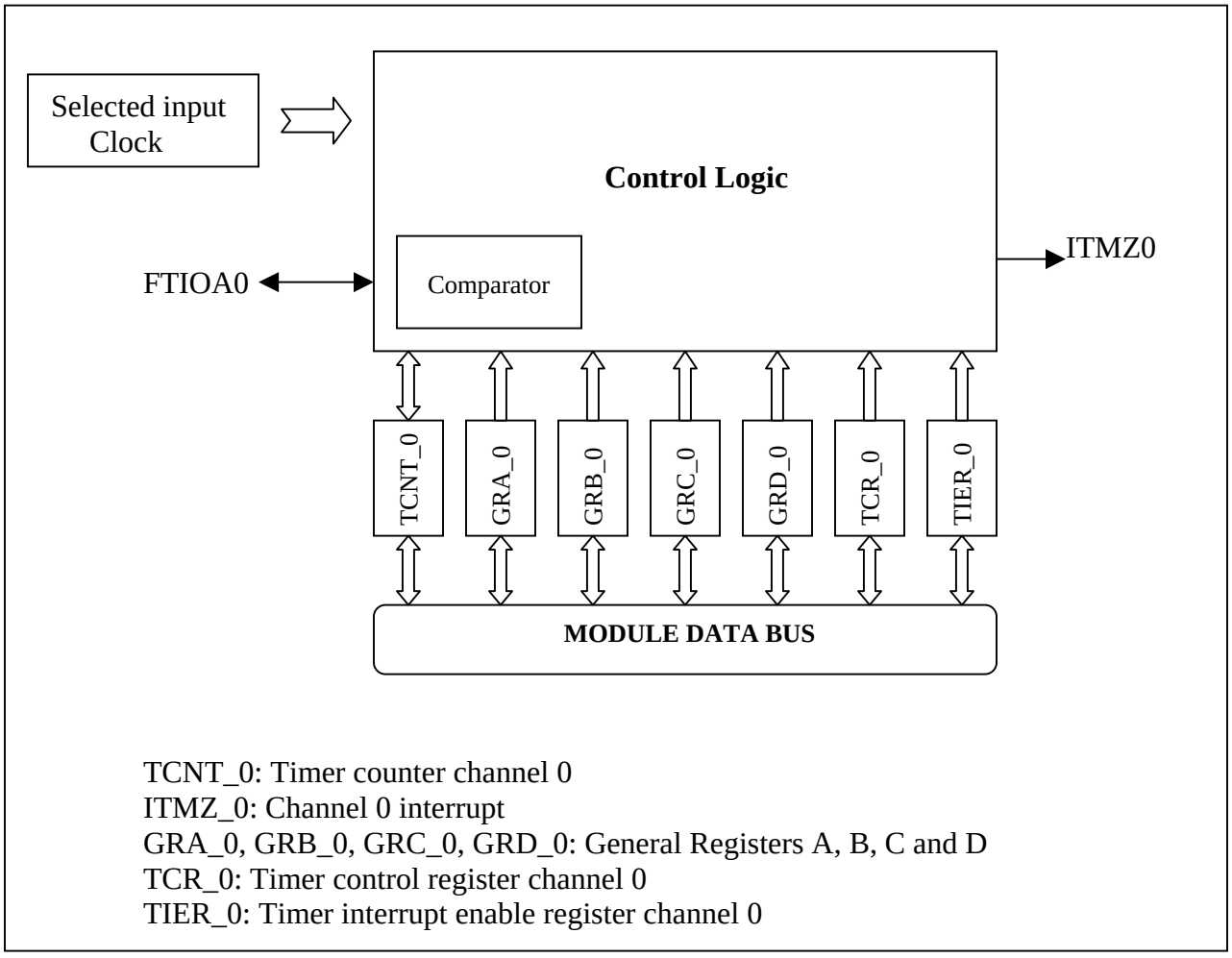
Figure 1 - Measurement of input pulse width

2. Description of Functions used

Figure 2.1 shows the Timer Z input-capture block diagram. The block diagram is summarized as:

- The system clock (ϕ) is a 16 MHz oscillator clock that is used as the reference clock.
- Prescaler S (PSS) is a 13-bit counter using the system clock (ϕ) as its input clock. It cannot be read from or written to by the CPU.

- For this AN, the input clock chosen is ϕ/8. Timer control register of channel 0 (TCR_0) selects the input clock and TCNT_0 as clearing method. TCNT_0 counts up on rising edge of the selected input clock. TCNT_0 is to be cleared on compare-output/input-capture with general register GRA_0.
- GRA_0 is used as an input capture register. TCNT_0 value is transferred to GRA_0 using FTIOA0 pin's rising and falling edges. For that matter, timer I/O control register A0 (TIORA0) controls both GRA_0 and GRB_0 registers.
- GRC_0 is used as an input capture register. Timer I/O control register C0 (TIORC0) controls both GRC_0 and GRD_0 registers.
- In this AN, overflow flag OVF is set to 1 when TCNT_0 overflows, GRA_0 input capture is detected and the input capture/output compare (IMFA) is set to 1. Timer status register channel 0 (TSR_0) shows Timer Z status.
- In this AN, all interrupts are disabled except TSR_0 and IMFA flags. Timer interrupt enable register channel 0 (TIER_0) enables different interrupt requests.
- In this AN, TCNT_0 is incremented at the rising edge of the selected input clock. Timer counter channel 0 (TCNT_0) is a 16-bit readable/writable upward counter that gets incremented by either internal or external input clocks.
- In this AN, GRA_0 operates as an input capture register. The TCNT_0 value is transferred to GRA_0 register using rising and falling edges of the FTIOA0 pin. General register A channel 0 (GRA_0) is a 16-bit read/write register.
- In this AN, GRC_0 operates as a buffer register for GRA_0. The GRA_0 value is transferred to GRC_0 register using rising and falling edges of the FTIOA0 pin. General register C channel 0 (GRC_0) is a 16-bit read-write register.
- In this AN, TCNT_0 is for start counting and TCNT_1 for stop counting. Timer start register (TSTR) starts/stops TCNT_0 and TCNT_1 operation.
- In this AN, TCNT_0 and TCNT_1 are independent channels. GRA_0 is in synch with GRC_0. Timer mode register (TMDR) selects synchronous or independent operations.
- In this AN, channels 0 and 1 are for normal operations. Timer function control register (TFCR) selects the output level based on the operation mode.
- Input capture/output compare pin A0 (FTIOA0) acts as an input capture pin this case and the value of TCNT_0 is transferred to GRA_0 at its rising and falling edges.



TCNT_0: Timer counter channel 0
ITMZ_0: Channel 0 interrupt
GRA_0, GRB_0, GRC_0, GRD_0: General Registers A, B, C and D
TCR_0: Timer control register channel 0
TIER_0: Timer interrupt enable register channel 0

Figure 2 – Timer Z (channel 0) Block diagram

3. Description of Operations

The operation described in this application note is shown in figure 3.1.

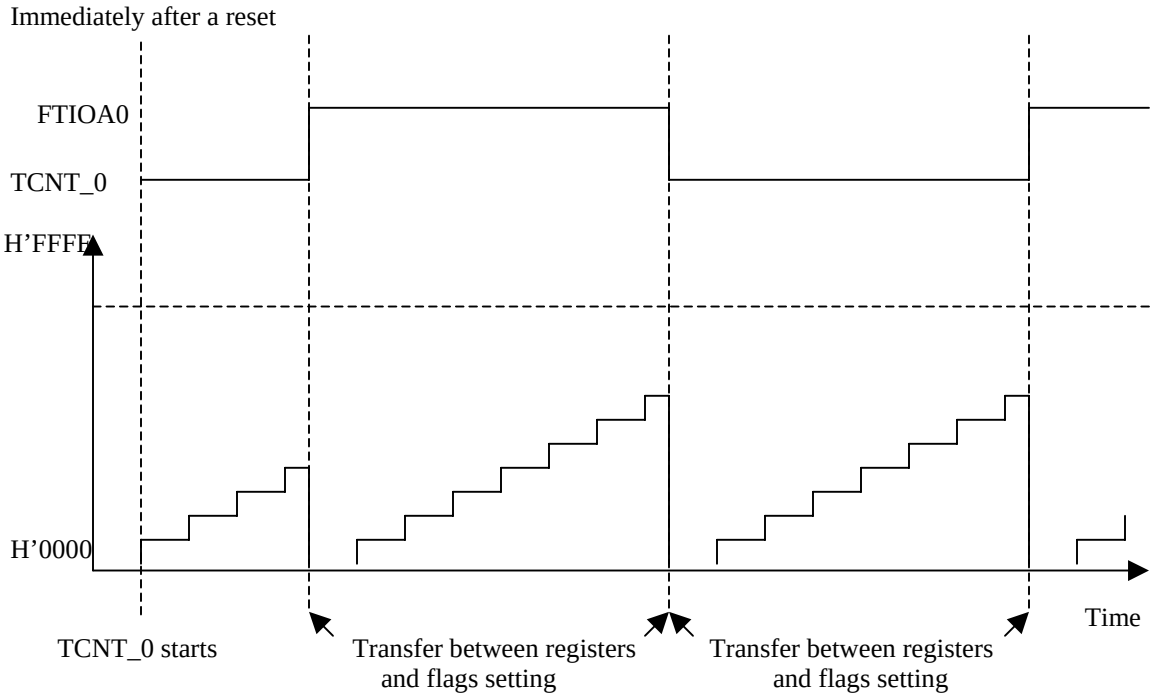


Figure 3.1 – Timer Z (Channel 0) Operation

4. Description of Software

4.1 Description of Modules

Table 4.1 describes the software used in this App. Note.

Module Name	Label Name	Function
Main	main	Sets Timer Z of channel 0, enables interrupts and sets registers
Pulse Measurement	tz0int	Timer Z channel 0 interrupt handling, stores in RAM

Table 4.1 - Description of Modules

4.2 Description of Arguments

None are used for this AN.

4.3 Description of Internal Registers

Table 4.2 describes the internal registers used in this AN.

Register	Name	Function	Address	Bit	Setting
TCR0	CCLR2	Timer control register channel 0 Counter is cleared to 0	H'F700	7	0
	CCLR1	CCLR2 = 0, CCLR1 = 0, CCLR1 = 1	H'F700	6	0
	CCLR0	TCNT_0 is cleared with GRA_0 enabled	H'F700	5	1
	CKEG1	Clock Edge of channel 1 is cleared If TCSWRE is set to 1, writing to WDON and WRST is enabled.	H'F700	4	0
	CKEG0	Counts at the rising edge of the clock CKEG1 = 0, CKEG0 = 0	H'F700	3	0
	TPSC2	Timer Prescaler is cleared to 0	H'F700	2	0
	TPSC1	TPSC2 = 0, TPSC1 = 1, TPSC0 = 0 Counts by selected input clock	H'F700	1	1
	TPSC0	Timer Prescaler is set to 1	H'F700	0	1
TIORA0	IOA2	Timer I/O control register A0 I/O control A2 to A0	H'F701	2	1
	IOA1	IOA2 = 1, IOA1 = 1, IOA0 = X System clock/8192 is selected as TCWD input clock	H'F701	1	1
	IOA0	GRA0 register is used as input capture TCNT_0 value is transferred to GRA_0 X: don't care	H'F701	0	0
TIORC0	IOC2	Timer I/O control register C0 I/O control C2 to C0	H'F702	2	1
	IOC1	IOC2 = 1, IOC1 = 1, IOC0 = X	H'F702	1	1
	IOC0	GRC_0 is used as input capture GRA_0 value is transferred to GRC_0	H'F702	0	0

Register	Name	Function	Address	Bit	Setting
TSR0	OVF	Timer status register 0 When OVF is cleared, TCNT_0 is not overflown When OVF is set to 1, TCNT_0 has overflown	H'F703	4	0
	IMFA	Input capture/output compare flag A When IMFA = 0, TCNT_0 value has not transferred to GRA_0 When IMFA =1, TCNT_0 value has transferred to GRA_0	H'F703	0	1
TIER0	OVIE	Timer interrupt enable register 0 OVIE = 0, TSR_0 interrupt requests OVF &UDF are disabled OVIE = 1, TSR_0 interrupt requests OVF &UDF are enabled	H'F704	4	1
	IMIEA	IMEA = 0, TSR_0 interrupt request IMFA is disabled IMEA = 1, TSR_0 interrupt request IMFA is enabled	H'F704	0	1
TCNT_0	TCNT_0	Timer counter channel 0 16-bit counter counting up at input clock's rate	H'F706	-	0000
GRA_0	GRA_0	General register A0 TCNT_0 value is transferred to this register	H'F708	-	N/A
GRC_0	GRC_0	General register C0 GRA_0 value is transferred to this register	H'F70C	-	N/A
TSTR	STR_0	Timer start register channel 0 When STR_0 is cleared, TCNT_0 stops When STR_0 is set to 1, TCNT_0 starts	H'F720	0	0
TMDR	BFC_0	Timer mode register (Buffer operation C) When BFC_0 is cleared, GRC_0 is set for normal operation When BFC_0 is set to 1, GRA_0 and GRC_0 are set for normal operation	H'F721	4	1
	SYNC	Timer Synchronization When SYNC = 0, TCNT_0 and TCNT_1 are independent When SYNC = 1, TCNT_0 and TCNT_1 are in synch	H'F721	0	0

Register	Name	Function	Address	Bit	Setting
TFCR	CMD_1	Timer function control register channel 1 Combination mode channel 1 is cleared	H'F723	1	0
	CMD_0	Timer function control register channel 0 CMD_1 = 0, CMD_0 = 0 channels 0 and 1 are set for normal operation	H'F723	0	0

Table 4.2 - Internal Registers

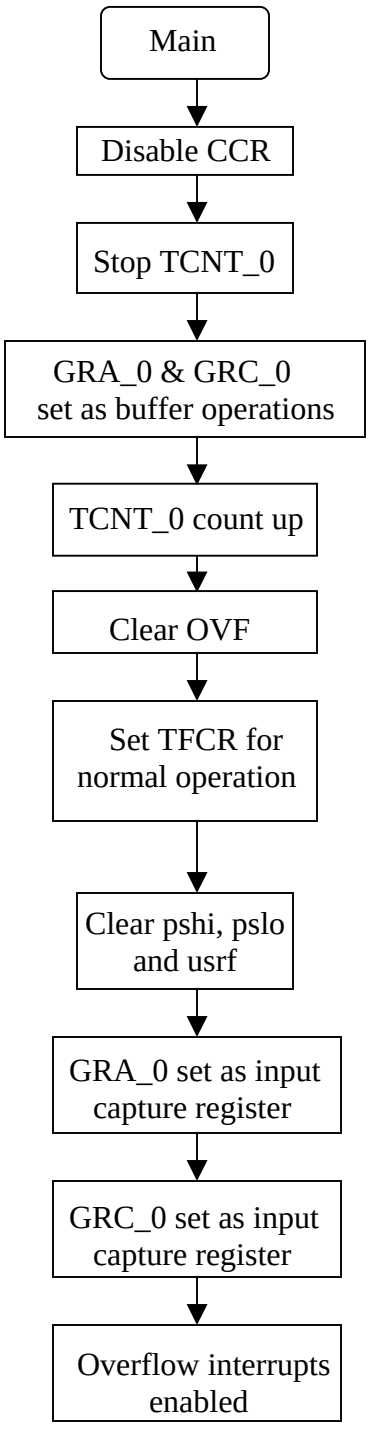
4.4 Description of RAM

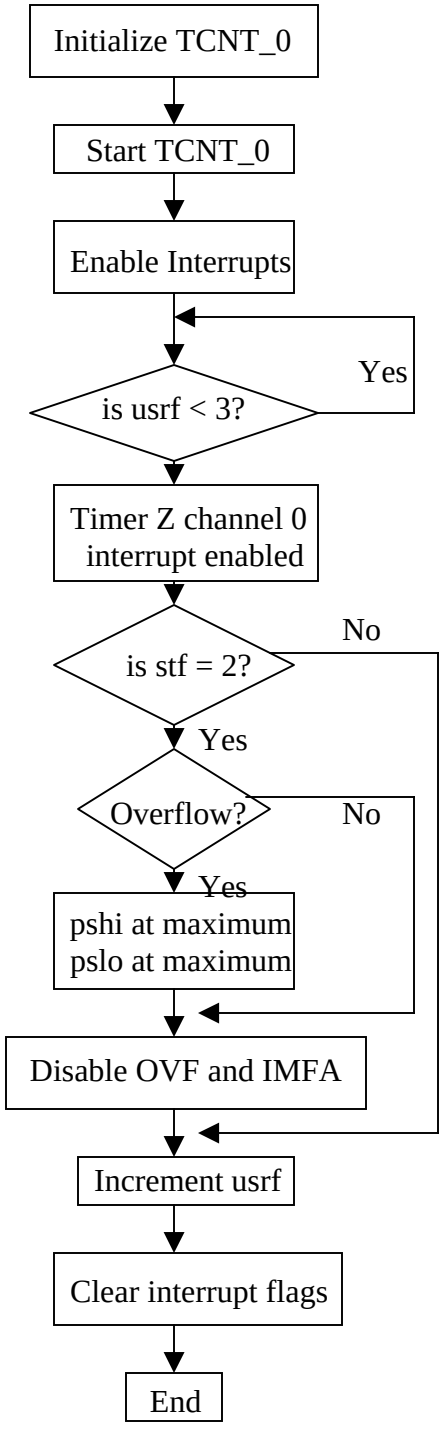
Table 4.3 describes the functionality of RAM in this AN.

Name	Function	Size	Used in
pshi	Measured high level PW	2 bytes	Main
pslo	Measured low level PW	2 bytes	Main
usrf	Timer Z interrupt status indicator	1 byte	Main

Table 4.3 - Functionality of RAM

5. Flowcharts





6. Program Listing

```
/************************************************************************************************************************/
/* */
/* H8/36077 Group MCU-                                                  */
/* Application Note                                                     */
/*                                                                      */
/* Buffer Operation of Timer Z '                                        */
/*                                                                      */
/* Function                                                             */
/* Timer Z Input Capture                                                */
/*                                                                      */
/* External Clock : 16MHz                                               */
/* Internal Clock : 16MHz                                               */
/* Sub Clock : 32.768kHz                                                */
/*                                                                      */
#include    <machine.h>

/* Symbol Definition                                                    */
/************************************************************************************************************************/
struct BIT {
unsigned char   b7:1;                                   /* bit7 */
unsigned char   b6:1;                                   /* bit6 */
unsigned char   b5:1;                                   /* bit5 */
unsigned char   b4:1;                                   /* bit4 */
unsigned char   b3:1;                                   /* bit3 */
unsigned char   b2:1;                                   /* bit2 */
unsigned char   b1:1;                                   /* bit1 */
unsigned char   b0:1;                                   /* bit0 */
};
#define TCR_0 *(volatile unsigned char *)H'F700         /* Timer control register channel 0 */
#define TIORA0 *(volatile unsigned char *)H'F701        /* Timer I/O Control Register A channel 0 */
#define TIORC0 *(volatile unsigned char *)H'F702        /* Timer I/O Control Register C channel 0 */
#define TSR0 *(volatile unsigned char *)H'F703  /* Timer status register channel 0 */
#define TSR0_BIT (*(struct BIT *)H'F703)                /* Timer status register channel 0 */
#define OVF TSR0_BIT.b4                                 /* Overflow flag */


#define IMFA TSR0_BIT.b0                                /* Input Capture/output compare Flag A */
#define TIER0 *(volatile unsigned char *)H'F704         /* Timer interrupt enable register channel 0 */
#define TIER0_BIT (*(struct BIT *)H'F704)               /* Timer interrupt enable register channel 0 */
#define IMIEA TIER0_BIT.b0                      /* Input Capture/Output compare */
```

```
                                                    /* Interrupt Enable A */
#define TCNT_0 *(volatile unsigned short *)H'F706   /* Timer counter channel 0 */
#define GRA_0 *(volatile unsigned short *)H'F708    /* General register A channel 0 */
#define GRC_0 *(volatile unsigned short *)H'F70C    /* General register C channel 0 */
#define TSTR *(volatile unsigned char *)H'F720  /* Timer start register */
#define TMDR *(volatile unsigned char *)H'F721      /* Timer mode register */
#define TPMR *(volatile unsigned char *)H'F722      /* Timer PWM mode register */
#define TFCR *(volatile unsigned char *)H'F723      /* Timer function control register */
#define TOER *(volatile unsigned char *)H'F724  /* Timer output master enable register */
#define TOCR *(volatile unsigned char *)H'F725      /* Timer output control register */
#pragma interrupt   (tz0int)                    /* Timer Z channel 0 interrupt */

extern void INIT (void);                        /* Stack Pointer Set */
void main (void);
void tz0int (void);
                                                /* RAM define */
volatile unsigned short pshi;                   /*High Pulse time data */
volatile unsigned short pslo;                   /*Low Pulse time data */
volatile unsigned char  usrf;                   /* User flag */
                                                /* Vector Address */
#pragma section     V1                          /* VECTOR SECTOIN SET */
void (*const VEC_TBL1[])(void) = {              /* 0x00 - 0x0f address register */
INIT                                            /* 00 Reset */
};
#pragma section     V2                          /* Vector section set */
void (*const VEC_TBL2[])(void) = {
tz0int                                          /* 34 Timer Z channel 0 Interrupt */
};
#pragma section                                 /* Delay */
/* Main Program */
void main ( void )
{
unsigned char tmp;
set_imask_ccr(1);                               /* Interrupt Disabled */


TSTR = 0xFC;                                    /* TCNT channel 0 count stop */
TMDR = 0x1E;                                    /* TCNT channel 0 & channel 1 Single Mode */
                                                /* GRC0 is used as buffer of GRA channel 0 */
TCR_0 = 0x23;                                   /* Rising edge, phi/2 Clock count */
tmp = TSR_0;
TSR_0 = 0xE0;                                   /* Interrupt Flag Clear */
TFCR = 0x80;                                    /* Channel 0 & 1 operate normally */
pshi = 0;                                   /* Clear RAM */
```

```
pslo = 0;                                   /* Clear RAM */
usrf = 0;                                   /* Clear Flag */
TIORA_0 = 0x8E;                                     /* Input capture to GRA channel 0 at both */
TIORC_0 = 0x8E;                                     /* Input capture to GRC channel 0 */
TIER_0 = 0xF1;                                      /* IMFA Interrupt Enable */
TCNT_0 = H'0000;                                    /* Clear TCNT channel 0 */
TSTR = 0xFD;                                        /* TCNT channel 0 count start  */
set_imask_ccr(0);                                   /* Interrupt Enable */
while(usrf < 3);
while(1);
}
/* Timer Z channel 0 Interrupt */
void tz0int ( void )
{
unsigned char tmp;
if(usrf == 2){
if(OVF == 1){
pshi = H'FFFF;                                      /* Overflow set */
pslo = H'FFFF;
}
else{
plhigh = GRC_0;                                     /* copy RAM to GRC channel 0 */
pllow = GRA_0;                                      /* copy RAM to GRA channel 0 */
}
TIER_0 = 0xE0;                                      /* OVF, IMFA Interrupt Disable */
}
usrf++;                                     /* increment User flag */
tmp = TSR_0;
TSR_0 = 0xE0;                                       /* Clear Interrupt Flag */
}
```

Important Notes

1. This document is provided for reference purposes only. Brazen Tek neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights.
2. Brazen Tek shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. This application note or the technology described in this document is not for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use.
4. Brazen Tek assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
5. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Brazen Tek makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Brazen Tek.