Section 37. Charge Time Measurement Unit (CTMU)

HIGHLIGHTS

This section of the manual contains the following major topics:

> **Note:** This family reference manual section is meant to serve as a complement to device data sheets. Depending on the device variant, this manual section may not apply to all PIC32 devices.
>
> Please consult the note at the beginning of the **"Charge Time Measurement Unit (CTMU)"** chapter in the current device data sheet to check whether this document supports the device you are using.
>
> Device data sheets and family reference manual sections are available for download from the Microchip Worldwide Web site at: http://www.microchip.com

37.1 INTRODUCTION

The Charge Time Measurement Unit (CTMU) is a flexible analog module that has a configurable current source with a digital configuration circuit built around it. The CTMU can be used for differential time measurement between pulse sources and can be used for generating an asynchronous pulse. By working with other on-chip analog modules, the CTMU can be used for high resolution time measurement, measure capacitance, measure relative changes in capacitance or generate output pulses with a specific time delay. The CTMU is ideal for interfacing with capacitive-based sensors.

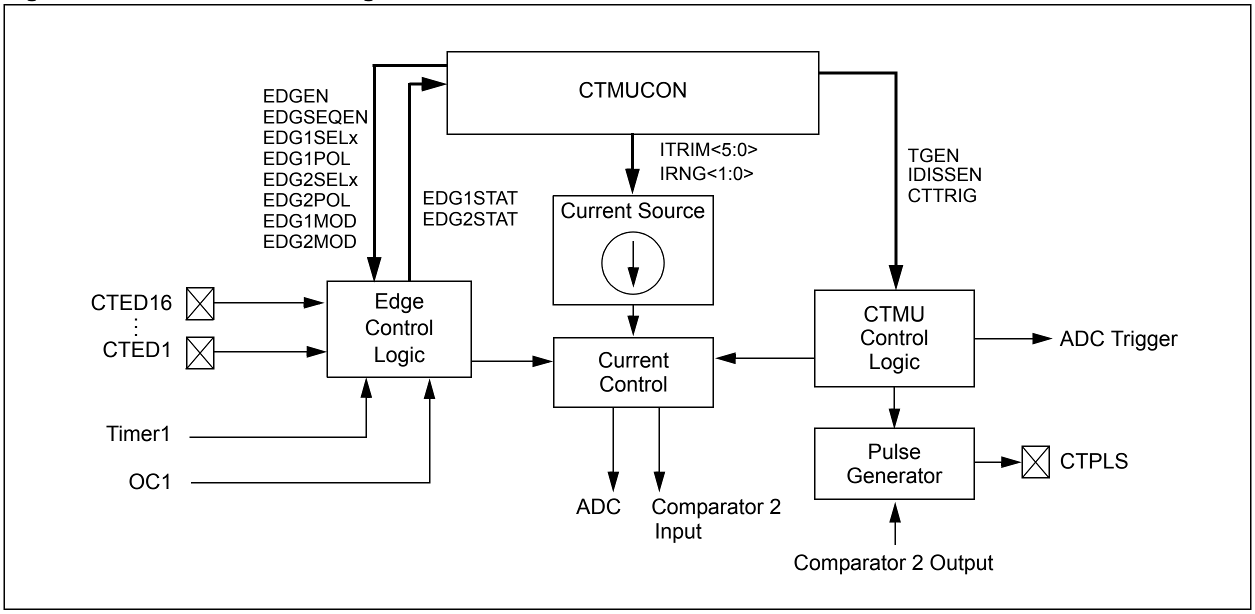
The module includes the following key features:

- Up to 32 channels available for capacitive or time measurement input
- On-chip precision current source
- 16-edge input trigger sources
- Selection of edge or level-sensitive inputs
- Polarity control for each edge source
- Control of edge sequence
- Control of response to edges
- High precision time measurement
- Time delay of external or internal signal asynchronous to system clock
- Integrated temperature sensing diode
- Control of current source during auto-sampling
- Four current source ranges
- Time measurement resolution of one nanosecond

The CTMU works in conjunction with the Analog-to-Digital Converter (ADC) to provide up to 32 channels for time or charge measurement, depending on the specific device and the number of ADC channels available. When configured for time delay, the CTMU is connected to one of the analog comparators. The level-sensitive input edge sources can be selected from four sources: two external inputs, Timer1 or Output Compare Module 1. For information on available input sources, refer to the specific device data sheet.

A block diagram of the CTMU is shown in Figure 37-1.

Figure 37-1: CTMU Block Diagram



37.2 REGISTERS

The CTMUCON register contains control bits for configuring the CTMU module edge source selection, edge source polarity selection, edge sequencing, ADC trigger, analog circuit capacitor discharge and enables. In addition, this register has bits for selecting the current source range and current source trim.

Table 37-1 summarizes the CTMU-related register. A detailed description of the register follows the summary.

Table 37-1: CTMU SFR Summary(1)

Name	Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
CTMUCON	31:24	EDG1MOD	EDG1POL	EDG1SEL<3:0>				EDG2STAT	EDG1STAT
	23:16	EDG2MOD	EDG2POL	EDG2SEL<3:0>				—	—
	15:8	ON	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG
	7:0	ITRIM<5:0>						IRNG<1:0>	

Legend: — = unimplemented, read as '0'. Address offset values are shown in hexadecimal.
Note 1: Not all registers have associated SET, CLR, and INV registers. Refer to the specific device data sheet for details.

Register 37-1: CTMUCON: CTMU Control Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	EDG1MOD	EDG1POL	EDG1SEL<3:0>(1)				EDG2STAT	EDG1STAT
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
	EDG2MOD	EDG2POL	EDG2SEL<3:0>(1)				—	—
15:8	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ON	—	CTMUSIDL	TGEN	EDGEN	EDGSEQEN	IDISSEN	CTTRIG
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	ITRIM<5:0>						IRNG<1:0>	

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31 **EDG1MOD:** Edge1 Edge Sampling Select bit
1 = Input is edge-sensitive
0 = Input is level-sensitive

bit 30 **EDG1POL:** Edge 1 Polarity Select bit
1 = Edge1 programmed for a positive edge response
0 = Edge1 programmed for a negative edge response

bit 29-26 **EDG1SEL<3:0>:** Edge 1 Source Select bits(1)
1111 = CTED16 selected
•
•
•
0000 = CTED1 selected

bit 25 **EDG2STAT:** Edge2 Status bit
Indicates the status of Edge2 and can be written to control edge source
1 = Edge2 has occurred
0 = Edge2 has not occurred

bit 24 **EDG1STAT:** Edge1 Status bit
Indicates the status of Edge1 and can be written to control edge source
1 = Edge1 has occurred
0 = Edge1 has not occurred

bit 23 **EDG2MOD:** Edge2 Edge Sampling Select bit
1 = Input is edge-sensitive
0 = Input is level-sensitive

bit 22 **EDG2POL:** Edge 2 Polarity Select bit
1 = Edge2 programmed for a positive edge response
0 = Edge2 programmed for a negative edge response

bit 21-18 **EDG2SEL<3:0>:** Edge 2 Source Select bits(1)
1111 = CTED16 selected
•
•
•
0000 = CTED1 selected

bit 17-16 **Unimplemented:** Read as '0'

bit 15 **ON:** ON Enable bit
1 = Module is enabled
0 = Module is disabled

Note 1: Refer to the specific device data sheet for the list of available trigger sources.

Register 37-1: CTMUCON: CTMU Control Register (Continued)

bit 14 **Unimplemented:** Read as '0'

bit 13 **CTMUSIDL:** Stop in Idle Mode bit
1 = Discontinue module operation when device enters Idle mode
0 = Continue module operation in Idle mode

bit 12 **TGEN:** Time Generation Enable bit
1 = Enables edge delay generation
0 = Disables edge delay generation

bit 11 **EDGEN:** Edge Enable bit
1 = Edges are not blocked
0 = Edges are blocked

bit 10 **EDGSEQEN:** Edge Sequence Enable bit
1 = Edge1 must occur before Edge2 can occur
0 = No edge sequence is needed

bit 9 **IDISSEN:** Analog Current Source Control bit
1 = Analog current source output is grounded
0 = Analog current source output is not grounded

bit 8 **CTTRIG:** Trigger Control bit
1 = Trigger output is enabled
0 = Trigger output is disabled

bit 7-2 **ITRIM<5:0>:** Current Source Trim bits
111111 = Minimum negative change from nominal current
•
•
•
100010
100001 = Maximum negative change from nominal current
011111 = Maximum positive change from nominal current
011110
•
•
•
000001 = Minimum positive change from nominal current
000000 = Nominal current output specified by IRNG<1:0>

bit 1-0 **IRNG<1:0>:** Current Range Select bits
11 = 100 times base current
10 = 10 times base current
01 = Base current level (0.55 µA nominal)
00 = 1000 times base current

Note 1: Refer to the specific device data sheet for the list of available trigger sources.

37.3 CTMU OPERATION

The CTMU works by using a fixed current source to charge a circuit. The type of circuit depends on the type of measurement being made. In the case of charge measurement, the current is fixed and the amount of time the current is applied to the circuit is fixed. The amount of voltage read by the ADC is then a measurement of the capacitance of the circuit. In the case of time measurement, the current, as well as the capacitance of the circuit, is fixed. In this case, the voltage read by the ADC is then representative of the amount of time elapsed from the time the current source starts and stops charging the circuit.

If the CTMU is being used as a time delay, both capacitance and current source are fixed, as well as the voltage supplied to the comparator circuit. The delay of a signal is determined by the amount of time taken for the voltage to charge to the comparator threshold voltage.

37.3.1 Theory of Operation

The operation of the CTMU is based on the equation for charge, as shown in Equation 37-1.

Equation 37-1:

$$I = C \cdot \frac{dV}{dt}$$

More simply, the amount of charge measured in coulombs in a circuit is defined as current in amperes (I) multiplied by the amount of time in seconds that the current flows (t). Charge is also defined as the capacitance in farads (C) multiplied by the voltage of the circuit (V), as shown in Equation 37-2.

Equation 37-2:

$$I \cdot t = C \cdot V$$

The CTMU module provides a constant, known current source. The ADC is used to measure (V) in the equation, leaving two unknowns: capacitance (C) and time (t). Equation 37-2 can be used to calculate capacitance or time, by either the relationship shown in Equation 37-3 and using the known fixed capacitance of the circuit, or by Equation 37-4 using a fixed time that the current source is applied to the circuit.

Equation 37-3:

$$t = \frac{(C \cdot V)}{I}$$

Equation 37-4:

$$C = \frac{(I \cdot t)}{V}$$

37.3.2 Current Source

At the heart of the CTMU is a precision current source, designed to provide a constant reference for measurements. The level of current is user-selectable across four ranges, or a total of two orders of magnitude, with the ability to trim the output in ±2% increments (nominal). The current range is selected by the IRNG<1:0> bits (CTMUCON<1:0>) with a value of '01' representing the lowest range.

Current trim is provided by the ITRIM<5:0> bits (CTMUCON<7:2>). These six bits allow trimming of the current source in steps of approximately 2% per step. Note that half of the range adjusts the current source positively and the other half reduces the current source. A value of '000000' is the neutral position (no change). A value of '100000' is the maximum negative adjustment (approximately -62%) and '011111' is the maximum positive adjustment (approximately +62%).

37.3.3 Edge Selection and Control

CTMU measurements are controlled by edge events occurring on the module's two input channels. Each channel, referred to as Edge 1 and Edge 2, can be configured to receive input pulses from one of the sixteen edge input pins. The inputs are selected using the EDG1SEL<3:0> and EDG2SEL<3:0> bit pairs (CTMUCON<29:26> and CTMUCON<21:18>).

In addition to source, each channel can be configured for event polarity using the EDG1POL and EDG2POL bits (CTMUCON<30>and CTMUCON<22>). The input channels can also be filtered for an edge event sequence (Edge 1 occurring before Edge 2) by setting the EDGSEQEN bit (CTMUCON<10>).

37.3.4 Edge Status

The CTMUCON register also contains two status bits: EDG1STAT (CTMUCON<24>) and EDG2STAT (CTMUCON<25>). Their primary function is to show if an edge response has occurred on the corresponding channel. The CTMU automatically sets a particular bit when an edge response is detected on its channel. The level-sensitive, or edge-sensitive nature of the input channels also means that the status bits are set immediately if the channel's configuration is changed and is the same as the channel's current state.

The CTMU module uses the edge status bits to control the current source output to external analog modules (such as the ADC). Current is only supplied to external modules when only one (but not both) of the status bits is set and shuts current off when both bits are either set or cleared. This allows the CTMU to measure current only during the interval between edges. After both status bits are set, it is necessary to clear them before another measurement is taken. Both bits should be cleared simultaneously, if possible, to avoid re-enabling the CTMU current source.

In addition to being set by the CTMU hardware, the edge status bits can also be set by software. This allows the user's application to manually enable or disable the current source. Setting either one (but not both) of the bits enables the current source. Setting or clearing both bits at once disables the source.

37.3.5 Interrupts

The CTMU sets its interrupt flag (CTMUIF) whenever the current source is enabled and then disabled. If edge sequencing is not enabled (i.e., Edge 1 must occur before Edge 2), it is necessary to monitor the edge status bits and determine which edge occurred last and caused the interrupt.

37.4 CTMU MODULE INITIALIZATION

The following sequence is a general guideline used to initialize the CTMU module:

1. Select the current source range using the IRNG<1:0> bits (CTMUCON<1:0>).
2. Adjust the current source trim using the ITRIM<5:0> bits (CTMUCON<7:2>).
3. Configure the edge input sources for Edge 1 and Edge 2 by setting the EDG1SEL and EDG2SEL bits (CTMUCON<29:26> and CTMUCON<21:18>).
4. Configure the input polarities for the edge inputs using the EDG1POL bit (CTMUCON<30>) and EDG2POL bit (CTMUCON<22>). The default configuration is for negative edge polarity (high-to-low transitions).
5. Enable edge sequencing using the EDGSEQEN bit (CTMUCON<10>). By default, edge sequencing is disabled.
6. Select the operating mode (Measurement or Time Delay) with the TGEN bit (CTMUCON<12>). By default, the time delay mode is disabled.
7. Configure the module to automatically trigger an analog-to-digital conversion when the second edge event has occurred using the CTTRIG bit (CTMUCON<8>). The conversion trigger is disabled by default.
8. Discharge the connected circuit by setting the IDISSEN bit (CTMUCON<9>). After waiting a sufficient time for the circuit to discharge, clear the IDISSEN bit.
9. Disable the module by clearing the ON bit (CTMUCON<15>).
10. Clear the Edge Status bits, EDG2STAT<3:0> and EDG1STAT<3:0> (CTMUCON<29:26> and CTMUCON<21:18>).
11. Enable both edge inputs by setting the EDGEN bit (CTMUCON<11>).
12. Enable the module by setting the ON bit (CTMUCON<15>).

Depending on the type of measurement, or pulse generation being performed, one or more additional modules may also need to be initialized and configured with the CTMU module:

- Edge Source Generation: In addition to the external edge input pins, other modules, such as ICx, OCx, and Timer1 can be used as edge sources for the CTMU. Refer to the specific device data sheet for available sources.
- Capacitance or Time Measurement: The CTMU module uses the ADC to measure the voltage across a capacitor that is connected to one of the analog input channels.
- Pulse Generation: When generating system clock independent output pulses, the CTMU module uses Comparator 2 and the associated comparator voltage reference.

For specific information on initializing these modules, refer to the applicable section in the *"PIC32 Family Reference Manual"* for the appropriate module.

37.5 CALIBRATING THE CTMU MODULE

The CTMU requires calibration for precise measurements of capacitance and time, as well as for accurate time delay. If the application only requires measurement of a relative change in capacitance or time, calibration is usually not necessary. An example of this type of application would include a capacitive touch switch, in which the touch circuit has a baseline capacitance and the added capacitance of the human body changes the overall capacitance of a circuit.

If actual capacitance or time measurement is required, two hardware calibrations must take place: the current source needs calibration to set it to a precise current, and the circuit being measured needs calibration to measure and/or nullify all other capacitance other than that to be measured.

37.5.1 Current Source Calibration

The current source on-board the CTMU module has a range of ±62% nominal for each of four current ranges. Therefore, for precise measurements, it is possible to measure and adjust this current source by placing a high precision resistor, RCAL, onto an unused analog channel. An example circuit is shown in Figure 37-2. The current source measurement is performed using the following steps:

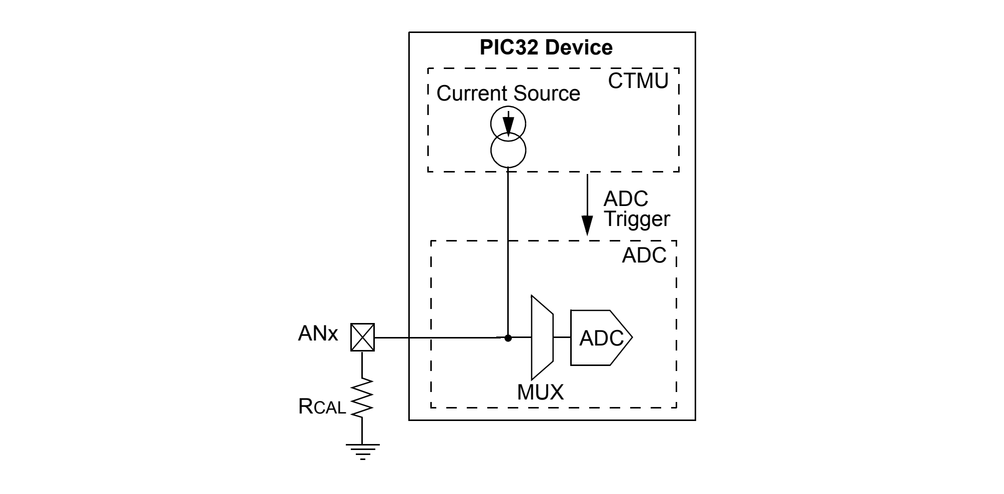
1. Initialize the ADC.
2. Initialize the CTMU by configuring the module for Pulse Generation (TGEN = `1`) mode.
3. Enable the current source by setting the EDG1STAT bit (CTMUCON<24>).
4. Issue settling time delay.
5. Perform analog-to-digital conversion.
6. Calculate the current source current using $I = V/R_{CAL}$, where R_{CAL} is a high precision resistance and V is measured by performing an analog-to-digital conversion.

The CTMU current source may be trimmed with the trim bits in the CTMUCON register using an iterative process to get an exact desired current. Alternatively, the nominal value without adjustment may be used. It may be stored by the software for use in all subsequent capacitive or time measurements.

Figure 37-2 shows the external connections for current source calibration, as well as the relationship of the different analog modules required.

To calculate the value for R_{CAL}, the nominal current must be chosen and then the resistance can be calculated. For example, if the ADC reference voltage is 3.3V, use 70% of full scale, or 2.31V as the desired approximate voltage to be read by the ADC. If the range of the CTMU current source is selected to be 0.55 µA, the resistor value needed is calculated as $R_{CAL} = 2.31\text{V}/0.55\ \mu\text{A}$, for a value of 4.2 MΩ. Similarly, if the current source is chosen to be 5.5 µA, R_{CAL} would be 420,000Ω and 42,000Ω if the current source is set to 55 µA.

Figure 37-2: CTMU Current Source Calibration Circuit



 Preliminary

A value of 70% of full-scale voltage is chosen to make sure that the ADC is in a range that is well above the noise floor. Keep in mind that if an exact current is chosen that is to incorporate the trimming bits from the CTMUCON register, the resistor value of R_{CAL} may need to be adjusted accordingly. R_{CAL} may also be adjusted to allow for available resistor values. R_{CAL} should be of the highest precision available, keeping in mind the amount of precision needed for the circuit that the CTMU will be used to measure. A recommended minimum would be 0.1% tolerance.

The following examples show one typical method for performing a CTMU current calibration. Example 37-1 shows how to initialize the ADC and the CTMU; this routine is typical for applications using both modules. Example 37-2 shows one method for the actual calibration routine. Note that this method manually triggers the ADC; this is done to demonstrate the entire stepwise process. It is also possible to automatically trigger the conversion by setting the CTTRIG bit (CTMUCON<8>).

Example 37-1: Setup for CTMU Calibration Routines

```
#include “plib.h”

/**************************************************************************/
/ Set up CTMU ************************************************************/
/**************************************************************************/

void setup(void)
{   //CTMUCON - CTMU Control register
    //make sure CTMU is disabled
    // CTMU continues to run when emulator is stopped, CTMU continues
    // to run in idle mode, Time Generation mode disabled, Edges are
    // blocked. No edge sequence order, Analog current source not
    // grounded, trigger output disabled, Edge2 polarity = positive level,
    // Edge2 source = source 0, Edge1 polarity = positive level,
    // Edge1 source = source 0, Set Edge status bits to zero
    000;
    CTMUCON = 0x0000;      // 0.55 uA, Nominal - No Adjustment
/**************************************************************************/
/ Set up Analog-to-Digital Converter **************************************/
/**************************************************************************/
    TRISB = 0x0001;        // Set channel 2 as an input
    AD1PCFGbits.AN2 = 1;   // Make AN2 as analog
    AD1CHSbits.CHOSA = 2;  // Select the analog channel(2)
    AD1RES1 = 0x0000;      // Skip the analog channels for input scan
    AD1CON1 = 0x8000;      // Turn On the ADC, continue in Idle mode,
                           // Unsigned fractional format, Clear SAMP bit to
                           // start conversion, Sample when SAMP bit is set,
                           // sampling on hold
    AD1CON2 = 0x0000;      // VR+ = AVDD, V- = AVSS, Don't scan,
                           // interrupts at end of conversion for each sample
    AD1CON3 = 0x0000;      // ADC uses system clock, conversion clock = 1xTcy
}
```

Example 37-2: Current Calibration Routine

```
#include "plib.h"

#define     COUNT   500         //@ 8 MHz = 125 uS.
#define     DELAY   DelayMs(2)
#define     RCAL    .027        //R value is 4200000 (4.2M)
                                //scaled so that result is in
                                //1/100th of uA
#define     ADSCALE 1023        //for unsigned conversion 10 sig bits
#define     ADREF   3.3         //VDD connected to ADC Vr+

int main(void)
{
    int i;
    int j = 0;                           //index for loop
    unsigned int Vread = 0;
    double VTot = 0;
    float Vavg=0, Vcal=0, CTMUISrc = 0;  //float values stored for calcs

    //assume CTMU and ADC have been set up correctly
    //see Example 37-1 for CTMU & ADC setup
    setup();

    CTMUCONbits.ON = 1;                  //Enable the CTMU

    for(j=0;j<10;j++)
    {
        AD1CON1bits.SAMP = 1;            //Manual sampling start
        CTMUCONbits.IDISSEN = 1;         //drain charge on the circuit
        DELAY;                           //wait 125 us
        CTMUCONbits.IDISSEN = 0;         //end drain of circuit

        CTMUCONbits.EDG1STAT = 1;        //Begin charging the circuit
                                         //using CTMU current source
        DELAY;                           //wait for 125 us

        IFS0bits.AD1IF = 0;              //make sure ADC Int not set
        AD1CON1bits.SAMP = 0;            //and begin analog-to-digital conv.
        while(!IFS0bits.AD1IF);          //Wait for ADC convert complete
        AD1CON1bits.DONE = 0;
        CTMUCONbits.EDG1STAT = 0;        //Stop charging circuit
        Vread = ADC1BUF0;                //Get the value from the ADC
        IFS0bits.AD1IF = 0;              //Clear ADC Interrupt Flag
        VTot += Vread;                   //Add the reading to the total

    }

        Vavg = (float)(VTot/10.000);     //Average of 10 readings
        Vcal = (float)(Vavg/ADSCALE*ADREF);
        CTMUISrc = Vcal/RCAL;            //CTMUISrc is in 1/100ths of uA
}
```


37.5.2 Capacitance Calibration

There is a small amount of capacitance from the internal ADC sample capacitor as well as stray capacitance from the circuit board traces and pads that affect the precision of capacitance measurements. A measurement of the stray capacitance can be taken by making sure the desired capacitance to be measured has been removed. The measurement is then performed using the following steps:

1. Initialize the ADC and the CTMU.
2. Set the EDG1STAT bit (= `1`).
3. Wait for a fixed delay of time, t.
4. Clear the EDG1STAT bit.
5. Perform an analog-to-digital conversion.
6. Calculate the stray and analog-to-digital sample capacitances using Equation 37-5.

Equation 37-5:

$$C_{OFFSET} = C_{STRAY} + C_{AD} = \frac{(I \cdot t)}{V}$$

Where :

I is known from the current source measurement step
t is a fixed delay
V is measured by performing an analog-to-digital conversion.

This measured value is then stored and used for calculations of time measurement or subtracted for capacitance measurement. For calibration, it is expected that the capacitance of $C_{STRAY} + C_{AD}$ is approximately known. C_{AD} is approximately 4 pF.

An iterative process may need to be used to adjust the time, t, that the circuit is charged to obtain a reasonable voltage reading from the ADC. The value of t may be determined by setting C_{OFFSET} to a theoretical value, and then solving for t. For example, if C_{STRAY} is theoretically calculated to be 11 pF, and V is expected to be 70% of V_{DD} or 2.31V, t would be equal to Equation 37-6 or 63 µs.

Equation 37-6:

$$t = (4\ pF + 11\ pF) \bullet \frac{2.31 V}{0.55\ \mu A}$$

A typical routine for CTMU capacitance calibration is shown in Example 37-3.

Example 37-3: Capacitance Calibration Routine

```
#include "plib.h"

#define COUNT 25            //@ 8 MHz INTFRC = 62.5 us.
#define ETIME COUNT*2.5     //time in uS
#define DELAY  DelayMs(2)
#define ADSCALE 1023        //for unsigned conversion 10 sig bits
#define ADREF 3.3           //VDD connected to ADC Vr+


int main(void)
{
    int i;
    int j = 0;              //index for loop
    unsigned int Vread = 0;
    float CTMUISrc, CTMUCap, Vavg, VTot, Vcal;

                            //assume CTMU and ADC have been set up correctly
                            //see Example 37-1 for CTMU & ADC setup
    setup();

    CTMUCONbits.CTMUEN = 1;//Enable the CTMU

    for(j=0;j<10;j++)
    {
        AD1CON1bits.SAMP = 1;       //Manual sampling start
        CTMUCONbits.IDISSEN= 1;     //drain any charge on the circuit
        DELAY;                      //wait 62.5 us
        CTMUCONbits.IDISSEN = 0;    //end drain of circuit
        CTMUCONbits.EDG1STAT = 1;   //Begin charging the circuit
                                    //using the CTMU current source
        DELAY;                      //wait for 62.5 us for circuit
                                    //to charge
        CTMUCONbits.EDG1STAT = 0;   //Stop charging circuit and begin
                                    //Analog-to-digital conversion
        AD1CON1bits.SAMP = 0;
        while(!IFS0bits.AD1IF);     //Wait for conversion to complete
        Vread = ADC1BUF0;           //Get the value from the ADC
        IFS0bits.AD1IF = 0;         //Clear AD1IF
        VTot += Vread;              //Add the reading to the total
    }

    Vavg = (VTot/10); //Average of 10 readings
    Vcal = (Vavg/ADSCALE*ADREF);
    CTMUCap = (CTMUISrc*ETIME/Vcal)/100;
    //CTMUISrc is in 1/100ths of uA,
    //calculated in Example 37-2
    //time is in us
    //CTMUCap is in pF
}
```


37.6 MEASURING CAPACITANCE WITH THE CTMU

There are two separate methods of measuring capacitance with the CTMU. The first is the absolute method, in which the actual capacitance value is desired. The second is the relative method, in which the actual capacitance is not needed, rather an indication of a change in capacitance is required.

37.6.1 Absolute Capacitance Measurement

Note:	The ADC must be configured correctly for proper CTMU functionality. If necessary, make sure that the ADC is pointing to an unused pin.

For absolute capacitance measurements, both the current and capacitance calibration steps found in **37.5 "Calibrating the CTMU Module"** should be followed. Capacitance measurements are then performed using the following steps:

1. Initialize the ADC.
2. Initialize the CTMU.
3. Set the EDG1STAT bit.
4. Wait for a fixed delay, T.
5. Clear the EDG1STAT bit.
6. Perform an analog-to-digital conversion.
7. Calculate the total capacitance, $C_{TOTAL} = (I * T)/V$, where I is known from the current source measurement step (**37.5.1 "Current Source Calibration"**), T is a fixed delay and V is measured by performing an analog-to-digital conversion.
8. Subtract the stray and analog-to-digital capacitance (C_{OFFSET} from **37.5.2 "Capacitance Calibration"**) from C_{TOTAL} to determine the measured capacitance.

37.6.2 Relative Charge Measurement

Note:	The ADC must be configured correctly for proper CTMU functionality. If necessary, make sure that the ADC is pointing to an unused pin.

An application may not require precise capacitance measurements. For example, when detecting a valid press of a capacitance-based switch, detecting a relative change of capacitance is of interest. In this type of application, when the switch is open (or not touched), the total capacitance is the capacitance of the combination of the board traces, the ADC, etc. A larger voltage will be measured by the ADC. When the switch is closed (or is touched), the total capacitance is larger due to the addition of the capacitance of the human body to the above listed capacitances and a smaller voltage will be measured by the ADC.

Detecting capacitance changes is easily accomplished with the CTMU using these steps:

1. Initialize the ADC and the CTMU.
2. Set the EDG1STAT bit.
3. Wait for a fixed delay.
4. Clear the EDG1STAT bit.
5. Perform an analog-to-digital conversion.

The voltage measured by performing the analog-to-digital conversion is an indication of the relative capacitance. Note that in this case, no calibration of the current source or circuit capacitance measurement is needed. A sample software routine for a capacitive touch switch is shown in Example 37-4.

Example 37-4: Routine for Capacitive Touch Switch

```
#include "plib.h"

#define     COUNT   500     //@ 8 MHz = 125 uS.
#define     DELAY   DelayMs(2)
#define     OPENSW  1000    //Unpressed switch value
#define     TRIP    300     //Difference between pressed
                            //and unpressed switch
#define     HYST    65      //amount to change
                            //from pressed to unpressed
#define     PRESSED 1
#define     UNPRESSED0

int main(void)
{
    unsigned int Vread;         //storage for reading
    unsigned int switchState;
    int i;

                                //assume CTMU & ADC have been set up correctly
                                //see Example 37-1 for CTMU & ADC setup
    setup();

    CTMUCONbits.ON = 1;         //Enable the CTMU

    AD1CON1bits.SAMP = 1;       //Manual sampling start
    CTMUCONbits.IDISSEN = 1;    //drain charge on the circuit
    DELAY;                      //wait 125 us
    CTMUCONbits.IDISSEN = 0;    //end drain of circuit

    CTMUCONbits.EDG1STAT = 1;   //Begin charging the circuit
                                //using CTMU current source
    DELAY;                      //wait for 125 us
    CTMUCONbits.EDG1STAT = 0;   //Stop charging circuit

    IFS0bits.AD1IF = 0;         //make sure ADC Int not set
    AD1CON1bits.SAMP = 0;       //and begin conversion
    while(!IFS0bits.AD1IF);     //Wait for conversion complete
    AD1CON1bits.DONE = 0;
    Vread = ADC1BUF0;           //Get the value from the ADC
    if(Vread < OPENSW - TRIP)
    {
        switchState = PRESSED;
    }
    else if(Vread > OPENSW - TRIP + HYST)
    {
        switchState = UNPRESSED;
    }

}
```


37.7 MEASURING TIME WITH THE CTMU MODULE

Time can be precisely measured after the ratio (C/I) is measured from the current and capacitance calibration step by following these steps:

1. Initialize the ADC and the CTMU.
2. Set the EDG1STAT bit.
3. Set the EDG2STAT bit.
4. Perform an analog-to-digital conversion.
5. Calculate the time between edges as $T = (C/I) * V$, where I is calculated in the current calibration step (**37.5.1 "Current Source Calibration"**), C is calculated in the capacitance calibration step (**37.5.2 "Capacitance Calibration"**) and V is measured by performing the analog-to-digital conversion.

It is assumed that the time measured is small enough that the capacitance, C_{OFFSET}, provides a valid voltage to the ADC. For the smallest time measurement, always set the ADC Channel Select register (AD1CHS) to an unused ADC channel; the corresponding pin that is not connected to any circuit board trace. This minimizes added stray capacitance, keeping the total circuit capacitance close to that of the ADC itself (4-5 pF). To measure longer time intervals, an external capacitor may be connected to an ADC channel and this channel is selected when making a time measurement.

37.8 CREATING A DELAY WITH THE CTMU MODULE

A unique feature on board the CTMU module is its ability to generate system clock independent output pulses based on an external capacitor value. This is accomplished using the internal comparator voltage reference module, Comparator 2 input pin and an external capacitor. The pulse is output onto the CTPLS pin. To enable this mode, set the TGEN bit.

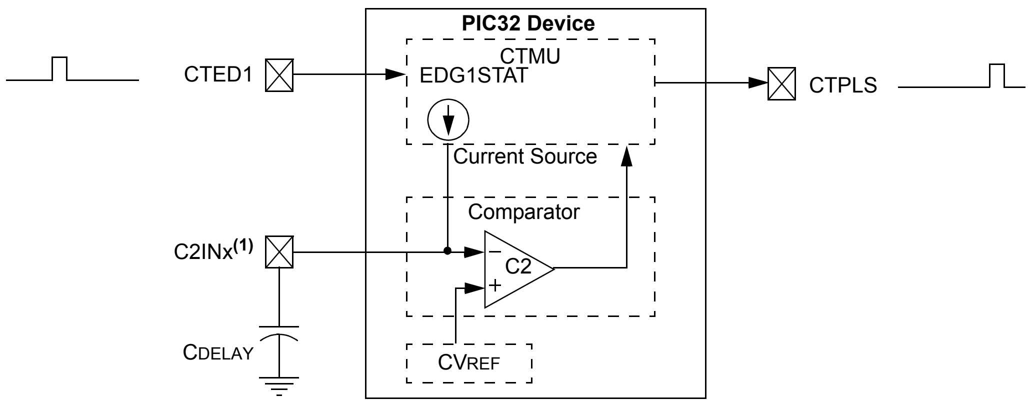
An example circuit is shown in Figure 37-3. C_{DELAY} is chosen by the user to determine the output pulse width on CTPLS. The pulse width is calculated by $T = (C_{DELAY}/I) * V$, where I is known from the current source measurement step (**37.5.1 "Current Source Calibration"**) and V is the internal reference voltage (CVREF).

An example use of this feature is for interfacing with variable capacitive-based sensors, such as a humidity sensor. As the humidity varies, the pulse-width output on CTPLS will vary. The CTPLS output pin can be connected to an input capture pin and the varying pulse width is measured to determine the humidity in the application.

Follow these steps to use this feature:

1. Initialize Comparator 2.
2. Initialize the comparator voltage reference.
3. Initialize the CTMU and enable time delay generation by setting the TGEN bit.
4. Set the EDG1STAT bit.
5. When C_{DELAY} charges to the value of the voltage reference trip point, an output pulse is generated on CTPLS.

Figure 37-3: Typical Connections and Internal Configuration for Pulse Delay Generation



Note 1: Please refer to the specific device data sheet for information related to the comparator input used in this mode.

37.9 MEASURING ON-CHIP TEMPERATURE WITH THE CTMU

The CTMU module can be used to measure the internal temperature of the device through an internal diode that is available for such purposes. When EDG1STAT is not equal to EDG2STAT and TGEN = 0, the current is steered into the temperature sensing diode. The voltage across the diode is available as an input to the ADC module.

Figure 37-4 shows how this module can be used for temperature measurement. As the temperature rises, the voltage across the diode will drop by about 300 mV over a 150ºC range. Selecting a higher current drive strength will raise the voltage value by a few 100 mV.

37.9.1 Basic Principle

We can show that the forward voltage (Vf) of a P-N junction, such as a diode, is an extension of the equation for the junction's thermal voltage, as shown in Equation 37-7:

Equation 37-7:

$$Vf = (kT/q)\ ln\ (1 - If / Is)$$

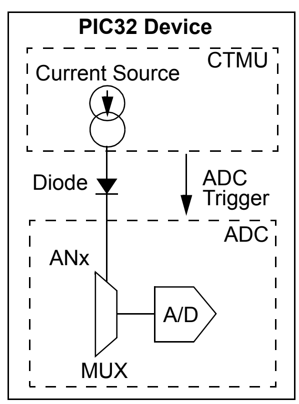
Where :

k is the Boltzmann constant (1.38 x 10-23 J K-1)
T is the absolute junction temperature in kelvin
q is the electron charge (1.6 x 10-19 C)
If is the forward current applied to the diode
Is is the diode's characteristic saturation current

Since k and q are physical constants, and Is is a constant for the device, this only leaves T and If as independent variables. If If is held constant, it follows from the equation that Vf will vary as a function of T. As the natural log term of the equation will always be negative, the temperature will be negatively proportional to Vf.

In other words, as temperature increases, Vf decreases.

Figure 37-4: CTMU Temperature Measurement Circuit



 Preliminary

37.10 OPERATION DURING SLEEP/IDLE MODES

37.10.1 Sleep Mode

When the device enters any Sleep mode, the CTMU module current source is always disabled. If the CTMU is performing an operation that depends on the current source when Sleep mode is invoked, the operation may not terminate correctly. Capacitance and time measurements may return erroneous values.

37.10.2 Idle Mode

The behavior of the CTMU in Idle mode is determined by the CTMUSIDL bit (CTMUCON<13>). If the CTMUSIDL bit is cleared, the module will continue to operate in Idle mode. If the CTMUSIDL bit is set, the module's current source is disabled when the device enters Idle mode. If the module is performing an operation when Idle mode is invoked, in this case, the results will be similar to those with Sleep mode.

37.11 EFFECTS OF A RESET ON CTMU

Upon Reset, all registers of the CTMU are cleared. This leaves the CTMU module disabled, its current source is turned off and all configuration options return to their default settings. The module needs to be re-initialized following any Reset.

If the CTMU is in the process of taking a measurement at the time of Reset, the measurement will be lost. A partial charge may exist on the circuit that was being measured and should be properly discharged before the CTMU makes subsequent attempts to make a measurement. The circuit is discharged by setting, and then clearing, the IDISSEN bit (CTMUCON<9>) while the ADC is connected to the appropriate channel.

37.12 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the PIC32 device family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Charge Time Measurement Unit (CTMU) module are:

Title	Application Note #
No related application notes at this time.	N/A

Note:	Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the PIC32 family of devices.

37.13 REVISION HISTORY

Revision A (March 2011)

This is the initial released revision of this document.

NOTES:

ISBN: 978-1-60932-939-6

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-213-7830
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

02/18/11