Location desktop ex:uMite CFunction Counter23Timer45

CLOCKTRIM Revisited

Introduction

Going back several years, there have been Backshed posts on the subject of CLOCKTRIM, a built-in umite option that allows trimming the clock for maximium accuracy. Some of the authors have been MatherP, TassyJim, Robert Rozee and Twofingers among others. The main problem is what trim value in the range [-31 to 31] decimal to use for maximum accuracy at a given temperature. I find -7 optimum for my 44 pin,@ 65f and -8 at 60F These values are close to -8 that PeterM reported several years ago. When the chip is heated in warm air I have observed optimum clocktrims up to 1. In general there is a range of temperature,about 5-6 deg F, where a single clicktrim is obtained from frequency error measurements. Therefore the response is more step-like near room temperature than smoothly changing. Nevertheless when a first order estimate of optimum clicktrim is needed the following equations has proven useful as a starting point for scanning clocktrim values to fin the optimum.

Approx optimum CLOCKTRIM = 0.2 * Tf – 20

This was derived from optrim- temp transients using the CFUNCTION, quickscan, on the 44 pin Micromite that I will describe.

Before discussing the code, there is a point to be made about how much the oscillator frequency can be varied with clock trim values

-31 to +31. In the PIC32MX170 data sheet and understandably in the Micromite manual it states that the frequency can vary about +-12%. I believe that is not correct for two reasons. I have measured the range to be about +-1.2% @ 65 F. In the microchip forums you will find a post entitled “Abusing FRC Oscillator Tunning” . In the replies near the end there is a post by Larry Standage who is a Microchip forum moderator. He explains that the +-12% figure is valid for older parts such as MX340, 69x,79x (Snad Pic), and others. Further he said they were trying to update the appropriate manuals (that was back in 2016). So the discrepancy is not just a decimal point error, rather Microchip must have copied the incorrect older spec in the newer part (MX170) manual. Pic 32MZ parts datasheets have a +/-2% tuning range in line with what I have found for the 170MX. Also 16F1455 used in USB bridge has tuning range +/- 1.25 %. I have seen quotes that +-2% is typical in newer 16F chips.

Method

Recently I have been working with PWM generated sine waves. I wanted to measure the frequency accurately within the limits of the Micromite MX170 28 and 44 pin chips. The sine waves are being generated by an 8 bit satellite PIC from a group of newer 16F and 18F that have the possibility of 16bit PWM. Naturally when measuring the frequency with the uMite the accuracy of the internal FRC oscillator with PLL's becomes of interest. As you would expect, the measured PWM frequency accuracy with the Micromite is improved by oscillator tuning. I usually measure standard frequency with an oscilloscope either a small Hantek or Rigol depending on where I am in code development. I will report more on the PWM work later.

This post concerns methods to quickly determine the best clock trim using two CFunctions. The first, called CTMU_Temp, determines the diode drop in the temperature measurement module of the CTMU. The diode drop was measured against ambient temp in the range 60 to 105F. This means the diode drop can be used to find an approximate temperature. It is important to note that it is not necessary to know the either the die temp or the ambient temp to determine the optimum clocktrim. All that is required is a accurate clock reference source. It this work I used a crystal based AD9850 DDS and checked the output freq using the six figures counter on a Rigol 1054 digital scope. Two frequencies of 100 Hz and 100 kHz were used in these experiments and gave 100.003Hz and 99.9995 kHz with the scope counter. Temperature is valuable as a guide to narrow the range of clocktrim values that need to be scanned for the optimum value. This range restriction helps to lessen the time required to find the optimum Clocktrim. (Plots of diode drop versus temperature at two different CTMU currents can be found in the CTMU section of the PIC24 device Family Reference Manual.)

Software

Using the another CFunction the entire range of clocktrim,[-31 to +31] was automatically scanned n about 6 sec and these optimum values were plotted against ambient temperature. With the these two functions which were approximately linear, temperature was eliminated and a new relationship between diode drop and optimum clocktrim was derived. The trial optimum clock calculated from the measured diode drop was passed to the second CFunction allowing a restricted range of clocktrim values 5 steps above and below the trial value were used. The input capture module was use to measure the DDS reference freq. Becasue the input caapture works by capturing timer values that are determined by the peripheral bus clock wre ae in effect meauring the error in the CPU frequency. The Micromite is set to have a pbclock divider of unity. In this way the minimum freq error in the measurement of a DDS derived reference freq, usually 100Hz, was the indicator for optimum clocktrim.

This approach reduced the time for scanning clicktrim by about 1/3 to approximatley 2 sec wall clock time. The measurement time for a single clocktrim value in the scan was about 0.1 sec

In general this is not a new topic but the newer method offers fast tunning. The entire clocktrim range [-31, +31] can be scanned and the errors written to the console in about 6 sec. The goal is to find the min absolute error of a DDS reference frequncy as a function of clock trim.

The Cfunction ICap32_ClkTrim is used to determine the optimum CLOCKTRIM value in the allowable range [-31,+31]. The program uses the Input Capture module set to capture two successive rising edges of a DDS generated square wave reference signal. When the captures take place in hardware, a 32 bit timer value is stored in FIFO buffers for each edge. Because we are using a 32 bit timer and there are four 16 bit FIFO buffers, we may store data on two edges giving a single value for the frequency. These results can be read and used to calculate the time between edges which gives the period and thus frequency. The deviation between the measure freq and the reference freq is taken to be the error in the FRC clock in the microprocessor. It should be mentioned that the timer runs at the peripheral bus frequency and can be lower that than the CPU freq. Since the hardware capture takes time > 1.5 periods, we had to put a delay in the code to wait for the FIFO buffers to fill. Once filled the data contained is held until read. Then the buffers can be reset to take more data at the next trim value in the scan loop. This delay time was determined empirically . For a 60Hz FqDDS, o.1 sec per clocktrin worked well. In this case, the scan of all 63 trim values taked about 6-7 sec. The console results are a print out of the error in counts vs CLOCKTRIM. By inspection we could see that at temp = 64F, the optimum value was -7 or sometimes -8 for a 44 pin uMite. At my ambient temp in Feb (Santa Clara, Ca), I believe we are on the border between -7 and -8.

A reference freqs of 100 Hz and 1000Hz were tested with simliar results, but overall shorter total acquisition times, several seconds, as expected. In the accompanying basic program the FqDDS and loop delay are left as input parameters for now. If any users want to test the code and report, I could make the best conditions permanent

Rough temperature testing was performed @60Hz and 0.1 sec delay by means of a stream of warm air focused on the PIC and nearby thermcouple. A clear trend toward less negative trim values was seen. For example, at 64F , -7; at 80 F, - and at 100F, - . Since a single trim value is obtained over a small range of temp, there is not simple linear behavior. A plot of trim value vs temp has step-like character. Nevertheless a linear estimate gave the equation,

trim = 0.2*TempF - 21.

This eq is taken to be a rough guide for the user.
The method described here is meant to find the optimum trim. To determine the best estimate of the relative error, the optimum clocktrim should be used with a long count timer. I have done this and find the realtive errors in the range 0.0001 and 0.0002, that is 100 to 200 ppm. At CLOCKTRIM = 0, the relative error is +0.0038 to +.0040 and very reproduicible.

Results

captions

Fig 1

Absolute error (T23 counts) vs clocktrim using quickscan. T = 64F