Design of a Portable Fast Scan Cyclic Voltammetry Device Utilizing Pulse Width Modulation for Waveform Generation

Nora Szymkowski

A Thesis Submitted to the Graduate Faculty of

GRAND VALLEY STATE UNIVERSITY

In

Partial Fulfillment of the Requirements

For the Degree of

Master of Science

School of Engineering

August 2024

Thesis Approval Form

GRAND VALLEY STATE UNIVERSITY

The signatories of the committee below indicate that they have read and approved the thesis of Nora Cole in partial fulfillment of the requirements for the degree of Master of Electrical and Computer Engineering.

Brian Krug 7/10/24

Dr. Brian Krug, Thesis committee chair Date

Nabeeh Kandalaft

7/10/24

Dr. Nabeeh Kandalaft, Committee member Date

7/10/24

Dr. Eric Ramsson, Committee member Date

Accepted and approved on behalf of the Padnos College of Engineering

Timothy L. B

Dean of the College

7/25/2024

Date

Accepted and approved on behalf of the Graduate Faculty

Dean of The Graduate School

8/5/2024

Date

Abstract

Fast Scan Cyclic Voltammetry (FSCV) is a widely used electrochemical technique for real-time measurement of the brain's chemical messengers, including the molecule dopamine, with high temporal resolution. Currently the financial burden of performing FSCV is quite high, ranging from $8,000 to $20,000+ making the barrier to entry nearly insurmountable for laboratories and classrooms at small institutions. The purpose of this project was to develop a Do-It-Yourself (DIY), portable, and cost-effective FSCV system for use in laboratory and classroom settings. The project aimed to create a compact and cost-effective system that could be used by researchers and educators to study dopamine levels and dynamics in various settings. By utilizing a Hercules Launchpad by Texas Instruments, a device was designed to perform FSCV of dopamine. The device was able to successfully produce the required waveform and collect the results. This project offers proof of concept for the use of Pulse Width Modulation (PWM) in the generation of a FSCV triangle waveform for dopamine concentration measurements.

Table of Contents

List of Tables

List of Figures

List of Equations

Chapter 1 Introduction

Introduction

The field of neuroscience and research into chemical signaling in the brain is of great importance for several reasons including: understanding brain function, studying neurological disorders and mental health conditions, designing new treatments for those disorders and conditions, and developing new technology [1].

Neuroscience research aims to uncover the complex workings of the brain, including how it processes information, controls behavior, and regulates various bodily functions. By studying chemical signaling, researchers can gain insights into the fundamental mechanisms underlying brain function, such as learning, memory, emotions, decision-making, and sensory processing [2]. Their research also helps shed light on the causes, mechanisms, and potential treatments for neurological disorders and mental health conditions. Chemical signaling abnormalities, such as imbalances in neurotransmitters are often implicated in conditions like Parkinson's disease, Alzheimer's disease, depression, anxiety disorders, and schizophrenia [1]. Understanding the chemical signaling disruptions associated with these conditions can aid in the development of targeted interventions and therapies.

Investigating chemical signaling in the brain provides critical insights for drug discovery and development. Many drugs act on specific receptors or modulate neurotransmitter levels to treat neurological and psychiatric disorders. Research into chemical signaling pathways helps identify potential drug targets and aids in the development of more effective and specific pharmacological interventions. Additionally, advancements in understanding chemical signaling contribute to the development of brain-computer interfaces (BCIs). BCIs enable direct communication between the brain and external devices, offering potential solutions for

individuals with paralysis or sensory impairments. Research in chemical signaling assists in designing interfaces that can interpret and modulate neuronal activity, facilitating bidirectional communication between the brain and external devices [3].

Fast Scan Cyclic Voltammetry (FSCV) is a widely used electrochemical technique for real-time measurement of the brain's chemical messengers, including the molecule dopamine, with high temporal resolution. FSCV involves applying a voltage waveform to a carbon-fiber electrode and measuring the resulting electrical currents as the molecules undergo oxidation or reduction reactions [4].

Currently the financial burden of performing FSCV is quite high, ranging from $8,000 to $20,000+. This makes the barrier to entry nearly insurmountable for laboratories and classrooms at small institutions. A low cost, portable FSCV device would make classroom demonstrations and proof of concept research for grant proposals possible in these more resource-restricted environments.

Purpose

The purpose of this project was to develop a DIY, portable, and cost-effective FSCV system for use in laboratory and classroom settings. The project aimed to create a compact and cost-effective system that could be used by researchers and educators to study dopamine levels and dynamics in various settings, including resource-constrained environments such as small university laboratories and classrooms. The project sought to achieve the following objectives:

1. Affordability and Accessibility: The focus of the project was on utilizing low-cost components and creating easily implementable, open-source software to develop a 'DIY' FSCV device that is affordable and accessible to a wide range of users. By reducing the

production cost, the project aimed to remove financial barriers and promote widespread access to FSCV technology.

2. Portability and Compact Design: The device was designed to be portable and lightweight, allowing for easy transportation and use in different settings. Its compact form factor was intended to enable researchers and educators to conduct experiments and demonstrations outside the confines of a traditional laboratory.
3. Real-Time Data Acquisition and Analysis: The device incorporated hardware and software designed to enable real-time data acquisition, analysis, and visualization of dopamine levels. In order to accomplish this the device was required to perform the following tasks:
    a. Generation of the stimulus waveform: The present device was designed with the ability to produce just one triangle waveform. The waveform selected is a very commonly used one when measuring dopamine concentrations.
    b. Measurement of the current flow in the electrode: The device was designed with the intention to be compatible with a single carbon fiber ‘working’ electrode and a single carbon fiber ‘reference’ electrode.
    c. Amplification of the recorded current waveform: Due to the miniscule nature of the current waveform produced, the device required circuitry to amplify this waveform to make it interpretable.
    d. Conversion of the current waveform to a digital signal: In order for the data to be retrieved by the device, the analog signal would have to be converted into a digital one.

e. Communication of the digital signal to a PC: Once the signal had been converted to be digital, it was then required to be sent to a PC for analysis.

f. Analysis and display of the resulting current signal: The final signal would then have to be interpreted by an algorithm and displayed in order for it to be useful to researchers.

4. Versatility and Compatibility: The open-source, 'DIY' nature of the FSCV device was intended to allow for future users and developers to build on the current work. With further development, future iterations of the device could be made compatible with a variety of electrode configurations and be made capable of producing multiple waveforms in order to support the measurement of different neurotransmitters, such as serotonin and glutamate.

5. Educational and Research Applications: The project aimed to foster educational and research opportunities by providing an affordable tool for studying neurotransmitter dynamics. The device will empower students, educators, and researchers to explore the principles of neuroscience promoting scientific discovery and understanding.

The final objective of this project was to attempt to address the limitations in the portable FSCV system developed by Foster at Grand Valley State University. The intent was to consolidate the system to just one microcontroller capable of performing both the waveform generation and data acquisition simultaneously. It was also a goal to, if possible, ensure the system performs well thermally at ambient temperatures without the need for makeshift cooling methods.

Chapter 2 Review of Literature

Mechanism of Neurotransmitters

The human body uses its nervous system to think, move, feel, see, hear, speak, taste, and smell. The brain, spinal cord, and a network of peripheral neurons make up the nervous system and are able to communicate and process massive amounts of sensory data, control the voluntary muscular system, ensure life supporting organ functions work seamlessly, store memories, think critically, and much more [2]. It is able to accomplish all of this simultaneously and at incredible speeds. Much of this nervous tissue is constructed of a powerful cell type called neurons [1]. A neuron and its basic anatomy are shown below in Figure 1.

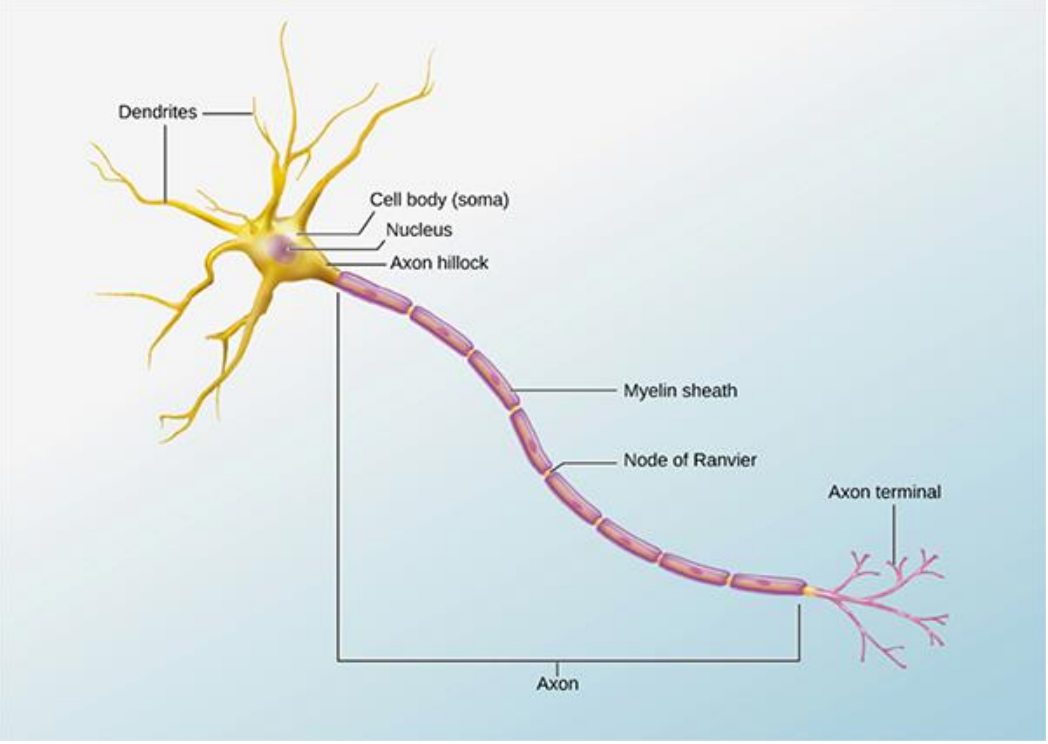


Figure 1. Neuron Diagram [5]

Neurons are similar to other cell types in that they have a cell body which contains a nucleus responsible for the cell's maintenance; however, they have other highly specialized structures which allow them to participate in the momentous task of operating the human body.

These structures include the dendrites (long tendrils protruding from the cell body), the axon (the long cable like structure that makes up much of the cell), the axon hillock (the junction where the cell body meets the axon), and the axon terminal (the branched end of the axon) [2]. Each of these structures is identified above in Figure 1.

Neurons transmit data to, from, and throughout the brain via electrical impulses that propagate along the length of each cell. When the dendrites of a cell are stimulated, ion channels are opened which create a local change in the potential difference across the cell membrane. This impulse, known as a graded potential, travels through the cell body toward the axon. When a great enough graded potential arrives at the axon hillock, a strong impulse known as an action potential is triggered and travels down the length of the axon to the axon terminal. This terminal end does not physically connect to the following neurons, so in order for the signal to pass from one neuron to the next, the signal must be converted into a chemical one. Specialized molecules called neurotransmitters are released from the axon terminal into the extracellular space between cells known as the synapse. These molecules bind to receptors on the dendrites of the postsynaptic neuron, stimulating another graded potential and thus the signal continues. Once the signal has been successfully passed, the neurotransmitter molecules are taken back up by the presynaptic neuron. This allows the synapse and the postsynaptic receptors to be cleared so that a future signal can be transmitted. [1].

There are many dozens of neurotransmitters that have been identified in the human body to date. Some function as excitatory, meaning they aid in triggering further action potentials as discussed above, and others as inhibitory, meaning they work to stop or reduce further action potentials, and still others can do both of these depending on the specific circumstances [1]. Both are critical to keeping all functions of the body working properly and in balance. Although there

are many neurotransmitters present and working in the nervous system, there are just several key ones which do much of the work and are specialized to regulate and govern certain functions within the brain and body. Some of these molecules are Glutamate, Gamma-Aminobutyric Acid (GABA), Epinephrine, Norepinephrine, Histamine, Serotonin, Oxytocin, Acetylcholine, and Dopamine [2]. For the purposes of this project, the focus will be on dopamine.

Dopamine

The importance of the role of the dopamine molecule in the human body cannot be overstated. Considered both a neurotransmitter and a hormone, dopamine plays a vital role in many bodily functions [1]. To name a few: movement, attention, motivation/reward, learning, and mood [2]. Additionally, the dopamine molecule functions as a precursor to epinephrine and norepinephrine [6], two pivotal neurotransmitters/hormones in the human body responsible for producing the critical "fight-or-flight" response to perceived danger [2].

One of the first known roles of dopamine was its impact on coordination and smooth movement [7]; however, in recent years, the findings suggest that one of the primary roles of dopamine is governing motivation and reward [8]. It is thought that the release of this molecule is necessary for the initiation of an action as well as the reinforcement of that action [9]. If the outcome of an action is as expected, the dopamine release in response to the outcome reflects that. Likewise, if the outcome is either better or worse than expected, the dopamine release increases or decreases correspondingly to either positively or negatively reinforce that action [10].

It is also hypothesized that the release of dopamine in the hippocampus in response to a novel event induces long term memory storage [11], and that dopamine plays a role in memory deletion as well [12]. In this way, dopamine has a massive impact on how the human brain learns

from actions and experiences. It is well known that drugs like amphetamines form dependencies because they act directly on dopamine terminals, but it is now also suggested that any substance that triggers dopamine's reward system can become habit forming including things like cannabis and caffeine [13]. All of these roles of dopamine reveal just how important it is when it comes to human behavior at large. There is still plenty that neuroscientists do not understand when it comes to dopamine and its functions, and it is an ongoing area of research to understand how to manipulate and maintain proper dopamine function.

Measurement Techniques

Over the years of research into neurotransmitters and more specifically dopamine, many techniques have been developed and used to measure dopamine levels in the brains of animals and humans. Some of these methods include:

1. Mircodialysis: This method of neurotransmitter measurement is invasive and is most commonly used in animal models. This method involves inserting a semipermeable membrane on the end of a probe into a particular brain region through which a sample of the extracellular fluid can be collected and then analyzed [14]. While not considered the preferred method, this technique has been historically used to measure dopamine concentrations in the interstitial space of a brain structure or region [15].
2. Positron Emission Tomography (PET): This method utilizes a PET scan and is therefore considered a non-invasive measurement technique. This procedure, "enables the direct measurement of components of the dopamine system in the living human brain. It relies on radiotracers which label dopamine receptors, dopamine transporters, precursors of dopamine or compounds which have specificity for the enzymes which

degrade dopamine," [16], all of which give researchers an abundance of information about the dopamine system at large.

3. Functional Magnetic Resonance Imaging (fMRI): This method employs an MRI which has the advantages of being non-invasive as well as non-radiological. However, it is more limited than its more invasive counterparts in that it cannot directly measure dopamine levels. This technique instead utilizes proxy measures that can then be extrapolated to a high-level understanding of "long term dopamine function or degeneration of dopaminergic neurons", [17].
4. Single-Cell Recordings: This method, while more cutting edge, is considered invasive and highly technical. This technique involves the insertion of very fine microelectrodes directly into individual neurons in order measure changes in voltage and current. This allows single cells to be monitored for activity in response to various external stimuli [18]. This makes single cell recordings highly detailed and granular, but also cost-prohibitive as well as technically prohibitive.
5. Fast Scan Cyclic Voltammetry (FSCV): This invasive method can measure dopamine concentrations with a high level of sensitivity in real time. This technique involves the placement of two single carbon-fiber electrodes into a brain sample or brain region. Changes in the electrical current as dopamine is oxidized and then reduced can be measured and then used to determine the concentration of dopamine present in the sample with great accuracy and temporal resolution [4]. It is noteworthy that this method is also quite cost-prohibitive.

While not comprehensive, this list represents some of the most common measurement techniques for neurotransmitters and for dopamine specifically. When to employ a given technique can only

be decided based on the requirements of the experiment in light of the strengths and weaknesses of each method. For the purposes of this project, the focus will be on FSCV.

FSCV Overview

Since only discussed very briefly above, this section will be used to delve deeper into the measurement technique known as Fast Scan Cyclic Voltammetry or FSCV. As mentioned previously, this method involves the placement of two single carbon-fiber electrodes into a brain sample or region; one is known as the working electrode and the other is known as the reference electrode. These electrodes are then used both to incite an oxidation-reduction reaction of dopamine and to measure the changes in the electrical current that result from that reaction. Systems that perform these tasks typically need the following components:

1. Electrode: A thin carbon fiber around 5-30 μm in diameter that is usually insulated aside from the very end which is placed into the solution or sample. These electrodes are highly conductive in order to be sensitive enough for neurotransmitter detection.
2. Potentiostat: An electronic device that is used to apply the correct voltage waveform to the working electrode and to measure and amplify the very small electrical currents produced by the experiment.
3. Data Acquisition System: Equipment used to capture and record the measurements from FSCV using an analog-to-digital converter (ADC). This takes the true analog measurements and digitizes them so they can be processed and stored by a computer.
4. Control and Analysis Software: Specialized software to control the initiation, recording, and visualization of incoming data in real time. May also include additional tools and algorithms for data analysis and neurotransmitter quantification.

The most common waveform used for FSCV of dopamine is a triangular waveform. The triangular waveform is simple, effective, and widely employed due to its favorable characteristics for dopamine oxidation and reduction measurements. The specific voltage parameters associated with the triangular waveform include: the initial potential (the starting voltage of the triangular waveform commonly set to a slightly negative potential such as -0.4V to establish a baseline and prepare for the oxidation phase of dopamine), the peak potential (the highest voltage reached during the positive phase of the triangular waveform, commonly 1.3V), the final potential (the voltage at the end of the negative phase of the triangular waveform, typically the same value as the initial potential), and the scan rate (the speed at which the triangular waveform is scanned, for instance, 400V/s).

This waveform is typically repeated 10 times per second for the duration of the measurement. During the positive phase of the triangular waveform, the voltage applied to the carbon-fiber electrode increases. As the voltage exceeds around 0.6V, dopamine molecules near the electrode undergo oxidation to form dopamine-*o*-quinone. During this reaction dopamine loses 2 electrons which incites a measurable electrical current at the electrode. This electrical current is directly proportional to the rate of dopamine oxidation, which can then be extrapolated to calculate the concentration of dopamine in the electrodes immediate proximity. During the negative phase of the triangular waveform, the voltage applied to the electrode decreases. At around -0.1V, the reduction of dopamine-*o*-quinone occurs and the molecule gains 2 electrons to once again become dopamine. The reduction reaction also generates an electrical current, which is proportional to the rate of dopamine reduction and, hence, the concentration of dopamine-*o*-quinone near the electrode.

The electrical currents generated during both the oxidation and reduction reactions are measured by the FSCV setup which amplifies and records these currents as a function of the applied voltage. By analyzing the recorded current data, researchers can determine how dopamine concentration changes over time. An image of the waveform as well as the chemical reactions and the resulting current measurements can be seen in Figure 2.

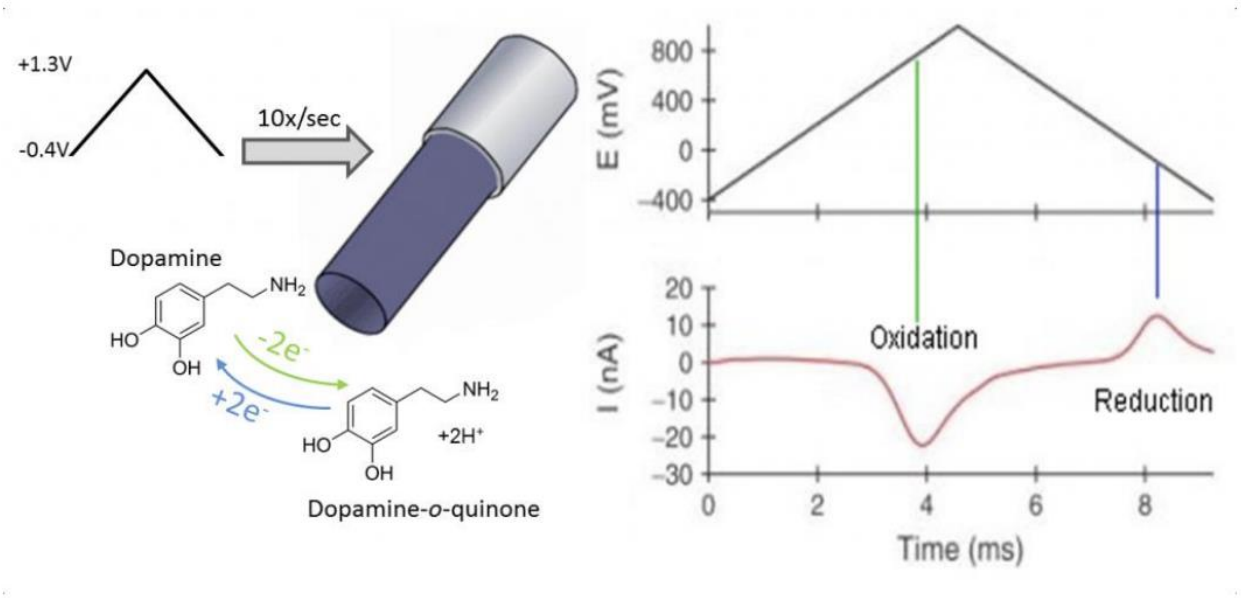


Figure 2. Waveform and Oxidation-Reduction Reactions [19]

Existing Devices and Previous Work

Devices such as the WaveNeuro Potentiostat by Pine Research are typically used to perform FSCV. National Instruments data acquisition cards as well as their LABVIEW® software are also required to perform the experiment. The cost of all of this equipment can range from around $8000 on the low end for simple setups all the way up to $20000+ for more complex setups. With this sort of price tag, FSCV is incredibly cost-prohibitive for smaller institutions, researchers, and educators who wish to demonstrate and utilize the effectiveness of

FSCV for the measurement of dopamine. This reveals an opportunity to provide a lower cost option.

The effort to lower the financial burden of this measurement technique is not novel. Previous work in this area was completed by Jayson Foster of Grand Valley State University in which he designed an Arduino based system to perform the waveform generation and data acquisition and utilized MATLAB to analyze the data and display the result. In the course of that project, it was determined that the microcontroller was unable to simultaneously perform the signal generation for the triangle waveform and the data collection through the ADC. In order to combat this, the tasks of waveform generation and data acquisition were divided between two separate microcontrollers. Additionally, it was discovered that the thermal load resulting in the system was producing inaccurate data and makeshift cooling methods were required to perform the measurements.

Chapter 3 Methodology

The development of the device was broken down into two distinct areas: hardware and software. Tasks categorized under the hardware category included: researching and selecting a microcontroller and designing the circuit for the waveform signal conditioning and result signal amplification. Tasks categorized under the software category included: writing the firmware for the microcontroller to perform the waveform generation and data collection (written in C) and writing the script for the data analysis and display (written in python).

Hardware

The board chosen for this project was the Hercules RM42x LaunchPad, a development kit offered by Texas Instruments (TI) specifically designed for the Hercules RM42x series of microcontrollers. The Hercules line of microcontrollers is designed for safety-critical designs and as such is commonly used in biomedical applications. The Hercules microcontrollers often include specialized peripherals and modules to support real-time operations making them a good option for this application. The launchpad utilizes the Hercules 100-pin RM42L432 microcontroller. Key specs for this launchpad that are most relevant to this project are a 12-bit analog to digital converter (ADC), and a programmable high-end timer (HET) module designed for highly accurate real-time operations. The HET module is capable of using Pulse Width Modulation (PWM) to generate complex waveforms independently of the main CPU. This makes it an ideal solution for the current project in that all of the signal generation can be done separately from the data acquisition, ensuring the two processes do not conflict without the need for a second microcontroller.

PWM is a method of encoding a voltage onto a fixed frequency carrier wave. The frequency of the PWM will be fixed while the duty cycle will vary between 0% and 100%. The

percentage of the on-time will be proportional to the output signal voltage. For example, a 0% duty cycle produces a 0 V output while a 100% duty cycle produces a peak-to-peak voltage Vp-p equal to the Vccio, which is the I/O power supply voltage to the microcontroller. The nominal Vccio is 3.3 V in Hercules microcontrollers so a 50% duty cycle would have produced an output voltage equal to 1.65 V. The PWM method is a low-cost way of implementing a digital-to-analog converter (DAC). By time-varying the duty cycle percentage, it is possible to generate an arbitrary analog waveform. The output signal needed to go through a simple analog low-pass filter to remove the high frequency components. This filter is constructed from a 220Ω resistor and a 10µF capacitor, which can be seen in Figure 3. The full schematic for the project can be found in Appendix I. The filtered signal was the expected triangle shape; however, it is also noteworthy that the microcontroller is only capable of producing signals between 0V and 3.3V. As discussed previously, the parameters for the present triangle wave require the signal to range from -0.4V to 1.3 V. The purpose of the remainder of the circuit shown in Figure 3 is to shift the triangle wave signal down to the required level utilizing a voltage divider.

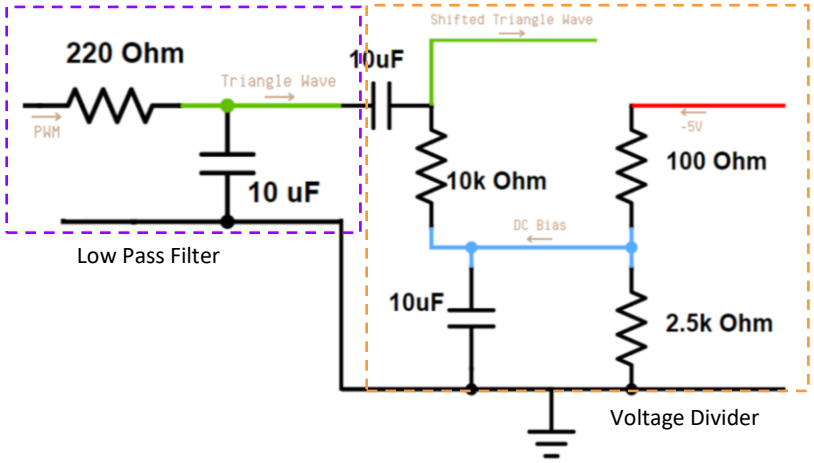


Figure 3. Waveform Generation Signal Conditioning Circuit

The first strategy to shift the signal down was simply to run the signal through a 10µF capacitor since this would essentially equalize the amount of signal that exists above and below 0V. It was

determined that this strategy did not shift the signal enough to satisfy the requirements, so a voltage divider was utilized to further bias the signal down by almost 0.2V. Equation 1 was used to design the voltage divider circuit.

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2} \quad (1)$$

Where V_{out} is the output voltage, V_{in} is the input voltage, R_1 is the voltage side resistor, and R_2 is the ground side resistor. It was found that an input voltage of -5V, a voltage side resistor of 2.5kΩ, and a ground side resistor of 100Ω produced the required output voltage of -0.192V.

Once the triangle waveform had been sufficiently modified by the circuitry to meet the FSCV parameters for dopamine oxidization and reduction, the signal could be sent to the working electrode, where the reference electrode could pick up the resulting fluctuations in the current present in the sample. The resulting signal was then passed through a current-to-voltage converter and amplifier circuit shown in Figure 4. This circuitry was left largely unchanged from the previous work completed by Jayson Foster at GVSU.

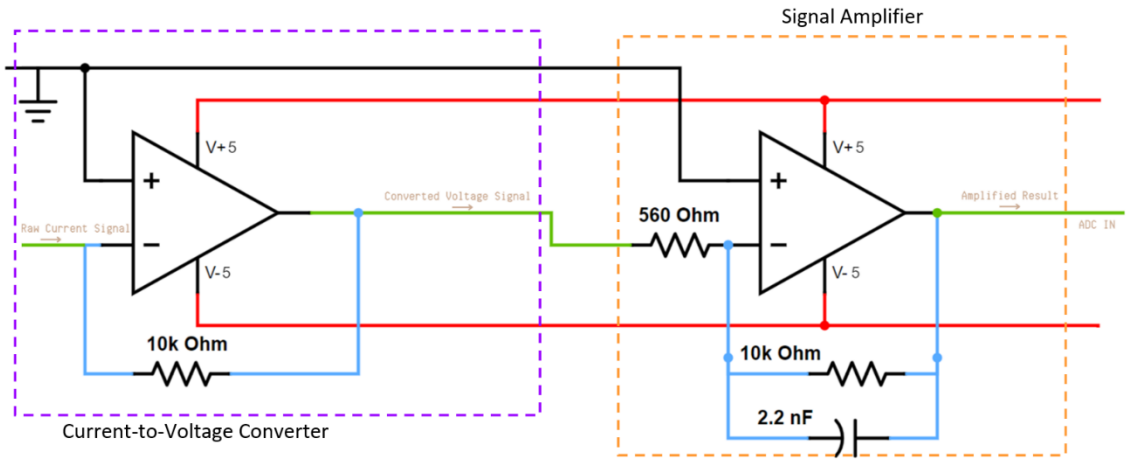


Figure 4. Current-to-Voltage Converter and Amplifier Circuit

The current follower equation shown below would then allow the current at the reference electrode to be calculated.

$$V_{out} = -iR_f \tag{2}$$

Where V_{out} is the output voltage of the current-to-voltage converter, i is the current at the reference electrode, and R_f is the feedback resistor. For this project a feedback resistor of 10kΩ was used. The signal next needed to pass through the amplifier portion of the circuit so that the very small signal could be read by the board. This part of the circuit is governed by equation 3:

$$V_{out} = -V_{in}\frac{R_f}{R_i} \tag{3}$$

Where V_{out} is the output voltage of the amplifier, V_{in} is the input voltage, R_f is the feedback resistor, and R_i is the input resistor. For this portion of the circuit a feedback resistor of 10kΩ and an input resistor of 560Ω were used. This signal was then passed back to the Hercules board to be measured by the onboard ADC.

Software

The first software step for this project was the development of the code to produce the PWM signal for the custom triangle waveform required for the oxidation and reduction reactions. The block diagram for this code is shown below in Figure 5.

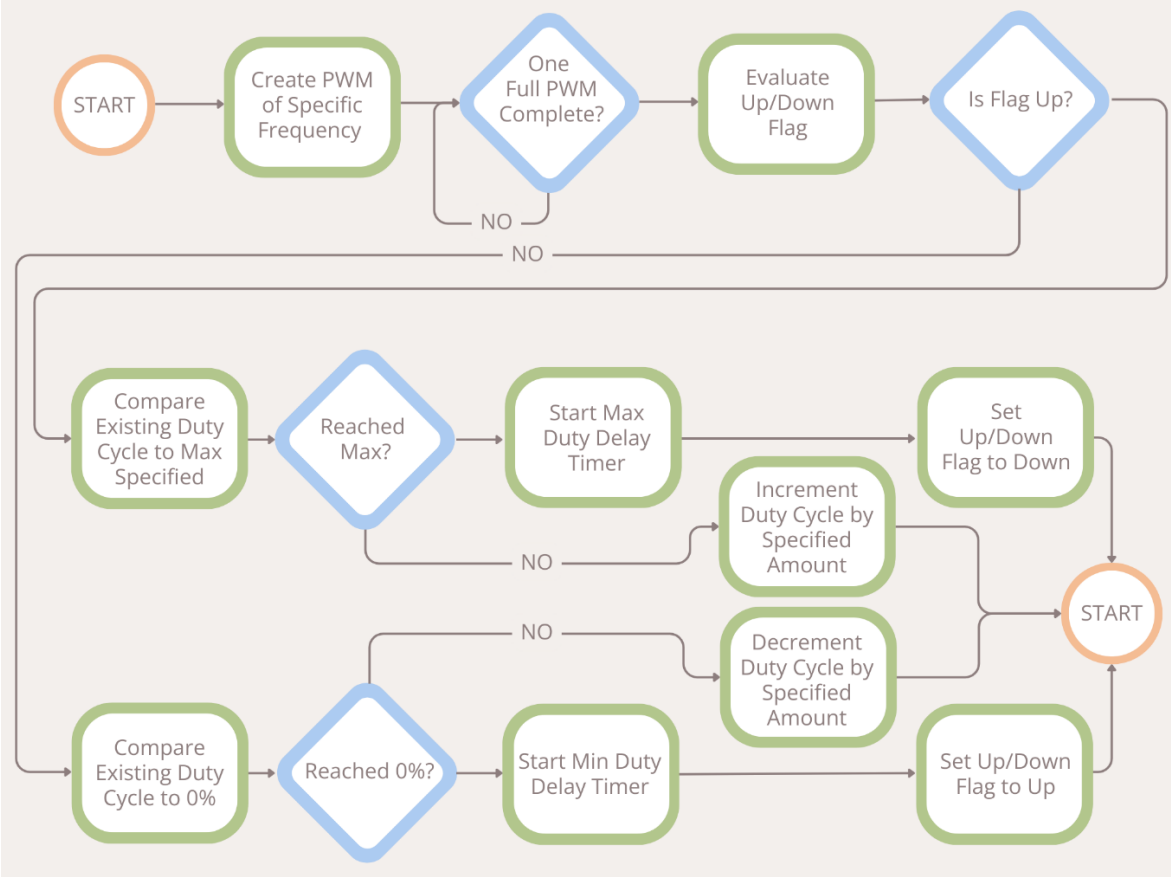


Figure 5. Block Diagram for PWM Triangle Wave Generation Code

The implementation is as follows:

1. The specified PWM is generated starting with 0% duty cycle. The duty cycle will be self-modified by the N2HET1. Wait until one full PWM period is generated before changing to a new duty cycle.
2. The up/down flag is evaluated to determine whether the duty cycle should increase or decrease. The flag will be initialized to zero after reset, meaning to increment the duty cycle, thus increasing the resulting PWM voltage. If the flag is 1, the duty cycle will decrement instead, thus decreasing the resulting PWM voltage.
3. The existing duty cycle is compared to the programmed maximum duty cycle. This parameter determines the maximum voltage the PWM signal will reach. The maximum

duty cycle is a parameter changeable by the host CPU before the N2HET program starts. If the maximum is reached, go to step 5.

4. The next duty cycle percentage is incremented by the programmed amount. This parameter determines the slope of the incline of the PWM wave.
5. A programmed timer is started. The timer is used to hold the PWM at the maximum duty cycle for a programmable amount of time. This step would be needed for creating trapezoid waveforms. In the present case, since a triangle wave is required, this timer delay was set to zero.
6. The existing duty cycle is compared to the 0% duty cycle. If 0% is reached, go to step 8.
7. The next duty cycle is decremented by the programmed amount. This parameter determines the slope of the decline of the PWM wave. Note that the amount to decrement can be different than the amount to increment to generate a sawtooth; however, this was not required in the present work.
8. A programmed timer is started. The timer is used to hold the PWM at the 0% duty cycle for a programmable amount of time. Note that this minimum duty cycle timer length can be different from the maximum duty cycle timer length. This was used to implement the time delay between individual triangle waves.

In order to implement the HET code, the N2HET Assembler from TI was used. The assembler translates the higher-level HET code, which can be found in Appendix II, so that it could be used by the HALCoGen software from TI to generate all the needed c and header files for the project. HALCoGen stands for Hardware Abstraction Layer Code Generator. Once HALCoGen had generated the project it was opened in Code Composer Studio where the main.c file could be written. This file was also pre-generated by HALCoGen with a skeleton code

consisting mainly of comments outlining where various parts of the code should be written so the file can remain interpretable by the HALCoGen tool. The code added to the main.c file includes 8 configurable macros which control the parameters of the triangle wave to be produced. They are: LRPFC, PWM_PERIOD, PWM_DUTY, DUTY_INCREMENT, DUTY_DECREMENT, MAX_DUTY_TIMER, MIN_DUTY_TIMER, and NHET1_PIN_PWM.

The carrier frequency of the PWM signal can be expressed in terms of the Loop Resolution Period (LRP). This is the frequency at which the HET module will loop through the HET code.

$$LRP = VCLK2 \times 2^{LRPFC} \times PWM\_PERIOD \quad (4)$$

In the equation shown above the VCLK2 is the clock speed which equals 11.11ns. LRPFC is the first configurable macro and stands for Loop Resolution Pre-scaler Factor Coefficient. The possible values for this macro are 5, 6, or 7. PWM_PERIOD is the second configurable macro and can be set to an integer in the range of 1 to 32. So if, for instance, LRPFC is set to 6 and PWM_PERIOD is set to 1, the NHET has 64 clock cycles for 1 LRP. This means the NHET can modulate the duty cycle between these 64 steps, giving a 6-bit resolution for the amplitude of the output signal. This resolution can be improved by increasing either LRPFC or PWM_PERIOD. For example, with and LRPFC set to 7 and PWM_PERIOD set to 32, 1 LRP is equivalent to 4096 clock cycles giving a 12-bit resolution. The higher the resolution the less noise present in the final analog signal. However, this comes with a tradeoff; the higher the resolution, the lower the possible output frequency that can be generated. If a higher frequency output signal is required, some resolution would have to be sacrificed. For the purposes of this project LRPFC

was set to 6 and PWM_PERIOD was set to 32 giving an LRP of 2048 clock cycles or an 11-bit resolution.

The next configurable macro is PWM_DUTY, which represents the maximum duty cycle the NHET will be allowed to reach. Practically speaking this is the max amplitude of the filtered output signal. This variable is expressed as a percentage of the total possible duty cycle and has a possible range of 0.1% to 100% which would allow the output signal to have a max amplitude anywhere in the range of 0.33mV to 3.3V.

DUTY_INCREMENT and DUTY_DECREMENT are the next configurable macros, and they represent the amount to increase or decrease the duty cycle from one step to the next. This variable is also given as a percentage in the range 0.1% to 100%. In the output signal these control the slope of the rise and the fall. These two macros can be set to different values to generate a sawtooth wave if that is required. The present work required a triangle wave so they were both set to the same value.

MAX_DUTY_TIMER and MIN_DUTY_TIMER represent the amount of time for which the system will maintain either the maximum or minimum duty cycle and is expressed in terms of number of LRP. This is used to change the amount of time the output signal will hold at either the maximum or minimum value and is thus useful for generating a trapezoid wave. The signal produced in this project is a triangle pulse with a delay between each during which the signal remains at baseline. In order to accomplish this, the MAX_DUTY_TIMER was set to 0 and the MIN_DUTY_TIMER was set to around 4000 LRP which produced a delay of around 90ms.

The final macro is NHET1_PIN_PWM which defines the pin number at which the PWM signal will be produced. For this project, PIN_HET_0 was used.

The next software to be written was the code to read the signal coming into the ADC for measurement. All the necessary include files for this portion of the code were also generated with the HALCoGen tool. These included the files for utilizing both the ADC module to read the voltage at the input pin, as well as the SCI module to print the values in a terminal window on the PC. Additional code was added to the main.c file to implement both of these tasks. The key pieces of code for this part of the main function are shown below in Figure 6. The full main.c file with all of these parameters and functions can be found in Appendix II.

```
adcData_t adc_data; //ADC data structure
adcData_t *adc_data_ptr = &adc_data; //ADC data pointer
unsigned int NumberOfChars, value; //Declare variables

sciInit(); //Initializes the SCI (UART) module
adcInit(); //Initializes the ADC module

while(1)
{
adcStartConversion(adcREG1, adcGROUP1); //Start ADC conversion
while(!adcIsConversionComplete(adcREG1, adcGROUP1)); //Wait for ADC conversion
adcGetData(adcREG1, 10, adc_data_ptr); //Store conversion into ADC pointer
value = (unsigned_int)adc_data_ptr->value;
NumberOfChars = ltoa(value,(char*)command);
sciSend(sci1inREG, 2, (unsigned char *)"0x"); //Sends hex designation
sciSend(sci1inREG, NumberOfChars, command); //Sends the data
sciSend(sci1inREG, 2, (unsigned char *)"\r\n"); //Sends new line character
}
```

Figure 6. Main Function Code for ADC and SCI Modules

The final software step in this project was the development of a python script to process the raw data from the board and make the necessary calculations to visualize the current change in the sample. This script utilized both equations 2 and 3 to perform these calculations and then used a basic digital lowpass Butterworth filter to remove the noise. The simple block diagram for this script is shown below in Figure 7.

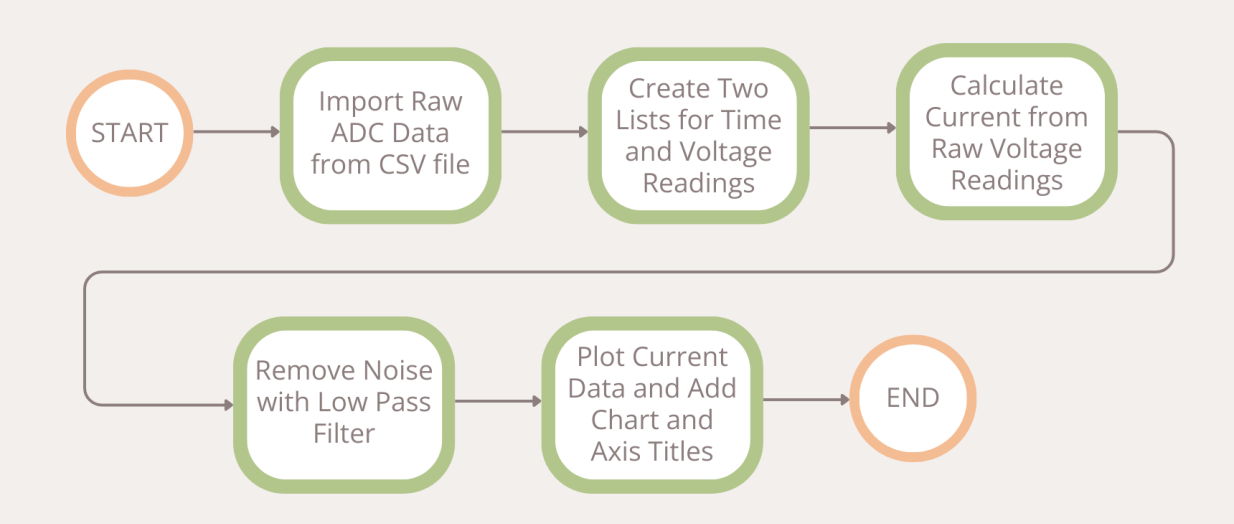


Figure 7. Block Diagram for Data Processing Python Script

Chapter 4 Results

The PWM output from the board as well as the resulting waveforms after both the initial analog lowpass filter as well as the voltage divider were visualized using an oscilloscope. These can be found below in Figure 8. Measurement A in Figure 8 shows a single, raw PWM signal coming from the Hercules board. This makes it possible to visualize the duty cycle increasing and then decreasing. Measurement B in Figure 8 shows three PWM signals both before (bottom) and after (top) the analog low pass filter. The low pass filter produced a triangle wave from the variable duty-cycle PWM signal. Measurement C in Figure 8 shows that the triangle wave, after the low pass filter, has a period of 99.9ms, a peak-to-peak voltage of 1.72V, and a rise time of 4.2ms. Measurement D in Figure 8 shows the triangle wave after the voltage divider biasing circuit has the same period, peak-to-peak voltage, and rise time as that of Measurement C but now with a minimum voltage of -0.4V and a maximum voltage of 1.32V.

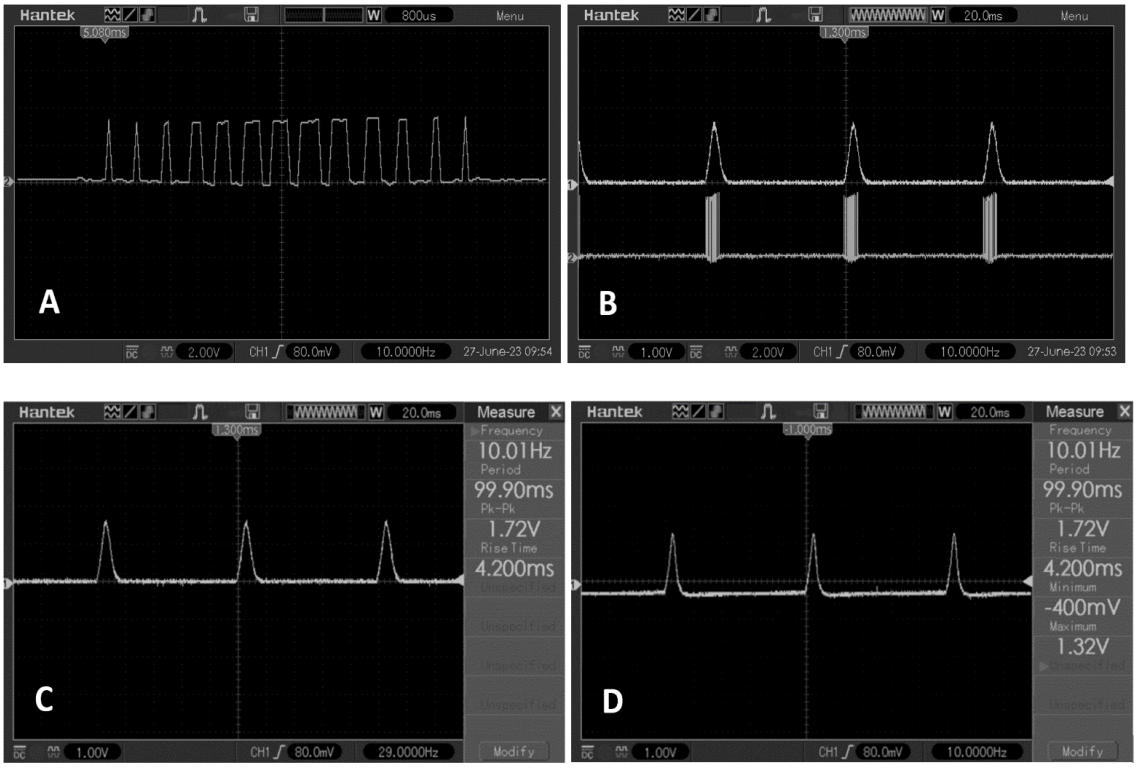


Figure 8. Oscilloscope Measurements of Triangle Wave Generation

In place of a dopamine solution, a dummy cell constructed from a 1nF capacitor and a 100kΩ resistor was used to test the device. Figure 9 below shows the dummy cell current waveform measurement taken with the present device both before and after the filtering done within the algorithm, compared to the same measurement made with the gold standard equipment. Chart A in Figure 9 shows the measurement taken with the Hercules device without the digital Butterworth filter applied, chart B in Figure 9 shows the measurement taken with the Hercules device with the digital Butterworth filter applied, and chart C in Figure 9 shows the measurement taken with the gold standard equipment.

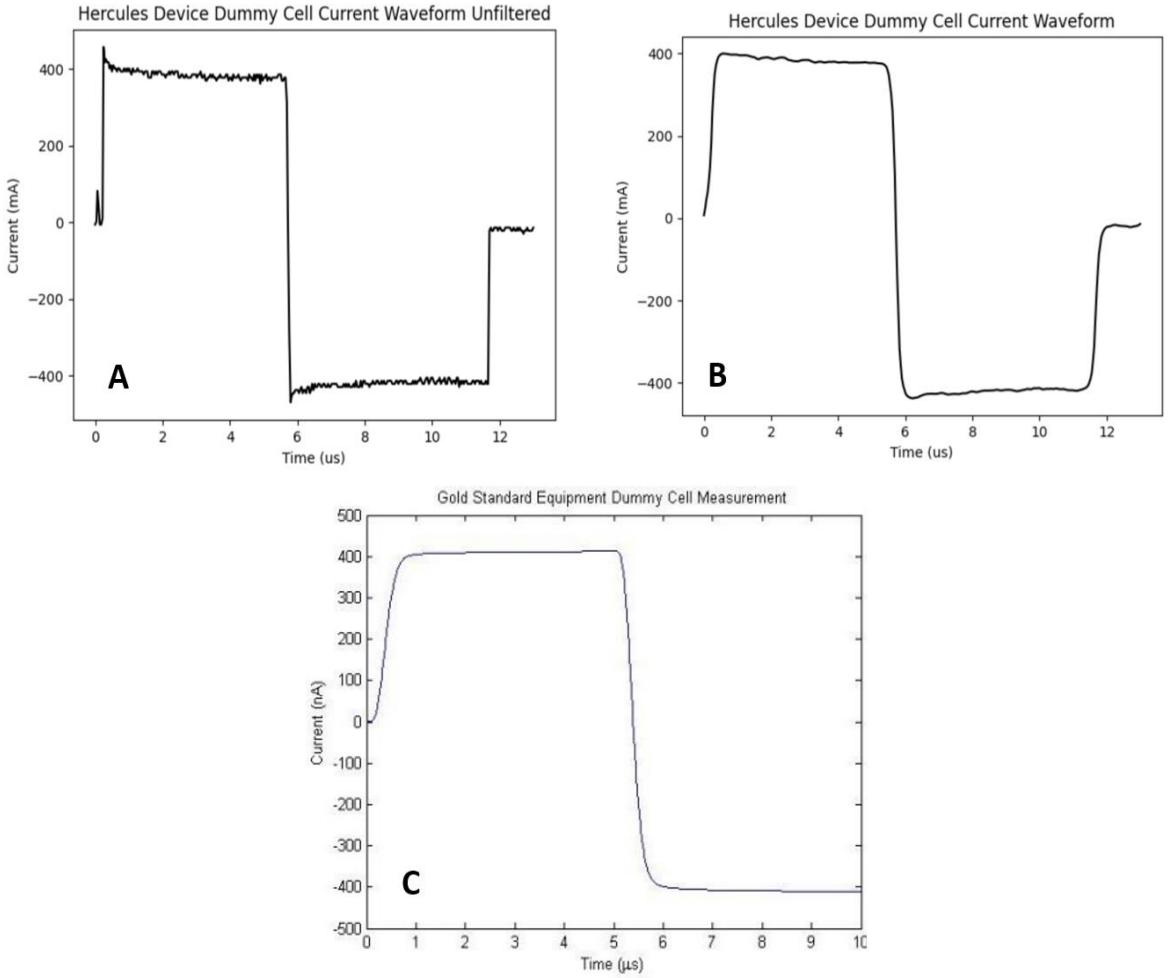


Figure 9. Current Waveform Outputs for Dummy Cell, (A) Hercules Device Unfiltered, (B) Hercules Device with Butterworth Filter, (C) Gold Standard [20]

All three measurements show the current steeply increasing from 0 to around 400nA, holding there for around 6µs, steeply decreasing to -400nA, and holding there for around 6µs. The measurement from the gold standard equipment held very steady at both the high and low ends, whereas the Hercules device was quite noisy in these areas. A digital lowpass Butterworth filter was used to combat some of this noise.

The price of all the components that were purchased to construct the device was tabulated so that the total cost could be calculated. The bill of materials can be found below in Table 1. The total cost to develop the device came to $108.

Table 1. Project Bill of Materials

Item	**Cost**
Hercules Development Kit	$50
Basic Electronics Kit	$15
Resistors Kit	$10
Op Amp Kit	$15
Capacitor Kit	$13
DC-DC Pos+Neg ±5V Reg	$5
Total	**$108**

For the sake of discussion, the cost of just the individual components used rather than the full cost of the electronics kits that were purchased for the development of the circuit was also

tabulated. This information can be found below in Table 2. The total cost to build the device as described in this paper is $61.29.

Table 2. Component Only Bill of Materials

Item	Cost
Hercules Development Kit	$50
LM2902n Op Amp	$0.43
10kΩ Resistor (X3)	$0.30
560 Ω Resistor	$0.10
220 Ω Resistor	$0.10
100 Ω Resistor	$0.10
2.5 kΩ Resistor	$0.64
2.2 nF ceramic Capacitor	$2.25
3 10 uF capacitor	$2.37
DC-DC Pos+Neg ±5V Reg	$5
Total	**$61.29**

Chapter 5 Discussion and Conclusion

Discussion

The Triangle wave produced met all the required parameters for FSCV. These parameters are: a period of 100ms, a rise time of 4.2ms, a resting voltage of -0.4V, and a peak voltage of 1.3V. The PWM signal produced by the board met the first two of these parameters once it was passed through the analog low pass filter. The voltage divider biasing circuit was then able to successfully shift the signal down to a resting voltage of -0.4V in order to achieve the final two parameters. The peak voltage this device was able to achieve was 1.32V, which varies just slightly from the ideal parameter. This 20mV error is likely due to simplicity of the circuitry and the nature of filtering a PWM signal into a triangle wave. Because FSCV of dopamine simply requires that dopamine's oxidation voltage of 0.6V and subsequently, its reduction voltage of -0.1V be reached within a certain timing, the maximum value is less critical and therefore the 20mV difference is acceptable. The key advantage of utilizing PWM to generate the triangle waveform as opposed to a DAC as seen in previous work is that the process requires nothing from the main CPU once the NHET is configured leaving all available processing power available for data collection and transmission to the PC. In addition, the PWM signal generation method remains highly configurable allowing for the possibility of configuring different triangle waves with ease within the code. A key disadvantage however is that due to the nature of variable duty cycle PWM there is an inherent tradeoff between noise and frequency: the higher the resolution (lower noise) the lower the possible output frequency. This disadvantage is acceptable given that the goal of the system isn't to achieve a perfect, noiseless signal, but to produce a passable signal at a low cost.

The Python script accurately calculated the current based on the data collected by the on-board analog to digital converter. The dummy cell measurement taken with the device matches the characteristic waveform seen with the gold standard equipment. This indicates that the device is capable of performing Fast Scan Cyclic Voltammetry for dopamine solutions. The discrepancies between the measurements taken with the Hercules device and the measurements taken with the gold standard equipment were to be expected. The simple, low cost, DIY nature of this project meant the resulting measurements would be noisier and less accurate than those taken on very expensive equipment. There are a couple of likely sources of this noise. The first is inherent noise within the ADC on the launchpad. This is likely a minor contributor to the noise. Another potential source would be noise introduced into the signal during waveform generation. It is the nature of PWM as a very high frequency signal of variable duty cycle which is then filtered into a much lower frequency signal to retain some of that high frequency component after the filtering process. The method utilized in the present work to mitigate the noise was the use of a digital lowpass Butterworth filter during the signal processing phase, but an analog lowpass filter prior to data collection by the ADC may also have been an effective noise reduction method to explore. All that said, this tradeoff between the quality of the measurement and the cost to perform the measurement is acceptable, provided that the information able to be obtained with the Hercules device is still accurate *enough* to give researchers an indication of what more expensive measurement techniques may reveal. Given the similarity between the measurements taken with both the gold standard equipment and the present device, it’s safe to say these measurements are indeed accurate enough.

The Device consists of just one microcontroller with total bill of materials for development coming in well under budget at $108. This means the project achieved the

objectives to simplify the device put forth by Jayson Foster, GVSU and to keep the device under $200. Additionally, if the device were to be built as specified in this paper, the need for the hardware ‘kits’ that were purchased would be eliminated and the total cost would be just $61.29.

Conclusion

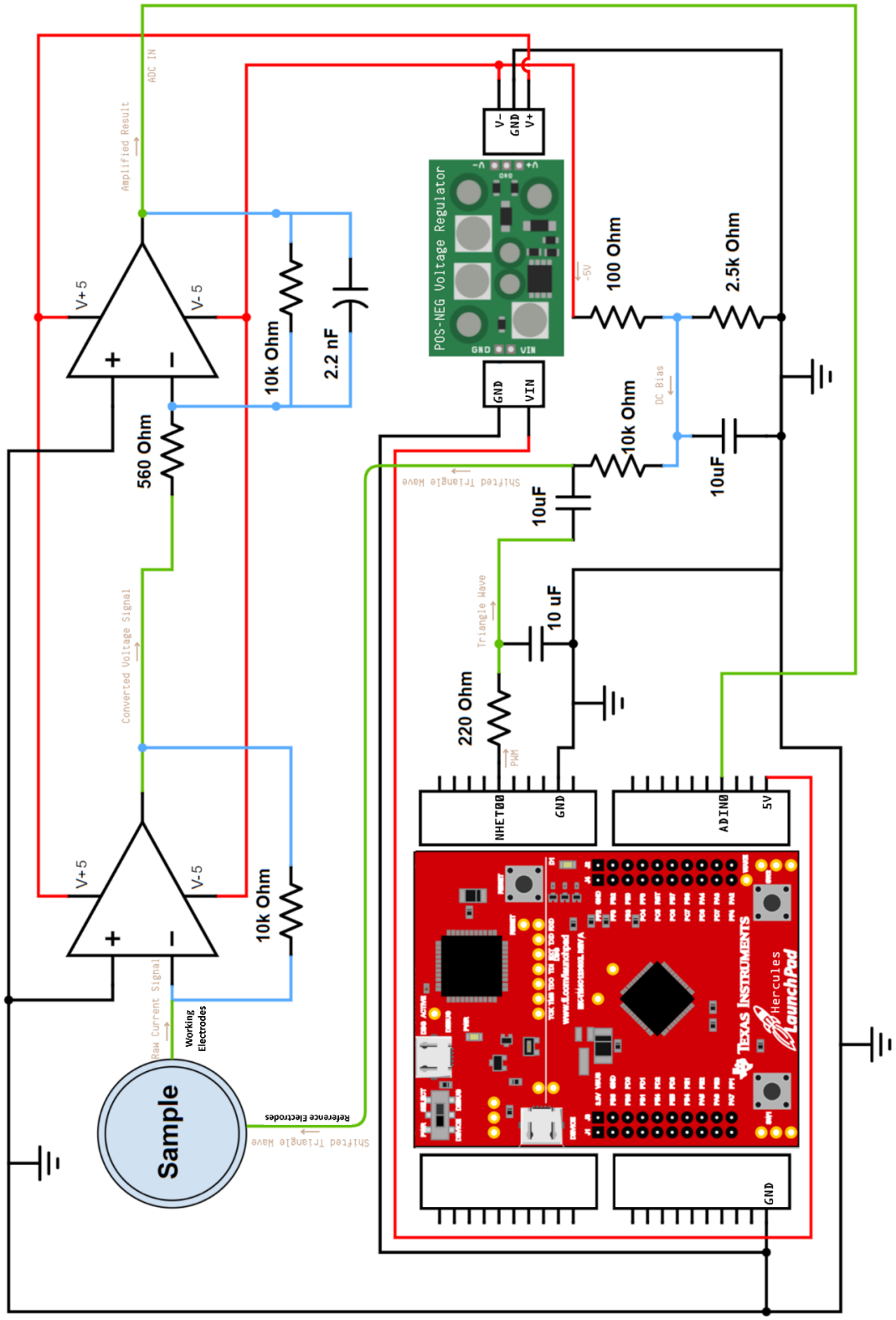
All of the objectives of this project were achieved. The device was able to successfully produce the required waveform and collect the results. This project offers proof of concept for the use of PWM in the generation of a FSCV triangle waveform for dopamine concentration measurement. By utilizing a PWM signal generated with a module independent from the main CPU to produce the triangle wave, the need for two microcontrollers, as seen in previous work, was eliminated, thus simplifying the device. All of the materials used in this project are readily available from online retailers, making this device highly accessible. The final cost to construct the device came to $108 making the project highly affordable.

Future Work

Due to COVID closures and personal constraints, it was not possible to test the system in a lab setting with dopamine solutions. A critical next step for this work would be to verify its effectiveness in lab use. Additionally, the current process requires the user to take the output from the microcontroller and manually run it through the python script in order to generate the output. An important next step would require the development of an app that’s capable of initiating the microcontroller, starting a cycle, and then automatically processing the data with the python algorithm. This would greatly improve the usability of the device. Finally, with further development, the device could be made compatible with a variety of electrode configurations and be made capable of producing multiple waveforms to allow the measurement of different neurotransmitters, such as serotonin, thus making the device more flexible.

Appendices

Appendix I – Schematic



Appendix II – Hercules Microcontroller Code

High Level HET code

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;
; This example code is to be loaded into N2HET1's RAM. This code will generate a
; time-varying PWM signal to render either triangle waves or trapezoid waves.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;
; PWM frequency to be generated
PWM_PERIOD .equ 2
; The pin number that will output the PWM signal
PWM_PIN_NUM .equ 9
; The initial maximum duty cycle (LR compare value) to be generated.
INIT_COMPARE .equ 3
; The initial maximum duty cycle (HR compare value) to be generated.
INIT_COMPARE_HR .equ 0
; amount of increment in terms of LRP. Note the total amount to increment is
; equal to DUTY_INCREMENT + DUTY_INCREMENT_HR.
DUTY_INCREMEMT .equ 0
; amount of increment in terms of HR.
DUTY_INCREMEMT_HR .equ 1
; amount of decrement in terms of LRP
DUTY_DECREMEMT .equ 0
; amount of decrement in terms of HR
DUTY_DECREMEMT_HR .equ 1
; amount of timer delay to keep the PWM at 0% duty cycle
LOW_DELAY .equ 0
; amount of timer delay to keep the PWM at maximum duty cycle
HIGH_DELAY .equ 0
; key to unlock N2HET
UNLOCK_KEY .equ 0xA
; The data field of the MOV32 instruction contains an initial value (0x5) that
; is not equal to the key to unlock the N2HET program. First the MOV32
; instruction moves the initial value to a temporary register T
L00 MOV32 { remote=DUMMY,type=IMTOREG,reg=T,data=0x5};
; Compare the register T value with the key to unlock N2HET. The key to unlock
; is 0xA. If the key is not matched then go back to L00. The CPU is supposed
; to write the proper key (0xA) to unlock the N2HET
L01 ECMP { next=L00,hr_lr=LOW,cond_addr=L02,pin=0,reg=T,data=UNLOCK_KEY};
; Creating a virtual counter using CNT which will determine the period of
; the PWM to be generated. The initial small max count allows for quick
; simulation which can later be changed by the host CPU.
L02 CNT { reg=A,irq=OFF,max=PWM_PERIOD};
; Use ECMP to determine the duty cycle of the PWM on the specified pin. The
; pin field and the duty cycle are changeable by the CPU.
L03 ECMP { hr_lr=HIGH,en_pin_action=ON,cond_addr=L04,pin=PWM_PIN_NUM,

action=PULSELO,reg=A,irq=OFF,data=0,hr_data=0};
; Only when the CNT reaches the max count will the program go to the
; conditional address. We want to wait for one complete PWM waveform to be
; generated before changing the duty cycle. When CNT reaches the max
; value it will set the Z flag.
```

```
L04 BR { next=L00,cond_addr=L05,event=Z};
; the data field in this ADD acts as a up/down flag. To create either a triangle
; or a trapezoid wave we first need to create a ramp up waveform. The PWM will first
; increase the duty cycle until it reaches the specified maximum duty cycle before
; it starts to decrease the duty or stay at the maximum duty. The up/down flag is
; used to create two different paths in the flow to alternate before increasing duty
; cycle vs decreasing duty cycle.
L05 ADD { src1=ZERO,src2=ZERO,dest=NONE,data=0};
; Move the up/down flag to a temp register T.
L06 MOV32 { remote=L05,type=REMTOREG,reg=T};
; Compare this up/down flag to 0. 0 means to increase the duty cycle and 1
; means to decrease the duty cycle.
L07 ECMP { next=L16,cond_addr=L08,pin=0,reg=T,data=0};
; move the ECMP DF which contains the compare value for duty cycle creation
; to register R
L08 MOV32 { remote=L03,type=REMTOREG,reg=R};
; Subtract the current compare value from the max duty cycle stored in
; REM_DUTY. The result will be stored in register S.
L09 SUB { src1=REM,src2=R,dest=S,remote=REM_DUTY,data=0};
; If the subtraction result is more than 0 then it means it has not
; reached the max duty cycle we will increase the duty cycle. If it is
; zero or less than zero then we have reached the max duty cycle and we
; will change the up/down flag to down position.
L10 BR { next=L12,cond_addr=L11,event=GT};
; Add specified amount to the existing compare value (duty cycle) to specify the new
; duty cycle. The amount to increment is changeable by CPU before the N2HET program
starts.
; After the addition, jump back to the beginning of the program
L11 ADD { next=L15,src1=R,src2=IMM,dest=S,rdest=REM,remote=L03,data=DUTY_INCREMEMT,

hr_data=DUTY_INCREMEMT_HR};

; Insert a timer delay after the maximum duty cycle is reached. A timer delay here
has the
; effect of creating the high side of a trapezoid waveform. If the timer is zero then
it
; becomes a triangle wave.
L12 DJZ { next=L00,cond_addr=L13,reg=NONE,data=HIGH_DELAY};
; After the above DJZ expires on its counter we need to reload the DJZ counter to the
; specified amount of delay.
L13 MOV32 { next=L14,remote=L12,type=IMTOREG&REM,reg=NONE,data=HIGH_DELAY};
; Now change the up/down flag to down by moving a 1 to the up/down flag
L14 MOV32 { remote=L05,type=IMTOREG&REM,reg=NONE,data=1};
; Branch to the beginning
L15 BR { next=L00,cond_addr=L00,event=NOCOND};
; move the ECMP DF to register R which contains the current compare value
; (duty cycle)
L16 MOV32 { remote=L03,type=REMTOREG,reg=R};
; Subtract the current duty cycle by the specified amount. This amount of decrement
is
; changeable by CPU before the N2HET program starts. When this instruction is
executed
; the first time, the current duty cycle is at the maximum duty cycle. Here we are
creating
; the ramp down part of the triangle/trapezoid waveforms.
```

```
L17 SUB { src1=R,src2=IMM,dest=S,rdest=NONE,data=DUTY_DECREMEMT,

hr_data=DUTY_DECREMEMT_HR};

; As long as the subtraction result is greater than zero, we will keep
; decreasing the duty cyle or otherwise we will again change the up/down
; flag to up position. The destination register is S which contains the
; subtraction result.
L18 BR { next=L19,cond_addr=L20,event=N};
; Move the subtraction result to the ECMP DF as the new duty cycle
L19 MOV32 { next=L00,remote=L03,type=REGTOREM,reg=S};
; Insert a timer delay after the 0% duty cycle is reached. A timer delay here has the
; effect of creating the low side of a trapezoid waveform. If the timer is zero then
it
; becomes a triangle wave.
L20 DJZ { next=L00,cond_addr=L21,reg=NONE,data=LOW_DELAY};
; Reset the increment delay to the specified amount.
L21 MOV32 {remote=L20,type=IMTOREG&REM,reg=NONE,data=LOW_DELAY};
; Move the value 0 to the up/down flag so in the next LRP the program
; flow will execute the path to increase duty cycle.
L22 MOV32 { remote=L05,type=IMTOREG&REM,reg=NONE,data=0};
; Branch to beginning
L23 BR { next=L00,cond_addr=L00,event=NOCOND};
; REM_DUTY data field stores the maximum duty cycle the PWM to be generated.
; The host CPU can change this value.
REM_DUTY ECMP { next=REM_DUTY,cond_addr=REM_DUTY,pin=0,reg=A,data=INIT_COMPARE,

hr_data=INIT_COMPARE_HR};

DUMMY BR { next=DUMMY,cond_addr=DUMMY,event=NOCOND,irq=OFF};
```

main.c

```
/** @file sys_main.c
*   @brief Application main file
*   @date 16.Feb.2015
*   @version 04.03.00
*
*   This file contains an empty main function,
*   which can be used for the application.
*/

/*
* Copyright (C) 2009-2015 Texas Instruments Incorporated - www.ti.com
*
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*    Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
```

```
*
*    Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the
*    distribution.
*
*    Neither the name of Texas Instruments Incorporated nor the names of
*    its contributors may be used to endorse or promote products derived
*    from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
*  "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
*  LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
*  A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
*  OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
*  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
*  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
*  OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*
*/


/* USER CODE BEGIN (0) */
#include "sci.h"
#include "adc.h"
#include "stdlib.h"
#include "sys_core.h"
#include "Trapezoid_Wave.h"
#include "het.h"
#include "system.h"
void calculate_ecmp_compare();
void configNHET1();
/* USER CODE END */

/* Include Files */

#include "sys_common.h"

/* USER CODE BEGIN (1) */
/*****************************************************************************
 * Below 8 macros are changeable by user to generate various different      *
 * waveforms.                                                               *
 *****************************************************************************/

/* PWM_PERIOD defines the period of the carrier PWM frequency. PWM_PERIOD is
 * expressed in terms of number of LRP (Loop Resolution Period). The frequency
 * of the carrier PWM will remain constant. However, the N2HET will
 * modulate the duty cycle to generate a time-varying PWM signal. This
 * time-varying PWM signal is then passed through a low pass filter to remove
 * unwanted high frequency components, an analog waveform is thus rendered.
 *
 * The LRP period depends on the reference VCLK2 frequency to the N2HET
```

```
 * module as well as the programmable LR Prescaler factor (lr). It can be
 * expressed as:
 * 1 LRP = VCLK2 * lr
 * If VCLK2 = 11.11ns and lr = 128 then
 * 1 LRP = 11.11 ns * 128 = 1.42us
 * Note that for this example, the high resolution prescale factor (hr)
 * is always confed to 1.
 *
 * The width of the PWM_PERIOD also defines the resolution of the output
 * analog signal to be rendered. With PWM_PERIOD = 1, there are a total
 * of 128 VCLK2 cycles. This means that the N2HET will module its duty
 * cycle between these 128 steps. Thefore, with PWM_PERIOD = 1, the
 * output analog signal will have a log2(128) = 8-bit resolution. If
 * user desires higher resolution, the PWM_PERIOD can be changed to a
 * higher value such as 32. With PWM_PERIOD = 32, the duty cycle can be
 * modulated among a total of 32 * 128 = 4096 steps and hence render a
 * log2(4096) = 12-bit of resolution. The higher the resolution the
 * less noise on the final output. However, the higher the resolution,
 * the lower the final output frequency of the signal to be generated.
 */
#define PWM_PERIOD     32

/* PWM_DUTY defines the maximum duty cycle the N2HET is allowed to
 * modulate to. The PWM_DUTY is expressed in terms of percent (%).
 * With PWM_DUTY = 100%, the maximum amplitude on the output signal
 * will be equal to Vccio. Setting PWM_DUTY = 50% will create a
 * maximum amplitude of Vccio / 2. Note that the maximum amplitude
 * will also depend on the type of filter used and its respective
 * RC values. */
#define PWM_DUTY      77.5

/* allowable LR Prescaler factors are 32, 64 and 128. Anything less
 * than 32 will not have enough time slots for the N2HET program to
 * run.

  LRPFC can be either 5, 6 or 7.
 * 7 -> one lr = 128 VCLK2
 * 6 -> one lr = 64 VCLK2
 * 5 -> one lr = 32 VCLK2
 */
#define LRPFC 6

/* The NH2ET1 program will automatically increase the PWM
 * modulation from 0% duty cycle to maximum duty cycle
 * specified in PWM_DUTY. When PWM_DUTY is reached it starts
 * to decrease the duty cycle from PWM_DUTY to 0%.
 * DUTY_INCREMENT specifies the delta amount of duty cycle to
 * change from one duty cycle to the next duty cycle while
 * the duty cycle is increasing. Increasing DUTY_INCREMENT
 * has the effect of decreasing the duty cycle resolution for
 * a given PWM_PERIOD. This is expressed in terms of (%).
 * For example specifying DUTY_INCREMENT equal to 5 will mean
 * the duty cycle will start at 0% and the next duty cycle
 * will be 5% at a 5% increment. If 0 is specified, then the
 * N2HET1 will increment the duty cycle at 1 VCLK2 clock resolution */
```

```
#define DUTY_INCREMENT   20

/* DUTY_DECREMEMT specifies the delta amount of duty cycle to
 * change from one duty cycle to the next duty cycle while
 * the duty cycle is decreasing. This is expressed in terms
 * of (%). */
#define DUTY_DECREMENT   20

/* The MAX_DUTY_TIMER is the amount of wait time to stay at
 * the maximum duty cycle. Change to a non-zero value will
 * create trapezoid waveform. A zero value will create a
 * triangle wave
 *
 * The MAX_DUTY_TIMER is expressed in terms of number of
 * LRP. */
#define MAX_DUTY_TIMER  0

/* The MIN_DUTY_TIMER is the amount of wait time
 * to stay at the minimum duty cycle. Change to
 * a non-zero value will create trapezoid waveform.
 * A zero value will create a triangle wave */
#define MIN_DUTY_TIMER  4480

/* Pin number in N2HET1 to generate the PWM. */
#define NHET1_PIN_PWM PIN_HET_0


/*************************************************************************/
/* Below macros are for internal calculation. Do not change.             */
/*************************************************************************/

/* The PWM Period to be loaded to N2HET1 CNT instruction */
#define CNT_MAX_PERIOD (PWM_PERIOD - 1)

/* The max PWM Duty to be loaded to N2HET1 ECMP Instruction */
#define ECMP_MAX_DUTY (PWM_PERIOD * PWM_DUTY / 100.0)

/* DELTA_INCREMENT is the amount to be loaded to N2HET1 to increment the
 * duty cycle */
#define DELTA_INCREMENT (CNT_MAX_PERIOD * DUTY_INCREMENT / 100)

/* DELTA_DECREMENT is amount to be loaded to N2HET1 to decrement the
 * duty cycle */
#define DELTA_DECREMENT (CNT_MAX_PERIOD * DUTY_DECREMENT / 100)


/* Unlock key to for N2HET2 */
#define UNLOCK_KEY 0xAU

unsigned char command[8];
/* USER CODE END */

/** @fn void main(void)
*   @brief Application main function
*   @note This function is empty by default.
```

```
*
*   This function is called after startup.
*   The user can use this function to implement the application.
*/

/* USER CODE BEGIN (2) */
uint32 ecmp_compare_value = 0;
/* USER CODE END */

void main(void)
{
/* USER CODE BEGIN (3) */
       /* This example uses the N2HET1 to generate either a triangle wave
        * or a trapozoid wave by modulating the PWM duty cycle.
        * N2HET1 program: Trapezoid_Wave.het
        */

    /* initialize N2HET1 based on HalCoGen settings */
       hetInit();

       /* Configure additional settings of N2HET1 based on the macros settings */
       configNHET1();


       adcData_t adc_data; //ADC data structure
       adcData_t *adc_data_ptr = &adc_data; //ADC data pointer
       unsigned int NumberOfChars, value; //Declare variables

       sciInit(); //Initializes the SCI (UART) module
       adcInit(); //Initializes the ADC module

       while(1)
       {
       adcStartConversion(adcREG1, adcGROUP1); //Start ADC conversion
       while(!adcIsConversionComplete(adcREG1, adcGROUP1)); //Wait for ADC conversion
       adcGetData(adcREG1, 10, adc_data_ptr); //Store conversion into ADC pointer
       value = (unsigned_int)adc_data_ptr->value;
       NumberOfChars = ltoa(value,(char*)command);
       sciSend(sci1inREG, 2, (unsigned char *)"0x"); //Sends hex designation
       sciSend(sci1inREG, NumberOfChars, command); //Sends the data
       sciSend(sci1inREG, 2, (unsigned char *)"\r\n"); //Sends new line character
       }

/* USER CODE END */
}

/* USER CODE BEGIN (4) */

/* this function is to configure additional settings for the N2HET1 */
void configNHET1()
{

       /* calculate_ecmp_compare() will calculate the compare value as well as
        * the high resolution delay values to be loaded into the N2HET1 ECMP1
        * instruction for PWM Duty generation based on the input ECMP_MAX_DUTY */
```

```
	calculate_ecmp_compare();

	/* Set NHET1_PIN_PWM to output */
	hetREG1->DIR = 1 << NHET1_PIN_PWM;

	/* Change the LRPFC according to user input */
	hetREG1->PFR = (LRPFC << 8);

	/* Initialize the PWM period and duty cycle based on the defined parameters */
	hetRAM1->Instruction[pHET_L02_0].Control = (uint32)CNT_MAX_PERIOD |
                                                   (hetRAM1-
>Instruction[pHET_L02_0].Control & 0xFD0000);
	hetRAM1->Instruction[pHET_REM_DUTY_0].Data = ecmp_compare_value;

	/* Configure the N2HET1 pin to output the PWM */
	hetRAM1->Instruction[pHET_L03_0].Control = (hetRAM1-
>Instruction[pHET_L03_0].Control & 0xFFFFE0FF) |
                                                     (NHET1_PIN_PWM << 8);

	/* Configure the amount of delay to stay at the maximum duty cycle */
	hetRAM1->Instruction[12].Data = ((uint32)MAX_DUTY_TIMER << 7);
	hetRAM1->Instruction[13].Data = ((uint32)MAX_DUTY_TIMER << 7);

	/* Configure the amount of delay to stay at the minumum duty cycle */
	hetRAM1->Instruction[20].Data = ((uint32)MIN_DUTY_TIMER << 7);
	hetRAM1->Instruction[21].Data = ((uint32)MIN_DUTY_TIMER << 7);

	/* Duty cycle increment amount */
	if ( (uint32)DELTA_INCREMENT == 0){
		hetRAM1->Instruction[pHET_L11_0].Data = 1 << (7 - LRPFC);
	} else {
		hetRAM1->Instruction[pHET_L11_0].Data = (uint32)DELTA_INCREMENT << (7 -
LRPFC);
	}

	/* Duty cycle decrement amount */
	if ((uint32)DELTA_DECREMENT == 0) {

		hetRAM1->Instruction[pHET_L17_0].Data = 1 << (7 - LRPFC);
	} else {
		hetRAM1->Instruction[pHET_L17_0].Data = (uint32)DELTA_DECREMENT << (7 -
LRPFC);
	}

	/* Unlock the N2HET program. Initially after reset the N2HET program is locked
*/
	hetRAM1->Instruction[pHET_L00_0].Data = UNLOCK_KEY << 7;

}
/* This function calculates the compare value and the high resolution delay
 * to be loaded into the N2HET1 ECMP instruction for generating the PWM
 * DUTY cycle according to the user supplied % in PWM_DUTY
 */
void calculate_ecmp_compare()
{
```

```
        uint32 uint32_high_phase_width;
        float float_high_phase_width;

        float_high_phase_width = ECMP_MAX_DUTY;
        uint32_high_phase_width = ECMP_MAX_DUTY;

        /* the (float_high_phase_width - uint32_high_phase_width) expression will
         * obtain the decimal value of one LRP (loop resolution period)
         * clock. Once we get the deciminal value we will multiply by 128 to
         * obtain the number of HR (high resolution) clocks. 1 LRP = 128 HR as
         * configured in the HalcoGen for this example.
         *
         * (uint32_high_phase_width << 7) will form the LRP compare value
         * ((float_high_phase_width - uint32_high_phase_width) * 128) forms the
         * high resolution delay
         */
        ecmp_compare_value = (uint32)(uint32_high_phase_width << 7) |
                                    (uint32)((float_high_phase_width -
uint32_high_phase_width) * 128);
}


/* USER CODE END */
```

Python Script

```
import csv
import matplotlib.pyplot as plot
import numpy as np
from scipy.signal import butter,filtfilt


with open('CurrentData.csv', 'r') as file:
    data = list(csv.reader(file))
#print(data)

xchar = [i[0] for i in data]
ychar = [i[1] for i in data]
x = [eval(i) for i in xchar]
y = [eval(i) for i in ychar]
#print(y)

y_current = y * 0.0000056 # Calculate Current

# Filter requirements.
T = 12.0         # Sample Period
fs = 30.0       # sample rate, Hz
cutoff = 2      # desired cutoff frequency of the filter, Hz
nyq = 0.5 * fs  # Nyquist Frequency
order = 1
n = int(T * fs) # total number of samples
```

```
def butter_lowpass_filter(y, cutoff, fs, order):
    normal_cutoff = cutoff / nyq
    # Get the filter coefficients
    b, a = butter(order, normal_cutoff, btype='low', analog=False)
    ynew = filtfilt(b, a, y)
    return ynew


ynew = butter_lowpass_filter(y_current, cutoff, fs, order)
plot.plot(x, ynew, color='black')
plot.xlabel('Time (us)')
plot.ylabel('Current (nA)')
plot.title('Hercules Device Dummy Cell Current Waveform Unfiltered')

plot.show()
```

Appendix III – Data Sheets

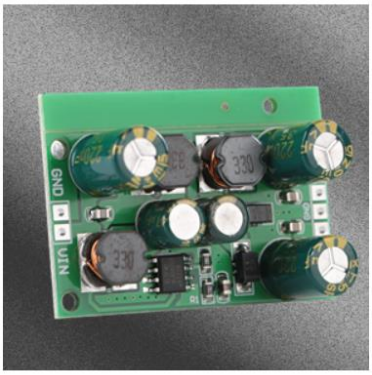
Q & A:

Q1 : Why output voltage is less than the nominal voltage

A1: Input power supply power is too low.Test the input voltage with a multimeter,a t this time of the input voltage is very low

Q2 : Why is the negative voltage not output or the output too low?

A2 : The negative voltage cannot be used alone. When using a negative voltage, the same load must be connected at the positive voltage terminal.



Specification:

Input voltage: 3~24V

Output voltage: ±5V/±6V/±9V/±10V/±12V/±15V/±18V/±24V (Optional)

Maximum output power: 1-8W

Conversion efficiency :70-90%

Quiescent current: 3-4mA

Accuracy : Positive voltage ±3%, Negative voltage ±5%

Working frequency: 400kHz

Operating temperature: -40~85

Size: 42×24×15mm / 1.65×0.94×0.59in Weight: 12g (approx.)

Package List: 1 × Positive Negative Voltage Converter

Application Report
SPNA220–May 2015

Triangle/Trapezoid Wave Generation Using PWM With Hercules™ N2HET

Charles Tsai

ABSTRACT

This application report illustrates how to generate various forms of triangle and trapezoid waves using the versatile programmable high-end timer (N2HET). The examples can be run in either the Hercules hardware development kit (HDK) or the LaunchPad™ development kit. The application report shows the N2HET program examples, the steps to setting up the N2HET registers as well as basic system settings utilizing the HalCoGen.

This document assumes that you have some basic understanding of the N2HET terms as well as some understanding of both the HET integrated development environment (IDE) and HalCoGen tools.

Project collateral and source code discussed in this application can be downloaded from the following URL: http://www.ti.com/lit/zip/spna220.

Contents

List of Figures

List of Tables

1 Introduction

N2HET is a fifth-generation Texas Instruments (TI) advanced intelligent timer coprocessor module based on the very long instruction word (VLIW) instruction set architecture. The instruction set, based mostly on very simple, but comprehensive instructions provides sophisticated timing functions for real-time applications. The high resolution hardware channels allow greater accuracy for widely used timing functions such as period and pulse measurements, output compare and PWMs.

Pulse Width Modulation (PWM) is a method of encoding a voltage onto a fixed frequency carrier wave. The frequency of the PWM will be fixed while the duty cycle will vary between 0% and 100%. The percentage of the on-time will be proportional to the output signal voltage. For example, a 0% duty cycle produces a 0 V output while a 100% duty cycle produces a peak-to-peak voltage V_{P-P} equal to the Vccio, which is the I/O power supply voltage to the microcontroller. The nominal Vccio is 3.3 V in Hercules microcontrollers. A 50% duty cycle would have produced an output voltage equal to 1.65 V. The PWM method is a low cost way of implementing a digital-to-analog converter (DAC). By time-varying the duty cycle percentage, it is possible to generate an arbitrary analog waveform.

Using the PWM method as a DAC is nothing new. Many microcontrollers on the market can accomplish this simple task. What is needed is a PWM capable hardware in the microcontrollers. This normally involves a simple time-based counter that will either count up or count down until the counter expires. The programmable width of the counter will determines the frequency of the PWM. While the counter is either counting up or down, the current counter value is compared against a programmable match value that defines the duty cycle. The output PWM signal could be setup to remain set while a match is not found. As soon as the counter value matches the programmable match value, the PWM will clear the PWM output. The pin remains clear until the counter expires and sets the pin again. Figure 1 illustrates the generation of an un-modulated PWM signal.

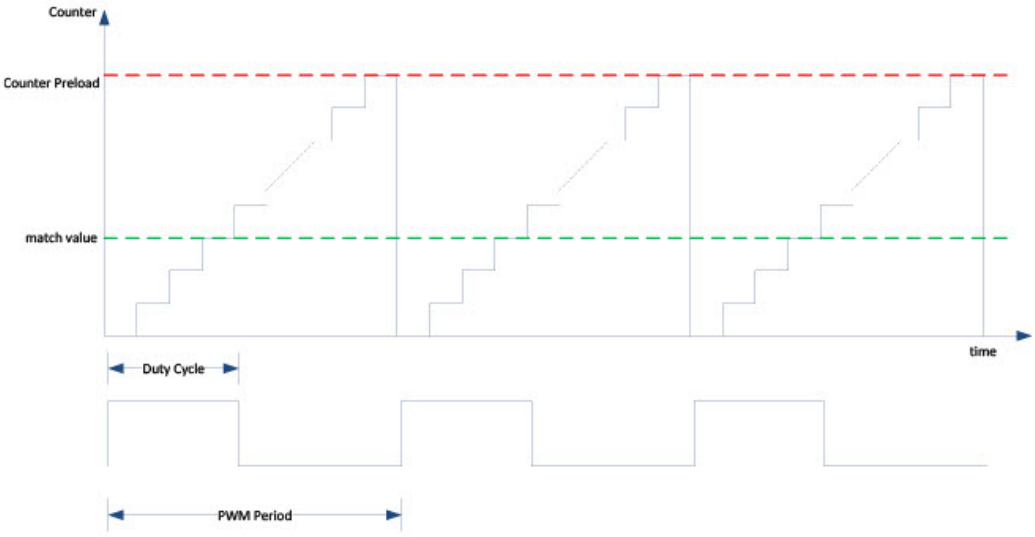


Figure 1. An Unmodulated PWM Signal

However, for a time-vary PWM output, the duty cycle needs to be changed. This normally involves generating an interrupt to the CPU upon a compare match or when the counter reaches its pre-loaded count. For example, to generate a ramp waveform, this could mean generating an interrupt to the CPU each time the counter expires at its programmable pre-load value. In the interrupt service routine, the CPU increments the duty cycle by a desire amount and updates the compare value register, see Figure 2. Generating interrupts to the CPU has some drawbacks; it can slow down the CPU from doing other tasks. The interrupt latency can also impact the maximum achievable output signal frequency.

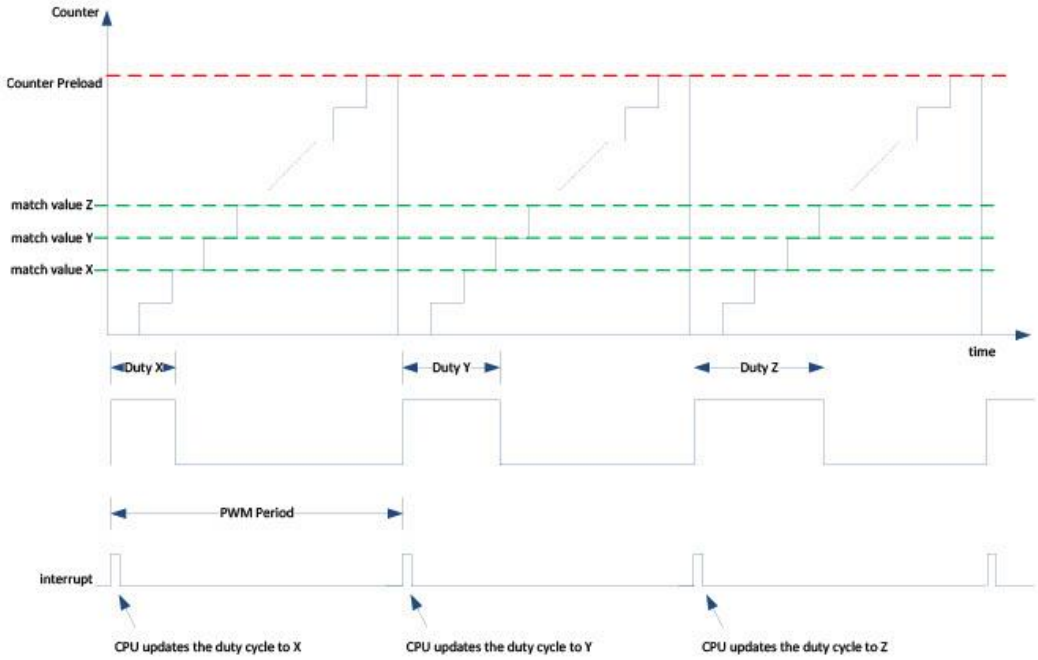


Figure 2. A Modulated PWM Signal

On the other hand, some of the mentioned drawbacks can be avoided using the N2HET since it is a coprocessor. You can write N2HET code to modulate the duty cycle of the PWM without involving the CPU at all. How and when to modulate the duty cycle is under the N2HET's control. This capability is the main focus of this application note.

In this application note, we will use the N2HET to generate time-varying PWM signals. The output signal will go through a simple analog low-pass filter to remove the high frequency components. The filter output renders various forms of triangle and trapezoid waves. The versatility of the N2HET allows you to control the slope of the ramp up and ramp down when forming a triangle. By controlling the amount of time the PWM stays at its maximum duty cycle and at 0% duty, an arbitrary trapezoid waveform can also be generated. Figure 3 shows the waveforms that are demonstrated in this application report.

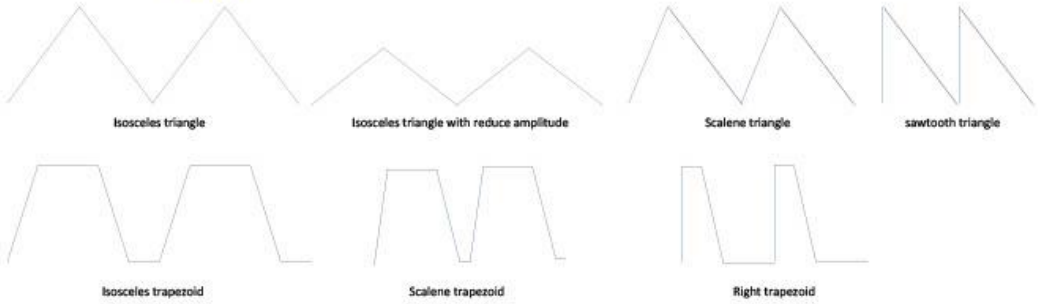


Figure 3. Triangle and Trapezoid Waves

2 Low-Pass Filter

The low-pass filter is used to remove the high frequency components that are produced on the PWM output. For simplicity reason, a first order passive low-pass filter using only the low-cost RC components are used, as shown in Figure 4. The focus of this application report is to show how to write the N2HET code to produce various waveforms. This document does not touch on the optimum filter to use. Extra information on the analysis of filters is provided in Section 7.

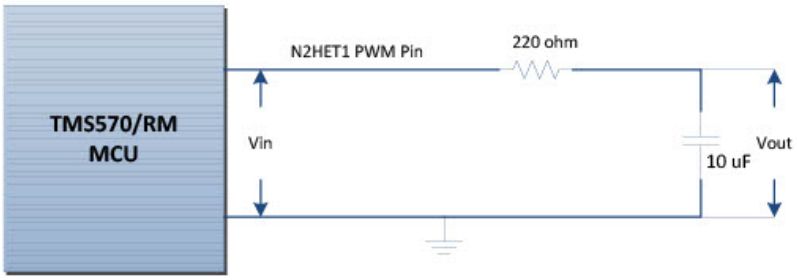


Figure 4. Low-Pass Filter

3 N2HET Implementation

3.1 N2HET1 Triangle/Trapezoid Wave Generation Flow Chart

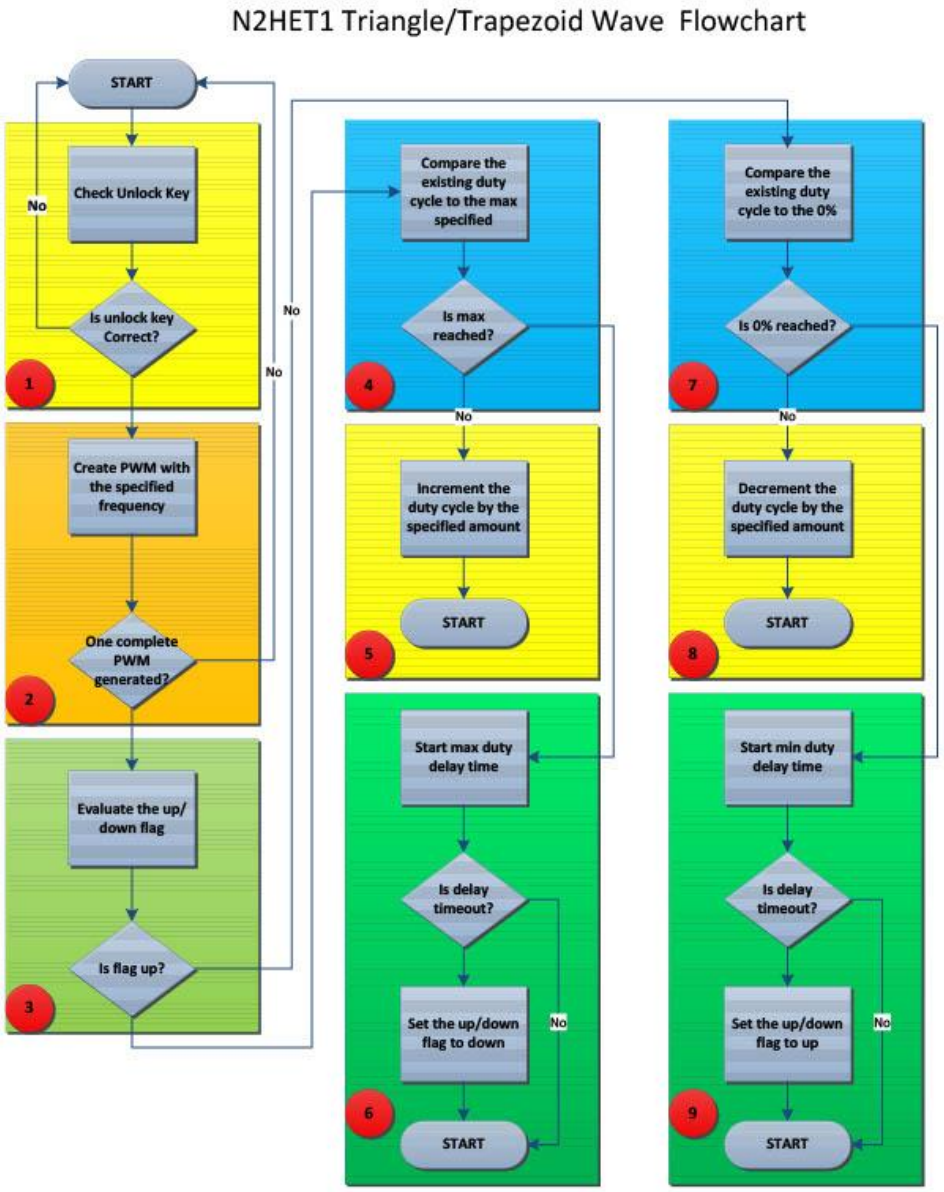


Figure 5. N2HET1 Triangle/Trapezoid Wave Flow Chart

1. The N2HET1 is put into a self-loop until a proper unlock key is written to the N2HET1. While the N2HET1 is still locked, the host CPU can setup various parameters for the N2HET1 program.
2. Generate the specified PWM starting with 0% duty cycle. The duty cycle will be self modified by the N2HET1 itself either in increasing order or decreasing order. Wait until one full PWM period is generated before changing to a new duty cycle.
3. Evaluate the up/down flag to determine whether the duty cycle should increase or decrease. The flag will be initialized to zero after reset, meaning to increment the duty cycle.
4. Compare the existing duty cycle to the programmable maximum duty cycle. The maximum duty cycle is a parameter changeable by the host CPU before the N2HET program starts. If the maximum is reached, go to step 6.
5. Increment to the next duty cycle percentage by the amount that is programmable by the host CPU before the N2HET program starts.
6. Start a programmable timer. The timer is used to hold the PWM at the maximum duty cycle for a programmable amount of time. This step is needed for creating trapezoid waveforms. To generate a triangle wave, this timer delay will be set to zero. The timer length is programmable by the host CPU before the N2HET program starts.
7. Compare the existing duty cycle to the 0% duty cycle. If the 0% is reached, go to step 9.
8. Decrement to the next duty cycle by the amount that is programmable by the host CPU. Note that the amount to decrement can be different than the amount to increment.
9. Start a programmable timer. The timer is used to hold the PWM at the 0% duty cycle for a programmable amount of time. Note that this minimum duty cycle timer length can be different from the maximum duty cycle timer length. This step is needed for creating trapezoid waveforms. To generate a triangle wave this timer delay will be set to zero. The timer length is programmable by the host CPU before the N2HET program starts. *Delay Between Cycles

3.2 N2HET1 Triangle/Trapezoid Wave Program

The example N2HET1 program code is illustrated below. Directives using .equ are parameters used to configure the program; you can change these parameters. By default, these parameters have initial values that are small for quick simulation using HET IDE. The host CPU will overwrite these parameters in the host side application code.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; This example code is to be loaded into N2HET1's RAM. This code will generate a
; time-varying PWM signal to render either triangle waves or trapezoid waves.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; PWM frequency to be generated
PWM_PERIOD              .equ 2
; The pin number that will output the PWM signal
PWM_PIN_NUM             .equ 9
; The initial maximum duty cycle (LR compare value) to be generated.
INIT_COMPARE            .equ 3
; The initial maximum duty cycle (HR compare value) to be generated.
INIT_COMPARE_HR         .equ 0
; amount of increment in terms of LRP. Note the total amount to increment is
; equal to DUTY_INCREMENT + DUTY_INCREMENT_HR.
DUTY_INCREMEMT          .equ 0
; amount of increment in terms of HR.
DUTY_INCREMEMT_HR       .equ 1
; amount of decrement in terms of LRP
DUTY_DECREMEMT          .equ 0
; amount of decrement in terms of HR
DUTY_DECREMEMT_HR       .equ 1
; amount of timer delay to keep the PWM at 0% duty cycle
LOW_DELAY               .equ 0
; amount of timer delay to keep the PWM at maximum duty cycle
HIGH_DELAY              .equ 0
; key to unlock N2HET
UNLOCK_KEY              .equ 0xA
```

```
; The data field of the MOV32 instruction contains an initial value (0x5) that
; is not equal to the key to unlock the N2HET program. First the MOV32
; instruction moves the initial value to a temporary register T
L00   MOV32 { remote=DUMMY,type=IMTOREG,reg=T,data=0x5};

; Compare the register T value with the key to unlock N2HET. The key to unlock
; is 0xA. If the key is not matched then go back to L00. The CPU is supposed
; to write the proper key (0xA) to unlock the N2HET
L01   ECMP { next=L00,hr_lr=LOW,cond_addr=L02,pin=0,reg=T,data=UNLOCK_KEY};

; Creating a virtual counter using CNT which will determine the period of
; the PWM to be generated. The initial small max count allows for quick
; simulation which can later be changed by the host CPU.
L02   CNT { reg=A,irq=OFF,max=PWM_PERIOD};

; Use ECMP to determine the duty cycle of the PWM on the specified pin. The
; pin field and the duty cycle are changeable by the CPU.
L03   ECMP { hr_lr=HIGH,en_pin_action=ON,cond_addr=L04,pin=PWM_PIN_NUM,
             action=PULSELO,reg=A,irq=OFF,data=0,hr_data=0};

; Only when the CNT reaches the max count will the program go to the
; conditional address. We want to wait for one complete PWM waveform to be
; generated before changing the duty cycle. When CNT reaches the max
; value it will set the Z flag.
L04   BR { next=L00,cond_addr=L05,event=Z};

; the data field in this ADD acts as a up/down flag. To create either a triangle
; or a trapezoid wave we first need to create a ramp up waveform. The PWM will first
; increase the duty cycle until it reaches the specified maximum duty cycle before
; it starts to decrease the duty or stay at the maximum duty. The up/down flag is
; used to create two different paths in the flow to alternate before increasing duty
; cycle vs decreasing duty cycle.
L05   ADD { src1=ZERO,src2=ZERO,dest=NONE,data=0};

; Move the up/down flag to a temp register T.
L06   MOV32 { remote=L05,type=REMTOREG,reg=T};

; Compare this up/down flag to 0. 0 means to increase the duty cycle and 1
; means to decrease the duty cycle.
L07   ECMP { next=L16,cond_addr=L08,pin=0,reg=T,data=0};

; move the ECMP DF which contains the compare value for duty cycle creation
; to register R
L08   MOV32 { remote=L03,type=REMTOREG,reg=R};

; Subtract the current compare value from the max duty cycle stored in
; REM_DUTY. The result will be stored in register S.
L09   SUB { src1=REM,src2=R,dest=S,remote=REM_DUTY,data=0};

; If the subtraction result is more than 0 then it means it has not
; reached the max duty cycle we will increase the duty cycle. If it is
; zero or less than zero then we have reached the max duty cycle and we
; will change the up/down flag to down position.
L10   BR { next=L12,cond_addr=L11,event=GT};

; Add specified amount to the existing compare value (duty cycle) to specify the new
; duty cycle. The amount to increment is changeable by CPU before the N2HET program starts.
; After the addition, jump back to the beginning of the program
L11   ADD { next=L15,src1=R,src2=IMM,dest=S,rdest=REM,remote=L03,data=DUTY_INCREMEMT,
            hr_data=DUTY_INCREMEMT_HR};

; Insert a timer delay after the maximum duty cycle is reached. A timer delay here has the
; effect of creating the high side of a trapezoid waveform. If the timer is zero then it
; becomes a triangle wave.
L12   DJZ { next=L00,cond_addr=L13,reg=NONE,data=HIGH_DELAY};
```

```
; After the above DJZ expires on its counter we need to reload the DJZ counter to the
; specified amount of delay.
L13   MOV32 { next=L14,remote=L12,type=IMTOREG&REM,reg=NONE,data=HIGH_DELAY};

; Now change the up/down flag to down by moving a 1 to the up/down flag
L14   MOV32 { remote=L05,type=IMTOREG&REM,reg=NONE,data=1};

; Branch to the beginning
L15   BR { next=L00,cond_addr=L00,event=NOCOND};

; move the ECMP DF to register R which contains the current compare value
; (duty cycle)
L16   MOV32 { remote=L03,type=REMTOREG,reg=R};

; Subtract the current duty cycle by the specified amount. This amount of decrement is
; changeable by CPU before the N2HET program starts. When this instruction is executed
; the first time, the current duty cycle is at the maximum duty cycle. Here we are creating
; the ramp down part of the triangle/trapezoid waveforms.
L17   SUB { src1=R,src2=IMM,dest=S,rdest=NONE,data=DUTY_DECREMEMT,
            hr_data=DUTY_DECREMEMT_HR};

; As long as the subtraction result is greater than zero, we will keep
; decreasing the duty cyle or otherwise we will again change the up/down
; flag to up position. The destination register is S which contains the
; subtraction result.
L18   BR { next=L19,cond_addr=L20,event=N};

; Move the subtraction result to the ECMP DF as the new duty cycle
L19   MOV32 { next=L00,remote=L03,type=REGTOREM,reg=S};

; Insert a timer delay after the 0% duty cycle is reached. A timer delay here has the
; effect of creating the low side of a trapezoid waveform. If the timer is zero then it
; becomes a triangle wave.
L20   DJZ { next=L00,cond_addr=L21,reg=NONE,data=LOW_DELAY};

; Reset the increment delay to the specified amount.
L21   MOV32 {remote=L20,type=IMTOREG&REM,reg=NONE,data=LOW_DELAY};

; Move the value 0 to the up/down flag so in the next LRP the program
; flow will execute the path to increase duty cycle.
L22   MOV32 { remote=L05,type=IMTOREG&REM,reg=NONE,data=0};

; Branch to beginning
L23   BR { next=L00,cond_addr=L00,event=NOCOND};

; REM_DUTY data field stores the maximum duty cycle the PWM to be generated.
; The host CPU can change this value.
REM_DUTY   ECMP { next=REM_DUTY,cond_addr=REM_DUTY,pin=0,reg=A,data=INIT_COMPARE,
                  hr_data=INIT_COMPARE_HR};
DUMMY   BR { next=DUMMY,cond_addr=DUMMY,event=NOCOND,irq=OFF};
```

3.3 N2HET Assembler

The N2HET code needs to be translated into the opcode that the N2HET can execute. This is done with the N2HET assembler *hetp*. The assembler can be executed on the command line. Here is an example of the command line to use for assembling the code for N2HET1 instance:

```
hetp -n0 -hc32 Trapezoid_Wave.het
```

The -hc32 argument produces C header file Trapezoid_Wave.h and source file Trapezoid_Wave.c for the Texas Instruments TI's C compiler. Specifying the -n0 argument will allow the assembler to produce unique header and source files for the N2HET1 instance.

4 Analog Output Waveform Frequency Calculation

The analog output frequency after the low-pass filter depends on the reference clock frequency (VCLK2) for the N2HET as well the duty cycle resolution. The resolution is the smallest increment in the analog output voltage that can be divided between 0 V and the power rail (Vccio). For a 10-bit resolution, there are a total of 1024 steps that can be divided between 0 V and Vccio = 3.3 V. Each step is equal to 3.3 V / 1024 ~= 3.2 mV. The higher the resolution, the less errors on the output voltage. However, the higher the resolution, the lower the output frequency.

The triangle wave frequency Fo can be calculated using Equation 1.

$$F_O = \frac{F_{PWM}}{(2 \times DCR)} \tag{1}$$

where:

F_{PWM} = the carrier PWM frequency

DCR = duty cycle resolution or 2^N where N is the number of bits

F_{PWM} can also be expressed as shown in Equation 2.

$$F_{PWM} = \frac{1}{T_{PWM}} \tag{2}$$

where:

$$T_{PWM} = VCLK2 \times DCR \tag{3}$$

VCLK2 = the reference clock cycle to the N2HET module

Therefore,

$$F_O = \frac{1}{(T_{PWM} \times 2 \times DCR)} \tag{4}$$

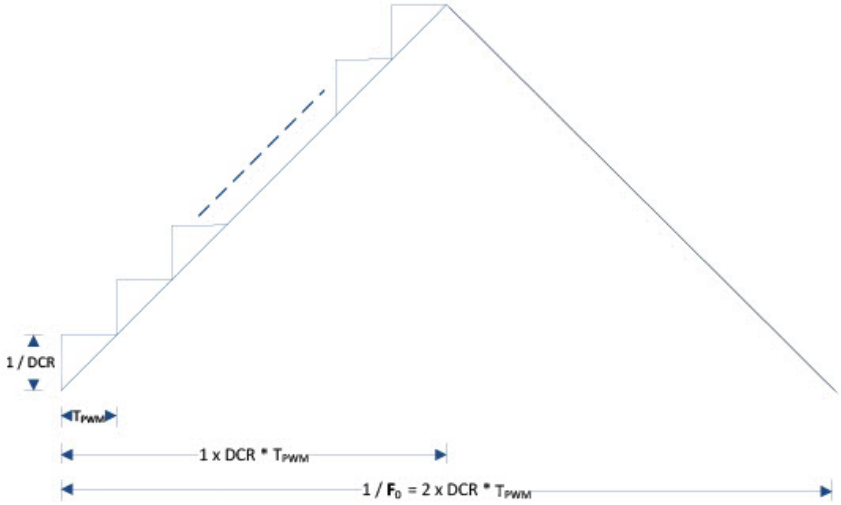
$$F_O = \frac{1}{(VCLK2 \times DCR \times 2 \times DCR)} = \frac{1}{\left(2 \times VCLK2 \times DCR^2\right)} \tag{5}$$

For example,

VCLK2 = 90 MHz or 11.11 ns and

DCR = 1024 steps of resolution

$$F_O = \frac{1}{\left(2 \times 11.11 ns \times 1024^2\right)} = 42.9 Hz \tag{6}$$



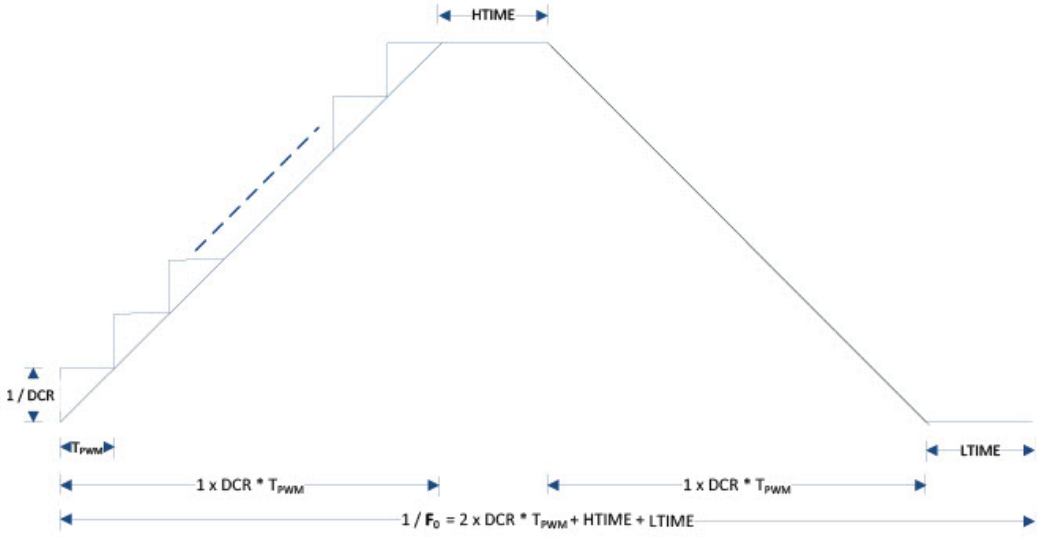
Calculating the output frequency for a trapezoid wave is similar. The triangle wave frequency F_o can be calculated using Equation 7.

$$F_0 = \frac{1}{\left(2 \times VCLK2 \times DCR^2 + HTIME + LTIME\right)} \quad (7)$$

where,

HTIME = time duration on the high phase of the trapezoid wave

LTIME = time duration of the low phase of the trapezoid wave

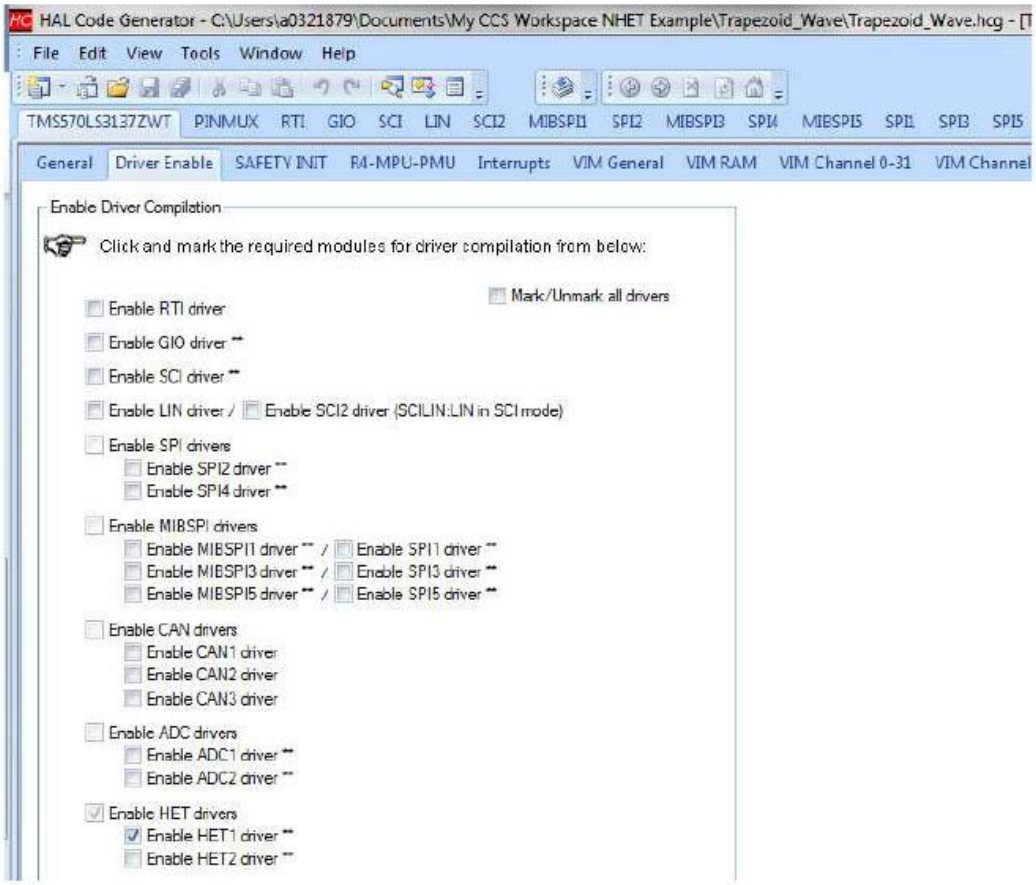


5 CPU Side Setup

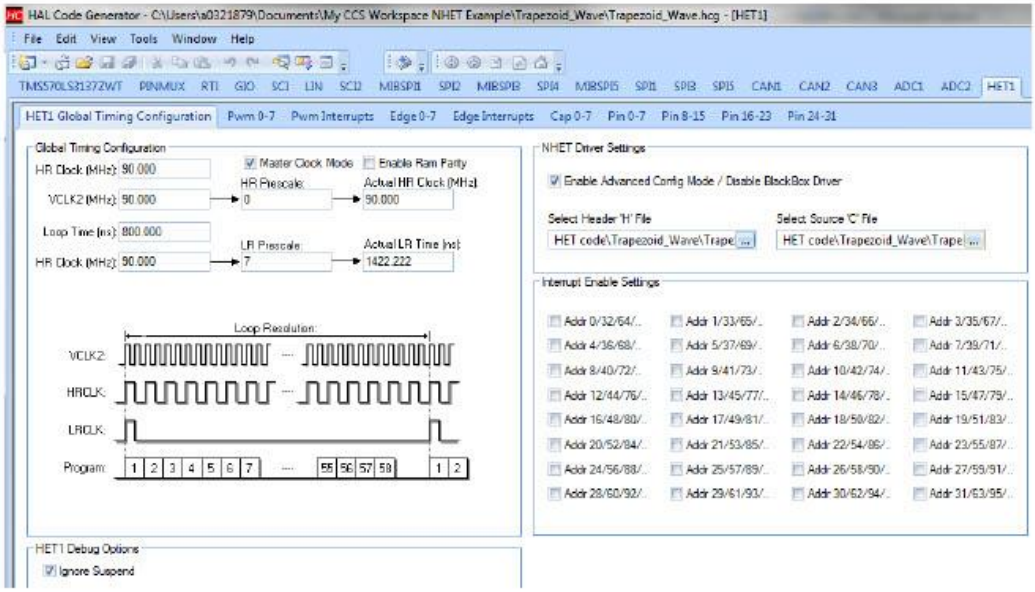
5.1 HalcoGen Setup

This example utilizes the HalCogen tool to configure the device. The target device selected in the HalCoGen is the TMS570LS3137. It can easily be ported to other devices by following the below steps.

1. Create a new project: File → New → Project. Name the project Trapezoid_Wave.
2. Enable the N2HET1 driver and disable the rest.



3. Configure the N2HET1. Make sure to enable the checkbox for "Enable Advanced Config Mode/Disable BlackBox Driver" and provide the header file (Trapezoid_Wave.h) and source file (Trapezoid_Wave.c) generated in Section 3.3. This step bypasses the default blackbox N2HET code provided by HalCoGen so that the custom N2HET code in Section 3.2 can be loaded to the N2HET module.



4. Select File → Generate Code to generate the code.

5.2 CPU Main() Code Setup

On the host CPU side, its main task is to first initialize the device and configure both the N2HET1 module. Various macros are utilized to configure the N2HET1 module. Below are the list of changeable macros.

```
/*****************************************************************************
 * Below 8 macros are changeable by user to generate various different       *
 * waveforms.                                                                *
 *****************************************************************************/

/* PWM_PERIOD defines the period of the carrier PWM frequency. PWM_PERIOD is
 * expressed in terms of number of LRP (Loop Resolution Period). The frequency
 * of the carrier PWM will remain constant. However, the N2HET will
 * modulate the duty cycle to generate a time-varying PWM signal. This
 * time-varying PWM signal is then passed through a low pass filter to remove
 * unwanted high frequency components, an analog waveform is thus rendered.
 *
 * The LRP period depends on the reference VCLK2 frequency to the N2HET
 * module as well as the programmable LR Prescaler factor (lr). It can be
 * expressed as:
 * 1 LRP = VCLK2 * lr
 * If VCLK2 = 11.11ns and lr = 128 then
 * 1 LRP = 11.11 ns * 128 = 1.42us
 * Note that for this example, the high resolution prescale factor (hr)
 * is always configured to 1.
 *
 * The width of the PWM_PERIOD also defines the resolution of the output
 * analog signal to be rendered. With PWM_PERIOD = 1, there are a total
 * of 128 VCLK2 cycles. This means that the N2HET will module its duty
 * cycle between these 128 steps. Thefore, with PWM_PERIOD = 1, the
 * output analog signal will have a log2(128) = 8-bit resolution. If
 * user desires higher resolution, the PWM_PERIOD can be changed to a
 * higher value such as 32. With PWM_PERIOD = 32, the duty cycle can be
 * modulated among a total of 32 * 128 = 4096 steps and hence render a
```

```
 * log2(4096) = 12-bit of resolution. The higher the resolution the
 * less noise on the final output. However, the higher the resolution,
 * the lower the final output frequency of the signal to be generated.
 */
#define PWM_PERIOD     32

/* PWM_DUTY defines the maximum duty cycle the N2HET is allowed to
 * modulate to. The PWM_DUTY is expressed in terms of percent (%).
 * With PWM_DUTY = 100%, the maximum amplitude on the output signal
 * will be equal to Vccio. Setting PWM_DUTY = 50% will create a
 * maximum amplitude of Vccio / 2. Note that the maximum amplitude
 * will also depend on the type of filter used and its respective
 * RC values. */
#define PWM_DUTY      100.0

/* allowable LR Prescaler factors are 32, 64 and 128. Anything less
 * than 32 will not have enough time slots for the N2HET program to
 * run.

  LRPFC can be either 5, 6 or 7.
 * 7 -> one lr = 128 VCLK2
 * 6 -> one lr = 64 VCLK2
 * 5 -> one lr = 32 VCLK2
 */
#define LRPFC 7

/* The NH2ET1 program will automatically increase the PWM
 * modulation from 0% duty cycle to maximum duty cycle
 * specified in PWM_DUTY. When PWM_DUTY is reached it starts
 * to decrease the duty cycle from PWM_DUTY to 0%.
 * DUTY_INCREMENT specifies the delta amount of duty cycle to
 * change from one duty cycle to the next duty cycle while
 * the duty cycle is increasing. Increasing DUTY_INCREMENT
 * has the effect of decreasing the duty cycle resolution for
 * a given PWM_PERIOD. This is expressed in terms of (%).
 * For example specifying DUTY_INCREMENT equal to 5 will mean
 * the duty cycle will start at 0% and the next duty cycle
 * will be 5% at a 5% increment. If 0 is specified, then the
 * N2HET1 will increment the duty cycle at 1 VCLK2 clock resolution */
#define DUTY_INCREMENT   0.0

/* DUTY_DECREMEMT specifies the delta amount of duty cycle to
 * change from one duty cycle to the next duty cycle while
 * the duty cycle is decreasing. This is expressed in terms
 * of (%). */
#define DUTY_DECREMENT   0.0

/* The MAX_DUTY_TIMER is the amount of wait time to stay at
 * the maximum duty cycle. Change to a non-zero value will
 * create traposoid waveform. A zero value will create a
 * trianagle wave
 *
 * The MAX_DUTY_TIMER is expressed in terms of number of
 * LRP. */
#define MAX_DUTY_TIMER  0

/* The MIN_DUTY_TIMER is the amount of wait time
 * to stay at the minimum duty cycle. Change to
 * a non-zero value will create trapesoid waveform.
 * A zero value will create a trainable wave */
#define MIN_DUTY_TIMER  0

/* Pin number in N2HET1 to generate the PWM. */
#define NHET1_PIN_PWM PIN_HET_9

/*****************************************************************************/
```

```
/* Below macros are for internal calculation. Do not change.                */
/***************************************************************************/

/* The PWM Period to be loaded to N2HET1 CNT instruction */
#define CNT_MAX_PERIOD (PWM_PERIOD - 1)

/* The max PWM Duty to be loaded to N2HET1 ECMP Instruction */
#define ECMP_MAX_DUTY (PWM_PERIOD * PWM_DUTY / 100.0)

/* DELTA_INCREMENT is the amount to be loaded to N2HET1 to increment the
 * duty cycle */
#define DELTA_INCREMENT (CNT_MAX_PERIOD * DUTY_INCREMENT / 100)

/* DELTA_DECREMENT is amount to be loaded to N2HET1 to decrement the
 * duty cycle */
#define DELTA_DECREMENT (CNT_MAX_PERIOD * DUTY_DECREMENT / 100)


/* Unlock key to for N2HET2 */
#define UNLOCK_KEY 0xAU
```

Table 1. Triangle Wave Frequency for Different LRPFC and PWM_PERIOD With VCLK2 = 90MHz

	PWM_PERIOD							
LRPFC	1	2	4	8	16	32	64	128
5	43.9KHz	11KHz	2.7KHz	686.7Hz	171.7Hz	42.9Hz	10.7Hz	2.7Hz
6	11KHz	2.7KHz	686.7Hz	171.7Hz	42.9Hz	10.7Hz	2.7Hz	0.67Hz
7	2.7KHz	686.7Hz	171.7Hz	42.9Hz	10.7Hz	2.7Hz	0.67Hz	0.17Hz

Table 2. Triangle Wave Resolution for Different LRPFC and PWM_PERIOD

	PWM_PERIOD							
LRPFC	1	2	4	8	16	32	64	128
5	5-bit	6-bit	7-bit	8-bit	9-bit	10-bit	11-bit	12-bit
6	6-bit	7-bit	8-bit	9-bit	10-bit	11-bit	12-bit	13-bit
7	7-bit	8-bit	9-bit	10-bit	11-bit	12-bit	13-bit	14bit

NOTE: Table 1 and Table 2 do not take into account the type of analog low-pass filters used.

The triangle and trapezoid waveform generation mainly handles the N2HET itself without CPU intervention. The host CPU's job is to first initialize the device and configure the N2HET1. The rest of the time the host CPU stays in a loop.

Data Collection

```
void main(void)
{
/* USER CODE BEGIN (3) */
    /* This example uses the N2HET1 to generate either a triangle wave
     * or a trapezoid wave by modulating the PWM duty cycle.
     * N2HET1 program: Trapezoid_Wave.het
     */

    /* initialize N2HET1 based on HalCoGen settings */
    hetInit();

    /* Configure additional settings of N2HET1 based on the macros settings */
    configNHET1();

    while(1);

/* USER CODE END */
```

```
}
/* this function is to configure additional settings for the N2HET1 */
void configNHET1()
{

    /* calculate_ecmp_compare() will calculate the compare value as well as
     * the high resolution delay values to be loaded into the N2HET1 ECMP1
     * instruction for PWM Duty generation based on the input ECMP_MAX_DUTY */
    calculate_ecmp_compare();

    /* Set NHET1_PIN_PWM to output */
    hetREG1->DIR = 1 << NHET1_PIN_PWM;

    /* Change the LRPFC according to user input */
    hetREG1->PFR = (LRPFC << 8);

    /* Initialize the PWM period and duty cycle based on the defined parameters */
    hetRAM1->Instruction[pHET_L02_0].Control = (uint32)CNT_MAX_PERIOD |
                                               (hetRAM1-
>Instruction[pHET_L02_0].Control & 0xFD0000);
    hetRAM1->Instruction[pHET_REM_DUTY_0].Data = ecmp_compare_value;

    /* Configure the N2HET1 pin to output the PWM */
    hetRAM1->Instruction[pHET_L03_0].Control = (hetRAM1-
>Instruction[pHET_L03_0].Control & 0xFFFFE0FF) |
                                               (NHET1_PIN_PWM << 8);

    /* Configure the amount of delay to stay at the maximum duty cycle */
    hetRAM1->Instruction[12].Data = ((uint32)MAX_DUTY_TIMER << 7);
    hetRAM1->Instruction[13].Data = ((uint32)MAX_DUTY_TIMER << 7);

    /* Configure the amount of delay to stay at the minumum duty cycle */
    hetRAM1->Instruction[20].Data = ((uint32)MIN_DUTY_TIMER << 7);
    hetRAM1->Instruction[21].Data = ((uint32)MIN_DUTY_TIMER << 7);

    /* Duty cycle increment amount */
    if ( (uint32)DELTA_INCREMENT == 0){
        hetRAM1->Instruction[pHET_L11_0].Data = 1 << (7 - LRPFC);
    } else {
        hetRAM1->Instruction[pHET_L11_0].Data = (uint32)DELTA_INCREMENT << (7 - LRPFC);
    }

    /* Duty cycle decrement amount */
    if ((uint32)DELTA_DECREMENT == 0) {

        hetRAM1->Instruction[pHET_L17_0].Data = 1 << (7 - LRPFC);
    } else {
        hetRAM1->Instruction[pHET_L17_0].Data = (uint32)DELTA_DECREMENT << (7 - LRPFC);
    }

    /* Unlock the N2HET program. Initially after reset the N2HET program is locked */
    hetRAM1->Instruction[pHET_L00_0].Data = UNLOCK_KEY << 7;

}
```

The complete main() can be found in the project folder under Trapezoid_Wave/source/sys_main.c.

5.3 Project Directory Structure

This example project is named Async_NHET1_PWM_NHET2_Monitoring; Figure 6 shows the project directory structure. The two N2HET programs are contained under the HET code folder.

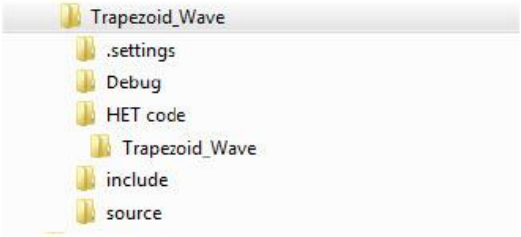


Figure 6. Trapezoid_Wave Project Directory Structure

6 Examples

By changing the parameters listed in Section 5.2, various forms of triangle and trapezoid waveforms can be rendered.

6.1 An Isosceles Triangle 1

In this example, a basic isosceles triangle (triangle with two equal sides) is created. With PWM_DUTY set to 100%, the output amplitude is equal to the rail voltage.

```
#define PWM_PERIOD        32
#define PWM_DUTY          100.0
#define LRPFC             7
#define DUTY_INCREMENT    0.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    0
#define MIN_DUTY_TIMER    0
```

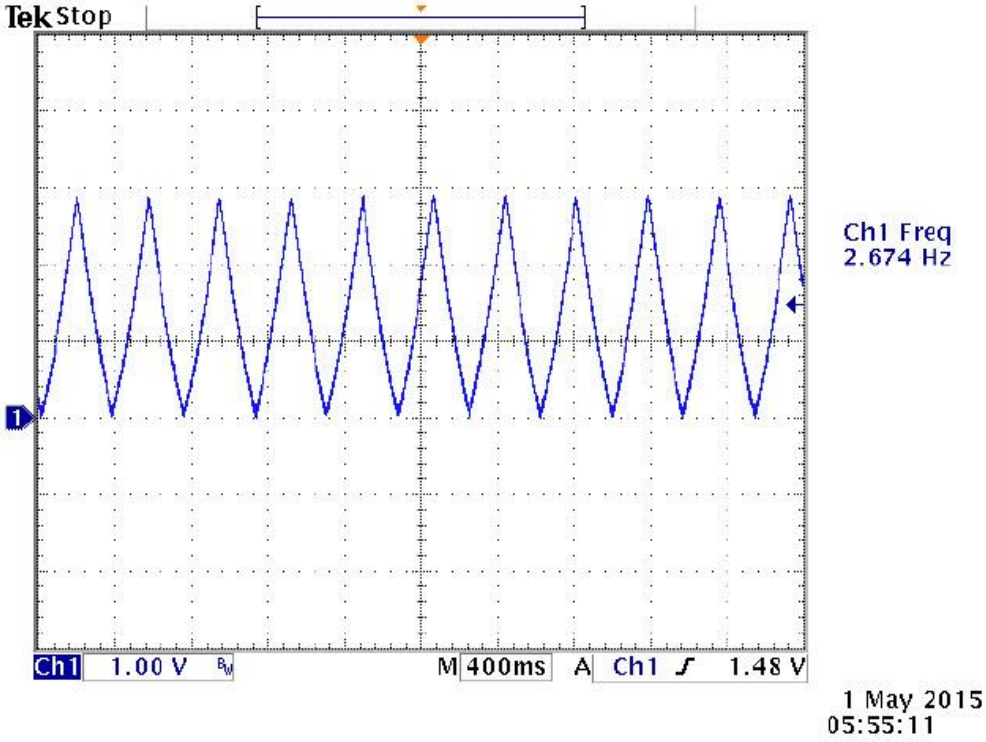


Figure 7. An Isosceles Triangle 1

6.2 An Isosceles Triangle 2

In this example, a basic isosceles triangle (triangle with two equal sides) is created. With PWM_DUTY set to 50%, the output amplitude is equal to half of the rail voltage (Vccio / 2). Reducing the maximum duty cycle to 50% also has the effect of reducing the duty cycle resolution by half. Reducing the resolution by half increases the output frequency by 2X.

```
#define PWM_PERIOD        32
#define PWM_DUTY          50.0
#define LRPFC             7
#define DUTY_INCREMENT    0.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    0
#define MIN_DUTY_TIMER    0
```

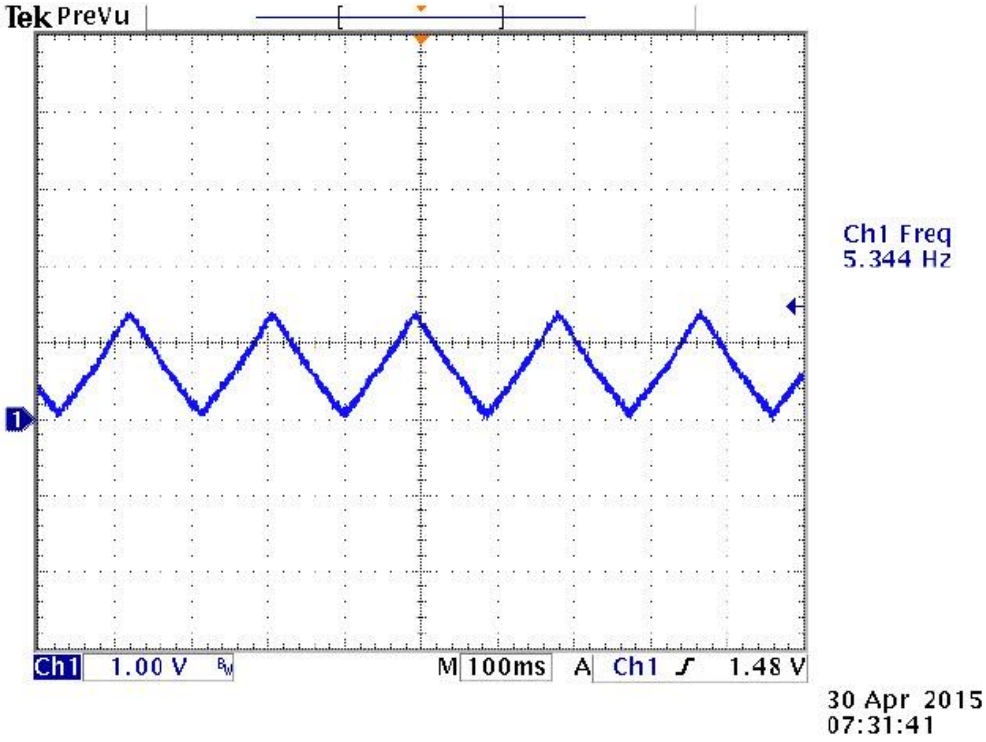


Figure 8. An Isosceles Triangle 2

6.3 A Scalene Triangle

In this example, a scalene triangle (triangle with no equal sides) is created.

```
#define PWM_PERIOD        32
#define PWM_DUTY          100.0
#define LRPFC             7
#define DUTY_INCREMENT    10.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    0
#define MIN_DUTY_TIMER    0
```

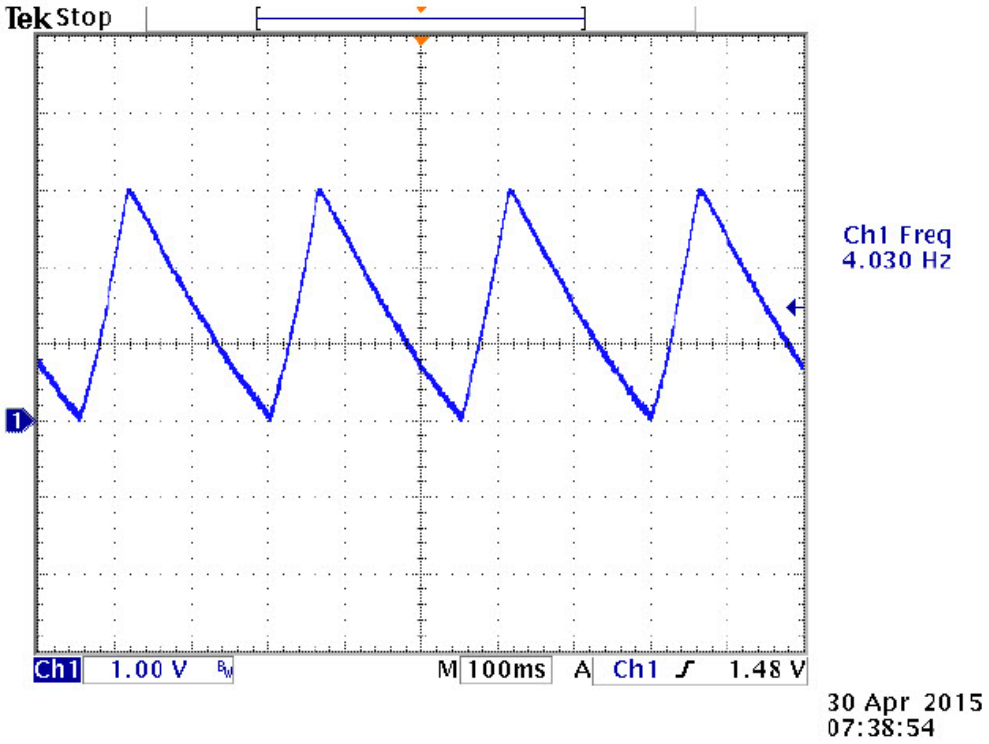


Figure 9. A Scalene Triangle

6.4 A Scalene Triangle

When DUTY_INCREMENT is increased to 100%, it is approaching a right triangle that is also a sawtooth wave.

```
#define PWM_PERIOD        32
#define PWM_DUTY          100.0
#define LRPFC             7
#define DUTY_INCREMENT    100.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    0
#define MIN_DUTY_TIMER    0
```

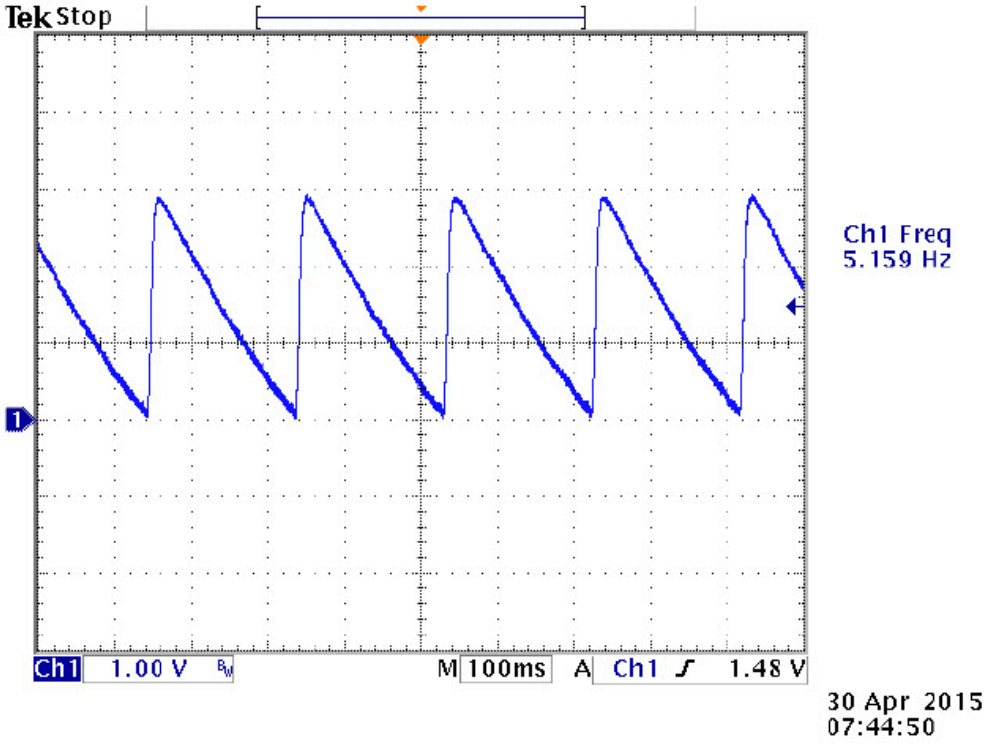


Figure 10. A Sawtooth Triangle

6.5 An Isosceles Trapezoid

In this example, a basic isosceles trapezoid (trapezoid with two equal sides) is created. With PWM_DUTY set to 100%, the output amplitude is equal to the rail voltage. Reducing the LRPFC to 5 is equivalent to generating 10-bit resolution on the ramp up and ramp down.

```
#define PWM_PERIOD        32
#define PWM_DUTY          100.0
#define LRPFC             5
#define DUTY_INCREMENT    0.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    2048
#define MIN_DUTY_TIMER    2048
```

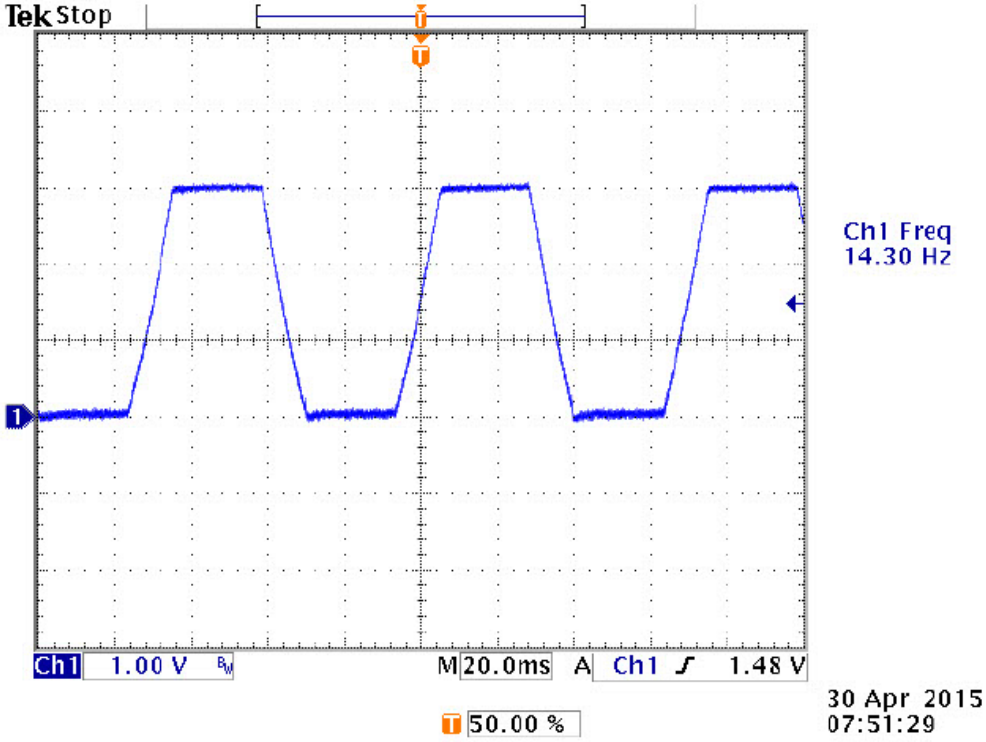


Figure 11. An Isosceles Trapezoid

6.6 An Right Trapezoid

```
#define PWM_PERIOD        32
#define PWM_DUTY          100.0
#define LRPFC             5
#define DUTY_INCREMENT    100.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    4096
#define MIN_DUTY_TIMER    1024
```

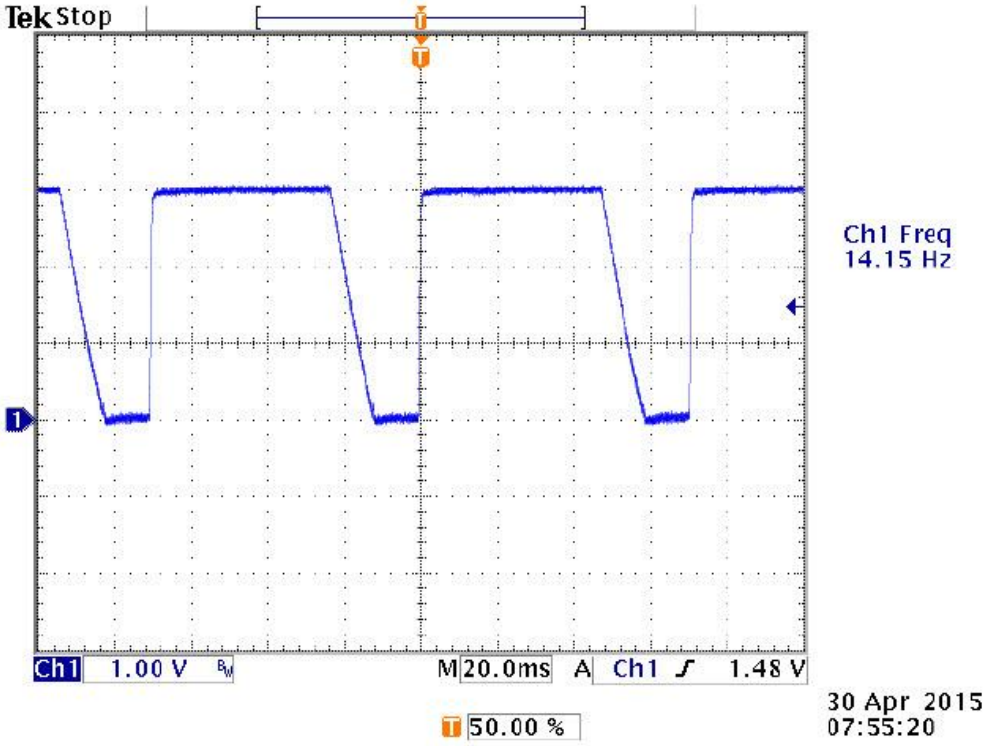


Figure 12. A Right Trapezoid

6.7 An Scalene Trapezoid

```
#define PWM_PERIOD        32
#define PWM_DUTY          100.0
#define LRPFC             5
#define DUTY_INCREMENT    10.0
#define DUTY_DECREMENT    0.0
#define MAX_DUTY_TIMER    1024
#define MIN_DUTY_TIMER    2048
```

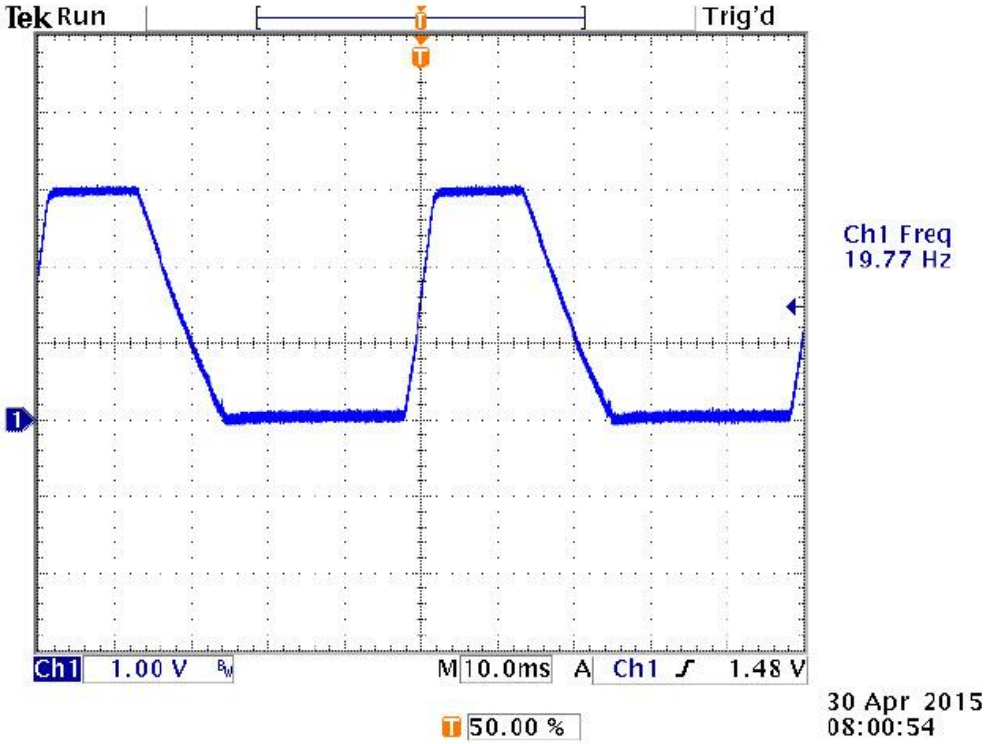


Figure 13. A Scalene Trapezoid

7 References

- *HET Integrated Development Environment User's Guide* (SPNU483)
- *NHET Getting Started* (SPRABA0B)
- *Enhanced High-End Timer (NHET) Assembler User's Guide* (SPNU490)
- *Using PWM Output as a Digital-to-Analog Converter on a TMS320F280x Digital Signal Controller* (SPRAA88)
- *PWM DAC Using MSP430 High-Resolution Timer* (SLAA497)

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Welcome to the Hercules™ LaunchPad

Additional resources at:
www.ti.com/launchpad

SPNU588

Hercules RM42x LaunchPad Quick Start Guide

Welcome to the Hercules RM42x LaunchPad Evaluation Kit. The Hercules LaunchPad is a USB-based evaluation platform that provides everything you need to start evaluation and development with Hercules MCUs.

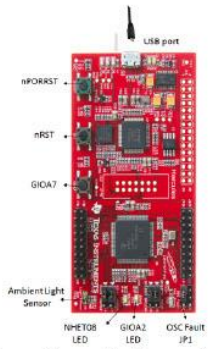
1. Software and Driver Installation

Go to www.ti.com/launchpad. Select Launchpads tab and then select Hercules. Here you can download and install Code Composer Studio™ (CCS). This will install the necessary drivers for LaunchPad. If you choose the custom install option of CCS, select 'Cortex-R4F MCUs' support at a minimum. Select 'Free CCS License – For use with XDS100 emulators'. **Note:** Complete the CCS installation before connecting the board.

Additional software and documentation can be found on the Hercules LaunchPad wiki page: http://processors.wiki.ti.com/index.php/Hercules_LaunchPad

2. Connecting the Hardware

Connect the LaunchPad using the included USB cable to a Windows PC (XP or 7). The board will be powered via the PC's USB port. If prompted, allow Windows to automatically install the driver software for the on-board XDS100v2 JTAG emulator and the Virtual COM Port.



3. Quick Start Application

The MCU on the Hercules LaunchPad comes pre-programmed with the Hercules Safety MCU Demo Software. This software can be used stand alone on the LaunchPad or in conjunction with the PC application shown in section 4 of this guide. When the board is powered on via the USB port the demo software will show a startup blinking sequence on the GIOA2 and NHET08 LEDs. The demo also lets you toggle the GIOA2 LED through the push button GIOA7.

You can start learning about the Hercules MCU's built-in safety features right out of the box. Inject an Oscillator fault by connecting OSCIN to GND (close jumper JP1). Upon detecting the fault, on-board Hercules MCU will respond by asserting the error pin (nERROR) low, indicated by the red LED on the bottom right corner of the board. **Note:** Open jumper JP1 and reset the board before continuing with other demos.

4. Hercules Safety MCU Demos

Go to the Hercules LaunchPad wiki page to download and install the Hercules Safety MCU Demos. Once the installation is complete, start the Hercules demo software. The software will be available in 'Start->All Programs->Texas Instruments->Hercules->Hercules Safety MCU Demos'.

It includes a safety features demo and other demos using LEDs and ambient light sensor that let you interact with and learn about features on Hercules MCUs.

5. Project 0

When you are ready to take the next step, complete Project 0. For more information go to www.ti.com/launchpad and click on the Project 0 link for Hercules LaunchPad.

Explore LaunchPad BoosterPacks at www.ti.com/boosterpacks
Watch Hercules 'How-to' videos at www.ti.com/herculestraining. Technical support: www.ti.com/hercules-support

STANDARD TERMS AND CONDITIONS FOR EVALUATION MODULES

1. *Delivery:* TI delivers TI evaluation boards, kits, or modules, including any accompanying demonstration software, components, or documentation (collectively, an "EVM" or "EVMs") to the User ("User") in accordance with the terms and conditions set forth herein. Acceptance of the EVM is expressly subject to the following terms and conditions.

   1.1 EVMs are intended solely for product or software developers for use in a research and development setting to facilitate feasibility evaluation, experimentation, or scientific analysis of TI semiconductors products. EVMs have no direct function and are not finished products. EVMs shall not be directly or indirectly assembled as a part or subassembly in any finished product. For clarification, any software or software tools provided with the EVM ("Software") shall not be subject to the terms and conditions set forth herein but rather shall be subject to the applicable terms and conditions that accompany such Software

   1.2 EVMs are not intended for consumer or household use. EVMs may not be sold, sublicensed, leased, rented, loaned, assigned, or otherwise distributed for commercial purposes by Users, in whole or in part, or used in any finished product or production system.

2 *Limited Warranty and Related Remedies/Disclaimers*:

   2.1 These terms and conditions do not apply to Software. The warranty, if any, for Software is covered in the applicable Software License Agreement.

   2.2 TI warrants that the TI EVM will conform to TI's published specifications for ninety (90) days after the date TI delivers such EVM to User. Notwithstanding the foregoing, TI shall not be liable for any defects that are caused by neglect, misuse or mistreatment by an entity other than TI, including improper installation or testing, or for any EVMs that have been altered or modified in any way by an entity other than TI. Moreover, TI shall not be liable for any defects that result from User's design, specifications or instructions for such EVMs. Testing and other quality control techniques are used to the extent TI deems necessary or as mandated by government requirements. TI does not test all parameters of each EVM.

   2.3 If any EVM fails to conform to the warranty set forth above, TI's sole liability shall be at its option to repair or replace such EVM, or credit User's account for such EVM. TI's liability under this warranty shall be limited to EVMs that are returned during the warranty period to the address designated by TI and that are determined by TI not to conform to such warranty. If TI elects to repair or replace such EVM, TI shall have a reasonable time to repair such EVM or provide replacements. Repaired EVMs shall be warranted for the remainder of the original warranty period. Replaced EVMs shall be warranted for a new full ninety (90) day warranty period.

3 *Regulatory Notices:*

   3.1 *United States*

      3.1.1 *Notice applicable to EVMs not FCC-Approved:*

      This kit is designed to allow product developers to evaluate electronic components, circuitry, or software associated with the kit to determine whether to incorporate such items in a finished product and software developers to write software applications for use with the end product. This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of this chapter, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of this chapter.

      3.1.2 *For EVMs annotated as FCC – FEDERAL COMMUNICATIONS COMMISSION Part 15 Compliant:*

      CAUTION

      This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

      Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

      FCC Interference Statement for Class A EVM devices

      NOTE: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

FCC Interference Statement for Class B EVM devices

NOTE: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- *Reorient or relocate the receiving antenna.*
- *Increase the separation between the equipment and receiver.*
- *Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.*
- *Consult the dealer or an experienced radio/TV technician for help.*

3.2 *Canada*

3.2.1 *For EVMs issued with an Industry Canada Certificate of Conformance to RSS-210*

Concerning EVMs Including Radio Transmitters:

This device complies with Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Concernant les EVMs avec appareils radio:

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes: (1) l'appareil ne doit pas produire de brouillage, et (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

Concerning EVMs Including Detachable Antennas:

Under Industry Canada regulations, this radio transmitter may only operate using an antenna of a type and maximum (or lesser) gain approved for the transmitter by Industry Canada. To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the equivalent isotropically radiated power (e.i.r.p.) is not more than that necessary for successful communication. This radio transmitter has been approved by Industry Canada to operate with the antenna types listed in the user guide with the maximum permissible gain and required antenna impedance for each antenna type indicated. Antenna types not included in this list, having a gain greater than the maximum gain indicated for that type, are strictly prohibited for use with this device.

Concernant les EVMs avec antennes détachables

Conformément à la réglementation d'Industrie Canada, le présent émetteur radio peut fonctionner avec une antenne d'un type et d'un gain maximal (ou inférieur) approuvé pour l'émetteur par Industrie Canada. Dans le but de réduire les risques de brouillage radioélectrique à l'intention des autres utilisateurs, il faut choisir le type d'antenne et son gain de sorte que la puissance isotrope rayonnée équivalente (p.i.r.e.) ne dépasse pas l'intensité nécessaire à l'établissement d'une communication satisfaisante. Le présent émetteur radio a été approuvé par Industrie Canada pour fonctionner avec les types d'antenne énumérés dans le manuel d'usage et ayant un gain admissible maximal et l'impédance requise pour chaque type d'antenne. Les types d'antenne non inclus dans cette liste, ou dont le gain est supérieur au gain maximal indiqué, sont strictement interdits pour l'exploitation de l'émetteur

3.3 *Japan*

3.3.1 *Notice for EVMs delivered in Japan:* Please see http://www.tij.co.jp/lsds/ti_ja/general/eStore/notice_01.page 日本国内に輸入される評価用キット、ボードについては、次のところをご覧ください。
http://www.tij.co.jp/lsds/ti_ja/general/eStore/notice_01.page

3.3.2 *Notice for Users of EVMs Considered "Radio Frequency Products" in Japan:* EVMs entering Japan may not be certified by TI as conforming to Technical Regulations of Radio Law of Japan.

If User uses EVMs in Japan, not certified to Technical Regulations of Radio Law of Japan, User is required by Radio Law of Japan to follow the instructions below with respect to EVMs:

1. Use EVMs in a shielded room or any other test facility as defined in the notification #173 issued by Ministry of Internal Affairs and Communications on March 28, 2006, based on Sub-section 1.1 of Article 6 of the Ministry's Rule for Enforcement of Radio Law of Japan,
2. Use EVMs only after User obtains the license of Test Radio Station as provided in Radio Law of Japan with respect to EVMs, or
3. Use of EVMs only after User obtains the Technical Regulations Conformity Certification as provided in Radio Law of Japan with respect to EVMs. Also, do not transfer EVMs, unless User gives the same notice above to the transferee. Please note that if User does not follow the instructions above, User will be subject to penalties of Radio Law of Japan.

【無線電波を送信する製品の開発キットをお使いになる際の注意事項】 開発キットの中には技術基準適合証明を受けていないものがあります。 技術適合証明を受けていないもののご使用に際しては、電波法遵守のため、以下のいずれかの措置を取っていただく必要がありますのでご注意ください。

1. 電波法施行規則第6条第1項第1号に基づく平成18年3月28日総務省告示第173号で定められた電波暗室等の試験設備でご使用いただく。
2. 実験局の免許を取得後ご使用いただく。
3. 技術基準適合証明を取得後ご使用いただく。

なお、本製品は、上記の「ご使用にあたっての注意」を譲渡先、移転先に通知しない限り、譲渡、移転できないものとします。
上記を遵守頂けない場合は、電波法の罰則が適用される可能性があることをご留意ください。 日本テキサス・インスツルメンツ株式会社
東京都新宿区西新宿６丁目２４番１号
西新宿三井ビル

3.3.3 *Notice for EVMs for Power Line Communication:* Please see http://www.tij.co.jp/lsds/ti_ja/general/eStore/notice_02.page
電力線搬送波通信についての開発キットをお使いになる際の注意事項については、次のところをご覧ください。http://www.tij.co.jp/lsds/ti_ja/general/eStore/notice_02.page

4 *EVM Use Restrictions and Warnings:*

4.1 EVMS ARE NOT FOR USE IN FUNCTIONAL SAFETY AND/OR SAFETY CRITICAL EVALUATIONS, INCLUDING BUT NOT LIMITED TO EVALUATIONS OF LIFE SUPPORT APPLICATIONS.

4.2 User must read and apply the user guide and other available documentation provided by TI regarding the EVM prior to handling or using the EVM, including without limitation any warning or restriction notices. The notices contain important safety information related to, for example, temperatures and voltages.

4.3 *Safety-Related Warnings and Restrictions:*

4.3.1 User shall operate the EVM within TI's recommended specifications and environmental considerations stated in the user guide, other available documentation provided by TI, and any other applicable requirements and employ reasonable and customary safeguards. Exceeding the specified performance ratings and specifications (including but not limited to input and output voltage, current, power, and environmental ranges) for the EVM may cause personal injury or death, or property damage. If there are questions concerning performance ratings and specifications, User should contact a TI field representative prior to connecting interface electronics including input power and intended loads. Any loads applied outside of the specified output range may also result in unintended and/or inaccurate operation and/or possible permanent damage to the EVM and/or interface electronics. Please consult the EVM user guide prior to connecting any load to the EVM output. If there is uncertainty as to the load specification, please contact a TI field representative. During normal operation, even with the inputs and outputs kept within the specified allowable ranges, some circuit components may have elevated case temperatures. These components include but are not limited to linear regulators, switching transistors, pass transistors, current sense resistors, and heat sinks, which can be identified using the information in the associated documentation. When working with the EVM, please be aware that the EVM may become very warm.

4.3.2 EVMs are intended solely for use by technically qualified, professional electronics experts who are familiar with the dangers and application risks associated with handling electrical mechanical components, systems, and subsystems. User assumes all responsibility and liability for proper and safe handling and use of the EVM by User or its employees, affiliates, contractors or designees. User assumes all responsibility and liability to ensure that any interfaces (electronic and/or mechanical) between the EVM and any human body are designed with suitable isolation and means to safely limit accessible leakage currents to minimize the risk of electrical shock hazard. User assumes all responsibility and liability for any improper or unsafe handling or use of the EVM by User or its employees, affiliates, contractors or designees.

4.4 User assumes all responsibility and liability to determine whether the EVM is subject to any applicable international, federal, state, or local laws and regulations related to User's handling and use of the EVM and, if applicable, User assumes all responsibility and liability for compliance in all respects with such laws and regulations. User assumes all responsibility and liability for proper disposal and recycling of the EVM consistent with all applicable international, federal, state, and local requirements.

5. *Accuracy of Information:* To the extent TI provides information on the availability and function of EVMs, TI attempts to be as accurate as possible. However, TI does not warrant the accuracy of EVM descriptions, EVM availability or other information on its websites as accurate, complete, reliable, current, or error-free.

6. *Disclaimers:*

   6.1 EXCEPT AS SET FORTH ABOVE, EVMS AND ANY WRITTEN DESIGN MATERIALS PROVIDED WITH THE EVM (AND THE DESIGN OF THE EVM ITSELF) ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." TI DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING SUCH ITEMS, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADE SECRETS OR OTHER INTELLECTUAL PROPERTY RIGHTS.

   6.2 EXCEPT FOR THE LIMITED RIGHT TO USE THE EVM SET FORTH HEREIN, NOTHING IN THESE TERMS AND CONDITIONS SHALL BE CONSTRUED AS GRANTING OR CONFERRING ANY RIGHTS BY LICENSE, PATENT, OR ANY OTHER INDUSTRIAL OR INTELLECTUAL PROPERTY RIGHT OF TI, ITS SUPPLIERS/LICENSORS OR ANY OTHER THIRD PARTY, TO USE THE EVM IN ANY FINISHED END-USER OR READY-TO-USE FINAL PRODUCT, OR FOR ANY INVENTION, DISCOVERY OR IMPROVEMENT MADE, CONCEIVED OR ACQUIRED PRIOR TO OR AFTER DELIVERY OF THE EVM.

7. *USER'S INDEMNITY OBLIGATIONS AND REPRESENTATIONS.* USER WILL DEFEND, INDEMNIFY AND HOLD TI, ITS LICENSORS AND THEIR REPRESENTATIVES HARMLESS FROM AND AGAINST ANY AND ALL CLAIMS, DAMAGES, LOSSES, EXPENSES, COSTS AND LIABILITIES (COLLECTIVELY, "CLAIMS") ARISING OUT OF OR IN CONNECTION WITH ANY HANDLING OR USE OF THE EVM THAT IS NOT IN ACCORDANCE WITH THESE TERMS AND CONDITIONS. THIS OBLIGATION SHALL APPLY WHETHER CLAIMS ARISE UNDER STATUTE, REGULATION, OR THE LAW OF TORT, CONTRACT OR ANY OTHER LEGAL THEORY, AND EVEN IF THE EVM FAILS TO PERFORM AS DESCRIBED OR EXPECTED.

8. *Limitations on Damages and Liability:*

   8.1 *General Limitations.* IN NO EVENT SHALL TI BE LIABLE FOR ANY SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF THESE TERMS ANDCONDITIONS OR THE USE OF THE EVMS PROVIDED HEREUNDER, REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCLUDED DAMAGES INCLUDE, BUT ARE NOT LIMITED TO, COST OF REMOVAL OR REINSTALLATION, ANCILLARY COSTS TO THE PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, RETESTING, OUTSIDE COMPUTER TIME, LABOR COSTS, LOSS OF GOODWILL, LOSS OF PROFITS, LOSS OF SAVINGS, LOSS OF USE, LOSS OF DATA, OR BUSINESS INTERRUPTION. NO CLAIM, SUIT OR ACTION SHALL BE BROUGHT AGAINST TI MORE THAN ONE YEAR AFTER THE RELATED CAUSE OF ACTION HAS OCCURRED.

   8.2 *Specific Limitations.* IN NO EVENT SHALL TI'S AGGREGATE LIABILITY FROM ANY WARRANTY OR OTHER OBLIGATION ARISING OUT OF OR IN CONNECTION WITH THESE TERMS AND CONDITIONS, OR ANY USE OF ANY TI EVM PROVIDED HEREUNDER, EXCEED THE TOTAL AMOUNT PAID TO TI FOR THE PARTICULAR UNITS SOLD UNDER THESE TERMS AND CONDITIONS WITH RESPECT TO WHICH LOSSES OR DAMAGES ARE CLAIMED. THE EXISTENCE OF MORE THAN ONE CLAIM AGAINST THE PARTICULAR UNITS SOLD TO USER UNDER THESE TERMS AND CONDITIONS SHALL NOT ENLARGE OR EXTEND THIS LIMIT.

9. *Return Policy.* Except as otherwise provided, TI does not offer any refunds, returns, or exchanges. Furthermore, no return of EVM(s) will be accepted if the package has been opened and no return of the EVM(s) will be accepted if they are damaged or otherwise not in a resalable condition. If User feels it has been incorrectly charged for the EVM(s) it ordered or that delivery violates the applicable order, User should contact TI. All refunds will be made in full within thirty (30) working days from the return of the components(s), excluding any postage or packaging costs.

10. *Governing Law:* These terms and conditions shall be governed by and interpreted in accordance with the laws of the State of Texas, without reference to conflict-of-laws principles. User agrees that non-exclusive jurisdiction for any dispute arising out of or relating to these terms and conditions lies within courts located in the State of Texas and consents to venue in Dallas County, Texas. Notwithstanding the foregoing, any judgment may be enforced in any United States or foreign court, and TI may seek injunctive relief in any United States or foreign court.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

RM42L432 16- and 32-Bit RISC Flash Microcontroller

1 Device Overview

1.1 Features

- High-Performance Microcontroller for Safety-Critical Applications
  - Dual CPUs Running in Lockstep
  - ECC on Flash and RAM Interfaces
  - Built-In Self-Test for CPU and On-Chip RAMs
  - Error Signaling Module With Error Pin
  - Voltage and Clock Monitoring
- ARM® Cortex®-R4 32-Bit RISC CPU
  - Efficient 1.66 DMIPS/MHz With 8-Stage Pipeline
  - 8-Region Memory Protection Unit (MPU)
  - Open Architecture With Third-Party Support
- Operating Conditions
  - 100-MHz System Clock
  - Core Supply Voltage (V_{CC}): 1.2-V Nominal
  - I/O Supply Voltage (V_{CCIO}): 3.3-V Nominal
  - ADC Supply Voltage (V_{CCAD}): 3.3-V Nominal
- Integrated Memory
  - 384KB of Program Flash With ECC
  - 32KB of RAM With ECC
  - 16KB of Flash for Emulated EEPROM With ECC
- Hercules™ Common Platform Architecture
  - Consistent Memory Map Across Family
  - Real-Time Interrupt (RTI) Timer (OS Timer)
  - 96-Channel Vectored Interrupt Module (VIM)
  - 2-Channel Cyclic Redundancy Checker (CRC)
- Frequency-Modulated Phase-Locked Loop (FMPLL) With Built-In Slip Detector
- IEEE 1149.1 JTAG Boundary Scan and ARM CoreSight™ Components
- Advanced JTAG Security Module (AJSM)
- Multiple Communication Interfaces
  - Two CAN Controllers (DCANs)
    - DCAN1 - 32 Mailboxes With Parity Protection
    - DCAN2 - 16 Mailboxes With Parity Protection
    - Compliant to CAN Protocol Version 2.0B
  - Multibuffered Serial Peripheral Interface (MibSPI) Module
    - 128 Words With Parity Protection
  - Two Standard Serial Peripheral Interface (SPI) Modules
  - UART (SCI) Interface With Local Interconnect Network (LIN 2.1) Interface Support
- Next Generation High-End Timer (N2HET) Module
  - Up to 19 Programmable Pins
  - 128-Word Instruction RAM With Parity Protection
  - Includes Hardware Angle Generator
  - Dedicated High-End Timer Transfer Unit (HTU) With MPU
- Enhanced Quadrature Encoder Pulse (eQEP) Module
  - Motor Position Encoder Interface
- 12-Bit Multibuffered Analog-to-Digital Converter (ADC) Module
  - 16 Channels
  - 64 Result Buffers With Parity Protection
- Up to 45 General-Purpose Input/Output (GPIO) Pins
  - 8 Dedicated Interrupt-Capable GPIO Pins
- Package
  - 100-Pin Quad Flatpack (PZ) [Green]

1.2 Applications

- Industrial Safety Applications
  - Industrial Automation
  - Safe Programmable Logic Controllers (PLCs)
  - Power Generation and Distribution
  - Turbines and Windmills
  - Elevators and Escalators
- Medical Applications
  - Ventilators
  - Defibrillators
  - Infusion and Insulin Pumps
  - Radiation Therapy
  - Robotic Surgery

1.3 Description

The RM42L432 device is a high-performance microcontroller for safety systems. The safety architecture includes dual CPUs in lockstep, CPU and Memory BIST logic, ECC on both the flash and the data SRAM, parity on peripheral memories, and loopback capability on peripheral I/Os.

The RM42L432 device integrates the ARM Cortex-R4 CPU. The CPU offers an efficient 1.66 DMIPS/MHz, and has configurations that can run up to 100 MHz, providing up to 166 DMIPS. The device operates in little-endian (LE) mode.

The RM42L432 device has 384KB of integrated flash and 32KB of data RAM. Both the flash and RAM have single-bit error correction and double-bit error detection. The flash memory on this device is a nonvolatile, electrically erasable, and programmable memory implemented with a 64-bit-wide data bus interface. The flash operates on a 3.3-V supply input (the same level as I/O supply) for all read, program, and erase operations. When in pipeline mode, the flash operates with a system clock frequency of up to 100 MHz. The SRAM supports single-cycle read and write accesses in byte, halfword, word, and double-word modes throughout the supported frequency range.

The RM42L432 device features peripherals for real-time control-based applications, including a Next Generation High-End Timer (N2HET) timing coprocessor with up to 19 I/O terminals and a 12-bit Analog-to-Digital Converter (ADC) supporting 16 inputs in the 100-pin package.

The N2HET is an advanced intelligent timer that provides sophisticated timing functions for real-time applications. The timer is software-controlled, using a small instruction set, with a specialized timer micromachine and an attached I/O port. The N2HET can be used for pulse-width-modulated outputs, capture or compare inputs, or GPIO. The N2HET is especially well suited for applications requiring multiple sensor information and drive actuators with complex and accurate time pulses. A High-End Timer Transfer Unit (HTU) can perform DMA-type transactions to transfer N2HET data to or from main memory. A Memory Protection Unit (MPU) is built into the HTU.

The Enhanced Quadrature Encoder Pulse (eQEP) module is used for direct interface with a linear or rotary incremental encoder to get position, direction, and speed information from a rotating machine as used in high-performance motion and position-control systems.

The device has a 12-bit-resolution MibADC with 16 channels and 64 words of parity-protected buffer RAM. The MibADC channels can be converted individually or can be grouped by software for sequential conversion sequences. There are three separate groupings. Each sequence can be converted once when triggered or configured for continuous conversion mode. The MibADC has a 10-bit mode for use when compatibility with older devices or faster conversion time is desired.

The device has multiple communication interfaces: one MibSPI, two SPIs, one UART/LIN, and two DCANs. The SPI provides a convenient method of serial high-speed communications between similar shift-register type devices. The UART/LIN supports the Local Interconnect standard 2.1 and can be used as a UART in full-duplex mode using the standard Non-Return-to-Zero (NRZ) format. The DCAN supports the CAN 2.0 (A and B) protocol standard and uses a serial, multimaster communication protocol that efficiently supports distributed real-time control with robust communication rates of up to 1 Mbps. The DCAN is ideal for applications operating in noisy and harsh environments (for example, automotive and industrial applications) that require reliable serial communication or multiplexed wiring.

The Frequency-Modulated Phase-Locked Loop (FMPLL) clock module is used to multiply the external frequency reference to a higher frequency for internal use. The FMPLL provides one of the five possible clock source inputs to the Global Clock Module (GCM). The GCM manages the mapping between the available clock sources and the device clock domains.

The device also has an External Clock Prescaler (ECP) module that when enabled, outputs a continuous external clock on the ECLK pin. The ECLK frequency is a user-programmable ratio of the peripheral interface clock (VCLK) frequency. This low-frequency output can be monitored externally as an indicator of the device operating frequency.

The Error Signaling Module (ESM) monitors all device errors and determines whether an interrupt is generated or the external nERROR pin is toggled when a fault is detected. The nERROR pin can be monitored externally as an indicator of a fault condition in the microcontroller.

The I/O Multiplexing and Control Module (IOMM) allows the configuration of the input/output pins to support alternate functions. See Table 4-17 for a list of the pins that support multiple functions on this device.

With integrated safety features and a wide choice of communication and control peripherals, the RM42L432 device is an ideal solution for real-time control applications with safety-critical requirements.

Device Information(1)

PART NUMBER	PACKAGE	BODY SIZE
RM42L432PZ	LQFP (100)	14.00 mm × 14.00 mm

(1) For more information, see Section 9, *Mechanical Packaging and Orderable Information*.

1.4 Functional Block Diagram

Figure 1-1 shows a functional block diagram of the device.

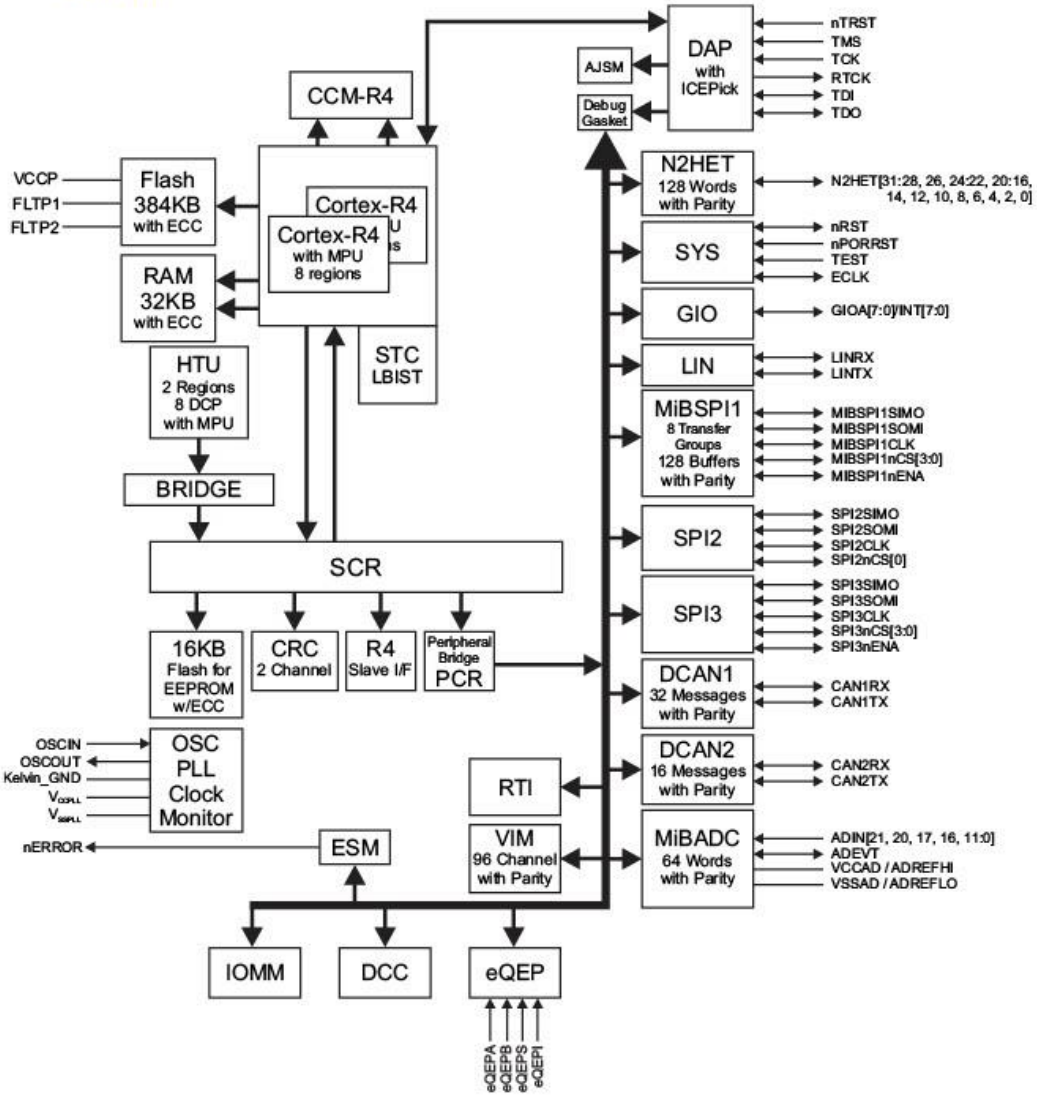


Figure 1-1. Functional Block Diagram

Table of Contents

2 Revision History

This data manual revision history highlights the technical changes made to the SPNS180A device-specific data manual to make it an SPNS180B revision.

Scope: Applicable updates to the Hercules™ MCU device family, specifically relating to the RM42L432 devices, which are now in the production data (PD) stage of development have been incorporated.

Changes from October 30, 2013 to June 30, 2015 (from A Revision (October 2013) to B Revision)	Page
• Updated/Changed section title to "Device Overview"	1
• Added Section 1.3 (Description): Added paragraph describing IOMM	4
• Section 1.3 (Description): Added the Device Information table	4
• Added Section 3, *Device Comparison*	8
• Section 5 (Specifications): Updated/Changed section title	18
• Section 5.1 (Absolute Maximum Ratings): Added Latch-up Performance Specification	18
• Section 5.2 (ESD Ratings): Added section	18
• Section 5.3 (Power-On Hours (POH)): Added table (new)	18
• Table 5-3 (Output Buffer Drive Strengths): Added the "SPI3nCS[0]" signal to the 2 mA zero-dominant signals row	23
• Table 5-4 (Selectable 8mA/2mA Control): Clarified impact of SPI2PC9 register on drive strength of SPI2SOMI pin in footnote	23
• Section 6.4.1 (Summary of ARM Cortex-R4 CPU Features): Added Quantity of Breakpoints and Watchpoints	31
• Section 6.20.3 (JTAG Identification Code): Added a table showing JTAG ID code for each silicon revision.	64
• Table 7-7 (MibADC Operating Characteristics): Added missing footnote for Z_{SET} 10-/12-bit modes.	73
• Section 7.7.1 (Features [MibSPI]): Updated/Changed size of SPI baud clock generator from "8-bit" to "11-bit"	83
• Section 8 (Device and Documentation Support): Updated/Changed section outline structure	95
• Section 8.1 (Device Support): Added section (new)	95
• Section 8.1.1 (Development Support, Hardware Development Tools): Updated/Changed the JTAG-based emulators specified	95
• Section 8.1.2 (Device Nomenclature): Updated/Changed section title	96
• Figure 8-1 (Device Numbering Conventions): Updated/Change figure to include "Die Revision"	96
• Section 8.7 (Device Identification Code Register): Added silicon revision B device identification code	97
• Section 8.8 (Die Identification Registers): Updated/Changed the DIEIDL and DIEIDH to point to the original registers at location 0xFFFFFF7C and 0xFFFFFF80	98

3 Device Comparison

Table 3-1 lists the features of the RM42L432 devices.

Table 3-1. RM42L432 Device Comparison(1)(2)

FEATURES	DEVICES							
Generic Part Number	RM46L852ZWT(3)	RM44L922ZWT	RM44L920PGE	RM44L920PZ	RM44L520PGE	RM44L520PZ	RM42L432PZ(3)	RM41L232PZ
Package	337 BGA	337 BGA	144 QFP	100 QFP	144 QFP	100 QFP	100 QFP	100 QFP
CPU	ARM Cortex-R4F	ARM Cortex-R4F	ARM Cortex-R4F	ARM Cortex-R4F	ARM Cortex-R4F	ARM Cortex-R4F	ARM Cortex-R4	ARM Cortex-R4
Frequency (MHz)	220	220	200	120	200	120	100	80
Flash (KB)	1260	1024	1024	1024	768	768	384	128
RAM (KB)	192	128	128	128	128	128	32	32
Data Flash [EEPROM] (KB)	64	64	64	64	64	64	16	16
USB OHCI + Device	2+0 or 1+1	–	–	–	–	–	–	–
EMAC	10/100	–	–	–	–	–	–	–
CAN	3	3	3	2	3	2	2	2
MibADC 12-bit (Ch)	2 (24ch)	2 (24ch)	2 (24ch)	2 (24ch)	2 (24ch)	2 (16ch)	1 (16ch)	1 (16ch)
N2HET (Ch)	2 (44)	2 (44)	2 (40)	2 (21)	2 (40)	2 (21)	1 (19)	1 (19)
ePWM Channels	14	14	14	8	14	8	–	–
eCAP Channels	6	6	6	4	6	4	–	–
eQEP Channels	2	2	2	1	2	1	1	1
MibSPI (CS)	3 (6 + 6 + 4)	3 (6 + 6 + 4)	3 (5 + 6 + 1)	2 (4 + 2)	3 (5 + 6 + 1)	2 (4 + 2)	1 (4)	1 (4)
SPI (CS)	2 (2 + 1)	2 (2 + 1)	1 (1)	1 (1)	1 (1)	1 (1)	2 (4 + 4)	2 (4 + 4)
SCI (LIN)	2 (1 with LIN)	2 (1 with LIN)	2 (1 with LIN)	1(with LIN)	2 (1 with LIN)	1(with LIN)	1(with LIN)	1(with LIN)
I2C	1	1	1	–	1	–	–	–
GPIO (INT)(4)	101 (with 16 interrupt capable)	101 (with 16 interrupt capable)	64 (with 16 interrupt capable)	45 (with 9 interrupt capable)	64 (with 10 interrupt capable)	45 (with 9 interrupt capable)	45 (with 8 interrupt capable)	45 (with 8 interrupt capable)
EMIF	16-bit data	–	–	–	–	–	–	–
ETM [Trace] (Data)	–	–	–	–	–	–	–	–
RTP/DMM (Data)	–	–	–	–	–	–	–	–
Operating Temperature	–40°C to 105°C	–40°C to 105°C	–40°C to 105°C	–40°C to 105°C	–40°C to 105°C	–40°C to 105°C	–40°C to 105°C	–40°C to 105°C
Core Supply (V)	1.14 V – 1.32 V	1.14 V – 1.32 V	1.14 V – 1.32 V	1.14 V – 1.32 V	1.14 V – 1.32 V	1.14 V – 1.32 V	1.14 V – 1.32 V	1.14 V – 1.32 V
I/O Supply (V)	3.0 V – 3.6 V	3.0 V – 3.6 V	3.0 V – 3.6 V	3.0 V – 3.6 V	3.0 V – 3.6 V	3.0 V – 3.6 V	3.0 V – 3.6 V	3.0 V – 3.6 V

(1) For additional device variants, see www.ti.com/rm
(2) This table reflects the maximum configuration for each peripheral. Some functions are multiplexed and not all pins are available at the same time.
(3) Superset device
(4) Total number of pins that can be used as general-purpose input or output when not used as part of a peripheral.

4 Terminal Configuration and Functions

4.1 PZ QFP Package Pinout (100-Pin)

Figure 4-1 shows the 100-pin PZ QFP package pinout.

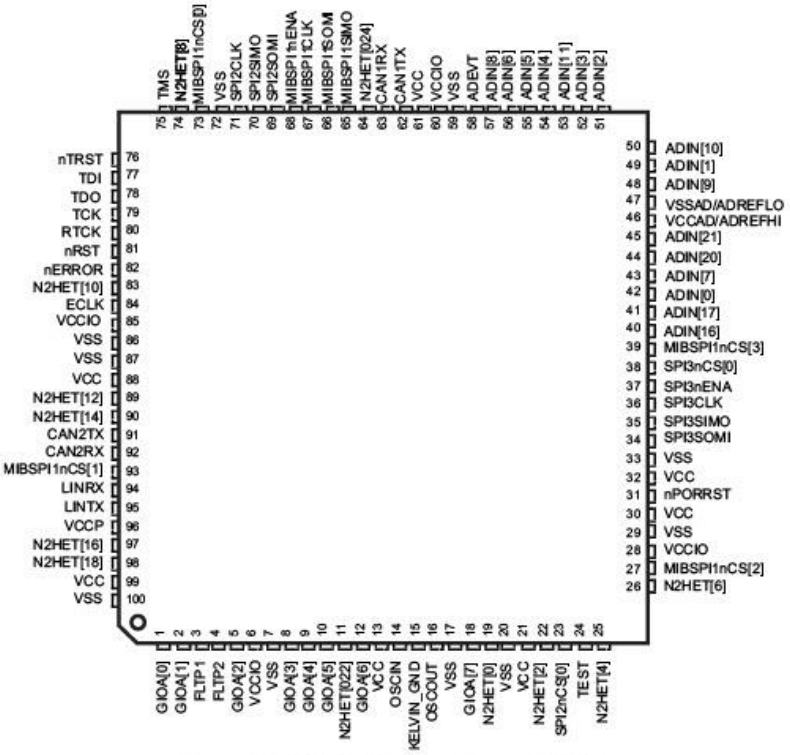


Figure 4-1. PZ QFP Package Pinout (100-Pin)

Note: Pins can have multiplexed functions. Only the default function is depicted in Figure 4-1.

4.2 Terminal Functions

Table 4-1 through Table 4-16 identify the external signal names, the associated pin numbers along with the mechanical package designator, the pin type (Input, Output, I/O, Power, or Ground), whether the pin has any internal pullup/pulldown, whether the pin can be configured as a GPIO, and a functional pin description.

NOTE

In the Terminal Functions table below, the "Reset Pull State" is the state of the pull applied to the terminal while nPORRST is low and immediately after nPORRST goes High. The default pull direction may change when software configures the pin for an alternate function. The "Pull Type" is the type of pull asserted when the signal name in bold is enabled for the given terminal by the IOMM control registers.

All I/O signals except nRST are configured as inputs while nPORRST is low and immediately after nPORRST goes High. While nPORRST is low, the input buffers are disabled, and the output buffers are disabled with the default pulls enabled.

All output-only signals have the output buffer disabled and the default pull enabled while nPORRST is low, and are configured as outputs with the pulls disabled immediately after nPORRST goes High.

4.2.1 High-End Timer (N2HET)

Table 4-1. High-End Timer (N2HET)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
N2HET[0]	19	I/O	Pulldown	Programmable, 20 µA	Timer input capture or output compare. The N2HET applicable terminals can be programmed as general-purpose input/output (GPIO). Each terminal has a suppression filter with a programmable duration.
N2HET[2]	22				
N2HET[4]	25				
N2HET[6]	26				
N2HET[8]	74				
N2HET[10]	83				
N2HET[12]	89				
N2HET[14]	90				
N2HET[16]	97				
MIBSPI1nCS[1]/EQEPS/ **N2HET[17]**	93				
N2HET[18]	98				
MIBSPI1nCS[2]/N2HET[20]/ **N2HET[19]**	27				
MIBSPI1nCS[2]/**N2HET[20]**/ N2HET[19]	27				
N2HET[22]	11				
N2HET[24]	64				
MIBSPI1nCS[3]/**N2HET[26]**	39				
ADEVT/**N2HET[28]**	58				
GIOA[7]/**N2HET[29]**	18				
MIBSPI1nENA/N2HET[23]/ **N2HET[30]**	68				
GIOA[6]/SPI2nCS[1]/**N2HET[31]**	12				

4.2.2 Enhanced Quadrature Encoder Pulse Modules (eQEP)

Table 4-2. Enhanced Quadrature Encoder Pulse Modules (eQEP)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
SPI3CLK/EQEPA	36	Input	Pullup	Fixed 20 µA	Enhanced QEP Input A
SPI3nENA/EQEPB	37	Input			Enhanced QEP Input B
SPI3nCS[0]/EQEPI	38	I/O			Enhanced QEP Index
MIBSPI1nCS[1]/EQEPS/N2HET [17]	93	I/O			Enhanced QEP Strobe

4.2.3 General-Purpose Input/Output (GPIO)

Table 4-3. General-Purpose Input/Output (GPIO)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
GIOA[0]/SPI3nCS[3]	1	I/O	Pulldown	Programmable, 20 µA	General-purpose input/output All GPIO terminals can generate interrupts to the CPU on rising/falling/both edges.
GIOA[1]/SPI3nCS[2]	2				
GIOA[2]/SPI3nCS[1]	5				
GIOA[3]/SPI2nCS[3]	8				
GIOA[4]/SPI2nCS[2]	9				
GIOA[5]/EXTCLKIN	10				
GIOA[6]/SPI2nCS[1]/N2HET[31]	12				
GIOA[7]/N2HET[29]	18				

4.2.4 Controller Area Network Interface Modules (DCAN1, DCAN2)

Table 4-4. Controller Area Network Interface Modules (DCAN1, DCAN2)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
CAN1RX	63	I/O	Pullup	Programmable, 20 µA	CAN1 Receive, or general-purpose I/O (GPIO)
CAN1TX	62				CAN1 Transmit, or GPIO
CAN2RX	92				CAN2 Receive, or GPIO
CAN2TX	91				CAN2 Transmit, or GPIO

4.2.5 Multibuffered Serial Peripheral Interface (MibSPI1)

Table 4-5. Multibuffered Serial Peripheral Interface (MibSPI1)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
MIBSPI1CLK	67	I/O	Pullup	Programmable, 20 µA	MibSPI1 Serial Clock, or GPIO
MIBSPI1nCS[0]	73				MibSPI1 Chip Select, or GPIO
MIBSPI1nCS[1]/EQEPS/N2HET[17]	93				
MIBSPI1nCS[2]/N2HET[20]/N2HET[19]	27				
MIBSPI1nCS[3]/N2HET[26]	39				
MIBSPI1nENA/N2HET[23]/N2HET[30]	68				MibSPI1 Enable, or GPIO
MIBSPI1SIMO	65				MibSPI1 Slave-In-Master-Out, or GPIO
MIBSPI1SOMI	66				MibSPI1 Slave-Out-Master-In, or GPIO

4.2.6 Standard Serial Peripheral Interface (SPI2)

Table 4-6. Standard Serial Peripheral Interface (SPI2)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
SPI2CLK	71	I/O	Pullup	Programmable, 20 µA	SPI2 Serial Clock, or GPIO
SPI2nCS[0]	23				SPI2 Chip Select, or GPIO
GIOA[6]/**SPI2nCS[1]**/N2HET[31]	12				
GIOA[4]/**SPI2nCS[2]**	9				
GIOA[3]/**SPI2nCS[3]**	8				
SPI2SIMO	70				SPI2 Slave-In-Master-Out, or GPIO
SPI2SOMI	69				SPI2 Slave-Out-Master-In, or GPIO
The drive strengths for the SPI2CLK, SPI2SIMO, and SPI2SOMI signals are selected individually by configuring the respective SRS bits of the SPIPC9 register fo SPI2. SRS = 0 for 8-mA drive (fast). This is the default mode as the SRS bits in the SPIPC9 register default to 0. SRS = 1 for 2-mA drive (slow)					
SPI3CLK/EQEPA	36	I/O	Pullup	Programmable, 20 µA	SPI3 Serial Clock, or GPIO
SPI3nCS[0]/EQEPI	38				SPI3 Chip Select, or GPIO
GIOA[2]/**SPI3nCS[1]**	5				
GIOA[1]/**SPI3nCS[2]**	2				
GIOA[0]/**SPI3nCS[3]**	1				
SPI3nENA/EQEPB	37				SPI3 Enable, or GPIO
SPI3SIMO	35				SPI3 Slave-In-Master-Out, or GPIO
SPI3SOMI	34				SPI3 Slave-Out-Master-In, or GPIO

4.2.7 Local Interconnect Network Controller (LIN)

Table 4-7. Local Interconnect Network Controller (LIN)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
LINRX	94	I/O	Pullup	Programmable, 20 µA	LIN Receive, or GPIO
LINTX	95				LIN Transmit, or GPIO

4.2.8 Multibuffered Analog-to-Digital Converter (MibADC)

Table 4-8. Multibuffered Analog-to-Digital Converter (MibADC)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
ADEVT/N2HET[28]	58	I/O	Pullup	Programmable, 20 µA	ADC event trigger or GPIO
ADIN[0]	42	Input	N/A	None	Analog inputs
ADIN[1]	49				
ADIN[2]	51				
ADIN[3]	52				
ADIN[4]	54				
ADIN[5]	55				
ADIN[6]	56				
ADIN[7]	43				
ADIN[8]	57				
ADIN[9]	48				
ADIN[10]	50				
ADIN[11]	53				
ADIN[16]	40				
ADIN[17]	41				
ADIN[20]	44				
ADIN[21]	45				
VCCAD/ADREFHI	46	Input/Power	N/A	None	ADC high reference level/ADC operating supply
VSSAD/ADREFLO	47	Input/Ground	N/A	None	ADC low reference level/ADC supply ground

4.2.9 System Module

Table 4-9. System Module

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
ECLK	84	I/O	Pulldown	Programmable, 20 µA	External prescaled clock output, or GPIO.
GIOA[5]/EXTCLKIN	10	Input	Pulldown	20 µA	External Clock In
nPORRST	31	Input	Pulldown	100 µA	Power-on reset, cold reset External power supply monitor circuitry must drive nPORRST low when any of the supplies to the microcontroller fall out of the specified range. This terminal has a glitch filter.
nRST	81	I/O	Pullup	100 µA	The external circuitry can assert a system reset by driving nRST low. To ensure that an external reset is not arbitrarily generated, TI recommends that an external pullup resistor is connected to this terminal. This terminal has a glitch filter.

4.2.10 Error Signaling Module (ESM)

Table 4-10. Error Signaling Module (ESM)

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
nERROR	82	I/O	Pulldown	20 µA	ESM error signal. Indicates error of high severity.

4.2.11 Main Oscillator

Table 4-11. Main Oscillator

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
OSCIN	14	Input	N/A	None	From external crystal/resonator, or external clock input
OSCOUT	16	Output	N/A	None	To external crystal/resonator
KELVIN_GND	15	Input	N/A	None	Dedicated ground for oscillator

4.2.12 Test/Debug Interface

Table 4-12. Test/Debug Interface

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
nTRST	76	Input	Pulldown	Fixed, 100 µA	JTAG test hardware reset
RTCK	80	Output	N/A	None	JTAG return test clock
TCK	79	Input	Pulldown	Fixed, 100 µA	JTAG test clock
TDI	77	I/O	Pullup	Fixed, 100 µA	JTAG test data in
TDO	78	Output	Fixed, 100-µA Pulldown	None	JTAG test data out
TMS	75	I/O	Pullup	Fixed, 100 µA	JTAG test select
TEST	24	I/O	Pulldown	Fixed, 100 µA	Test enable. This terminal must be connected to ground directly or through a pulldown resistor.

4.2.13 Flash

Table 4-13. Flash

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
FLTP1	3	Input	N/A	None	Flash test pins. For proper operation this terminal must connect only to a test pad or not be connected at all [no connect (NC)]. The test pad must not be exposed in the final product where it might be subjected to an ESD event.
FLTP2	4	Input	N/A	None	
VCCP	98	3.3-V Power	N/A	None	Flash external pump voltage (3.3 V). This terminal is required for both flash read and flash program and erase operations.

4.2.14 Core Supply

Table 4-14. Core Supply

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
VCC	13	1.2-V Power	N/A	None	Digital logic and RAM supply
VCC	21				
VCC	30				
VCC	32				
VCC	61				
VCC	88				
VCC	99				

4.2.15 I/O Supply

Table 4-15. I/O Supply

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
VCCIO	6	3.3-V Power	N/A	None	I/O supply
VCCIO	28				
VCCIO	60				
VCCIO	85				

4.2.16 Core and I/O Supply Ground Reference

Table 4-16. Core and I/O Supply Ground Reference

TERMINAL		SIGNAL TYPE	RESET PULL STATE	PULL TYPE	DESCRIPTION
SIGNAL NAME	100 PZ				
VSS	7	Ground	N/A	None	Device Ground Reference. This is a single ground reference for all supplies except for the ADC supply.
VSS	17				
VSS	20				
VSS	29				
VSS	33				
VSS	59				
VSS	72				
VSS	86				
VSS	87				
VSS	100				

4.3 Output Multiplexing and Control

Output multiplexing will be used in the device. The multiplexing is used to allow development of additional package and feature combinations as well as to maintain pinout compatibility with the marketing device family.

In all cases indicated as multiplexed, the output buffers are multiplexed.

4.3.1 Notes on Output Multiplexing

Table 4-17 shows the output signal multiplexing and control signals for selecting the desired functionality for each pin.

- The pins default to the signal defined by the DEFAULT FUNCTION column in Table 4-17
- The CONTROL 1, CONTROL 2, and CONTROL 3 columns indicate the multiplexing control register and the bit that must be set in order to select the corresponding functionality to be output on any particular pin.
  For example, consider the multiplexing on pin 18, shown in Table 4-18 .

Table 4-17. Output Mux Options

100 PZ PIN	DEFAULT FUNCTION	CONTROL 1	OPTION2	CONTROL 2	OPTION 3	CONTROL 3
1	GIOA[0]	PINMMR0[8]	SPI3nCS[3]	PINMMR0[9]	–	–
2	GIOA[1]	PINMMR1[0]	SPI3nCS[2]	PINMMR1[1]	–	–
5	GIOA[2]	PINMMR1[8]	SPI3nCS[1]	PINMMR1[9]	–	–
8	GIOA[3]	PINMMR1[16]	SPI2nCS[3]	PINMMR1[17]	–	–
9	GIOA[4]	PINMMR1[24]	SPI2nCS[2]	PINMMR1[25]	–	–
10	GIOA[5]	PINMMR2[0]	EXTCLKIN	PINMMR2[1]	–	–
12	GIOA[6]	PINMMR2[8]	SPI2nCS[1]	PINMMR2[9]	N2HET[31]	PINMMR2[10]
18	GIOA[7]	PINMMR2[16]	N2HET[29]	PINMMR2[17]	–	–
93	MIBSPI1nCS[1]	PINMMR6[8]	EQEPS	PINMMR6[9]	N2HET[17]	PINMMR6[10]
27	MIBSPI1nCS[2]	PINMMR3[0]	N2HET[20]	PINMMR3[1]	N2HET[19]	PINMMR3[2]
39	MIBSPI1nCS[3]	PINMMR4[8]	N2HET[26]	PINMMR4[9]	–	–
68	MIBSPI1nENA	PINMMR5[8]	N2HET[23]	PINMMR5[9]	N2HET[30]	PINMMR5[10]
36	SPI3CLK	PINMMR3[16]	EQEPA	PINMMR3[17]	–	–
38	SPI3nCS[0]	PINMMR4[0]	EQEPI	PINMMR4[1]	–	–
37	SPI3nENA	PINMMR3[24]	EQEPB	PINMMR3[25]	–	–
58	ADEVT	PINMMR4[16]	N2HET[28]	PINMMR4[17]	–	–

Table 4-18. Muxing Example

100 PZ PIN	DEFAULT FUNCTION	CONTROL 1	OPTION2	CONTROL 2	OPTION 3	CONTROL 3
18	GIOA[7]	PINMMR2[16]	N2HET[29]	PINMMR2[17]	–	–

- When GIOA[7] is configured as an output pin in the GPIO module control register, then the programmed output level appears on pin 18 by default. The PINMMR2[16] bit is set by default to indicate that the GIOA[7] signal is selected to be output.
- If the application must output the N2HET[29] signal on pin 18, it must clear PINMMR2[16] and set PINMMR2[17].
- The pin is connected as input to both the GPIO and N2HET modules. That is, there is no input multiplexing on this pin.

4.3.2 General Rules for Multiplexing Control Registers

- The PINMMR control registers can only be written in privileged mode. A write in a nonprivileged mode will generate an error response.
- If the application writes all 0s to any PINMMR control register, then the default functions are selected for the affected pins.
- Each byte in a PINMMR control register is used to select the functionality for a given pin. If the application sets more than 1 bit within a byte for any pin, then the default function is selected for this pin.
- Some bits within the PINMMR registers could be associated with internal pads that are not brought out in the 100-pin package. As a result, bits marked reserved should not be written as 1.

4.4 Special Multiplexed Options

Special controls are implemented to affect particular functions on this microcontroller. These controls are described in this section.

4.4.1 Filtering for eQEP Inputs

4.4.1.1 eQEPA Input

- When PINMMR8[0] = 1, the eQEPA input is double-synchronized using VCLK.
- When PINMMR8[0] = 0 and PINMMR8[1] = 1, the eQEPA input is double-synchronized and then qualified through a fixed 6-bit counter using VCLK.
- PINMMR8[0] = 0 and PINMMR8[1] = 0 is an illegal combination and behavior defaults to PINMMR8[0] = 1.

4.4.1.2 eQEPB Input

- When PINMMR8[8] = 1, the eQEPB input is double-synchronized using VCLK.
- When PINMMR8[8] = 0 and PINMMR8[9] = 1, the eQEPB input is double-synchronized and then qualified through a fixed 6-bit counter using VCLK.
- PINMMR8[8] = 0 and PINMMR8[9] = 0 is an illegal combination and behavior defaults to PINMMR8[8] = 1.

4.4.1.3 eQEPI Input

- When PINMMR8[16] = 1, the eQEPI input is double-synchronized using VCLK.
- When PINMMR8[16] = 0 and PINMMR8[17] = 1, the eQEPI input is double-synchronized and then qualified through a fixed 6-bit counter using VCLK.
- PINMMR8[16] = 0 and PINMMR8[17] = 0 is an illegal combination and behavior defaults to PINMMR8[16] = 1.

4.4.1.4 eQEPS Input

- When PINMMR8[24] = 1, the eQEPS input is double-synchronized using VCLK.
- When PINMMR8[24] = 0 and PINMMR8[25] = 1, the eQEPS input is double-synchronized and then qualified through a fixed 6-bit counter using VCLK.
- PINMMR8[24] = 0 and PINMMR8[25] = 0 is an illegal combination and behavior defaults to PINMMR8[24] = 1.

4.4.2 N2HET PIN_nDISABLE Input Port

- When PINMMR9[0] = 1, GIOA[5] is connected directly to N2HET PIN_nDISABLE input of the N2HET module.
- When PINMMR9[0] = 0 and PINMMR9[1] = 1, EQEPERR is inverted and double-synchronized using VCLK before connecting directly to the N2HET PIN_nDISABLE input of the N2HET module.
- PINMMR9[0] = 0 and PINMMR9[1] = 0 is an illegal combination and behavior defaults to PINMMR9[0] = 1.

5 Specifications

5.1 Absolute Maximum Ratings(1)

Over Operating Free-Air Temperature Range

		MIN	MAX	UNIT
Supply voltage	V_{CC}(2)	–0.3	1.43	V
	V_{CCIO}, V_{CCP}(2)	–0.3	4.6	
	V_{CCAD}	–0.3	3.6	
Input voltage	All input pins	–0.3	4.6	V
	ADC input pins	–0.3	4.6	
Input clamp current	I_{IK} (V_I < 0 or V_I > V_{CCIO}) All pins, except ADIN	–20	20	mA
	I_{IK} (V_I < 0 or V_I > V_{CCAD}) ADIN	–10	10	
	Total	–40	40	
Operating free-air temperature, T_A		–40	105	°C
Operating junction temperature, T_J		–40	130	°C
Latch-up performance	I-test, All I/O pins	–100	100	mA
Storage temperature, T_{stg}		–65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions* is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) Maximum-rated conditions for extended periods may affect device reliability. All voltage values are with respect to their associated grounds.

5.2 ESD Ratings

				VALUE	UNIT
$V_{(ESD)}$	Electrostatic discharge (ESD) performance:	Human Body Model (HBM), per ANSI/ESDA/JEDEC JS001(1)		±2	kV
		Charged Device Model (CDM), per JESD22-C101(2)	All pins	±250	V

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.
(2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process.

5.3 Power-On Hours (POH)(1)(2)

NOMINAL CORE VOLTAGE (V_{CC})	JUNCTION TEMPERATURE (Tj)	LIFETIME POH
1.2	105ºC	100K

(1) This information is provided solely for your convenience and does not extend or modify the warranty provided under TI's standard terms and conditions for TI semiconductor products.
(2) To avoid significant degradation, the device power-on hours (POH) must be limited to those specified in this table. To convert to equivalent POH for a specific temperature profile, see the *Calculating Equivalent Power-on-Hours for Hercules Safety MCUs* Application Report (SPNA207).

5.4 Recommended Operating Conditions[1]

		MIN	NOM	MAX	UNIT
V_{CC}	Digital logic supply voltage (Core)	1.14	1.2	1.32	V
V_{CCIO}	Digital logic supply voltage (I/O)	3	3.3	3.6	V
V_{CCAD} / $V_{ADREFHI}$	MibADC supply voltage / A-to-D high-voltage reference source	3	3.3	3.6	V
V_{CCP}	Flash pump supply voltage	3	3.3	3.6	V
V_{SS}	Digital logic supply ground		0		V
V_{SSAD} / $V_{ADREFLO}$	MibADC supply ground / A-to-D low-voltage reference source	–0.1		0.1	V
V_{SLEW}	Maximum positive slew rate for V_{CCIO}, V_{CCAD} and V_{CCP} supplies			1	V/µs
T_A	Operating free-air temperature	–40		105	°C
T_J	Operating junction temperature[2]	–40		130	°C

(1) All voltages are with respect to V_{SS}, except V_{CCAD}, which is with respect to V_{SSAD}
(2) Reliability data is based upon a temperature profile that is equivalent to 100,000 power-on hours at 105°C junction temperature.

5.5 Switching Characteristics Over Recommended Operating Conditions for Clock Domains

Table 5-1. Clock Domains Timing Specifications

	PARAMETER	CONDITIONS	MIN	MAX	UNIT
f_{HCLK}	HCLK - System clock frequency			100	MHz
f_{GCLK}	GCLK - CPU clock frequency (ratio f_{GCLK} : f_{HCLK} = 1:1)			f_{HCLK}	MHz
f_{VCLK}	VCLK - Primary peripheral clock frequency			100	MHz
f_{VCLK2}	VCLK2 - Secondary peripheral clock frequency			100	MHz
f_{VCLKA1}	VCLKA1 - Primary asynchronous peripheral clock frequency			100	MHz
f_{RTICLK}	RTICLK - clock frequency			f_{VCLK}	MHz

5.6 Wait States Required

The TCM RAM can support program and data fetches at full CPU speed without any address or data wait states required. There are no registers which need to be programmed for RAM wait states.

The TCM flash can support zero address and data wait states up to a CPU speed of 50 MHz in nonpipelined mode.The flash supports a maximum CPU clock speed of 100 MHz in pipelined mode with no address wait states and one data wait state.

The proper wait states should be set in the register fields *Address Setup Wait State Enable* (ASWSTEN 0xFFF87000[4]), *Random Wait states* (RWAIT 0xFFF87000[11:8]), and *Emulation Wait states* (EWAIT 0xFFF872B8[19:16]) as shown in Figure 5-1.

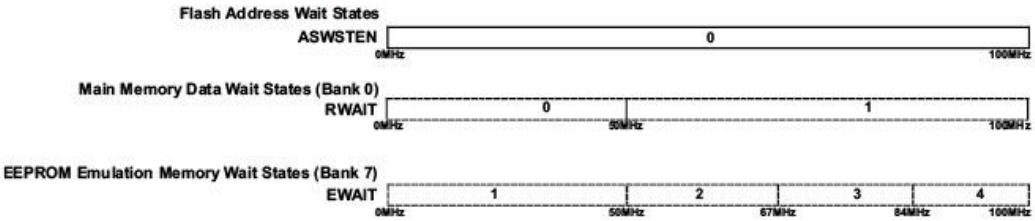


Figure 5-1. Wait States Scheme

The flash wrapper defaults to nonpipelined mode with address wait states disabled, ASWSTEN=0; the main memory random-read data wait state, RWAIT=1; and the emulation memory random-read wait states, EWAIT=1.

5.7 Power Consumption

Over Recommended Operating Conditions

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
I_{CC}	V_{CC} digital supply current (operating mode)	f_{HCLK} = 100 MHz f_{VCLK} = 100 MHz, Flash in pipelined mode, V_{CCmax}			150 (1)	mA
	V_{CC} digital supply current (LBIST mode)	LBIST clock rate = 50 MHz			165 (2)(3)	
	V_{CC} digital supply current (PBIST mode)	PBIST ROM clock frequency = 100 MHz			150 (2)(3)	
$I_{CCREFHI}$	AD_{REFHI} supply current (operating mode)	$AD_{REFHImax}$			3	mA
I_{CCAD}	V_{CCAD} supply current (operating mode)	$V_{CCADmax}$			45 (4)	mA
I_{CCIO}	V_{CCIO} digital supply current (operating mode)	No DC load, V_{CCmax}				
I_{CCP}	V_{CCP} pump supply current	Read mode				
I_{CCP}, I_{CCIO}, I_{CCAD}	3.3-V supply current	Read from one bank and program or erase another, V_{CCPmax}			65 (4)	mA

(1) The maximum I_{CC}, value can be derated
- linearly with voltage
- by 0.76 mA/MHz for lower operating frequency when $f_{HCLK} = f_{VCLK}$
- for lower junction temperature by the equation below where T_{JK} is the junction temperature in Kelvin and the result is in milliamperes. $36 - 0.001^{0.026T_{JK}}$

(2) The maximum I_{CC}, value can be derated
- linearly with voltage
- for lower junction temperature by the equation below where T_{JK} is the junction temperature in Kelvin and the result is in milliamperes. $36 - 0.001^{0.026T_{JK}}$

(3) LBIST and PBIST currents are for a short duration, typically less than 10 ms. They are usually ignored for thermal calculations for the device and the voltage regulator

(4) Maximum current requirement of the three combined supplies

5.8 Thermal Resistance Characteristics for PZ

Table 5-2 shows the thermal resistance characteristics for the PQFP - PZ mechanical packages.

Table 5-2. Thermal Resistance Characteristics (S-PQFP Package) [PZ]

PARAMETER	°C/W
$R_{\theta JA}$	48
$R_{\theta JC}$	5

5.9 Input/Output Electrical Characteristics[1]

Over Recommended Operating Conditions

PARAMETER			TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{hys}	Input hysteresis	All inputs		180			mV
V_{IL}	Low-level input voltage	All inputs[2]		–0.3		0.8	V
V_{IH}	High-level input voltage	All inputs[2]		2		V_{CCIO} + 0.3	V
V_{OL}	Low-level output voltage		$I_{OL} = I_{OLmax}$			0.2 V_{CCIO}	V
			I_{OL} = 50 µA, standard output mode			0.2	
V_{OH}	High-level output voltage		$I_{OH} = I_{OHmax}$	0.8 V_{CCIO}			V
			I_{OH} = 50 µA, standard output mode	V_{CCIO} - 0.3			
I_{IC}	Input clamp current (I/O pins)		$V_I < V_{SSIO}$ - 0.3 or $V_I > V_{CCIO}$ + 0.3	–3.5		3.5	mA
I_I	Input current (I/O pins)	I_{IH} 20-µA pulldown	$V_I = V_{CCIO}$	5		40	µA
		I_{IH} 100-µA pulldown	$V_I = V_{CCIO}$	40		195	
		I_{IL} 20-µA pullup	$V_I = V_{SS}$	–40		–5	
		I_{IL} 100-µA pullup	$V_I = V_{SS}$	–195		–40	
		All other pins	No pullup or pulldown	–1		1	
C_I	Input capacitance					2	pF
C_O	Output capacitance					3	pF

(1) Source currents (out of the device) are negative while sink currents (into the device) are positive.
(2) This does not apply to the nPORRST pin.

5.10 Output Buffer Drive Strengths

Table 5-3. Output Buffer Drive Strengths

LOW-LEVEL OUTPUT CURRENT, I_{OL} for $V_I=V_{OLmax}$ or HIGH-LEVEL OUTPUT CURRENT, I_{OH} for $V_I=V_{OHmin}$	SIGNALS
8 mA	EQEPI, EQEPS, TMS, TDI, TDO, RTCK, nERROR
4 mA	TEST, MIBSPI1SIMO, MIBSPI1SOMI, MIBSPI1CLK, SPI3CLK, SPI3SIMO, SPI3SOMI, nRST
2 mA zero-dominant	AD1EVT, CAN1RX, CAN1TX, CAN2RX, CAN2TX, GIOA[0-7], LINRX, LINTX, MIBSPI1nCS[0-3], MIBSPI1nENA N2HET[0], N2HET[2], N2HET[4], N2HET[6], N2HET[8], N2HET[10], N2HET[12], N2HET[14], N2HET[16], N2HET[18], N2HET[22], N2HET[24], SPI2nCS[0-3], SPI3nENA, SPI3nCS[0]
selectable 8 mA/ 2 mA	ECLK, SPI2CLK, SPI2SIMO, SPI2SOMI The default output buffer drive strength is 8 mA for these signals.

Table 5-4. Selectable 8 mA/ 2 mA Control

SIGNAL	CONTROL BIT	ADDRESS	8 mA	2 mA
ECLK	SYSPC10[0]	0xFFFF FF78	0	1
SPI2CLK	SPI2PC9[9]	0xFFF7 F668	0	1
SPI2SIMO	SPI2PC9[10]	0xFFF7 F668	0	1
SPI2SOMI	SPI2PC9[11](1)	0xFFF7 F668	0	1

(1) Either SPI2PC9[11] or SPI2PC9[24] can change the output strength of the SPI2SOMI pin. In case of a 32-bit write where these 2 bits differ, SPI2PC9[11] determines the drive strength.

5.11 Input Timings

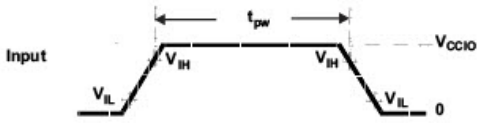


Figure 5-2. TTL-Level Inputs

Table 5-5. Timing Requirements for Inputs[1]

		MIN	MAX	UNIT
t_{pw}	Input minimum pulse width	$t_{c(VCLK)} + 10$[2]		ns

(1) $t_{c(VCLK)}$ = peripheral VBUS clock cycle time = 1 / $f_{(VCLK)}$
(2) The timing shown in Figure 5-2 is only valid for pin used in GIO mode.

5.12 Output Timings

Table 5-6. Switching Characteristics for Output Timings versus Load Capacitance (CL)

PARAMETER				MIN	MAX	UNIT
Rise time, t_r	8-mA pins		CL = 15 pF		2.5	ns
			CL = 50 pF		4	
			CL = 100 pF		7.2	
			CL = 150 pF		12.5	
Fall time, t_f			CL = 15 pF		2.5	
			CL = 50 pF		4	
			CL = 100 pF		7.2	
			CL = 150 pF		12.5	
Rise time, t_r	4-mA pins		CL = 15 pF		5.6	ns
			CL = 50 pF		10.4	
			CL = 100 pF		16.8	
			CL = 150 pF		23.2	
Fall time, t_f			CL = 15 pF		5.6	
			CL= 50 pF		10.4	
			CL = 100 pF		16.8	
			CL = 150 pF		23.2	
Rise time, t_r	2-mA-z pins		CL = 15 pF		8	ns
			CL = 50 pF		15	
			CL = 100 pF		23	
			CL = 150 pF		33	
Fall time, t_f			CL = 15 pF		8	
			CL = 50 pF		15	
			CL = 100 pF		23	
			CL = 150 pF		33	
Rise time, t_r	Selectable 8-mA/ 2-mA-z pins	8-mA mode	CL = 15 pF		2.5	ns
			CL = 50 pF		4	
			CL = 100 pF		7.2	
			CL = 150 pF		12.5	
Fall time, t_f			CL = 15 pF		2.5	
			CL = 50 pF		4	
			CL = 100 pF		7.2	
			CL = 150 pF		12.5	
Rise time, t_r		2-mA-z mode	CL = 15 pF		8	
			CL = 50 pF		15	
			CL = 100 pF		23	
			CL = 150 pF		33	
Fall time, t_f			CL = 15 pF		8	
			CL = 50 pF		15	
			CL = 100 pF		23	
			CL = 150 pF		33	

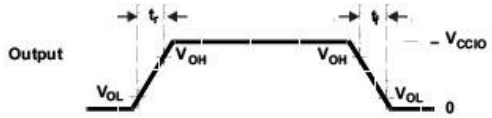


Figure 5-3. CMOS-Level Outputs

Table 5-7. Timing Requirements for Outputs[1]

PARAMETER		MIN	MAX	UNIT
$t_{d(parallel\_out)}$	Delay between low-to-high, or high-to-low transition of general-purpose output signals that can be configured by an application in parallel, for example, all signals in a GIOA port, or all N2HET signals.		5	ns

(1) This specification does not account for any output buffer drive strength differences or any external capacitive loading differences. Check Table 5-3 for output buffer drive strength information on each signal.

6 System Information and Electrical Specifications

6.1 Voltage Monitor Characteristics

A voltage monitor is implemented on this device. The purpose of this voltage monitor is to eliminate the requirement for a specific sequence when powering up the core and I/O voltage supplies.

6.1.1 Important Considerations

- The voltage monitor does not eliminate the need of a voltage supervisor circuit to ensure that the device is held in reset when the voltage supplies are out of range.
- The voltage monitor only monitors the core supply (VCC) and the I/O supply (VCCIO). The other supplies are not monitored by the VMON. For example, if the VCCAD or VCCP are supplied from a source different from that for VCCIO, then there is no internal voltage monitor for the VCCAD and VCCP supplies.

6.1.2 Voltage Monitor Operation

The voltage monitor generates the Power Good MCU signal (PGMCU) as well as the I/Os Power Good I/O signal (PGIO) on the device. During power up or power down, the PGMCU and PGIO are driven low when the core or I/O supplies are lower than the specified minimum monitoring thresholds. The PGIO and PGMCU being low isolates the core logic as well as the I/O controls during the power up or power down of the supplies. This allows the core and I/O supplies to be powered up or down in any order.

When the voltage monitor detects a low voltage on the I/O supply, it will assert a power-on reset. When the voltage monitor detects an out-of-range voltage on the core supply, it asynchronously makes all output pins high impedance, and asserts a power-on reset. The voltage monitor is disabled when the device enters a low power mode.

The VMON also incorporates a glitch filter for the nPORRST input. Refer to Section 6.2.3.1 for the timing information on this glitch filter.

Table 6-1. Voltage Monitoring Specifications

PARAMETER			MIN	TYP	MAX	UNIT
V_{MON}	Voltage monitoring thresholds	VCC low - VCC level below this threshold is detected as too low.	0.75	0.9	1.13	V
		VCC high - VCC level above this threshold is detected as too high.	1.40	1.7	2.1	
		VCCIO low - VCCIO level below this threshold is detected as too low.	1.85	2.4	2.9	

6.1.3 Supply Filtering

The VMON has the capability to filter glitches on the VCC and VCCIO supplies.

Table 6-2 shows the characteristics of the supply filtering. Glitches in the supply larger than the maximum specification cannot be filtered.

Table 6-2. VMON Supply Glitch Filtering Capability

PARAMETER	MIN	MAX	UNIT
Width of glitch on VCC that can be filtered	250	1000	ns
Width of glitch on VCCIO that can be filtered	250	1000	ns

6.2 Power Sequencing and Power-On Reset

6.2.1 Power-Up Sequence

There is no timing dependency between the ramp of the VCCIO and the VCC supply voltage. The power-up sequence starts with the I/O voltage rising above the minimum I/O supply threshold, (for more details, see Table 6-4), core voltage rising above the minimum core supply threshold, and the release of power-on reset. The high-frequency oscillator will start up first and its amplitude will grow to an acceptable level. The oscillator start-up time is dependent on the type of oscillator and is provided by the oscillator vendor. The different supplies to the device can be powered up in any order.

During power up, the device goes through the sequential phases listed in Table 6-3.

Table 6-3. Power-Up Phases

Oscillator start-up and validity check	1032 oscillator cycles
eFuse autoload	1160 oscillator cycles
Flash pump power up	688 oscillator cycles
Flash bank power up	617 oscillator cycles
Total	**3497 oscillator cycles**

The CPU reset is released at the end of this sequence and fetches the first instruction from address 0x00000000.

6.2.2 Power-Down Sequence

The different supplies to the device can be powered down in any order.

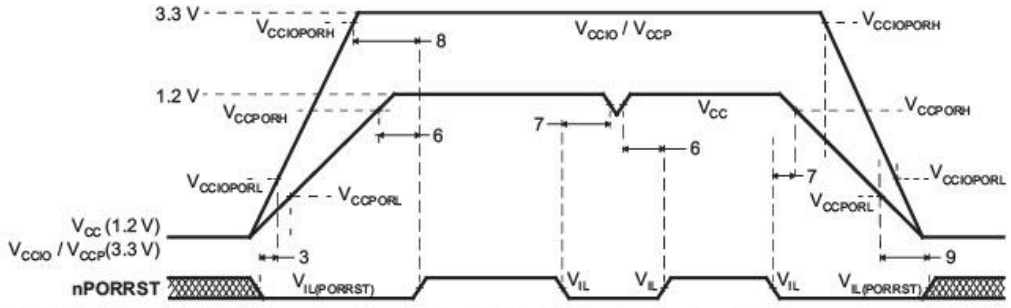
6.2.3 Power-On Reset: nPORRST

This reset must be asserted by an external circuitry whenever the I/O or core supplies are outside the recommended range. This signal has a glitch filter on it. It also has an internal pulldown.

6.2.3.1 nPORRST Electrical and Timing Requirements

Table 6-4. Electrical Requirements for nPORRST

NO.	PARAMETER		MIN	MAX	UNIT
	V_{CCPORL}	V_{CC} low supply level when nPORRST must be active during power up		0.5	V
	V_{CCPORH}	V_{CC} high supply level when nPORRST must remain active during power up and become active during power down	1.14		V
	$V_{CCIOPORL}$	V_{CCIO} / V_{CCP} low supply level when nPORRST must be active during power up		1.1	V
	$V_{CCIOPORH}$	V_{CCIO} / V_{CCP} high supply level when nPORRST must remain active during power up and become active during power down	3.0		V
	$V_{IL(PORRST)}$	Low-level input voltage of nPORRST V_{CCIO} > 2.5 V		0.2 * V_{CCIO}	V
		Low-level input voltage of nPORRST V_{CCIO} < 2.5 V		0.5	V
3	$t_{su(PORRST)}$	Setup time, nPORRST active before V_{CCIO} and V_{CCP} > $V_{CCIOPORL}$ during power up	0		ms
6	$t_{h(PORRST)}$	Hold time, nPORRST active after V_{CC} > V_{CCPORH}	1		ms
7	$t_{su(PORRST)}$	Setup time, nPORRST active before V_{CC} < V_{CCPORH} during power down	2		µs
8	$t_{h(PORRST)}$	Hold time, nPORRST active after V_{CCIO} and V_{CCP} > $V_{CCIOPORH}$	1		ms
9	$t_{h(PORRST)}$	Hold time, nPORRST active after V_{CC} < V_{CCPORL}	0		ms
	$t_{f(nPORRST)}$	Filter time nPORRST pin; Pulses less than MIN will be filtered out, pulses greater than MAX will generate a reset.	475	2000	ns



NOTE: There is no timing dependency between the ramp of the VCCIO and the VCC supply voltage; this is just an exemplary drawing.

Figure 6-1. nPORRST Timing Diagram

6.3 Warm Reset (nRST)

This is a bidirectional reset signal. The internal circuitry drives the signal low on detecting any device reset condition. An external circuit can assert a device reset by forcing the signal low. On this terminal, the output buffer is implemented as an open drain (drives low only). To ensure an external reset is not arbitrarily generated, TI recommends that an external pullup resistor is connected to this terminal.

This terminal has a glitch filter. It also has an internal pullup

6.3.1 Causes of Warm Reset

Table 6-5. Causes of Warm Reset

DEVICE EVENT	SYSTEM STATUS FLAG
Power-up reset	Exception Status Register, bit 15
Oscillator fail	Global Status Register, bit 0
PLL slip	Global Status Register, bits 8 and 9
Watchdog exception / Debugger reset	Exception Status Register, bit 13
CPU Reset (driven by the CPU STC)	Exception Status Register, bit 5
Software reset	Exception Status Register, bit 4
External reset	Exception Status Register, bit 3

6.3.2 nRST Timing Requirements

Table 6-6. nRST Timing Requirements

		MIN	MAX	UNIT
$t_{v(RST)}$	Valid time, nRST active after nPORRST inactive	$2256t_{c(OSC)}$ (1)		ns
	Valid time, nRST active (all other system reset conditions)	$32t_{c(VCLK)}$		
$t_{f(nRST)}$	Filter time nRST pin; Pulses less than MIN will be filtered out, pulses greater than MAX will generate a reset	475	2000	ns

(1) Assumes the oscillator has started up and stabilized before nPORRST is released.

6.4 ARM Cortex-R4 CPU Information

6.4.1 Summary of ARM Cortex-R4 CPU Features

The features of the ARM Cortex-R4 CPU include:

- An integer unit with integral Embedded ICE-RT logic.
- High-speed Advanced Microprocessor Bus Architecture (AMBA) Advanced eXtensible Interfaces (AXI) for Level two (L2) master and slave interfaces.
- Dynamic branch prediction with a global history buffer, and a 4-entry return stack
- Low interrupt latency.
- Nonmaskable interrupt.
- A Harvard Level one (L1) memory system with:
  - Tightly Coupled Memory (TCM) interfaces with support for error correction or parity checking memories
  - ARMv7-R architecture Memory Protection Unit (MPU) with 8 regions
- Dual core logic for fault detection in safety-critical applications.
- An L2 memory interface:
  - Single 64-bit master AXI interface
  - 64-bit slave AXI interface to TCM RAM blocks
- A debug interface to a CoreSight Debug Access Port (DAP).
- Six Hardware Breakpoints
- Two Watchpoints
- A Perfomance Monitoring Unit (PMU)
- A Vectored Interrupt Controller (VIC) port.

For more information on the ARM Cortex-R4 CPU, see www.arm.com.

6.4.2 ARM Cortex-R4 CPU Features Enabled by Software

The following CPU features are disabled on reset and must be enabled by the application if required.

- ECC On Tightly Coupled Memory (TCM) Accesses
- Hardware Vectored Interrupt (VIC) Port
- Memory Protection Unit (MPU)

6.4.3 Dual Core Implementation

The device has two Cortex-R4 cores, where the output signals of both CPUs are compared in the CCM-R4 unit. To avoid common mode impacts the signals of the CPUs to be compared are delayed by 2 clock cycles as shown in Figure 6-3.

The CPUs have a diverse CPU placement given by following requirements:

- Different orientation; for example, CPU1 = "north" orientation, CPU2 = "flip west" orientation
- Dedicated guard ring for each CPU

Figure 6-2. Dual - CPU Orientation

6.4.4 Duplicate clock tree after GCLK

The CPU clock domain is split into two clock trees, one for each CPU, with the clock of the 2nd CPU running at the same frequency and in phase to the clock of CPU1. See Figure 6-3.

6.4.5 ARM Cortex-R4 CPU Compare Module (CCM) for Safety

This device has two ARM Cortex-R4 CPU cores, where the output signals of both CPUs are compared in the CCM-R4 unit. To avoid common mode impacts the signals of the CPUs to be compared are delayed in a different way as shown in Figure 6-3.

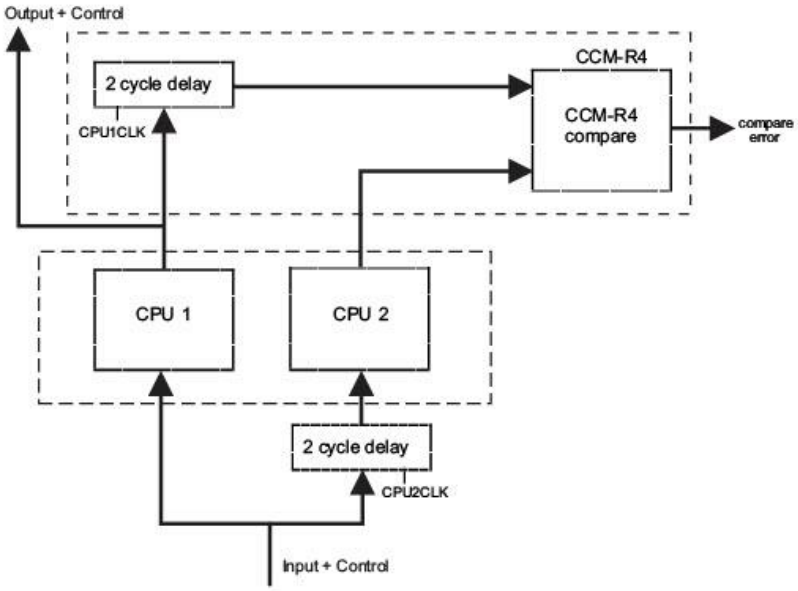


Figure 6-3. Dual Core Implementation

To avoid an erroneous CCM-R4 compare error, the application software must initialize the registers of both CPUs before the registers are used, including function calls where the register values are pushed onto the stack.

6.4.6 CPU Self-Test

The CPU STC (Self-Test Controller) is used to test the two Cortex-R4 CPU Cores using the Deterministic Logic BIST Controller as the test engine.

The main features of the self-test controller are:

- Ability to divide the complete test run into independent test intervals
- Capable of running the complete test or running a few intervals at a time
- Ability to continue from the last executed interval (test set) or to restart from the beginning (first test set)
- Complete isolation of the self-tested CPU core from the rest of the system during the self-test run
- Ability to capture the failure interval number
- Timeout counter for the CPU self-test run as a fail-safe feature

6.4.6.1 Application Sequence for CPU Self-Test

1. Configure clock domain frequencies.
2. Select the number of test intervals to be run.
3. Configure the timeout period for the self-test run.
4. Save the CPU state if required
5. Enable self-test.
6. Wait for CPU reset.
7. In the reset handler, read CPU self-test status to identify any failures.
8. Retrieve CPU state if required.

For more information, see the *RM42L42x 16/32-Bit RISC Flash Microcontroller Technical Reference Manual* (SPNU516).

6.4.6.2 CPU Self-Test Clock Configuration

The maximum clock rate for the self-test is 50 MHz. The STCCLK is divided down from the CPU clock, when necessary. This divider is configured by the STCCLKDIV register at address 0xFFFFE108.

6.4.6.3 CPU Self-Test Coverage

Table 6-7 shows CPU test coverage achieved for each self-test interval. It also lists the cumulative test cycles. The test time can be calculated by multiplying the number of test cycles with the STC clock period.

Table 6-7. CPU Self-Test Coverage

INTERVALS	TEST COVERAGE, %	TEST CYCLES
0	0	0
1	60.06	1365
2	68.71	2730
3	73.35	4095
4	76.57	5460
5	78.7	6825
6	80.4	8190
7	81.76	9555
8	82.94	10920
9	83.84	12285
10	84.58	13650
11	85.31	15015
12	85.9	16380
13	86.59	17745
14	87.17	19110
15	87.67	20475
16	88.11	21840
17	88.53	23205
18	88.93	24570
19	89.26	25935
20	89.56	27300
21	89.86	28665
22	90.1	30030
23	90.36	31395
24	90.62	32760

Table 6-7. CPU Self-Test Coverage (continued)

INTERVALS	TEST COVERAGE, %	TEST CYCLES
25	90.86	34125
26	91.06	35490

6.5 Clocks

6.5.1 Clock Sources

The table below lists the available clock sources on the device. Each of the clock sources can be enabled or disabled using the CSDISx registers in the system module. The clock source number in the table corresponds to the control bit in the CSDISx register for that clock source.

The table also shows the default state of each clock source.

Table 6-8. Available Clock Sources

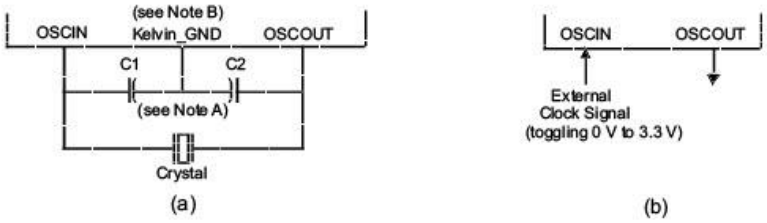
CLOCK SOURCE NO.	NAME	DESCRIPTION	DEFAULT STATE
0	OSCIN	Main Oscillator	Enabled
1	PLL1	Output From PLL1	Disabled
2	Reserved	Reserved	Disabled
3	EXTCLKIN1	External Clock Input #1	Disabled
4	CLK80K	Low-Frequency Output of Internal Reference Oscillator	Enabled
5	CLK10M	High-Frequency Output of Internal Reference Oscillator	Enabled
6	Reserved	Reserved	Disabled
7	Reserved	Reserved	Disabled

6.5.1.1 Main Oscillator

The oscillator is enabled by connecting the appropriate fundamental resonator/crystal and load capacitors across the external OSCIN and OSCOUT pins as shown in Figure 6-4. The oscillator is a single stage inverter held in bias by an integrated bias resistor. This resistor is disabled during leakage test measurement and low power modes.

TI strongly encourages each customer to submit samples of the device to the resonator/crystal vendors for validation. The vendors are equipped to determine what load capacitors will best tune their resonator/crystal to the microcontroller device for optimum start-up and operation over temperature/voltage extremes.

An external oscillator source can be used by connecting a 3.3 V clock signal to the OSCIN pin and leaving the OSCOUT pin unconnected (open) as shown in Figure 6-4.



Note A: The values of C1 and C2 should be provided by the resonator/crystal vendor.
Note B: Kelvin_GND should not be connected to any other GND.

Figure 6-4. Recommended Crystal/Clock Connection

6.5.1.1.1 Timing Requirements for Main Oscillator

Table 6-9. Timing Requirements for Main Oscillator

	PARAMETER	MIN	TYP	MAX	UNIT
tc(OSC)	Cycle time, OSCIN (when using a sine-wave input)	50		200	ns
tc(OSC_SQR)	Cycle time, OSCIN, (when input to the OSCIN is a square wave)	50		200	ns
tw(OSCIL)	Pulse duration, OSCIN low (when input to the OSCIN is a square wave)	15			ns
tw(OSCIH)	Pulse duration, OSCIN high (when input to the OSCIN is a square wave)	15			ns

6.5.1.2 Low-Power Oscillator

The Low-Power Oscillator (LPO) is comprised of two oscillators — HF LPO and LF LPO.

6.5.1.2.1 Features

The main features of the LPO are:

- Supplies a clock at extremely low power for power-saving modes. This is connected as clock source # 4 of the Global Clock Module.
- Supplies a high-frequency clock for nontiming-critical systems. This is connected as clock source # 5 of the Global Clock Module.
- Provides a comparison clock for the crystal oscillator failure detection circuit.

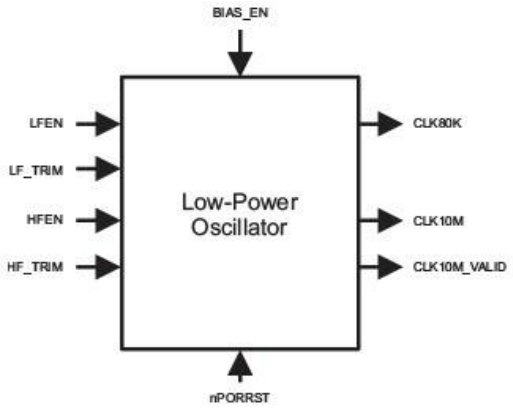


Figure 6-5. LPO Block Diagram

Figure 6-5 shows a block diagram of the internal reference oscillator. This is an LPO and provides two clock sources: one nominally 80 kHz and one nominally 10 MHz.

6.5.1.2.2 LPO Electrical and Timing Specifications

Table 6-10. LPO Specifications

PARAMETER		MIN	TYP	MAX	UNIT
Clock Detection	Oscillator fail frequency - lower threshold, using untrimmed LPO output	1.375	2.4	4.875	MHz
	Oscillator fail frequency - higher threshold, using untrimmed LPO output	22	38.4	78	
LPO - HF oscillator (f_{HFLPO})	Untrimmed frequency	5.5	9	19.5	MHz
	Trimmed frequency	8	9.6	11	MHz
	Start-up time from STANDBY (LPO BIAS_EN High for at least 900 µs)			10	µs
	Cold start-up time			900	µs
LPO - LF oscillator (f_{LFLPO})	Untrimmed frequency	36	85	180	kHz
	Start-up time from STANDBY (LPO BIAS_EN High for at least 900 µs)			100	µs
	Cold start-up time			2000	µs

6.5.1.3 Phase Locked Loop (PLL) Clock Modules

The PLL is used to multiply the input frequency to some higher frequency.

The main features of the PLL are:

- Frequency modulation can be optionally superimposed on the synthesized frequency of PLL.
- Configurable frequency multipliers and dividers.
- Built-in PLL Slip monitoring circuit.
- Option to reset the device on a PLL slip detection.

6.5.1.3.1 Block Diagram

Figure 6-6 shows a high-level block diagram of the PLL macro on this microcontroller.

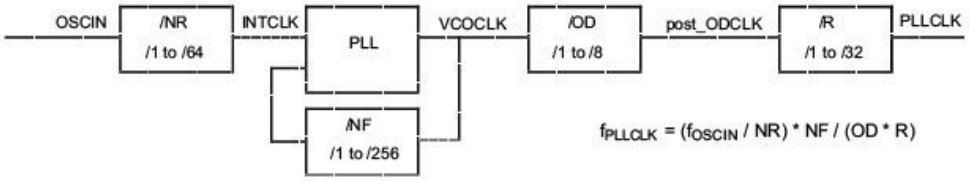


Figure 6-6. PLL Block Diagram

6.5.1.3.2 PLL Timing Specifications

Table 6-11. PLL Timing Specifications

PARAMETER		MIN	MAX	UNIT
f_{INTCLK}	PLL1 Reference Clock frequency	1	20	MHz
f_{post_ODCLK}	Post-ODCLK – PLL1 Post-divider input clock frequency		400	MHz
f_{VCOCLK}	VCOCLK – PLL1 Output Divider (OD) input clock frequency	150	550	MHz

6.5.2 Clock Domains

6.5.2.1 Clock Domain Descriptions

Table 6-12 lists the device clock domains and their default clock sources. The table also shows the system module control register that is used to select an available clock source for each clock domain.

Table 6-12. Clock Domain Descriptions

CLOCK DOMAIN NAME	DEFAULT CLOCK SOURCE	CLOCK SOURCE SELECTION REGISTER	DESCRIPTION
HCLK	OSCIN	GHVSRC	• Is disabled through the CDDISx registers bit 1
GCLK	OSCIN	GHVSRC	• Always the same frequency as HCLK • In phase with HCLK • Is disabled separately from HCLK through the CDDISx registers bit 0 • Can be divided by 1 up to 8 when running CPU self-test (LBIST) using the CLKDIV field of the STCCLKDIV register at address 0xFFFFE108
GCLK2	OSCIN	GHVSRC	• Always the same frequency as GCLK • 2 cycles delayed from GCLK • Is disabled along with GCLK • Gets divided by the same divider setting as that for GCLK when running CPU self-test (LBIST)
VCLK	OSCIN	GHVSRC	• Divided down from HCLK • Can be HCLK/1, HCLK/2, ... or HCLK/16 • Is disabled separately from HCLK through the CDDISx registers bit 2 • Can be disabled separately for eQEP using CDDISx registers bit 9
VCLK2	OSCIN	GHVSRC	• Divided down from HCLK • Can be HCLK/1, HCLK/2, ... or HCLK/16 • Frequency must be an integer multiple of VCLK frequency • Is disabled separately from HCLK through the CDDISx registers bit 3
VCLKA1	VCLK	VCLKASRC	• Defaults to VCLK as the source • Frequency can be as fast as HCLK frequency • Is disabled through the CDDISx registers bit 4
RTICLK	VCLK	RCLKSRC	• Defaults to VCLK as the source • If a clock source other than VCLK is selected for RTICLK, then the RTICLK frequency must be less than or equal to VCLK/3 – Application can ensure this by programming the RTI1DIV field of the RCLKSRC register, if necessary • Is disabled through the CDDISx registers bit 6

6.5.2.2 Mapping of Clock Domains to Device Modules

Each clock domain has a dedicated functionality as shown in the figure below.

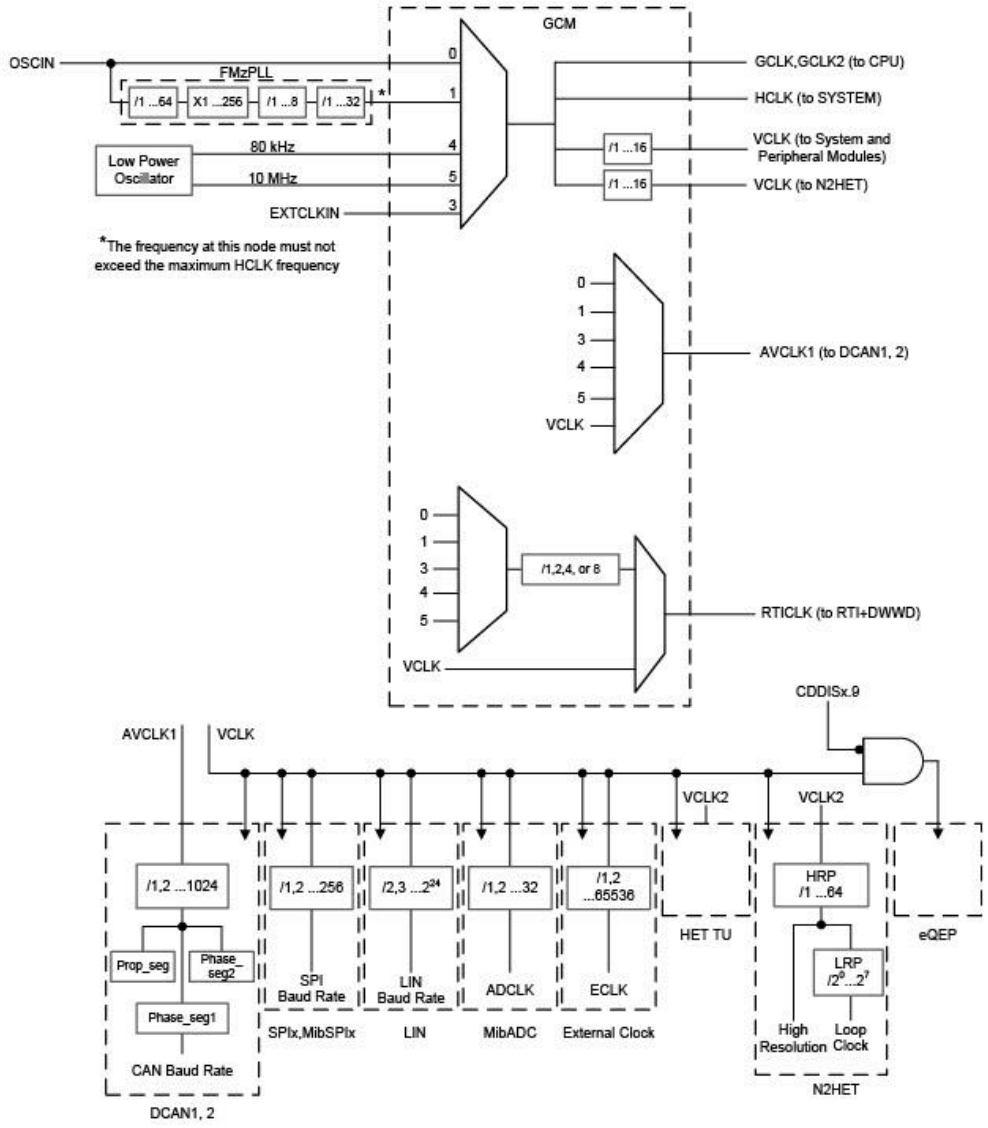


Figure 6-7. Device Clock Domains

6.5.3 Clock Test Mode

The RM4x platform architecture defines a special mode that allows various clock signals to be brought out on to the ECLK pin and N2HET[2] device outputs. This mode is called the Clock Test mode. It is very useful for debugging purposes and can be configured through the CLKTEST register in the system module.

Table 6-13. Clock Test Mode Options

CLKTEST[3-0]	SIGNAL ON ECLK		CLKTEST[11-8]	SIGNAL ON N2HET[2]
0000	Oscillator		0000	Oscillator Valid Status
0001	Main PLL free-running clock output (PLLCLK)		0001	Main PLL Valid status
0010	Reserved		0010	Reserved
0011	Reserved		0011	Reserved
0100	CLK80K		0100	Reserved
0101	CLK10M		0101	CLK10M Valid status
0110	Reserved		0110	Reserved
0111	Reserved		0111	Reserved
1000	GCLK		1000	CLK80K
1001	RTI Base		1001	Oscillator Valid status
1010	Reserved		1010	Oscillator Valid status
1011	VCLKA1		1011	Oscillator Valid status
1100	Reserved		1100	Oscillator Valid status
1101	Reserved		1101	Oscillator Valid status
1110	Reserved		1110	Oscillator Valid status
1111	Flash HD Pump Oscillator		1111	Oscillator Valid status

6.6 Clock Monitoring

The LPO Clock Detect (LPOCLKDET) module consists of a clock monitor (CLKDET) and an internal low-power oscillator (LPO).

The LPO provides two different clock sources – a low frequency (LFLPO) and a high frequency (HFLPO).

The CLKDET is a supervisor circuit for an externally supplied clock signal (OSCIN). In case the OSCIN frequency falls out of a frequency window, the CLKDET flags this condition in the global status register (GLBSTAT bit 0: OSC FAIL) and switches all clock domains sourced by OSCIN to the HFLPO clock (limp mode clock).

The valid OSCIN frequency range is defined as: $f_{HFLPO} / 4 < f_{OSCIN} < f_{HFLPO} * 4$.

6.6.1 Clock Monitor Timings

For more information on LPO and Clock detection, refer to Table 6-10.

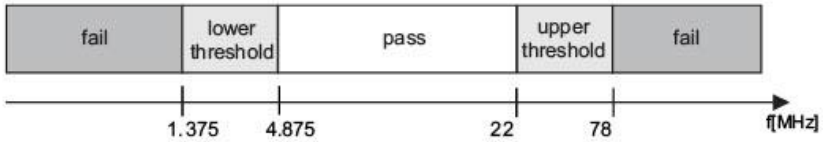


Figure 6-8. LPO and Clock Detection, Untrimmed HFLPO

6.6.2 External Clock (ECLK) Output Functionality

The ECLK pin can be configured to output a prescaled clock signal indicative of an internal device clock. This output can be externally monitored as a safety diagnostic.

6.6.3 Dual Clock Comparator

The Dual Clock Comparator (DCC) module determines the accuracy of selectable clock sources by counting the pulses of two independent clock sources (counter 0 and counter 1). If one clock is out of spec, an error signal is generated. For example, the DCC can be configured to use CLK10M as the reference clock (for counter 0) and VCLK as the "clock under test" (for counter 1). This configuration allows the DCC to monitor the PLL output clock when VCLK is using the PLL output as its source.

6.6.3.1 Features

- Takes two different clock sources as input to two independent counter blocks.
- One of the clock sources is the known-good, or reference clock; the second clock source is the "clock under test."
- Each counter block is programmable with initial, or seed values.
- The counter blocks start counting down from their seed values at the same time; a mismatch from the expected frequency for the clock under test generates an error signal which is used to interrupt the CPU.

6.6.3.2 Mapping of DCC Clock Source Inputs

Table 6-14. DCC Counter 0 Clock Sources

TEST MODE	CLOCK SOURCE [3:0]	CLOCK NAME
0	Others	Oscillator (OSCIN)
	0x5	High-frequency LPO
	0xA	Test clock (TCK)
1	X	VCLK

Table 6-15. DCC Counter 1 Clock Sources

TEST MODE	KEY [3:0]	CLOCK SOURCE [3:0]	CLOCK NAME
0	Others	–	N2HET[31]
	0xA	0x0	Main PLL free-running clock output
		0x1	n/a
		0x2	Low-frequency LPO
		0x3	High-frequency LPO
		0x4	Flash HD pump oscillator
		0x5	EXTCLKIN
		0x6	n/a
		0x7	Ring oscillator
		0x8 - 0xF	VCLK
1	X	X	HCLK

6.7 Glitch Filters

A glitch filter is present on the following signals.

Table 6-16. Glitch Filter Timing Specifications

PIN	PARAMETER		MIN	MAX	UNIT
nPORRST	$t_{h(nPORRST)}$	Filter time nPORRST pin: pulses less than MIN will be filtered out, pulses greater than MAX will generate a reset(1)	475	2000	ns
nRST	$t_{h(nRST)}$	Filter time nRST pin: pulses less than MIN will be filtered out, pulses greater than MAX will generate a reset	475	2000	ns
TEST	$t_{h(TEST)}$	Filter time TEST pin: pulses less than MIN will be filtered out, pulses greater than MAX will pass through	475	2000	ns

(1) The glitch filter design on the nPORRST signal is designed such that no size pulse will reset any part of the microcontroller (flash pump, I/O pins, and so forth) without also generating a valid reset signal to the CPU.

6.8 Device Memory Map

6.8.1 Memory Map Diagram

Figure 6-9 shows the device memory map.

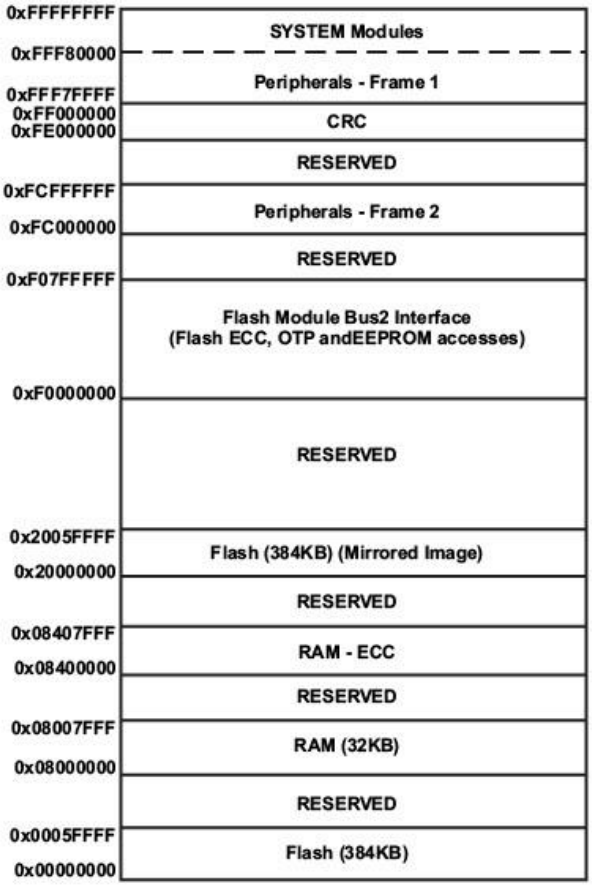


Figure 6-9. RM42LS432 Memory Map

The Flash memory in all configurations is mirrored to support ECC logic testing. The base address of the mirrored Flash image is 0x2000 0000.

6.8.2 Memory Map Table

See Figure 1-1 for a block diagram showing the device interconnects.

Table 6-17. Device Memory Map

MODULE NAME	FRAME CHIP SELECT	ADDRESS RANGE		FRAME SIZE	ACTUAL SIZE	RESPONSE FOR ACCESS TO UNIMPLEMENTED LOCATIONS IN FRAME
		START	END			
Memories tightly coupled to the ARM Cortex-R4 CPU						
TCM Flash	CS0	0x0000_0000	0x00FF_FFFF	16MB	384KB	Abort
TCM RAM + RAM ECC	CSRAM0	0x0800_0000	0x0BFF_3FFF	64MB	32KB	
Mirrored Flash	Flash mirror frame	0x2000_0000	0x20FF_FFFF	16MB	384KB	
Flash Module Bus2 Interface						
Customer OTP, TCM Flash Banks		0xF000_0000	0xF000_07FF	64KB	2KB	Abort
Customer OTP, EEPROM Bank		0xF000_E000	0xF000_E3FF		1KB	
Customer OTP–ECC, TCM Flash Banks		0xF004_0000	0xF004_00FF	8KB	256B	
Customer OTP–ECC, EEPROM Bank		0xF004_1C00	0xF004_1C7F		128B	
TI OTP, TCM Flash Banks		0xF008_0000	0xF008_07FF	64KB	2KB	
TI OTP, EEPROM Bank		0xF008_E000	0xF008_E3FF		1KB	
TI OTP–ECC, TCM Flash Banks		0xF00C_0000	0xF00C_00FF	8KB	256B	
TI OTP–ECC, EEPROM Bank		0xF00C_1C00	0xF00C_1C7F		128B	
EEPROM Bank–ECC		0xF010_0000	0xF010_07FF	256KB	2KB	
EEPROM Bank		0xF020_0000	0xF020_3FFF	2MB	16KB	
Flash Data Space ECC		0xF040_0000	0xF040_DFFF	1MB	48KB	
Cyclic Redundancy Checker (CRC) Module Registers						
CRC	CRC frame	0xFE00_0000	0xFEFF_FFFF	16MB	512B	Accesses above 0x200 generate abort.
Peripheral Memories						
MIBSPI1 RAM	PCS[7]	0xFF0E_0000	0xFF0F_FFFF	128KB	2KB	Abort for accesses above 2KB
DCAN2 RAM	PCS[14]	0xFF1C_0000	0xFF1D_FFFF	128KB	2KB	Wrap around for accesses to unimplemented address offsets lower than 0x7FF. Abort generated for accesses beyond offset 0x800.
DCAN1 RAM	PCS[15]	0xFF1E_0000	0xFF1F_FFFF	128KB	2KB	Wrap around for accesses to unimplemented address offsets lower than 0x7FF. Abort generated for accesses beyond offset 0x800.
MIBADC RAM	PCS[31]	0xFF3E_0000	0xFF3F_FFFF	128KB	8KB	Wrap around for accesses to unimplemented address offsets lower than 0x1FFF.
MIBADC Look-Up Table					384 bytes	Look-up table for ADC wrapper. Starts at offset 0x2000 ans ends at 0x217F. Wrap around for accesses between offsets 0x180 and 0x3FFF. Aborts generated for accesses beyond 0x4000

Table 6-17. Device Memory Map (continued)

MODULE NAME	FRAME CHIP SELECT	ADDRESS RANGE		FRAME SIZE	ACTUAL SIZE	RESPONSE FOR ACCESS TO UNIMPLEMENTED LOCATIONS IN FRAME
		START	END			
N2HET RAM	PCS[35]	0xFF46_0000	0xFF47_FFFF	128KB	16KB	Wrap around for accesses to unimplemented address offsets lower than 0x3FFF. Abort generated for accesses beyond 0x3FFF.
HTU RAM	PCS[39]	0xFF4E_0000	0xFF4F_FFFF	128KB	1KB	Abort
Debug Components						
CoreSight Debug ROM	CSCS0	0xFFA0_0000	0xFFA0_0FFF	4KB	4KB	Reads return zeros, writes have no effect
Cortex-R4 Debug	CSCS1	0xFFA0_1000	0xFFA0_1FFF	4KB	4KB	Reads return zeros, writes have no effect
Peripheral Control Registers						
HTU	PS[22]	0xFFF7_A400	0xFFF7_A4FF	256B	256B	Reads return zeros, writes have no effect
N2HET	PS[17]	0xFFF7_B800	0xFFF7_B8FF	256B	256B	Reads return zeros, writes have no effect
GIO	PS[16]	0xFFF7_BC00	0xFFF7_BCFF	256B	256B	Reads return zeros, writes have no effect
MIBADC	PS[15]	0xFFF7_C000	0xFFF7_C1FF	512B	512B	Reads return zeros, writes have no effect
DCAN1	PS[8]	0xFFF7_DC00	0xFFF7_DDFF	512B	512B	Reads return zeros, writes have no effect
DCAN2	PS[8]	0xFFF7_DE00	0xFFF7_DFFF	512B	512B	Reads return zeros, writes have no effect
LIN	PS[6]	0xFFF7_E400	0xFFF7_E4FF	256B	256B	Reads return zeros, writes have no effect
MibSPI1	PS[2]	0xFFF7_F400	0xFFF7_F5FF	512B	512B	Reads return zeros, writes have no effect
SPI2	PS[2]	0xFFF7_F600	0xFFF7_F7FF	512B	512B	Reads return zeros, writes have no effect
SPI3	PS[1]	0xFFF7_F800	0xFFF7_F9FF	512B	512B	Reads return zeros, writes have no effect
EQEP	PS[25]	0xFFF7_9900	0xFFF7_99FF	256B	256B	Reads return zeros, writes have no effect
EQEP (Mirrored)	PS2[25]	0xFCF7_9900	0xFCF7_99FF	256B	256B	Reads return zeros, writes have no effect
System Modules Control Registers and Memories						
VIM RAM	PPCS2	0xFFF8_2000	0xFFF8_2FFF	4KB	1KB	Wrap around for accesses to unimplemented address offsets lower than 0x3FF. Accesses beyond 0x3FF will be ignored.
Flash Wrapper	PPCS7	0xFFF8_7000	0xFFF8_7FFF	4KB	4KB	Abort
eFuse Farm Controller	PPCS12	0xFFF8_C000	0xFFF8_CFFF	4KB	4KB	Abort
PCR registers	PPS0	0xFFFF_E000	0xFFFF_E0FF	256B	256B	Reads return zeros, writes have no effect
System Module - Frame 2 (see device TRM)	PPS0	0xFFFF_E100	0xFFFF_E1FF	256B	256B	Reads return zeros, writes have no effect
PBIST	PPS1	0xFFFF_E400	0xFFFF_E5FF	512B	512B	Reads return zeros, writes have no effect
STC	PPS1	0xFFFF_E600	0xFFFF_E6FF	256B	256B	Reads return zeros, writes have no effect
IOMM Multiplexing control module	PPS2	0xFFFF_EA00	0xFFFF_EBFF	512B	512B	Generates address error interrupt if enabled.

Table 6-17. Device Memory Map (continued)

MODULE NAME	FRAME CHIP SELECT	ADDRESS RANGE		FRAME SIZE	ACTUAL SIZE	RESPONSE FOR ACCESS TO UNIMPLEMENTED LOCATIONS IN FRAME
		START	END			
DCC	PPS3	0xFFFF_EC00	0xFFFF_ECFF	256B	256B	Reads return zeros, writes have no effect
ESM	PPS5	0xFFFF_F500	0xFFFF_F5FF	256B	256B	Reads return zeros, writes have no effect
CCMR4	PPS5	0xFFFF_F600	0xFFFF_F6FF	256B	256B	Reads return zeros, writes have no effect
RAM ECC even	PPS8	0xFFFF_F800	0xFFFF_F8FF	256B	256B	Reads return zeros, writes have no effect
RAM ECC odd	PPS8	0xFFFF_F900	0xFFFF_F9FF	256B	256B	Reads return zeros, writes have no effect
RTI + DWWD	PPS7	0xFFFF_FC00	0xFFFF_FCFF	256B	256B	Reads return zeros, writes have no effect
VIM Parity	PPS7	0xFFFF_FD00	0xFFFF_FDFF	256B	256B	Reads return zeros, writes have no effect
VIM	PPS7	0xFFFF_FE00	0xFFFF_FEFF	256B	256B	Reads return zeros, writes have no effect
System Module - Frame 1 (see device TRM)	PPS7	0xFFFF_FF00	0xFFFF_FFFF	256B	256B	Reads return zeros, writes have no effect

6.8.3 Master/Slave Access Privileges

The table below lists the access permissions for each bus master on the device. A bus master is a module that can initiate a read or a write transaction on the device.

Each slave module on the main interconnect is listed in the table. A "Yes" indicates that the module listed in the "MASTERS" column can access that slave module.

Table 6-18. Master / Slave Access Matrix

MASTERS	ACCESS MODE	SLAVES ON MAIN SCR			
		Flash Module Bus2 Interface: OTP, ECC, EEPROM Bank	Non-CPU Accesses to Program Flash and CPU Data RAM	CRC	Peripheral Control Registers, All Peripheral Memories, And All System Module Control Registers And Memories
CPU READ	User/Privilege	Yes	Yes	Yes	Yes
CPU WRITE	User/Privilege	No	Yes	Yes	Yes
HTU	Privilege	No	Yes	Yes	Yes

6.9 Flash Memory

6.9.1 Flash Memory Configuration

Flash Bank: A separate block of logic consisting of 1 to 16 sectors. Each flash bank normally has a customer-OTP and a TI-OTP area. These flash sectors share input/output buffers, data paths, sense amplifiers, and control logic.

Flash Sector: A contiguous region of flash memory which must be erased simultaneously due to physical construction constraints.

Flash Pump: A charge pump which generates all the voltages required for reading, programming, or erasing the flash banks.

Flash Module: Interface circuitry required between the host CPU and the flash banks and pump module.

Table 6-19. Flash Memory Banks and Sectors

MEMORY ARRAYS (or BANKS)	SECTOR NO.	SEGMENT	LOW ADDRESS	HIGH ADDRESS
BANK0 (384KB)(1)	0	8KB	0x0000_0000	0x0000_1FFF
	1	8KB	0x0000_2000	0x0000_3FFF
	2	8KB	0x0000_4000	0x0000_5FFF
	3	8KB	0x0000_6000	0x0000_7FFF
	4	8KB	0x0000_8000	0x0000_9FFF
	5	8KB	0x0000_A000	0x0000_BFFF
	6	8KB	0x0000_C000	0x0000_DFFF
	7	8KB	0x0000_E000	0x0000_FFFF
	8	8KB	0x0001_0000	0x0001_1FFF
	9	8KB	0x0001_2000	0x0001_3FFF
	10	8KB	0x0001_4000	0x0001_5FFF
	11	8KB	0x0001_6000	0x0001_7FFF
	12	32KB	0x0001_8000	0x0001_FFFF
	13	128KB	0x0002_0000	0x0003_FFFF
	14	128KB	0x0004_0000	0x0005_FFFF
BANK7 (16KB) for EEPROM emulation(2)(3)	0	4KB	0xF020_0000	0xF020_0FFF
	1	4KB	0xF020_1000	0xF020_1FFF
	2	4KB	0xF020_2000	0xF020_2FFF
	3	4KB	0xF020_3000	0xF020_3FFF

(1) This Flash bank is 144-bit wide with ECC support.
(2) Flash bank7 is an FLEE bank and can be programmed while executing code from flash bank0. It is 72-bit wide with ECC support.
(3) Code execution is not allowed from flash bank7.

6.9.2 Main Features of Flash Module

- Support for multiple flash banks for program and/or data storage
- Simultaneous read access on a bank while performing program or erase operation on any other bank
- Integrated state machines to automate flash erase and program operations
- Software interface for flash program and erase operations
- Pipelined mode operation to improve instruction access interface bandwidth
- Support for Single Error Correction Double Error Detection (SECDED) block inside Cortex-R4 CPU
  - Error address is captured for host system debugging
- Support for a rich set of diagnostic features

6.9.3 ECC Protection for Flash Accesses

All accesses to the program flash memory are protected by Single Error Correction Double Error Detection (SECDED) logic embedded inside the CPU. The flash module provides 8 bits of ECC code for 64 bits of instructions or data fetched from the flash memory. The CPU calculates the expected ECC code based on the 64 bits received and compares it with the ECC code returned by the flash module. A single-bit error is corrected and flagged by the CPU, while a multibit error is only flagged. The CPU signals an ECC error through its Event bus. This signaling mechanism is not enabled by default and must be enabled by setting the "X" bit of the Performance Monitor Control Register, c9.

```
MRC p15,#0,r1,c9,c12,#0    ;Enabling Event monitor states
ORR r1, r1, #0x00000010
MCR p15,#0,r1,c9,c12,#0    ;Set 4th bit ('X') of PMNC register
MRC p15,#0,r1,c9,c12,#0
```

The application must also explicitly enable the CPU's ECC checking for accesses on the CPU's ATCM and BTCM interfaces. These are connected to the program flash and data RAM respectively. ECC checking for these interfaces can be done by setting the B1TCMPCEN, B0TCMPCEN and ATCMPCEN bits of the System Control coprocessor's Auxiliary Control Register, c1.

```
MRC p15, #0, r1, c1, c0, #1
ORR r1, r1, #0x0e000000        ;Enable ECC checking for ATCM and BTCMs
DMB
MCR p15, #0, r1, c1, c0, #1
```

6.9.4 Flash Access Speeds

For information on flash memory access speeds and the relevant wait states required, see Section 5.6.

6.10 Flash Program and Erase Timings for Program Flash

Table 6-20. Timing Specifications for Program Flash

PARAMETER			MIN	NOM	MAX	UNIT
t_{prog} (144bit)	Wide Word (144 bit) programming time			40	300	µs
t_{prog} (Total)	384KByte programming time(1)	-40°C to 105°C			4	s
		0°C to 60°C, for first 25 cycles		1	2	
t_{erase}	Sector/Bank erase time(2)	-40°C to 105°C		0.30	4	s
		0°C to 60°C, for first 25 cycles		16	100	ms
t_{wec}	Write/erase cycles with 15 year Data Retention requirement	-40°C to 105°C			1000	cycles

(1) This programming time includes overhead of state machine, but does not include data transfer time. The programming time assumes programming 144 bits at a time at the maximum specified operating frequency.
(2) During bank erase, the selected sectors are erased simultaneously. The time to erase the bank is specified as equal to the time to erase a sector.

6.11 Flash Program and Erase Timings for Data Flash

Table 6-21. Timing Specifications for Data Flash

PARAMETER			MIN	NOM	MAX	UNIT
t_{prog} (72 bit)	Wide Word (72 bit) programming time			47	300	µs
t_{prog} (Total)	16KB programming time(1)	–40°C to 105°C			330	ms
		0°C to 60°C, for first 25 cycles		100	165	
t_{erase}	Sector/Bank erase time(2)	–40°C to 105°C		0.200	8	s
		0°C to 60°C, for first 25 cycles		14	100	ms
t_{wec}	Write/erase cycles with 15 year Data Retention requirement	–40°C to 105°C			100000	cycles

(1) This programming time includes overhead of state machine, but does not include data transfer time. The programming time assumes programming 72 bits at a time at the maximum specified operating frequency.
(2) During bank erase, the selected sectors are erased simultaneously. The time to erase the bank is specified as equal to the time to erase a sector.

6.12 Tightly Coupled RAM Interface Module

Figure 6-10 illustrates the connection of the Tightly Coupled RAM (TCRAM) to the Cortex-R4 CPU.

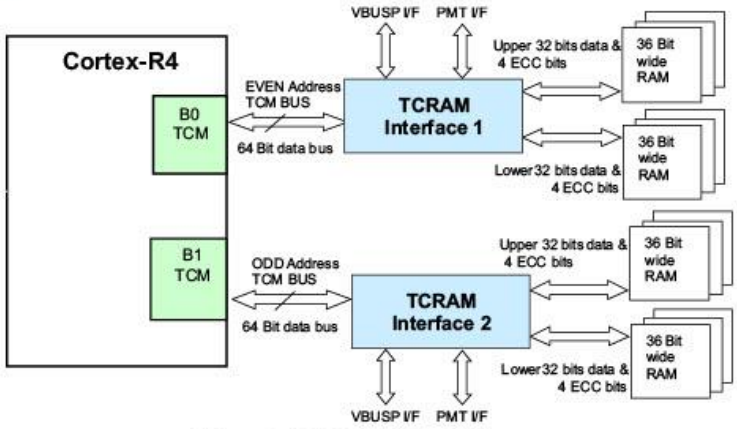


Figure 6-10. TCRAM Block Diagram

6.12.1 Features

The features of the Tightly Coupled RAM (TCRAM) module are:

- Acts as slave to the BTCM interface of the Cortex-R4 CPU
- Supports CPU's internal ECC scheme by providing 64-bit data and 8-bit ECC code
- Monitors CPU Event Bus and generates single-bit or multibit error interrupts
- Stores addresses for single-bit and multibit errors
- Provides CPU address bus integrity checking by supporting parity checking on the address bus
- Performs redundant address decoding for the RAM bank chip select and ECC select generation logic
- Provides enhanced safety for the RAM addressing by implementing two 36-bit wide byte-interleaved RAM banks and generating independent RAM access control signals to the two banks
- Supports auto-initialization of the RAM banks along with the ECC bits
- No support for bit-wise RAM accesses

6.12.2 TCRAMW ECC Support

The TCRAMW passes on the ECC code for each data read by the Cortex-R4 CPU from the RAM. It also stores the CPU's ECC port contents in the ECC RAM when the CPU does a write to the RAM. The TCRAMW monitors the CPU's event bus and provides registers for indicating single-bit and multibit errors and also for identifying the address that caused the single-bit or multibit error. The event signaling and the ECC checking for the RAM accesses must be enabled inside the CPU.

For more information see the device Technical Reference Manual.

6.13 Parity Protection for Accesses to peripheral RAMs

Accesses to some peripheral RAMs are protected by odd/even parity checking. During a read access the parity is calculated based on the data read from the peripheral RAM and compared with the good parity value stored in the parity RAM for that peripheral. If any word fails the parity check, the module generates a parity error signal that is mapped to the Error Signaling Module. The module also captures the peripheral RAM address that caused the parity error.

The parity protection for peripheral RAMs is not enabled by default and must be enabled by the application. Each individual peripheral contains control registers to enable the parity protection for accesses to its RAM.

NOTE

The CPU read access gets the actual data from the peripheral. The application can choose to generate an interrupt whenever a peripheral RAM parity error is detected.

6.14 On-Chip SRAM Initialization and Testing

6.14.1 On-Chip SRAM Self-Test Using PBIST

6.14.1.1 Features

- Extensive instruction set to support various memory test algorithms
- ROM-based algorithms allow the application to run TI production-level memory tests
- Independent testing of all on-chip SRAM

6.14.1.2 PBIST RAM Groups

Table 6-22. PBIST RAM Grouping

MEMORY	RAM GROUP	TEST CLOCK	MEM TYPE	TEST PATTERN (ALGORITHM)			
				TRIPLE READ SLOW READ	TRIPLE READ FAST READ	MARCH 13N(1) TWO PORT (CYCLES)	MARCH 13N(1) SINGLE PORT (CYCLES)
				ALGO MASK 0x1	ALGO MASK 0x2	ALGO MASK 0x4	ALGO MASK 0x8
PBIST_ROM	1	ROM CLK	ROM	X	X		
STC_ROM	2	ROM CLK	ROM	X	X		
DCAN1	3	VCLK	Dual Port			12720	
DCAN2	4	VCLK	Dual Port			6480	
RAM	6	HCLK	Single Port				133160
MIBSPI1	7	VCLK	Dual Port			33440	
VIM	10	VCLK	Dual Port			12560	
MIBADC	11	VCLK	Dual Port			4200	
N2HET1	13	VCLK	Dual Port			25440	
HTU1	14	VCLK	Dual Port			6480	

(1) There are several memory testing algorithms stored in the PBIST ROM. However, TI recommends the March13N algorithm for application testing.

The PBIST ROM clock can be divided down from HCLK. The divider is selected by programming the ROM_DIV field of the Memory Self-Test Global Control Register (MSTGCR) at address 0xFFFFFF58.

6.14.2 On-Chip SRAM Auto Initialization

This microcontroller allows some of the on-chip memories to be initialized through the Memory Hardware Initialization mechanism in the System module. This hardware mechanism allows an application to program the memory arrays with error detection capability to a known state based on their error detection scheme (odd/even parity or ECC).

The MINITGCR register enables the memory initialization sequence, and the MSINENA register selects the memories that are to be initialized.

For more information on these registers refer to the device Technical Reference Manual.

The mapping of the different on-chip memories to the specific bits of the MSINENA registers is shown in Table 6-23.

Table 6-23. Memory Initialization

CONNECTING MODULE	ADDRESS RANGE		MSINENA REGISTER BIT NO.(1)
	BASE ADDRESS	ENDING ADDRESS	
RAM	0x08000000	0x08007FFF	0
MIBSPI1 RAM	0xFF0E0000	0xFF0FFFFF	7(2)
DCAN2 RAM	0xFF1C0000	0xFF1DFFFF	6
DCAN1 RAM	0xFF1E0000	0xFF1FFFFF	5
MIBADC RAM	0xFF3E0000	0xFF3FFFFF	8
N2HET RAM	0xFF460000	0xFF47FFFF	3
HTU RAM	0xFF4E0000	0xFF4FFFFF	4
VIM RAM	0xFFF82000	0xFFF82FFF	2

(1) Unassigned register bits are reserved.
(2) The MibSPI1 module performs an initialization of the transmit and receive RAMs as soon as the module is brought out of reset using the SPI Global Control Register 0 (SPIGCR0). This is independent of whether the application chooses to initialize the MibSPI1 RAMs using the system module auto-initialization method.

6.15 Vectored Interrupt Manager

The vectored interrupt manager (VIM) provides hardware assistance for prioritizing and controlling the many interrupt sources present on this device. Interrupts are caused by events outside of the normal flow of program execution. Normally, these events require a timely response from the central processing unit (CPU); therefore, when an interrupt occurs, the CPU switches execution from the normal program flow to an interrupt service routine (ISR).

6.15.1 VIM Features

The VIM module has the following features:

- Supports 96 interrupt channels.
  - Provides programmable priority and enable for interrupt request lines.
- Provides a direct hardware dispatch mechanism for fastest IRQ dispatch.
- Provides two software dispatch mechanisms when the CPU VIC port is not used.
  - Index interrupt
  - Register vectored interrupt
- Parity protected vector interrupt table against soft errors.

6.15.2 Interrupt Request Assignments

Table 6-24. Interrupt Request Assignments

MODULES	INTERRUPT SOURCES	DEFAULT VIM INTERRUPT CHANNEL
ESM	ESM High level interrupt (NMI)	0
Reserved	Reserved	1
RTI	RTI compare interrupt 0	2
RTI	RTI compare interrupt 1	3
RTI	RTI compare interrupt 2	4
RTI	RTI compare interrupt 3	5
RTI	RTI overflow interrupt 0	6
RTI	RTI overflow interrupt 1	7
Reserved	Reserved	8
GIO	GIO interrupt A	9
N2HET	N2HET level 0 interrupt	10
HTU	HTU level 0 interrupt	11
MIBSPI1	MIBSPI1 level 0 interrupt	12
LIN	LIN level 0 interrupt	13
MIBADC	MIBADC event group interrupt	14
MIBADC	MIBADC sw group 1 interrupt	15
DCAN1	DCAN1 level 0 interrupt	16
SPI2	SPI2 level 0 interrupt	17
Reserved	Reserved	18
Reserved	Reserved	19
ESM	ESM Low level interrupt	20
SYSTEM	Software interrupt (SSI)	21
CPU	PMU interrupt	22
GIO	GIO interrupt B	23
N2HET	N2HET level 1 interrupt	24
HTU	HTU level 1 interrupt	25

Table 6-24. Interrupt Request Assignments (continued)

MODULES	INTERRUPT SOURCES	DEFAULT VIM INTERRUPT CHANNEL
MIBSPI1	MIBSPI1 level 1 interrupt	26
LIN	LIN level 1 interrupt	27
MIBADC	MIBADC sw group 2 interrupt	28
DCAN1	DCAN1 level 1 interrupt	29
SPI2	SPI2 level 1 interrupt	30
MIBADC	MIBADC magnitude compare interrupt	31
Reserved	Reserved	32-34
DCAN2	DCAN2 level 0 interrupt	35
Reserved	Reserved	36
SPI3	SPI3 level 0 interrupt	37
SPI3	SPI3 level 1 interrupt	38
Reserved	Reserved	39-41
DCAN2	DCAN2 level 1 interrupt	42
Reserved	Reserved	43-60
FMC	FSM_DONE interrupt	61
Reserved	Reserved	62-79
HWAG	HWA_INT_REQ_H	80
Reserved	Reserved	81
DCC	DCC done interrupt	82
Reserved	Reserved	83
eQEPINTn	eQEP Interrupt	84
PBIST	PBIST Done Interrupt	85
Reserved	Reserved	86-87
HWAG	HWA_INT_REQ_L	88
Reserved	Reserved	89-95

NOTE

Address location 0x00000000 in the VIM RAM is reserved for the phantom interrupt ISR entry; therefore only request channels 0..94 can be used and are offset by 1 address in the VIM RAM.

6.16 Real-Time Interrupt Module

The real-time interrupt (RTI) module provides timer functionality for operating systems and for benchmarking code. The RTI module can incorporate several counters that define the timebases needed for scheduling an operating system.

The timers also allow you to benchmark certain areas of code by reading the values of the counters at the beginning and the end of the desired code range and calculating the difference between the values.

6.16.1 Features

The RTI module has the following features:

- Two independent 64 bit counter blocks
- Four configurable compares for generating operating system ticks. Each event can be driven by either counter block 0 or counter block 1.
- Fast enabling/disabling of events
- Two time-stamp (capture) functions for system or peripheral interrupts, one for each counter block

6.16.2 Block Diagrams

Figure 6-11 shows a high-level block diagram for one of the two 64-bit counter blocks inside the RTI module. Both the counter blocks are identical.

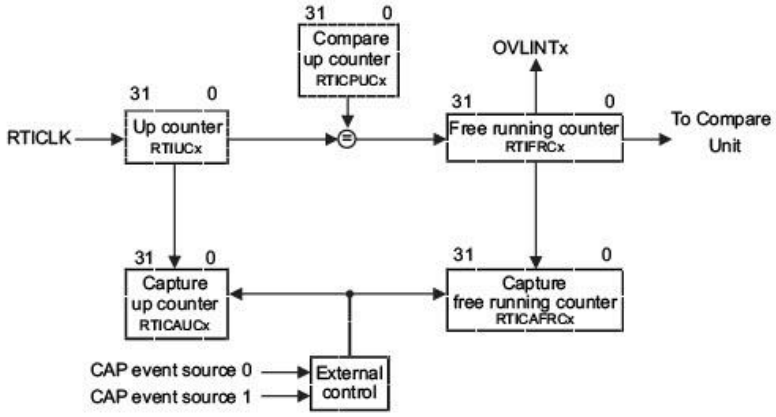


Figure 6-11. Counter Block Diagram

Figure 6-12 shows a typical high-level block diagram for one of the four compares inside the RTI module. Each of the four compares are identical.

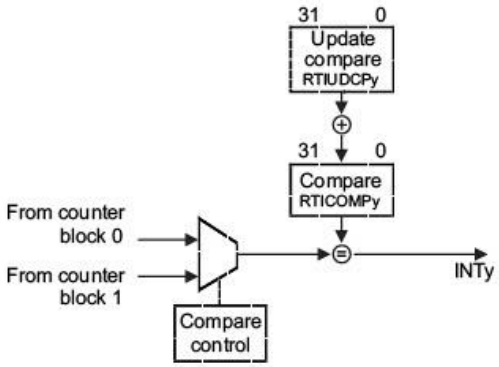


Figure 6-12. Compare Block Diagram

6.16.3 Clock Source Options

The RTI module uses the RTICLK clock domain for generating the RTI time bases.

The application can select the clock source for the RTICLK by configuring the RCLKSRC register in the System module at address 0xFFFFFF50. The default source for RTICLK is VCLK.

For more information, on the clock sources see Table 6-8 and Table 6-12.

6.17 Error Signaling Module

The Error Signaling Module (ESM) manages the various error conditions on the RM4x microcontroller. The error condition is handled based on a fixed severity level assigned to it. Any severe error condition can be configured to drive a low level on a dedicated device terminal called nERROR. This can be used as an indicator to an external monitor circuit to put the system into a safe state.

6.17.1 Features

The features of the Error Signaling Module are:

- 128 interrupt/error channels are supported, divided into 3 different groups
  - 64 channels with maskable interrupt and configurable error pin behavior
  - 32 error channels with nonmaskable interrupt and predefined error pin behavior
  - 32 channels with predefined error pin behavior only
- Error pin to signal severe device failure
- Configurable timebase for error signal
- Error forcing capability

6.17.2 ESM Channel Assignments

The Error Signaling Module (ESM) integrates all the device error conditions and groups them in the order of severity. Group1 is used for errors of the lowest severity while Group3 is used for errors of the highest severity. The device response to each error is determined by the severity group it is connected to. Table 6-26 shows the channel assignment for each group.

Table 6-25. ESM Groups

ERROR GROUP	INTERRUPT CHARACTERISTICS	INFLUENCE ON ERROR PIN
Group1	Maskable, low or high priority	Configurable
Group2	Nonmaskable, high priority	Fixed
Group3	No interrupt generated	Fixed

Table 6-26. ESM Channel Assignments

ERROR SOURCES	GROUP	CHANNELS
Reserved	Group1	0
Reserved	Group1	1
Reserved	Group1	2
Reserved	Group1	3
Reserved	Group1	4
Reserved	Group1	5
FMC - correctable error: bus1 and bus2 interfaces (does not include accesses to EEPROM bank)	Group1	6
N2HET - parity	Group1	7
HTU - parity	Group1	8
HTU - MPU	Group1	9
PLL - Slip	Group1	10
Clock Monitor - interrupt	Group1	11
Reserved	Group1	12
Reserved	Group1	13
Reserved	Group1	14
VIM RAM - parity	Group1	15
Reserved	Group1	16
MibSPI1 - parity	Group1	17
Reserved	Group1	18
MibADC - parity	Group1	19
Reserved	Group1	20
DCAN1 - parity	Group1	21
Reserved	Group1	22
DCAN2 - parity	Group1	23
Reserved	Group1	24
Reserved	Group1	25
RAM even bank (B0TCM) - correctable error	Group1	26
CPU - self-test	Group1	27
RAM odd bank (B1TCM) - correctable error	Group1	28
Reserved	Group1	29
DCC - error	Group1	30
CCM-R4 - self-test	Group1	31
Reserved	Group1	32
Reserved	Group1	33
Reserved	Group1	34
FMC - correctable error (EEPROM bank access)	Group1	35
FMC - uncorrectable error (EEPROM bank access)	Group1	36
IOMM - Mux configuration error	Group1	37
Reserved	Group1	38

Table 6-26. ESM Channel Assignments (continued)

ERROR SOURCES	GROUP	CHANNELS
Reserved	Group1	39
eFuse farm – this error signal is generated whenever any bit in the eFuse farm error status register is set. The application can choose to generate and interrupt whenever this bit is set in order to service any eFuse farm error condition.	Group1	40
eFuse farm - self test error. It is not necessary to generate a separate interrupt when this bit gets set.	Group1	41
Reserved	Group1	42
Reserved	Group1	43
Reserved	Group1	44
Reserved	Group1	45
Reserved	Group1	46
Reserved	Group1	47
Reserved	Group1	48
Reserved	Group1	49
Reserved	Group1	50
Reserved	Group1	51
Reserved	Group1	52
Reserved	Group1	53
Reserved	Group1	54
Reserved	Group1	55
Reserved	Group1	56
Reserved	Group1	57
Reserved	Group1	58
Reserved	Group1	59
Reserved	Group1	60
Reserved	Group1	61
Reserved	Group1	62
Reserved	Group1	63
Reserved	Group2	0
Reserved	Group2	1
CCMR4 - compare	Group2	2
Reserved	Group2	3
FMC - uncorrectable error (address parity on bus1 accesses)	Group2	4
Reserved	Group2	5
RAM even bank (B0TCM) - uncorrectable error	Group2	6
Reserved	Group2	7
RAM odd bank (B1TCM) - uncorrectable error	Group2	8
Reserved	Group2	9
RAM even bank (B0TCM) - address bus parity error	Group2	10
Reserved	Group2	11
RAM odd bank (B1TCM) - address bus parity error	Group2	12
Reserved	Group2	13
Reserved	Group2	14
Reserved	Group2	15
TCM - ECC live lock detect	Group2	16
Reserved	Group2	17
Reserved	Group2	18
Reserved	Group2	19

Table 6-26. ESM Channel Assignments (continued)

ERROR SOURCES	GROUP	CHANNELS
Reserved	Group2	20
Reserved	Group2	21
Reserved	Group2	22
Reserved	Group2	23
RTI_WWD_NMI	Group2	24
Reserved	Group2	25
Reserved	Group2	26
Reserved	Group2	27
Reserved	Group2	28
Reserved	Group2	29
Reserved	Group2	30
Reserved	Group2	31
Reserved	Group3	0
eFuse Farm - autoload error	Group3	1
Reserved	Group3	2
RAM even bank (B0TCM) - ECC uncorrectable error	Group3	3
Reserved	Group3	4
RAM odd bank (B1TCM) - ECC uncorrectable error	Group3	5
Reserved	Group3	6
FMC - uncorrectable error: bus1 and bus2 interfaces (does not include address parity error and errors on accesses to EEPROM bank)	Group3	7
Reserved	Group3	8
Reserved	Group3	9
Reserved	Group3	10
Reserved	Group3	11
Reserved	Group3	12
Reserved	Group3	13
Reserved	Group3	14
Reserved	Group3	15
Reserved	Group3	16
Reserved	Group3	17
Reserved	Group3	18
Reserved	Group3	19
Reserved	Group3	20
Reserved	Group3	21
Reserved	Group3	22
Reserved	Group3	23
Reserved	Group3	24
Reserved	Group3	25
Reserved	Group3	26
Reserved	Group3	27
Reserved	Group3	28
Reserved	Group3	29
Reserved	Group3	30
Reserved	Group3	31

6.18 Reset / Abort / Error Sources

Table 6-27. Reset/Abort/Error Sources

ERROR SOURCE	SYSTEM MODE	ERROR RESPONSE	ESM HOOKUP GROUP.CHANNEL
CPU TRANSACTIONS			
Precise write error (NCNB/Strongly Ordered)	User/Privilege	Precise Abort (CPU)	n/a
Precise read error (NCB/Device or Normal)	User/Privilege	Precise Abort (CPU)	n/a
Imprecise write error (NCB/Device or Normal)	User/Privilege	Imprecise Abort (CPU)	n/a
Illegal instruction	User/Privilege	Undefined Instruction Trap (CPU)(1)	n/a
MPU access violation	User/Privilege	Abort (CPU)	n/a
SRAM			
B0 TCM (even) ECC single error (correctable)	User/Privilege	ESM	1.26
B0 TCM (even) ECC double error (noncorrectable)	User/Privilege	Abort (CPU), ESM → nERROR	3.3
B0 TCM (even) uncorrectable error (that is, redundant address decode)	User/Privilege	ESM → NMI → nERROR	2.6
B0 TCM (even) address bus parity error	User/Privilege	ESM → NMI → nERROR	2.10
B1 TCM (odd) ECC single error (correctable)	User/Privilege	ESM	1.28
B1 TCM (odd) ECC double error (noncorrectable)	User/Privilege	Abort (CPU), ESM → nERROR	3.5
B1 TCM (odd) uncorrectable error (that is, redundant address decode)	User/Privilege	ESM → NMI → nERROR	2.8
B1 TCM (odd) address bus parity error	User/Privilege	ESM → NMI → nERROR	2.12
FLASH WITH CPU BASED ECC			
FMC correctable error - Bus1 and Bus2 interfaces (does not include accesses to EEPROM bank)	User/Privilege	ESM	1.6
FMC uncorrectable error - Bus1 accesses (does not include address parity error)	User/Privilege	Abort (CPU), ESM → nERROR	3.7
FMC uncorrectable error - Bus2 accesses (does not include address parity error and EEPROM bank accesses)	User/Privilege	ESM → nERROR	3.7
FMC uncorrectable error - address parity error on Bus1 accesses	User/Privilege	ESM → NMI → nERROR	2.4
FMC correctable error - Accesses to EEPROM bank	User/Privilege	ESM	1.35
FMC uncorrectable error - Accesses to EEPROM bank	User/Privilege	ESM	1.36
HIGH-END TIMER TRANSFER UNIT (HTU)			
NCNB (Strongly Ordered) transaction with slave error response	User/Privilege	Interrupt → VIM	n/a
External imprecise error (Illegal transaction with ok response)	User/Privilege	Interrupt → VIM	n/a
Memory access permission violation	User/Privilege	ESM	1.9
Memory parity error	User/Privilege	ESM	1.8
N2HET			
Memory parity error	User/Privilege	ESM	1.7
MIBSPI			
MibSPI1 memory parity error	User/Privilege	ESM	1.17
MIBADC			
MibADC Memory parity error	User/Privilege	ESM	1.19
DCAN			
DCAN1 memory parity error	User/Privilege	ESM	1.21

(1) The Undefined Instruction TRAP is NOT detectable outside the CPU. The trap is taken only if the instruction reaches the execute stage of the CPU.

Table 6-27. Reset/Abort/Error Sources (continued)

ERROR SOURCE	SYSTEM MODE	ERROR RESPONSE	ESM HOOKUP GROUP.CHANNEL
DCAN2 memory parity error	User/Privilege	ESM	1.23
PLL			
PLL slip error	User/Privilege	ESM	1.10
CLOCK MONITOR			
Clock monitor interrupt	User/Privilege	ESM	1.11
DCC			
DCC error	User/Privilege	ESM	1.30
CCM-R4			
Self test failure	User/Privilege	ESM	1.31
Compare failure	User/Privilege	ESM → NMI → nERROR	2.2
VIM			
Memory parity error	User/Privilege	ESM	1.15
VOLTAGE MONITOR			
VMON out of voltage range	n/a	Reset	n/a
CPU SELF-TEST (LBIST)			
CPU Self-test (LBIST) error	User/Privilege	ESM	1.27
PIN MULTIPLEXING CONTROL			
Mux configuration error	User/Privilege	ESM	1.37
eFuse CONTROLLER			
eFuse Controller Autoload error	User/Privilege	ESM → nERROR	3.1
eFuse Controller - Any bit set in the error status register	User/Privilege	ESM	1.40
eFuse Controller self-test error	User/Privilege	ESM	1.41
WINDOWED WATCHDOG			
WWD Nonmaskable Interrupt exception	n/a	ESM => NMI => nERROR	2.24
ERRORS REFLECTED IN THE SYSESR REGISTER			
Power-Up Reset	n/a	Reset	n/a
Oscillator fail / PLL slip(2)	n/a	Reset	n/a
Watchdog exception	n/a	Reset	n/a
CPU Reset (driven by the CPU STC)	n/a	Reset	n/a
Software Reset	n/a	Reset	n/a
External Reset	n/a	Reset	n/a

(2) Oscillator fail/PLL slip can be configured in the system register (SYS.PLLCTL1) to generate a reset.

6.19 Digital Windowed Watchdog

This device includes a digital windowed watchdog (DWWD) module that protects against runaway code execution.

The DWWD module allows the application to configure the time window within which the DWWD module expects the application to service the watchdog. A watchdog violation occurs if the application services the watchdog outside of this window, or fails to service the watchdog at all. The application can choose to generate a system reset or a nonmaskable interrupt to the CPU in case of a watchdog violation.

The watchdog is disabled by default and must be enabled by the application. Once enabled, the watchdog can only be disabled upon a system reset.

6.20 Debug Subsystem

6.20.1 Block Diagram

The device contains an ICEPICK module to allow JTAG access to the scan chains (see Figure 6-13).

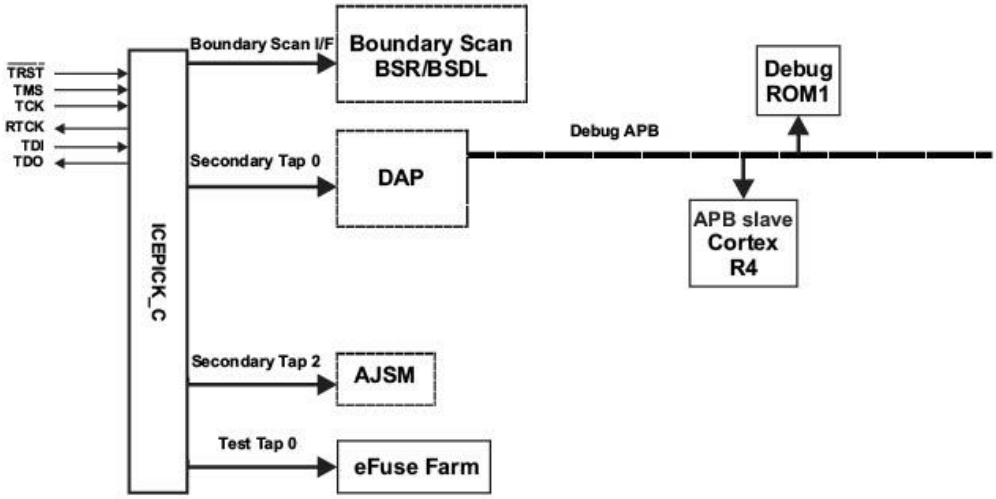


Figure 6-13. Debug Subsystem Block Diagram

6.20.2 Debug Components Memory Map

Table 6-28. Debug Components Memory Map

MODULE NAME	FRAME CHIP SELECT	FRAME ADDRESS RANGE		FRAME SIZE	ACTUAL SIZE	RESPONSE FOR ACCESS TO UNIMPLEMENTED LOCATIONS IN FRAME
		START	END			
CoreSight Debug ROM	CSCS0	0xFFA0_0000	0xFFA0_0FFF	4KB	4KB	Reads return zeros, writes have no effect
Cortex-R4 Debug	CSCS1	0xFFA0_1000	0xFFA0_1FFF	4KB	4KB	Reads return zeros, writes have no effect

6.20.3 JTAG Identification Code

The JTAG ID code for this device is the same as the device ICEPick Identification Code.

Table 6-29. JTAG Identification Code

SILICON REVISION	IDENTIFICATION CODE
Initial Silicon	0x0B97102F
Revision A	0x1B97102F
Revision B	0x2B97102F

6.20.4 Debug ROM

The Debug ROM stores the location of the components on the Debug APB bus:

Table 6-30. Debug ROM table

ADDRESS	DESCRIPTION	VALUE
0x000	Pointer to Cortex-R4	0x0000 1003
0x001	Reserved	0x0000 2002
0x002	Reserved	0x0000 3002
0x003	Reserved	0x0000 4002
0x004	End of table	0x0000 0000

6.20.5 JTAG Scan Interface Timings

Table 6-31. JTAG Scan Interface Timing(1)

NO.	PARAMETER		MIN	MAX	UNIT
	f_{TCK}	TCK frequency (at HCLKmax)		12	MHz
	f_{RTCK}	RTCK frequency (at TCKmax and HCLKmax)	10		MHz
1	$t_{d(TCK-RTCK)}$	Delay time, TCK to RTCK		24	ns
2	$t_{su(TDI/TMS-RTCKr)}$	Setup time, TDI, TMS before RTCK rise (RTCKr)	26		ns
3	$t_{h(RTCKr-TDI/TMS)}$	Hold time, TDI, TMS after RTCKr	0		ns
4	$t_{h(RTCKf-TDO)}$	Hold time, TDO after RTCKf	0		ns
5	$t_{d(TCKf-TDO)}$	Delay time, TDO valid after RTCK fall (RTCKf)		12	ns

(1) Timings for TDO are specified for a maximum of 50 pF load on TDO

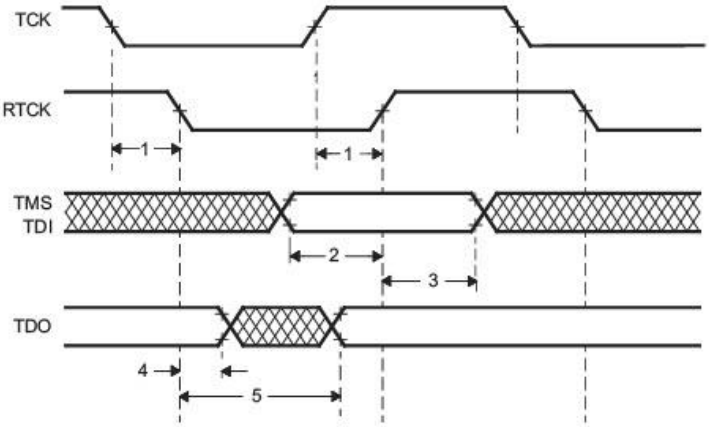


Figure 6-14. JTAG Timing

6.20.6 Advanced JTAG Security Module

This device includes a an Advanced JTAG Security Module (AJSM). which provides maximum security to the memory content of the device by allowing users to secure the device after programming.

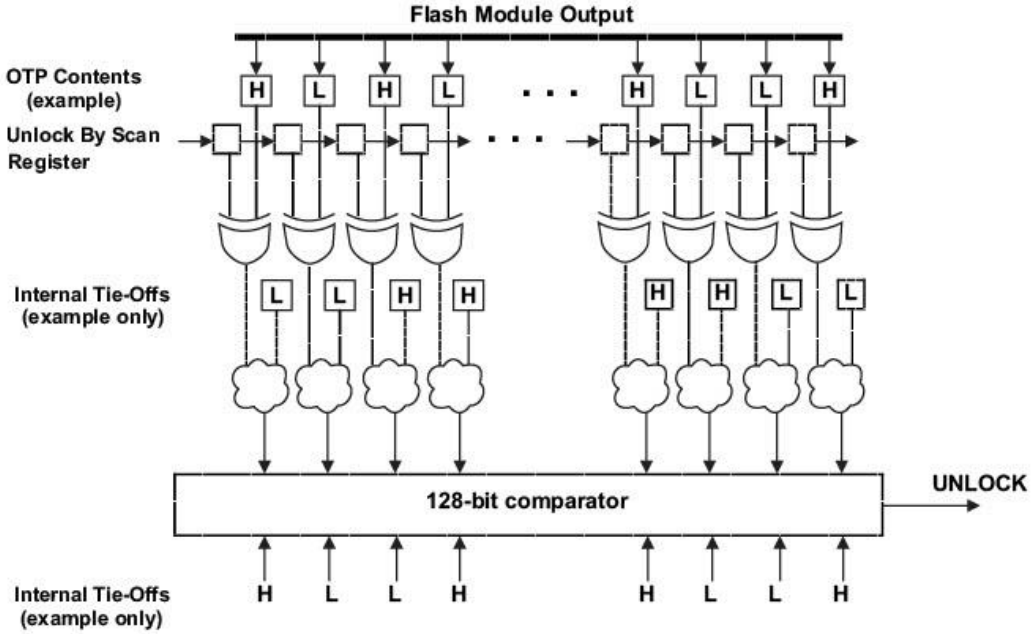


Figure 6-15. AJSM Unlock

The device is unsecure by default by virtue of a 128-bit visible unlock code programmed in the OTP address 0xF0000000.The OTP contents are XOR-ed with the "Unlock By Scan" register contents. The outputs of these XOR gates are again combined with a set of secret internal tie-offs. The output of this combinational logic is compared against a secret hard-wired 128-bit value. A match results in the UNLOCK signal being asserted, so that the device is now unsecure.

A user can secure the device by changing at least one bit in the visible unlock code from 1 to 0. Changing a 0 to 1 is not possible because the visible unlock code is stored in the One Time Programmable (OTP) flash region. Also, changing all the 128 bits to zeros is not a valid condition and will permanently secure the device.

Once secured, a user can unsecure the device by scanning an appropriate value into the "Unlock By Scan" register of the AJSM module. The value to be scanned is such that the XOR of the OTP contents and the Unlock-By-Scan register contents results in the original visible unlock code.

The Unlock-By-Scan register is reset only upon asserting power-on reset (nPORRST).

A secure device only permits JTAG accesses to the AJSM scan chain through the Secondary Tap # 2 of the ICEPick module. All other secondary taps, test taps and the boundary scan interface are not accessible in this state.

6.20.7 Boundary Scan Chain

The device supports BSDL-compliant boundary scan for testing pin-to-pin compatibility. The boundary scan chain is connected to the Boundary Scan Interface of the ICEPICK module.

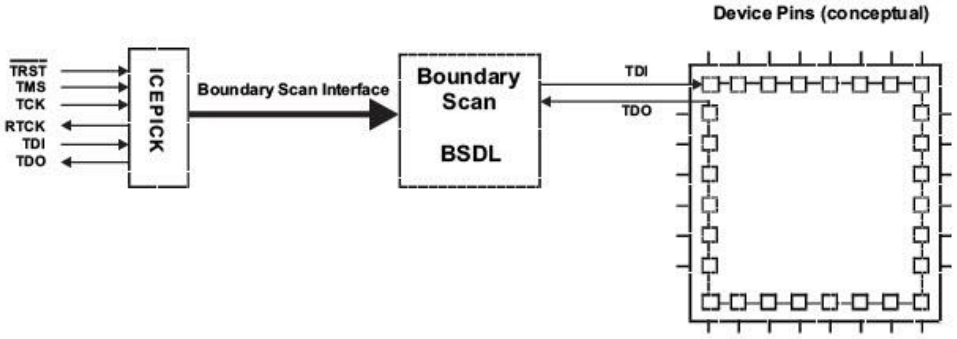


Figure 6-16. Boundary Scan Implementation (Conceptual Diagram)

Data is serially shifted into all boundary-scan buffers through TDI, and out through TDO.

7 Peripheral Information and Electrical Specifications

7.1 Peripheral Legend

Table 7-1. Peripheral Legend

ABBREVIATION	FULL NAME
MibADC	Multibuffered Analog-to-Digital Converter
CCM-R4	CPU Compare Module – Cortex-R4
CRC	Cyclic Redundancy Check
DCAN	Controller Area Network
DCC	Dual Clock Comparator
ESM	Error Signaling Module
GIO	General-Purpose Input/Output
HTU	High-End Timer Transfer Unit
LIN	Local Interconnect Network
MibSPI	Multibuffered Serial Peripheral Interface
N2HET	Platform High-End Timer
RTI	Real-Time Interrupt Module
SCI	Serial Communications Interface
SPI	Serial Peripheral Interface
VIM	Vectored Interrupt Manager
eQEP	Enhanced Quadrature Encoder Pulse

7.2 Multibuffered 12-Bit Analog-to-Digital Converter

The multibuffered A-to-D converter (MibADC) has a separate power bus for its analog circuitry that enhances the A-to-D performance by preventing digital switching noise on the logic circuitry which could be present on V_{SS} and V_{CC} from coupling into the A-to-D analog stage. All A-to-D specifications are given with respect to AD_{REFLO} unless otherwise noted.

Table 7-2. MibADC Overview

DESCRIPTION	VALUE
Resolution	12 bits
Monotonic	Assured
Output conversion code	00h to FFFh [00 for $V_{AI} \le AD_{REFLO}$; FFF for $V_{AI} \ge AD_{REFHI}$]

7.2.1 Features

- 12-bit resolution
- AD_{REFHI} and AD_{REFLO} pins (high and low reference voltages)
- Total Sample/Hold/Convert time: 600 ns Typical Minimum at 30 MHz ADCLK
- One memory region per conversion group is available (event, group 1, group 2)
- Allocation of channels to conversion groups is completely programmable
- Memory regions are serviced by interrupt
- Programmable interrupt threshold counter is available for each group
- Programmable magnitude threshold interrupt for each group for any one channel
- Option to read either 8-, or 10-, or 12-bit values from memory regions
- Single or continuous conversion modes
- Embedded self-test
- Embedded calibration logic
- Enhanced power-down mode
  - Optional feature to automatically power down ADC core when no conversion is in progress

- External event pin (ADEVT) programmable as general-purpose I/O

7.2.2 Event Trigger Options

The ADC module supports three conversion groups: Event Group, Group1, and Group2. Each of these three groups can be configured to be hardware event-triggered. In that case, the application can select from among eight event sources to be the trigger for the conversions of a group.

7.2.2.1 MIBADC Event Trigger Hookup

Table 7-3. MIBADC Event Trigger Hookup

EVENT NUMBER	SOURCE SELECT BITS For G1, G2, or EVENT (G1SRC[2:0], G2SRC[2:0], or EVSRC[2:0])	TRIGGER
1	000	ADEVT
2	001	N2HET[8]
3	010	N2HET[10]
4	011	RTI compare 0 interrupt
5	100	N2HET[12]
6	101	N2HET[14]
7	110	N2HET[17]
8	111	N2HET[19]

NOTE

For ADEVT, N2HET trigger sources, the connection to the MibADC module trigger input is made from the output side of the input buffer. This way, a trigger condition can be generated either by configuring the function as output onto the pad, or by driving the function from an external trigger source as input. If the mux controller module is used to select different functionality instead of ADEVT or N2HET[x], care must be taken to disable these signals from triggering conversions; there is no multiplexing on input connections.

NOTE

For the RTI compare 0 interrupt source, the connection is made directly from the output of the RTI module. That is, the interrupt condition can be used as a trigger source even if the actual interrupt is not signaled to the CPU.

7.2.3 ADC Electrical and Timing Specifications

Table 7-4. MibADC Recommended Operating Conditions

PARAMETER		MIN	MAX	UNIT
AD_{REFHI}	A-to-D high-voltage reference source	AD_{REFLO}	V_{CCAD}	V
AD_{REFLO}	A-to-D low-voltage reference source	V_{SSAD}	AD_{REFHI}	V
V_{AI}	Analog input voltage	AD_{REFLO}	AD_{REFHI}	V
I_{AIC}	Analog input clamp current ($VAI < V_{SSAD} - 0.3$ or $VAI > V_{CCAD} + 0.3$)	–2	2	mA

Table 7-5. MibADC Electrical Characteristics Over Full Ranges of Recommended Operating Conditions(1)

PARAMETER		DESCRIPTION/CONDITIONS		MIN	TYP	MAX	UNIT
R_{mux}	Analog input mux on-resistance	See Figure 7-1			95	250	Ω
R_{samp}	ADC sample switch on-resistance	See Figure 7-1			60	250	Ω
C_{mux}	Input mux capacitance	See Figure 7-1			7	16	pF
C_{samp}	ADC sample capacitance	See Figure 7-1			8	13	pF
I_{AIL}	Analog off-state input leakage current	V_{CCAD} = 3.6 V MAX	$V_{SSAD} < V_{IN} < V_{SSAD} + 100$ mV	–300	–1	200	nA
			$V_{SSAD} + 100$ mV $< V_{IN} < V_{CCAD} - 200$ mV	–200	–0.3	200	
			$V_{CCAD} - 200$ mV $< V_{IN} < V_{CCAD}$	–200	1	500	
I_{AOSB}	Analog on-state input bias	V_{CCAD} = 3.6 V MAX	$V_{SSAD} < V_{IN} < V_{SSAD} + 100$ mV	–8		2	µA
			$V_{SSAD} + 100$ mV $< V_{IN} < V_{CCAD} - 200$ mV	–4		2	
			$V_{CCAD} - 200$ mV $< V_{IN} < V_{CCAD}$	–4		12	
$I_{ADREFHI}$	AD_{REFHI} input current	$AD_{REFHI} = V_{CCAD}$, $AD_{REFLO} = V_{SSAD}$				3	mA
I_{CCAD}	Static supply current	Normal operating mode				(2)	mA
		ADC core in power-down mode				5	µA

(1) 1 LSB = $(AD_{REFHI} - AD_{REFLO})/ 2^n$ where n = 10 in 10-bit mode and 12 in 12-bit mode
(2) See Section 5.7.

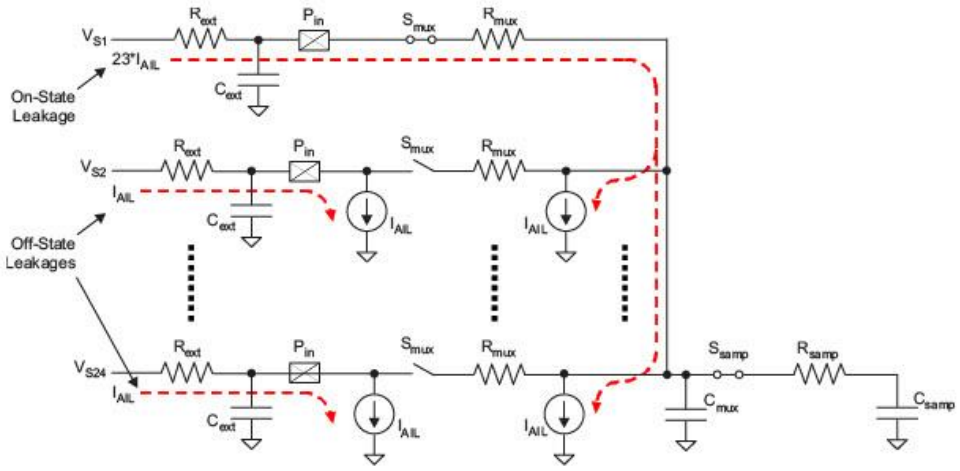


Figure 7-1. MibADC Input Equivalent Circuit

Table 7-6. MibADC Timing Specifications

PARAMETER		MIN	NOM	MAX	UNIT
$t_{c(ADCLK)}$ (1)	Cycle time, MibADC clock	33			ns
$t_{d(SH)}$ (2)	Delay time, sample and hold time	200			ns
$t_{d(PU-ADV)}$	Delay time from ADC power on until first input can be sampled	1			µs
12-BIT MODE					
$t_{d(C)}$	Delay time, conversion time	400			ns
$t_{d(SHC)}$ (3)	Delay time, total sample/hold and conversion time	600			ns
10-BIT MODE					
$t_{d(C)}$	Delay time, conversion time	330			ns
$t_{d(SHC)}$ (3)	Delay time, total sample/hold and conversion time	530			ns

(1) The MibADC clock is the ADCLK, generated by dividing down the VCLK by a prescale factor defined by the ADCLOCKCR register bits 4:0.
(2) The sample and hold time for the ADC conversions is defined by the ADCLK frequency and the AD<GP>SAMP register for each conversion group. The sample time must be determined by accounting for the external impedance connected to the input channel as well as the internal impedance of the ADC.
(3) This is the minimum sample/hold and conversion time that can be achieved. These parameters are dependent on many factors for example, the prescale settings.

Table 7-7. MibADC Operating Characteristics Over Full Ranges of Recommended Operating Conditions

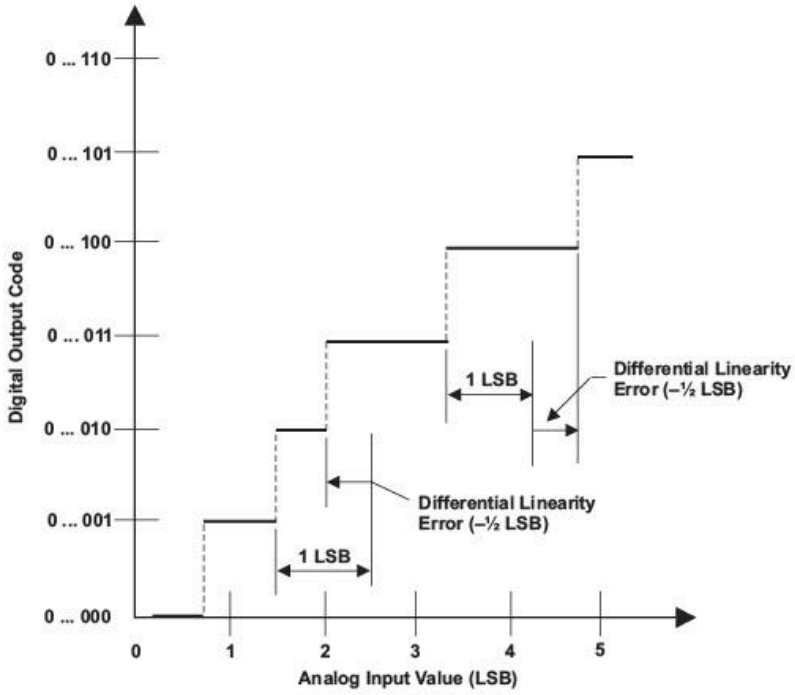
PARAMETER		DESCRIPTION/CONDITIONS			MIN	TYP	MAX	UNIT
CR	Conversion range over which specified accuracy is maintained	AD_{REFHI} - AD_{REFLO}			3		3.6	V
Z_{SET}	Offset Error	Difference between the first ideal transition (from code 000h to 001h) and the actual transition	10-bit mode	With ADC Calibration			1	LSB(1)
				Without ADC Calibration			2	
			12-bit mode	With ADC Calibration			2	
				Without ADC Calibration			4	
F_{SET}	Gain Error	Difference between the last ideal transition (from code FFEh to FFFh) and the actual transition minus offset.	10-bit mode				2	LSB
			12-bit mode				3	
E_{DNL}	Differential nonlinearity error	Difference between the actual step width and the ideal value. (See Figure 7-2)	10-bit mode				± 1.5	LSB
			12-bit mode				± 2	
E_{INL}	Integral nonlinearity error	Maximum deviation from the best straight line through the MibADC. MibADC transfer characteristics, excluding the quantization error. (See Figure 7-3)	10-bit mode				± 2	LSB
			12-bit mode				± 2	
E_{TOT}	Total unadjusted error	Maximum value of the difference between an analog value and the ideal midstep value. (See Figure 7-4)	10-bit mode	With ADC Calibration			± 2	LSB
				Without ADC Calibration			± 4	
			12-bit mode	With ADC Calibration			± 4	
				Without ADC Calibration			± 7	

(1) 1 LSB = $(AD_{REFHI} - AD_{REFLO})/ 2^n$ where n = 10 in 10-bit mode and 12 in 12-bit mode

7.2.4 Performance (Accuracy) Specifications

7.2.4.1 MibADC Nonlinearity Errors

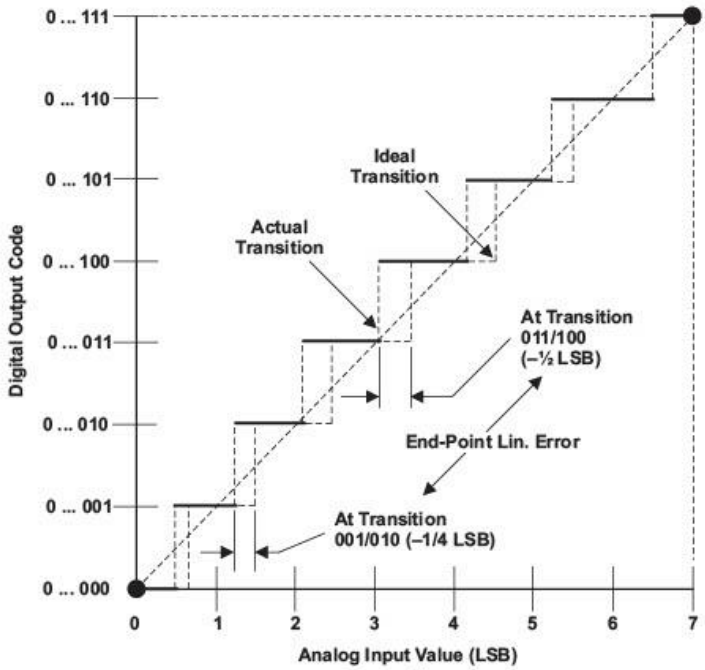
The differential nonlinearity error shown in Figure 7-2 (sometimes referred to as differential linearity) is the difference between an actual step width and the ideal value of 1 LSB.



NOTE A: 1 LSB = $(AD_{REFHI} - AD_{REFLO})/2^n$ where n=10 in 10-bit mode and 12 in 12-bit mode

Figure 7-2. Differential Nonlinearity (DNL) Error

The integral nonlinearity error shown in Figure 7-3 (sometimes referred to as linearity error) is the deviation of the values on the actual transfer function from a straight line.

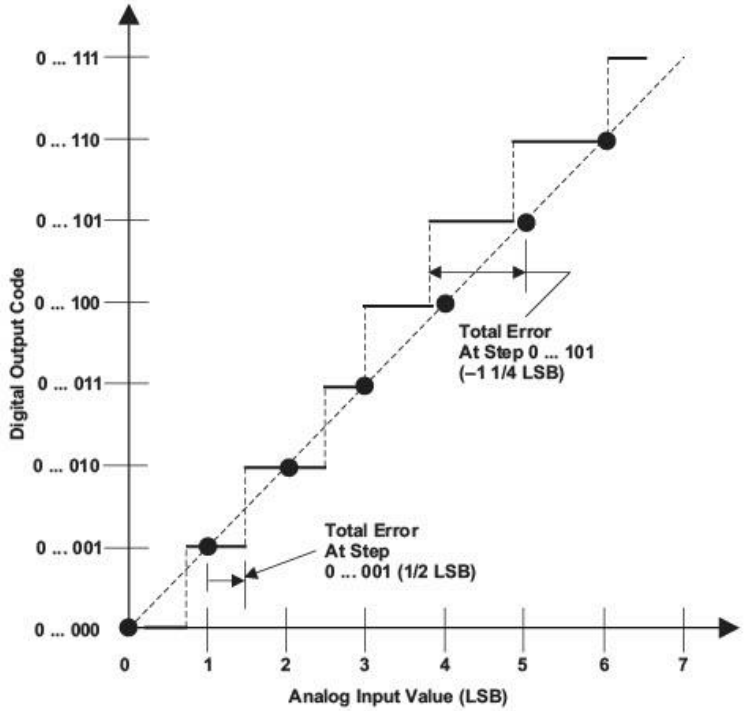


NOTE A: 1 LSB = $(AD_{REFHI} - AD_{REFLO})/2^n$ where n=10 in 10-bit mode and 12 in 12-bit mode

Figure 7-3. Integral Nonlinearity (INL) Error

7.2.4.2 MibADC Total Error

The absolute accuracy or total error of an MibADC as shown in Figure 7-4 is the maximum value of the difference between an analog value and the ideal midstep value.



NOTE A: 1 LSB = $(AD_{REFHI} - AD_{REFLO})/2^n$ where n=10 in 10-bit mode and 12 in 12-bit mode

Figure 7-4. Absolute Accuracy (Total) Error

7.3 General-Purpose Input/Output

The GPIO module on this device supports one port GIOA. The I/O pins are bidirectional and bit-programmable. GIOA supports external interrupt capability.

7.3.1 Features

The GPIO module has the following features:

- Each I/O pin can be configured as:
  - Input
  - Output
  - Open Drain
- The interrupts have the following characteristics:
  - Programmable interrupt detection either on both edges or on a single edge (set in GIOINTDET)
  - Programmable edge-detection polarity, either rising or falling edge (set in GIOPOL register)
  - Individual interrupt flags (set in GIOFLG register)
  - Individual interrupt enables, set and cleared through GIOENASET and GIOENACLR registers respectively
  - Programmable interrupt priority, set through GIOLVLSET and GIOLVLCLR registers
- Internal pullup/pulldown allows unused I/O pins to be left unconnected

For information on input and output timings see Section 5.11 and Section 5.12

7.4 Enhanced High-End Timer (N2HET)

The N2HET is an advanced intelligent timer that provides sophisticated timing functions for real-time applications. The timer is software-controlled, using a reduced instruction set, with a specialized timer micromachine and an attached I/O port. The N2HET can be used for pulse width modulated outputs, capture or compare inputs, or general-purpose I/O.. It is especially well suited for applications requiring multiple sensor information and drive actuators with complex and accurate time pulses.

7.4.1 Features

The N2HET module has the following features:

- Programmable timer for input and output timing functions
- Reduced instruction set (30 instructions) for dedicated time and angle functions
- 128 words of instruction RAM protected by parity
- User defined number of 25-bit virtual counters for timer, event counters and angle counters
- 7-bit hardware counters for each pin allow up to 32-bit resolution in conjunction with the 25-bit virtual counters
- Up to 19 pins usable for input signal measurements or output signal generation
- Programmable suppression filter for each input pin with adjustable limiting frequency
- Low CPU overhead and interrupt load
- Efficient data transfer to or from the CPU memory with dedicated High-End-Timer Transfer Unit (HTU)
- Diagnostic capabilities with different loopback mechanisms and pin status readback functionality

7.4.2 N2HET RAM Organization

The timer RAM uses 4 RAM banks, where each bank has two port access capability. This means that one RAM address may be written while another address is read. The RAM words are 96-bits wide, which are split into three 32-bit fields (program, control, and data).

7.4.3 Input Timing Specifications

The N2HET instructions PCNT and WCAP impose some timing constraints on the input signals.

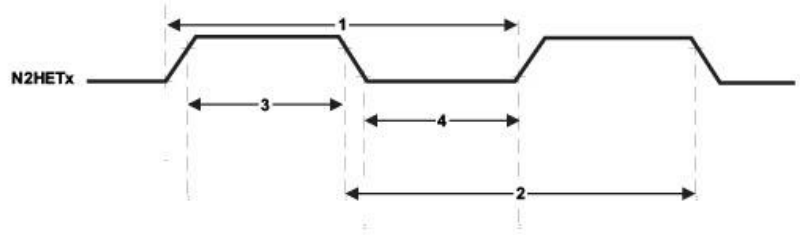


Figure 7-5. N2HET Input Capture Timings

Table 7-8. Dynamic Characteristics for the N2HET Input Capture Functionality

	PARAMETER	MIN(1) (2)	MAX(1) (2)	UNIT
1	Input signal period, PCNT or WCAP for rising edge to rising edge	(hr)(lr) $t_{c(VCLK2)}$ + 2	2^{25}(hr)(lr)$t_{c(VCLK2)}$ - 2	ns
2	Input signal period, PCNT or WCAP for falling edge to falling edge	(hr) (lr) $t_{c(VCLK2)}$ + 2	2^{25} (hr)(lr) $t_{c(VCLK2)}$ - 2	ns
3	Input signal high phase, PCNT or WCAP for rising edge to falling edge	2(hr) $t_{c(VCLK2)}$ + 2	2^{25} (hr)(lr) $t_{c(VCLK2)}$ - 2	ns
4	Input signal low phase, PCNT or WCAP for falling edge to rising edge	2(hr) $t_{c(VCLK2)}$ + 2	2^{25} (hr)(lr) $t_{c(VCLK2)}$ - 2	ns

(1) hr = High-resolution prescaler, configured using the HRPFC field of the Prescale Factor Register (HETPFR).
(2) lr = Loop-resolution prescaler, configured using the LFPRC field of the Prescale Factor Register (HETPFR).

7.4.4 N2HET Checking

7.4.4.1 Output Monitoring using Dual Clock Comparator (DCC)

N2HET[31] is connected as a clock source for counter 1 in DCC1. This allows the application to measure the frequency of the pulse-width modulated (PWM) signal on N2HET[31].

N2HET[31] can be configured to be an internal-only channel. That is, the connection to the DCC module is made directly from the output of the N2HET module (from the input of the output buffer).

For more information on DCC, see Section 6.6.3.

7.4.5 Disabling N2HET Outputs

Some applications require the N2HET outputs to be disabled under some fault condition. The N2HET module provides this capability through the "Pin Disable" input signal. This signal, when driven low, causes the N2HET outputs identified by a programmable register (HETPINDIS) to be tri-stated.

For more details on the "N2HET Pin Disable" feature, see the device-specific Technical Reference Manual listed in Section 8.2.1.

GIOA[5] and EQEPERR are connected to the "Pin Disable" input for N2HET. In the case of GIOA[5] connection, this connection is made from the output of the input buffer. In the case of EQEPERR, the EQEPERR output signal is asserted in the event of a phase error. This signal is inverted and double-synchronized to VCLK2 for input into the N2HET PIN_nDISABLE port.

The PIN_nDISABLE port input source is selectable between the GIOA[5] and EQEPERR sources. This is achieved through the PINMMR9[1:0] bits.

7.4.6 High-End Timer Transfer Unit (N2HET)

A High-End Timer Transfer Unit (N2HET) can perform DMA type transactions to transfer N2HET data to or from main memory. A Memory Protection Unit (MPU) is built into the N2HET.

7.4.6.1 Features

- CPU independent
- Master Port to access system memory
- 8 control packets supporting dual buffer configuration
- Control packet information is stored in RAM protected by parity
- Event synchronization (N2HET transfer requests)
- Supports 32- or 64-bit transactions
- Addressing modes for N2HET address (8 byte or 16 byte) and system memory address (fixed, 32-bit or 64-bit)
- One shot, circular, and auto switch buffer transfer modes
- Request lost detection

7.4.6.2 Trigger Connections

Table 7-9. N2HET Request Line Connection

MODULES	REQUEST SOURCE	HTU REQUEST
N2HET	HTUREQ[0]	HTU DCP[0]
N2HET	HTUREQ[1]	HTU DCP[1]
N2HET	HTUREQ[2]	HTU DCP[2]
N2HET	HTUREQ[3]	HTU DCP[3]
N2HET	HTUREQ[4]	HTU DCP[4]
N2HET	HTUREQ[5]	HTU DCP[5]
N2HET	HTUREQ[6]	HTU DCP[6]
N2HET	HTUREQ[7]	HTU DCP[7]

7.5 Controller Area Network (DCAN)

The DCAN supports the CAN 2.0B protocol standard and uses a serial, multimaster communication protocol that efficiently supports distributed real-time control with robust communication rates of up to 1 Mbps. The DCAN is ideal for applications operating in noisy and harsh environments (for example, automotive and industrial fields) that require reliable serial communication or multiplexed wiring.

7.5.1 Features

Features of the DCAN module include:

- Supports CAN protocol version 2.0 part A, B
- Bit rates up to 1 Mbps
- The CAN kernel can be clocked by the oscillator for baud-rate generation.
- 32 and 16 mailboxes on DCAN1 and DCAN2, respectively
- Individual identifier mask for each message object
- Programmable FIFO mode for message objects
- Programmable loop-back modes for self-test operation
- Automatic bus on after Bus-Off state by a programmable 32-bit timer
- Message RAM protected by parity
- Direct access to Message RAM during test mode
- CAN RX / TX pins configurable as general-purpose I/O pins
- Message RAM Auto Initialization

For more information on the DCAN, see the device-specific Technical Reference Manual listed in Section 8.2.1.

7.5.2 Electrical and Timing Specifications

Table 7-10. Dynamic Characteristics for the DCANx TX and RX pins

PARAMETER		MIN	MAX	UNIT
$t_{d(CANnTX)}$	Delay time, transmit shift register to CANnTX pin(1)		15	ns
$t_{d(CANnRX)}$	Delay time, CANnRX pin to receive shift register		5	ns

(1) These values do not include rise/fall times of the output buffer.

7.6 Local Interconnect Network Interface (LIN)

The SCI/LIN module can be programmed to work either as an SCI or as a LIN. The core of the module is an SCI. The SCI's hardware features are augmented to achieve LIN compatibility.

The SCI module is a universal asynchronous receiver-transmitter that implements the standard nonreturn to zero format. The SCI can be used to communicate, for example, through an RS-232 port or over a K-line.

The LIN standard is based on the SCI (UART) serial data link format. The communication concept is single-master/multiple-slave with a message identification for multicast transmission between any network nodes.

7.6.1 LIN Features

The following are features of the LIN module:

- Compatible to LIN 1.3, 2.0 and 2.1 protocols
- Multibuffered receive and transmit units
- Identification masks for message filtering
- Automatic Master Header Generation
  - Programmable Synch Break Field
  - Synch Field
  - Identifier Field
- Slave Automatic Synchronization
  - Synch break detection
  - Optional baudrate update
  - Synchronization Validation
- 2^{31} programmable transmission rates with 7 fractional bits
- Error detection
- 2 Interrupt lines with priority encoding

7.7 Multibuffered / Standard Serial Peripheral Interface

The MibSPI is a high-speed synchronous serial input/output port that allows a serial bit stream of programmed length (2 to 16 bits) to be shifted in and out of the device at a programmed bit-transfer rate. Typical applications for the SPI include interfacing to external peripherals, such as I/Os, memories, display drivers, and ADCs.

7.7.1 Features

Both Standard and MibSPI modules have the following features:

- 16-bit shift register
- Receive buffer register
- 11-bit baud clock generator
- SPICLK can be internally generated (master mode) or received from an external clock source (slave mode)
- Each word transferred can have a unique format
- SPI I/Os not used in the communication can be used as digital input/output signals

Table 7-11. MibSPI/SPI Default Configurations

MibSPIx/SPIx	I/Os
MibSPI1	MIBSPI1SIMO[0], MIBSPI1SOMI[0], MIBSPI1CLK, MIBSPI1nCS[3:0], MIBSPI1nENA
SPI2	SPI2SIMO, SPI2SOMI, SPI2CLK, SPI2nCS[0]
SPI3	SPI3SIMO, SPI3SOMI, SPI3CLK, SPI3nENA, SPI3nCS[0]

7.7.2 MibSPI Transmit and Receive RAM Organization

The Multibuffer RAM is comprised of 128 buffers. Each entry in the Multibuffer RAM consists of four parts: a 16-bit transmit field, a 16-bit receive field, a 16-bit control field, and a 16-bit status field. The Multibuffer RAM can be partitioned into multiple transfer group with variable number of buffers each.

7.7.3 MibSPI Transmit Trigger Events

Each of the transfer groups can be configured individually. For each of the transfer groups a trigger event and a trigger source can be chosen. A trigger event can be, for example, a rising edge or a permanent low level at a selectable trigger source. Up to 15 trigger sources are available which can be used by each transfer group. These trigger options are listed in Table 7-12.

7.7.3.1 MIBSPI1 Event Trigger Hookup

Table 7-12. MIBSPI1 Event Trigger Hookup

EVENT NO.	TGxCTRL TRIGSRC[3:0]	TRIGGER
Disabled	0000	No trigger source
EVENT0	0001	GIOA[0]
EVENT1	0010	GIOA[1]
EVENT2	0011	GIOA[2]
EVENT3	0100	GIOA[3]
EVENT4	0101	GIOA[4]
EVENT5	0110	GIOA[5]
EVENT6	0111	GIOA[6]
EVENT7	1000	GIOA[7]
EVENT8	1001	N2HET[8]
EVENT9	1010	N2HET[10]
EVENT10	1011	N2HET[12]
EVENT11	1100	N2HET[14]
EVENT12	1101	N2HET[16]
EVENT13	1110	N2HET[18]
EVENT14	1111	Internal Tick counter

NOTE

For N2HET trigger sources, the connection to the MibSPI1 module trigger input is made from the input side of the output buffer (at the N2HET module boundary). This way, a trigger condition can be generated even if the N2HET signal is not selected to be output on the pad.

NOTE

For GIOx trigger sources, the connection to the MibSPI1 module trigger input is made from the output side of the input buffer. This way, a trigger condition can be generated either by selecting the GIOx pin as an output pin, or by driving the GIOx pin from an external trigger source.

7.7.4 MibSPI/SPI Master Mode I/O Timing Specifications

Table 7-13. SPI Master Mode External Timing Parameters (CLOCK PHASE = 0, SPICLK = output, SPISIMO = output, and SPISOMI = input)(1)(2)(3)

NO.	PARAMETER			MIN	MAX	UNIT
1	$t_{c(SPC)M}$	Cycle time, SPICLK(4)		40	$256t_{c(VCLK)}$	ns
2(5)	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 0)		$0.5t_{c(SPC)M} - t_{r(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	ns
	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 1)		$0.5t_{c(SPC)M} - t_{f(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	
3(5)	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 0)		$0.5t_{c(SPC)M} - t_{f(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	ns
	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 1)		$0.5t_{c(SPC)M} - t_{r(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	
4(5)	$t_{d(SPCH-SIMO)M}$	Delay time, SPISIMO valid before SPICLK low (clock polarity = 0)		$0.5t_{c(SPC)M} - 6$		ns
	$t_{d(SPCL-SIMO)M}$	Delay time, SPISIMO valid before SPICLK high (clock polarity = 1)		$0.5t_{c(SPC)M} - 6$		
5(5)	$t_{v(SPCL-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 0)		$0.5t_{c(SPC)M} - t_{f(SPC)} - 4$		ns
	$t_{v(SPCH-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 1)		$0.5t_{c(SPC)M} - t_{r(SPC)} - 4$		
6(5)	$t_{su(SOMI-SPCL)M}$	Setup time, SPISOMI before SPICLK low (clock polarity = 0)		$t_{f(SPC)} + 2.2$		ns
	$t_{su(SOMI-SPCH)M}$	Setup time, SPISOMI before SPICLK high (clock polarity = 1)		$t_{r(SPC)} + 2.2$		
7(5)	$t_{h(SPCL-SOMI)M}$	Hold time, SPISOMI data valid after SPICLK low (clock polarity = 0)		10		ns
	$t_{h(SPCH-SOMI)M}$	Hold time, SPISOMI data valid after SPICLK high (clock polarity = 1)		10		
8(6)	$t_{C2TDELAY}$	Setup time CS active until SPICLK high (clock polarity = 0)	CSHOLD = 0	$C2TDELAY*t_{c(VCLK)} + 2*t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} - 7$	$(C2TDELAY+2) * t_{c(VCLK)} - t_{f(SPICS)} + t_{r(SPC)} + 5.5$	ns
			CSHOLD = 1	$C2TDELAY*t_{c(VCLK)} + 3*t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} - 7$	$(C2TDELAY+3) * t_{c(VCLK)} - t_{f(SPICS)} + t_{r(SPC)} + 5.5$	
		Setup time CS active until SPICLK low (clock polarity = 1)	CSHOLD = 0	$C2TDELAY*t_{c(VCLK)} + 2*t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} - 7$	$(C2TDELAY+2) * t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} + 5.5$	ns
			CSHOLD = 1	$C2TDELAY*t_{c(VCLK)} + 3*t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} - 7$	$(C2TDELAY+3) * t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} + 5.5$	
9(6)	$t_{T2CDELAY}$	Hold time SPICLK low until CS inactive (clock polarity = 0)		$0.5*t_{c(SPC)M} + T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{f(SPC)} + t_{f(SPICS)} - 7$	$0.5*t_{c(SPC)M} + T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{f(SPC)} + t_{f(SPICS)} + 11$	ns
		Hold time SPICLK high until CS inactive (clock polarity = 1)		$0.5*t_{c(SPC)M} + T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{r(SPC)} + tr(SPICS) - 7$	$0.5*t_{c(SPC)M} + T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{r(SPC)} + t_{r(SPICS)} + 11$	ns
10	t_{SPIENA}	SPIENAn Sample point		$(C2TDELAY+1) * t_{c(VCLK)} - t_{f(SPICS)} - 29$	$(C2TDELAY+1)*t_{c(VCLK)}$	ns
11	$t_{SPIENAW}$	SPIENAn Sample point from write to buffer			$(C2TDELAY+2)*t_{c(VCLK)}$	ns

(1) The MASTER bit (SPIGCR1.0) is set and the CLOCK PHASE bit (SPIFMTx.16) is cleared.
(2) $t_{c(VCLK)}$ = interface clock cycle time = 1 / $f_{(VCLK)}$
(3) For rise and fall timings, see Table 5-6.
(4) When the SPI is in Master mode, the following must be true:
For PS values from 1 to 255: $t_{c(SPC)M} \geq (PS + 1)t_{c(VCLK)} \geq 40$ ns, where PS is the prescale value set in the SPIFMTx.[15:8] register bits.
For PS values of 0: $t_{c(SPC)M} = 2t_{c(VCLK)} \geq 40$ ns.
The external load on the SPICLK pin must be less than 60 pF.
(5) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPIFMTx.17).
(6) C2TDELAY and T2CDELAY is programmed in the SPIDELAY register

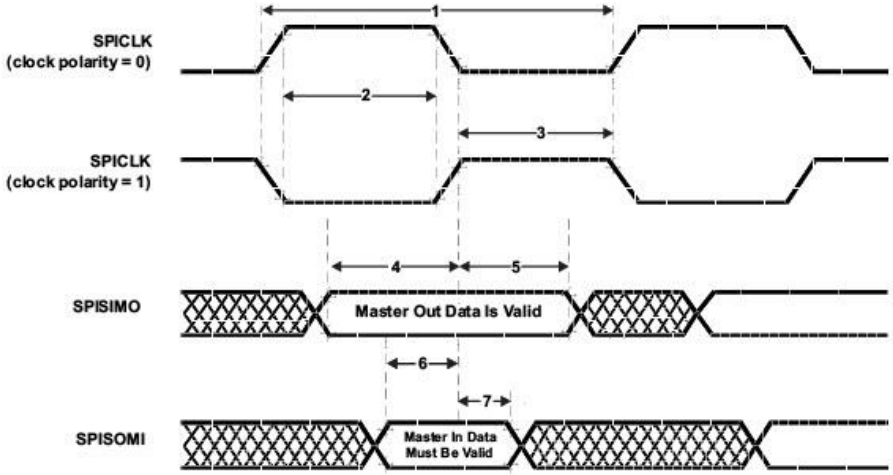


Figure 7-6. SPI Master Mode External Timing (CLOCK PHASE = 0)

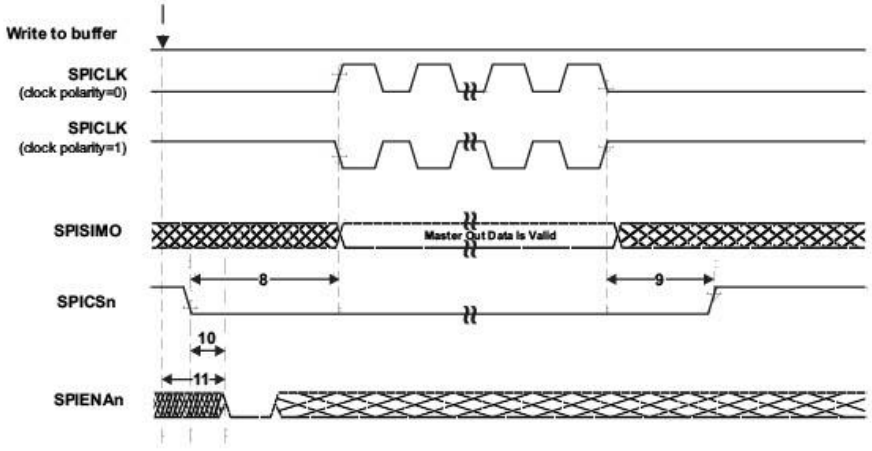


Figure 7-7. SPI Master Mode Chip Select Timing (CLOCK PHASE = 0)

Table 7-14. SPI Master Mode External Timing Parameters (CLOCK PHASE = 1, SPICLK = output, SPISIMO = output, and SPISOMI = input)(1)(2)(3)

NO.	PARAMETER			MIN	MAX	UNIT
1	$t_{c(SPC)M}$	Cycle time, SPICLK (4)		40	$256t_{c(VCLK)}$	ns
2(5)	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 0)		$0.5t_{c(SPC)M} - t_{r(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	ns
	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 1)		$0.5t_{c(SPC)M} - t_{f(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	
3(5)	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 0)		$0.5t_{c(SPC)M} - t_{f(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	ns
	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 1)		$0.5t_{c(SPC)M} - t_{r(SPC)M} - 3$	$0.5t_{c(SPC)M} + 3$	
4(5)	$t_{v(SIMO-SPCH)M}$	Valid time, SPICLK high after SPISIMO data valid (clock polarity = 0)		$0.5t_{c(SPC)M} - 6$		ns
	$t_{v(SIMO-SPCL)M}$	Valid time, SPICLK low after SPISIMO data valid (clock polarity = 1)		$0.5t_{c(SPC)M} - 6$		
5(5)	$t_{v(SPCH-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 0)		$0.5t_{c(SPC)M} - t_{r(SPC)} - 4$		ns
	$t_{v(SPCL-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 1)		$0.5t_{c(SPC)M} - t_{f(SPC)} - 4$		
6(5)	$t_{su(SOMI-SPCH)M}$	Setup time, SPISOMI before SPICLK high (clock polarity = 0)		$t_{r(SPC)} + 2.2$		ns
	$t_{su(SOMI-SPCL)M}$	Setup time, SPISOMI before SPICLK low (clock polarity = 1)		$t_{f(SPC)} + 2.2$		
7(5)	$t_{v(SPCH-SOMI)M}$	Valid time, SPISOMI data valid after SPICLK high (clock polarity = 0)		10		ns
	$t_{v(SPCL-SOMI)M}$	Valid time, SPISOMI data valid after SPICLK low (clock polarity = 1)		10		
8(6)	$t_{C2TDELAY}$	Setup time CS active until SPICLK high (clock polarity = 0)	CSHOLD = 0	$0.5*t_{c(SPC)M} + (C2TDELAY+2) * t_{c(VCLK)} - t_{f(SPICS)} + t_{r(SPC)} - 7$	$0.5*t_{c(SPC)M} + (C2TDELAY+2) * t_{c(VCLK)} - t_{f(SPICS)} + t_{r(SPC)} + 5.5$	ns
			CSHOLD = 1	$0.5*t_{c(SPC)M} + (C2TDELAY+3) * t_{c(VCLK)} - t_{f(SPICS)} + t_{r(SPC)} - 7$	$0.5*t_{c(SPC)M} + (C2TDELAY+3) * t_{c(VCLK)} - t_{f(SPICS)} + t_{r(SPC)} + 5.5$	
		Setup time CS active until SPICLK low (clock polarity = 1)	CSHOLD = 0	$0.5*t_{c(SPC)M} + (C2TDELAY+2) * t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} - 7$	$0.5*t_{c(SPC)M} + (C2TDELAY+2) * t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} + 5.5$	ns
			CSHOLD = 1	$0.5*t_{c(SPC)M} + (C2TDELAY+3) * t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} - 7$	$0.5*t_{c(SPC)M} + (C2TDELAY+3) * t_{c(VCLK)} - t_{f(SPICS)} + t_{f(SPC)} + 5.5$	
9(6)	$t_{T2CDELAY}$	Hold time SPICLK low until CS inactive (clock polarity = 0)		$T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{f(SPC)} + t_{r(SPICS)} - 7$	$T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{f(SPC)} + t_{r(SPICS)} + 11$	ns
		Hold time SPICLK high until CS inactive (clock polarity = 1)		$T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{r(SPC)} + t_{r(SPICS)} - 7$	$T2CDELAY*t_{c(VCLK)} + t_{c(VCLK)} - t_{r(SPC)} + t_{r(SPICS)} + 11$	ns
10	t_{SPIENA}	SPIENAn Sample Point		$(C2TDELAY+1)* t_{c(VCLK)} - t_{f(SPICS)} - 29$	$(C2TDELAY+1)*t_{c(VCLK)}$	ns
11	$t_{SPIENAW}$	SPIENAn Sample point from write to buffer			$(C2TDELAY+2)*t_{c(VCLK)}$	ns

(1) The MASTER bit (SPIGCR1.0) is set and the CLOCK PHASE bit (SPIFMTx.16) is set.
(2) $t_{c(VCLK)}$ = interface clock cycle time = 1 / $f_{(VCLK)}$
(3) For rise and fall timings, see the Table 5-6.
(4) When the SPI is in Master mode, the following must be true:
For PS values from 1 to 255: $t_{c(SPC)M} \geq (PS+1)t_{c(VCLK)} \geq 40$ ns, where PS is the prescale value set in the SPIFMTx.[15:8] register bits.
For PS values of 0: $t_{c(SPC)M} = 2t_{c(VCLK)} \geq 40$ ns.
The external load on the SPICLK pin must be less than 60 pF.
(5) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPIFMTx.17).
(6) C2TDELAY and T2CDELAY is programmed in the SPIDELAY register

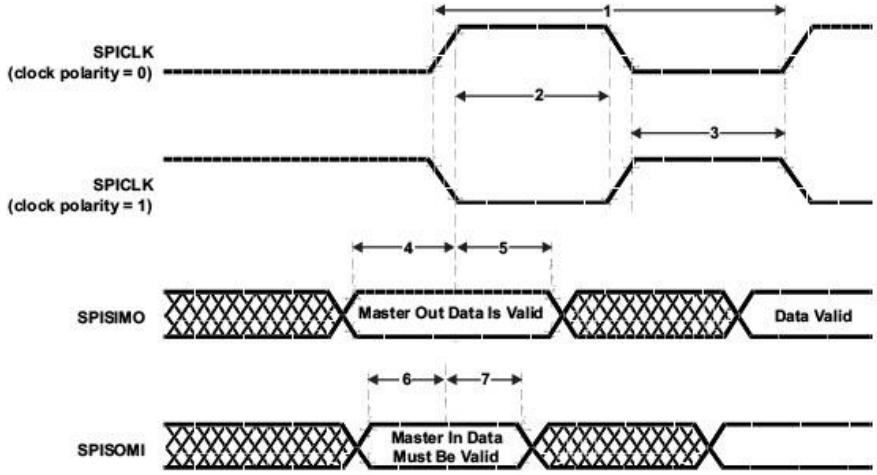


Figure 7-8. SPI Master Mode External Timing (CLOCK PHASE = 1)

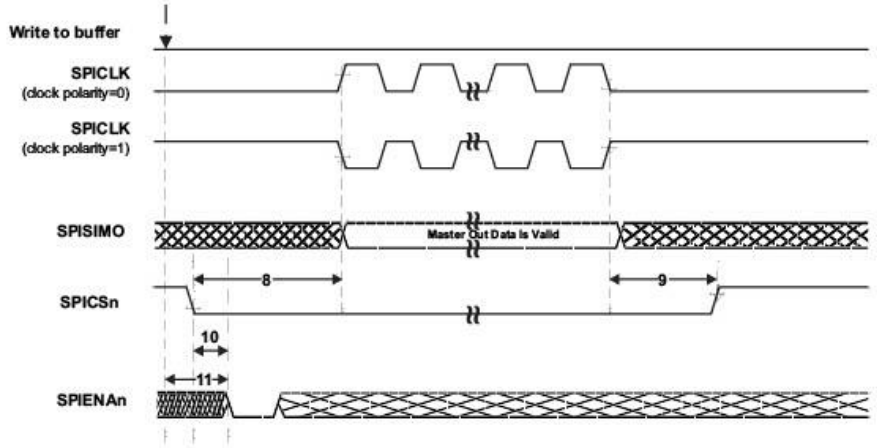


Figure 7-9. SPI Master Mode Chip Select Timing (CLOCK PHASE = 1)

7.7.5 SPI Slave Mode I/O Timings

Table 7-15. SPI Slave Mode External Timing Parameters (CLOCK PHASE = 0, SPICLK = input, SPISIMO = input, and SPISOMI = output)[1][2][3][4]

NO.	PARAMETER		MIN	MAX	UNIT
1	$t_{c(SPC)S}$	Cycle time, SPICLK[5]	40		ns
2[6]	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 0)	14		ns
	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 1)	14		
3[6]	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 0)	14		ns
	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 1)	14		
4[6]	$t_{d(SPCH-SOMI)S}$	Delay time, SPISOMI valid after SPICLK high (clock polarity = 0)		$t_{rf(SOMI)}$ + 20	ns
	$t_{d(SPCL-SOMI)S}$	Delay time, SPISOMI valid after SPICLK low (clock polarity = 1)		$t_{rf(SOMI)}$ + 20	
5[6]	$t_{h(SPCH-SOMI)S}$	Hold time, SPISOMI data valid after SPICLK high (clock polarity =0)	2		ns
	$t_{h(SPCL-SOMI)S}$	Hold time, SPISOMI data valid after SPICLK low (clock polarity =1)	2		
6[6]	$t_{su(SIMO-SPCL)S}$	Setup time, SPISIMO before SPICLK low (clock polarity = 0)	4		ns
	$t_{su(SIMO-SPCH)S}$	Setup time, SPISIMO before SPICLK high (clock polarity = 1)	4		
7[6]	$t_{h(SPCL-SIMO)S}$	Hold time, SPISIMO data valid after SPICLK low (clock polarity = 0)	2		ns
	$t_{h(SPCH-SIMO)S}$	Hold time, SPISIMO data valid after S PICLK high (clock polarity = 1)	2		
8	$t_{d(SPCL-SENAH)S}$	Delay time, SPIENAn high after last SPICLK low (clock polarity = 0)	$1.5t_{c(VCLK)}$	$2.5t_{c(VCLK)}+t_{r(ENAn)}+22$	ns
	$t_{d(SPCH-SENAH)S}$	Delay time, SPIENAn high after last SPICLK high (clock polarity = 1)	$1.5t_{c(VCLK)}$	$2.5t_{c(VCLK)}+t_{r(ENAn)}+22$	
9	$t_{d(SCSL-SENAL)S}$	Delay time, SPIENAn low after SPICSn low (if new data has been written to the SPI buffer)	$t_{f(ENAn)}$	$t_{c(VCLK)}+t_{f(ENAn)}+27$	ns

(1) The MASTER bit (SPIGCR1.0) is cleared and the CLOCK PHASE bit (SPIFMTx.16) is cleared.
(2) If the SPI is in slave mode, the following must be true: $t_{c(SPC)S} \geq (PS + 1)\ t_{c(VCLK)}$, where PS = prescale value set in SPIFMTx.[15:8].
(3) For rise and fall timings, see Table 5-6.
(4) $t_{c(VCLK)}$ = interface clock cycle time = $1 / f_{(VCLK)}$
(5) When the SPI is in Slave mode, the following must be true:
For PS values from 1 to 255: $t_{c(SPC)S} \geq (PS + 1)t_{c(VCLK)} \geq 40$ ns, where PS is the prescale value set in the SPIFMTx.[15:8] register bits.
For PS values of 0: $t_{c(SPC)S} = 2t_{c(VCLK)} \geq 40$ ns.
(6) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPIFMTx.17).

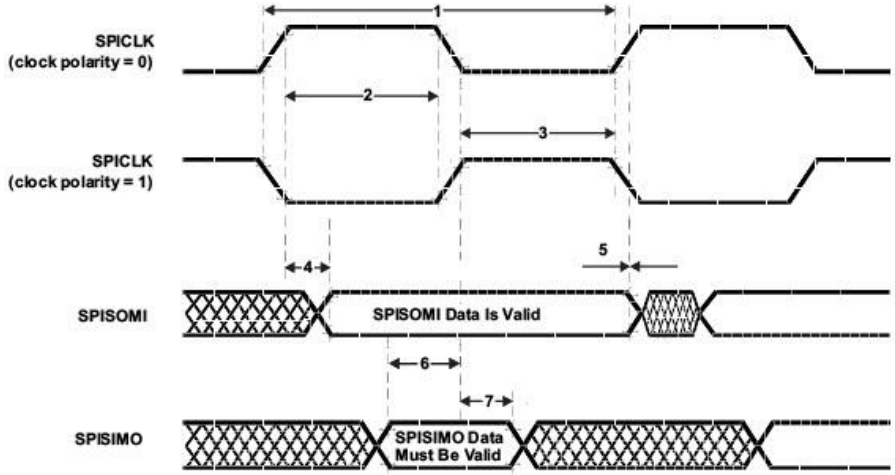


Figure 7-10. SPI Slave Mode External Timing (CLOCK PHASE = 0)

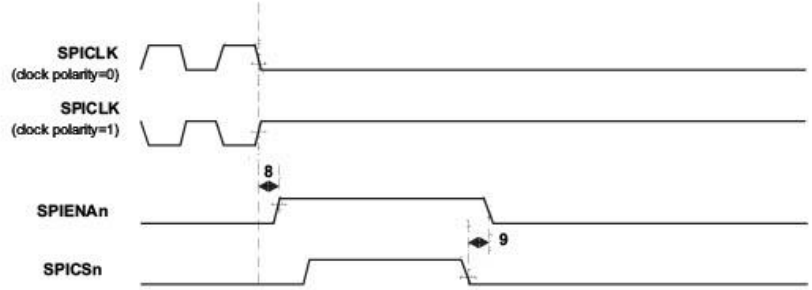


Figure 7-11. SPI Slave Mode Enable Timing (CLOCK PHASE = 0)

Table 7-16. SPI Slave Mode External Timing Parameters (CLOCK PHASE = 1, SPICLK = input, SPISIMO = input, and SPISOMI = output)[1][2][3][4]

NO.	PARAMETER		MIN	MAX	UNIT
1	$t_{c(SPC)S}$	Cycle time, SPICLK[5]	40		ns
2[6]	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 0)	14		ns
	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 1)	14		
3[6]	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 0)	14		ns
	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 1)	14		
4[6]	$t_{d(SOMI-SPCL)S}$	Dealy time, SPISOMI data valid after SPICLK low (clock polarity = 0)		$t_{rf(SOMI)}$ + 20	ns
	$t_{d(SOMI-SPCH)S}$	Delay time, SPISOMI data valid after SPICLK high (clock polarity = 1)		$t_{rf(SOMI)}$ + 20	
5[6]	$t_{h(SPCL-SOMI)S}$	Hold time, SPISOMI data valid after SPICLK high (clock polarity =0)	2		ns
	$t_{h(SPCH-SOMI)S}$	Hold time, SPISOMI data valid after SPICLK low (clock polarity =1)	2		
6[6]	$t_{su(SIMO-SPCH)S}$	Setup time, SPISIMO before SPICLK high (clock polarity = 0)	4		ns
	$t_{su(SIMO-SPCL)S}$	Setup time, SPISIMO before SPICLK low (clock polarity = 1)	4		
7[6]	$t_{v(SPCH-SIMO)S}$	High time, SPISIMO data valid after SPICLK high (clock polarity = 0)	2		ns
	$t_{v(SPCL-SIMO)S}$	High time, SPISIMO data valid after SPICLK low (clock polarity = 1)	2		
8	$t_{d(SPCH-SENAH)S}$	Delay time, SPIENAn high after last SPICLK high (clock polarity = 0)	$1.5t_{c(VCLK)}$	$2.5t_{c(VCLK)}+t_{f(ENAn)}$ + 22	ns
	$t_{d(SPCL-SENAH)S}$	Delay time, SPIENAn high after last SPICLK low (clock polarity = 1)	$1.5t_{c(VCLK)}$	$2.5t_{c(VCLK)}+t_{f(ENAn)}$ + 22	
9	$t_{d(SCSL-SENAL)S}$	Delay time, SPIENAn low after SPICSn low (if new data has been written to the SPI buffer)	$t_{f(ENAn)}$	$t_{c(VCLK)}+t_{f(ENAn)}$+ 27	ns
10	$t_{d(SCSL-SOMI)S}$	Delay time, SOMI valid after SPICSn low (if new data has been written to the SPI buffer)	$t_{c(VCLK)}$	$2t_{c(VCLK)}+t_{rf(SOMI)}$+ 28	ns

(1) The MASTER bit (SPIGCR1.0) is cleared and the CLOCK PHASE bit (SPIFMTx.16) is set.
(2) If the SPI is in slave mode, the following must be true: tc(SPC)S ≤ (PS + 1) tc(VCLK), where PS = prescale value set in SPIFMTx.[15:8].
(3) For rise and fall timings, see Table 5-6.
(4) $t_{c(VCLK)}$ = interface clock cycle time = 1 /$f_{(VCLK)}$
(5) When the SPI is in Slave mode, the following must be true:
For PS values from 1 to 255: $t_{c(SPC)S} \geq (PS + 1)t_{c(VCLK)} \geq 40$ ns, where PS is the prescale value set in the SPIFMTx.[15:8] register bits.
For PS values of 0: $t_{c(SPC)S} = 2t_{c(VCLK)} \geq 40$ ns.
(6) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPIFMTx.17).

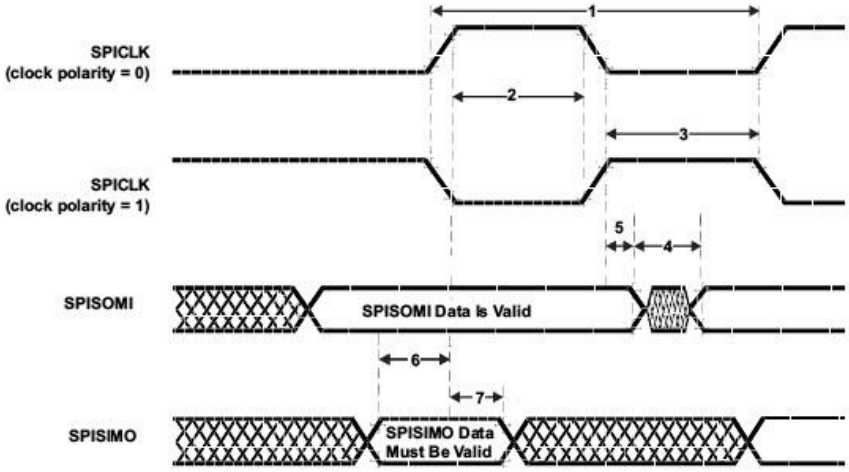


Figure 7-12. SPI Slave Mode External Timing (CLOCK PHASE = 1)

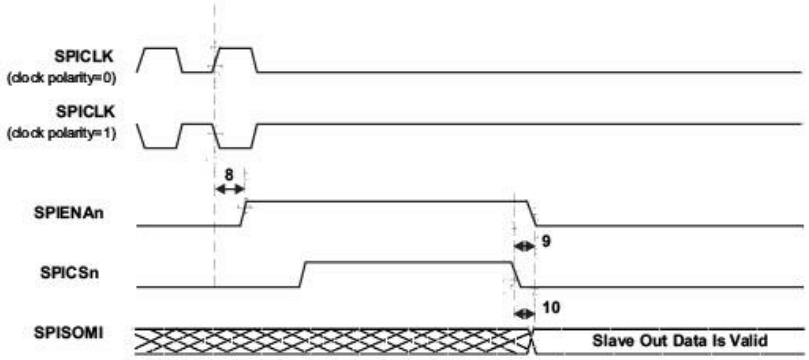


Figure 7-13. SPI Slave Mode Enable Timing (CLOCK PHASE = 1)

7.8 Enhanced Quadrature Encoder (eQEP)

Figure 7-14 shows the eQEP module interconnections on the device.

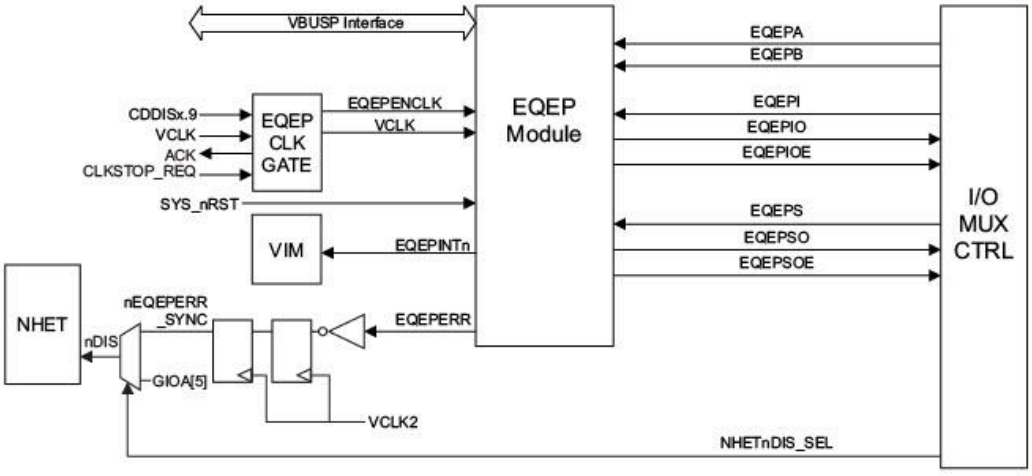


Figure 7-14. eQEP Module Interconnections

7.8.1 Clock Enable Control for eQEPx Modules

The device level control of the eQEP clock is accomplished through the enable/disable of the VCLK clock domain for eQEP only. This is realized using bit 9 of the CLKDDIS register. The eQEP clock source is enabled by default.

7.8.2 Using eQEPx Phase Error

The eQEP module sets the EQEPERR signal output whenever a phase error is detected in its inputs EQEPxA and EQEPxB. This error signal from both the eQEP modules is input to the connection selection multiplexor. As shown in Figure 7-14, the output of this selection multiplexor is inverted and connected to the N2HET module. This connection allows the application to define the response to a phase error indicated by the eQEP modules.

7.8.3 Input Connections to eQEPx Modules

The input connections to each of the eQEP modules can be selected between a double-VCLK-synchronized input or a double-VCLK-synchronized and filtered input, as shown in Table 7-17.

Table 7-17. Device-Level Input Synchronization

INPUT SIGNAL	CONTROL FOR DOUBLE-SYNCHRONIZED CONNECTION TO eQEPx	CONTROL FOR DOUBLE-SYNCHRONIZED AND FILTERED CONNECTION TO eQEPx
eQEPA	PINMMR8[0] = 1	PINMMR8[0] = 0 and PINMMR8[1] = 1
eQEPB	PINMMR8[8] = 1	PINMMR8[8] = 0 and PINMMR8[9] = 1
eQEPI	PINMMR8[16] = 1	PINMMR8[16] = 0 and PINMMR8[17] = 1
eQEPS	PINMMR8[24] = 1	PINMMR8[24] = 0 and PINMMR8[25] = 1

7.8.4 Enhanced Quadrature Encoder Pulse (eQEPx) Timing

Table 7-18. eQEPx Timing Requirements

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(QEPP)}$	QEP input period	Synchronous	2 $t_{c(VCLK)}$		cycles
		Synchronous, with input filter	2 $t_{c(VCLK)}$ + filter width		cycles
$t_{w(INDEXH)}$	QEP Index Input High Time	Synchronous	2 $t_{c(VCLK)}$		cycles
		Synchronous, with input filter	2 $t_{c(VCLK)}$ + filter width		cycles
$t_{w(INDEXL)}$	QEP Index Input Low Time	Synchronous	2 $t_{c(VCLK)}$		cycles
		Synchronous, with input filter	2 $t_{c(VCLK)}$ + filter width		cycles
$t_{w(STROBH)}$	QEP Strobe Input High Time	Synchronous	2 $t_{c(VCLK)}$		cycles
		Synchronous, with input filter	2 $t_{c(VCLK)}$ + filter width		cycles
$t_{w(STROBL)}$	QEP Strobe Input Low Time	Synchronous	2 $t_{c(VCLK)}$		cycles
		Synchronous, with input filter	2 $t_{c(VCLK)}$ + filter width		cycles

Table 7-19. eQEPx Switching Characteristics

PARAMETER		MIN	MAX	UNIT
$t_{d(CNTR)xin}$	Delay time, external clock to counter increment		4 $t_{c(VCLK)}$	cycles
$t_{d(PCS-OUT)QEP}$	Delay time, QEP input edge to position compare sync output		6 $t_{c(VCLK)}$	cycles

8 Device and Documentation Support

8.1 Device Support

8.1.1 Development Support

Texas Instruments (TI) offers an extensive line of development tools for the Hercules™ Safety generation of MCUs, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules.

The following products support development of Hercules™-based applications:

Software Development Tools

- Code Composer Studio™ Integrated Development Environment (IDE)
  - C/C++ Compiler
  - Code generation tools
  - Assembler/Linker
  - Cycle Accurate Simulator
- Application algorithms
- Sample applications code

Hardware Development Tools

- Development and evaluation boards
- JTAG-based emulators - XDS100™v2, XDS200, XDS560™ v2 emulator
- Flash programming tools
- Power supply
- Documentation and cables

8.1.1.1 Getting Started

This section gives a brief overview of the steps to take when first developing for a RM4x MCU device. For more detail on each of these steps, see the following:

- *Initialization of the TMS570LS043x, TMS570LS033x and RM42L432 Hercules ARM Cortex-R4 Microcontrollers* (SPNA163)
- *Compatibility Considerations: Migrating From RM48x or RM46x to RM42x Safety Microcontrollers* (SPNA174)

8.1.2 Device Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all devices.Each device has one of three prefixes: X, P, or null (no prefix) (for example, xRM46L852). These prefixes represent evolutionary stages of product development from engineering prototypes through fully qualified production devices/tools.

Device development evolutionary flow:

x Experimental device that is not necessarily representative of the final device's electrical specifications and may not use production assembly flow.

P Prototype device that is not necessarily the final silicon die and may not necessarily meet final electrical specifications.

null Fully-qualified production device.

x and P devices and TMDX development-support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

Production devices have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that prototype devices have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

Figure 8-1 illustrates the numbering and symbol nomenclature for the RM42L432.

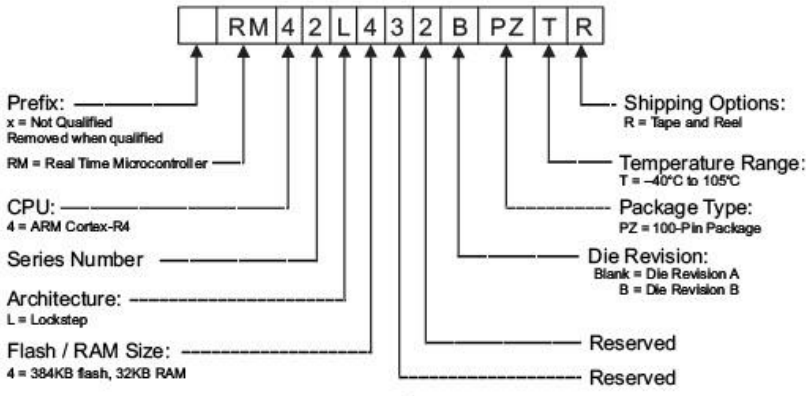


Figure 8-1. Device Numbering Conventions

8.2 Documentation Support

8.2.1 Related Documentation from Texas Instruments

The following documents describe the *RM42L432* microcontroller.

SPNU516 — ***RM42L42x 16/32-Bit RISC Flash Microcontroller Technical Reference Manual*** details the integration, the environment, the functional description, and the programming models for each peripheral and subsystem in the device.

SPNZ198 — ***RM42L432 Microcontroller, Silicon Revision A, Silicon Errata*** describes the usage notes and known exceptions to the functional specifications for the device silicon revision(s).

SPNZ227 — ***RM42x Microcontroller, Silicon Revision B, Silicon Errata*** describes the usage notes and known exceptions to the functional specifications for the device silicon revision(s).

SPNA207 — ***Calculating Equivalent Power-on-Hours for Hercules™ Safety MCUs*** details how to use the spreadsheet to calculate the aging effect of temperature on Texas Instruments Hercules Safety MCUs.

8.3 Community Resources

The following links connect to TI community resources. Linked contents are provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

TI E2E™ Online Community ***TI's Engineer-to-Engineer (E2E) Community.*** Created to foster collaboration among engineers. At e2e.ti.com, you can ask questions, share knowledge, explore ideas and help solve problems with fellow engineers.

TI Embedded Processors Wiki ***Texas Instruments Embedded Processors Wiki.*** Established to help developers get started with Embedded Processors from Texas Instruments and to foster innovation and growth of general knowledge about the hardware and software surrounding these devices.

8.4 Trademarks

Hercules, Code Composer Studio, XDS100, XDS560, E2E are trademarks of Texas Instruments.
CoreSight is a trademark of ARM Limited.
ARM, Cortex are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere.
All rights reserved.
All other trademarks are the property of their respective owners.

8.5 Electrostatic Discharge Caution

This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

8.6 Glossary

SLYZ022 — *TI Glossary*.

This glossary lists and explains terms, acronyms, and definitions.

8.7 Device Identification Code Register

The device identification code register identifies several aspects of the device including the silicon version. The details of the device identification code register are shown in Table 8-1. The device identification code register value for this device is:

- Rev 0 = 0x8048AD05
- Rev A = 0x8048AD0D
- Rev B = 0x8048AD15

Figure 8-2. Device ID Bit Allocation Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CP-15	UNIQUE ID														TECH
R-1	R-00000000100100														R-0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TECH			I/O VOLTAGE	PERIPH PARITY	FLASH ECC		RAM ECC	VERSION					1	0	1
R-101			R-0	R-1	R-10		R-1	R-00001					R-1	R-0	R-1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 8-1. Device ID Bit Allocation Register Field Descriptions

BIT	FIELD	VALUE	DESCRIPTION
31	CP15		Indicates the presence of coprocessor 15
		1	CP15 present
30-17	UNIQUE ID	100100	Silicon version (revision) bits. This bit field holds a unique number for a dedicated device configuration (die).
16-13	TECH		Process technology on which the device is manufactured.
		0101	F021
12	I/O VOLTAGE		I/O voltage of the device.
		0	I/O are 3.3v
11	PERIPHERAL PARITY		Peripheral Parity
		1	Parity on peripheral memories
10-9	FLASH ECC		Flash ECC
		10	Program memory with ECC
8	RAM ECC		Indicates if RAM memory ECC is present.
		1	ECC implemented
7-3	REVISION	0	Revision of the Device.
2-0	FAMILY ID	101	The platform family ID is always 0b101

8.8 Die Identification Registers

The two die ID registers at addresses 0xFFFFFF7C and 0xFFFFFF80 form a 64-bit die id with the information as shown in Table 8-2.

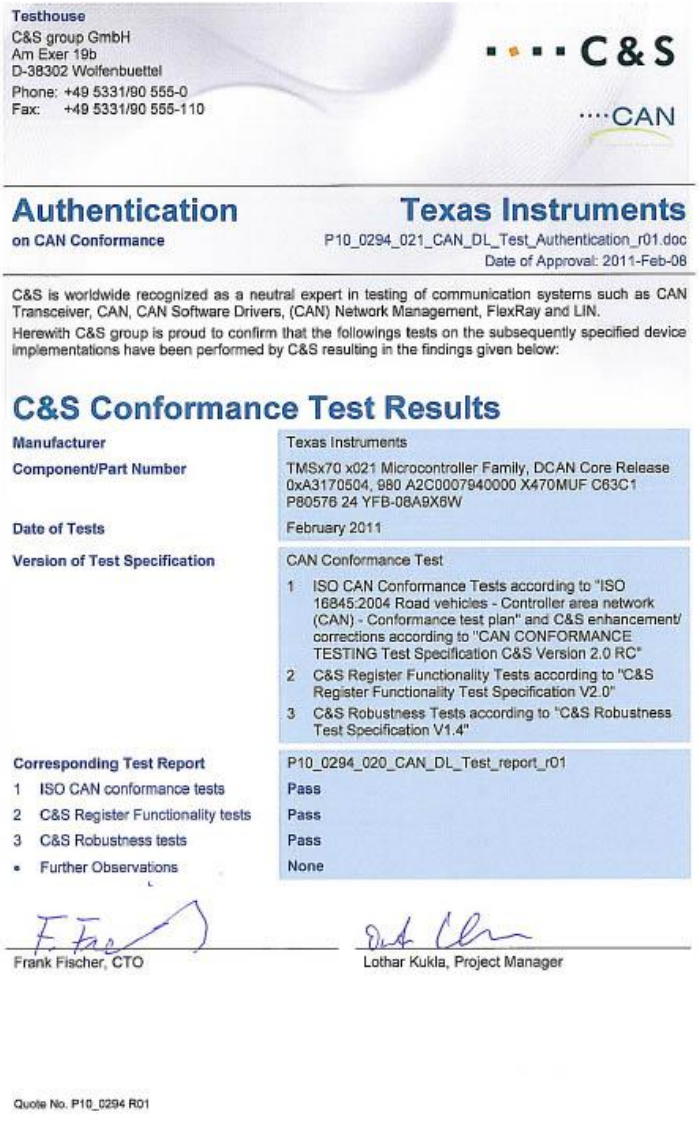
Table 8-2. Die-ID Registers

ITEM	NO. OF BITS	BIT LOCATION
X Coord. on Wafer	12	0xFFFFFF7C[11:0]
Y Coord. on Wafer	12	0xFFFFFF7C[23:12]
Wafer #	8	0xFFFFFF7C[31:24]
Lot #	24	0xFFFFFF80[23:0]
Reserved	8	0xFFFFFF80[31:24]

8.9 Module Certifications

The following communications modules have received certification of adherence to a standard.

8.9.1 DCAN Certification



Testhouse
C&S group GmbH
Am Exer 19b
D-38302 Wolfenbuettel
Phone: +49 5331/90 555-0
Fax: +49 5331/90 555-110

C&S
CAN

Authentication
on CAN Conformance

Texas Instruments
P10_0294_021_CAN_DL_Test_Authentication_r01.doc
Date of Approval: 2011-Feb-08

C&S is worldwide recognized as a neutral expert in testing of communication systems such as CAN Transceiver, CAN, CAN Software Drivers, (CAN) Network Management, FlexRay and LIN.

Herewith C&S group is proud to confirm that the followings tests on the subsequently specified device implementations have been performed by C&S resulting in the findings given below:

C&S Conformance Test Results

Manufacturer	Texas Instruments
Component/Part Number	TMSx70 x021 Microcontroller Family, DCAN Core Release 0xA3170504, 980 A2C0007940000 X470MUF C63C1 P80576 24 YFB-08A9X6W
Date of Tests	February 2011
Version of Test Specification	CAN Conformance Test 1 ISO CAN Conformance Tests according to "ISO 16845:2004 Road vehicles - Controller area network (CAN) - Conformance test plan" and C&S enhancement/ corrections according to "CAN CONFORMANCE TESTING Test Specification C&S Version 2.0 RC" 2 C&S Register Functionality Tests according to "C&S Register Functionality Test Specification V2.0" 3 C&S Robustness Tests according to "C&S Robustness Test Specification V1.4"
Corresponding Test Report	P10_0294_020_CAN_DL_Test_report_r01
1 ISO CAN conformance tests	Pass
2 C&S Register Functionality tests	Pass
3 C&S Robustness tests	Pass
• Further Observations	None

Frank Fischer, CTO

Lothar Kukla, Project Manager

Quote No. P10_0294 R01

Figure 8-3. DCAN Certification

8.9.2 LIN Certifications

8.9.2.1 LIN Master Mode

ihr

AUTOMOTIVE
Engineering
Services & Products

DAkkS
Deutsche Akkreditierungsstelle
D-PL-17201-01-01

Accredited test lab by DAkkS GmbH according DIN ISO/IEC 17025.
The accreditation is valid for all test methods stated in the certificate.

Test Summary
for
LIN 2.1 Conformance Test - Master

Client / Manufacturer:	**Texas Instruments**
	Automotive Applications
	12500 TI Boulevard
	75243 DALLAS, TEXAS
	UNITED STATES OF AMERICA
Implementation Under Test:	**Microcontroller TMS570LS3137**
Part Number:	**LIN Master Mode**
Revision:	SW: : 2013-05-31_IHR_LIN
Test Sample Marking:	TMX570LS 3137 CZWTQQ1 YFC-2AAQ32W
Test Board:	Gladiator 337 SN# 58 rev. B
Performed Tests:	**LIN OSI Layer 2 – Data Link Layer**
	Node Configuration / Network Management
Specification References:	**LIN Conformance Test Specification, 10-Oct-2008**
	for the LIN Specification Package Revision 2.1 , 24-Nov-2006
Result / Status:	**The device has *passed* the test.**
Version / Date:	1.0 / 13-May-2013
Document Number:	Test Summary_DLL21_Master_20121130_130513_TMS570LS_V1.0.doc
ihr Reference:	20121130

This test summary don't represent a complete test report according the LIN consortium.
It contains 7 pages and shall not be reproduced except in full without written approval of the *ihr* Test Center. All performed test results concerns the above mentioned IUT revision only.

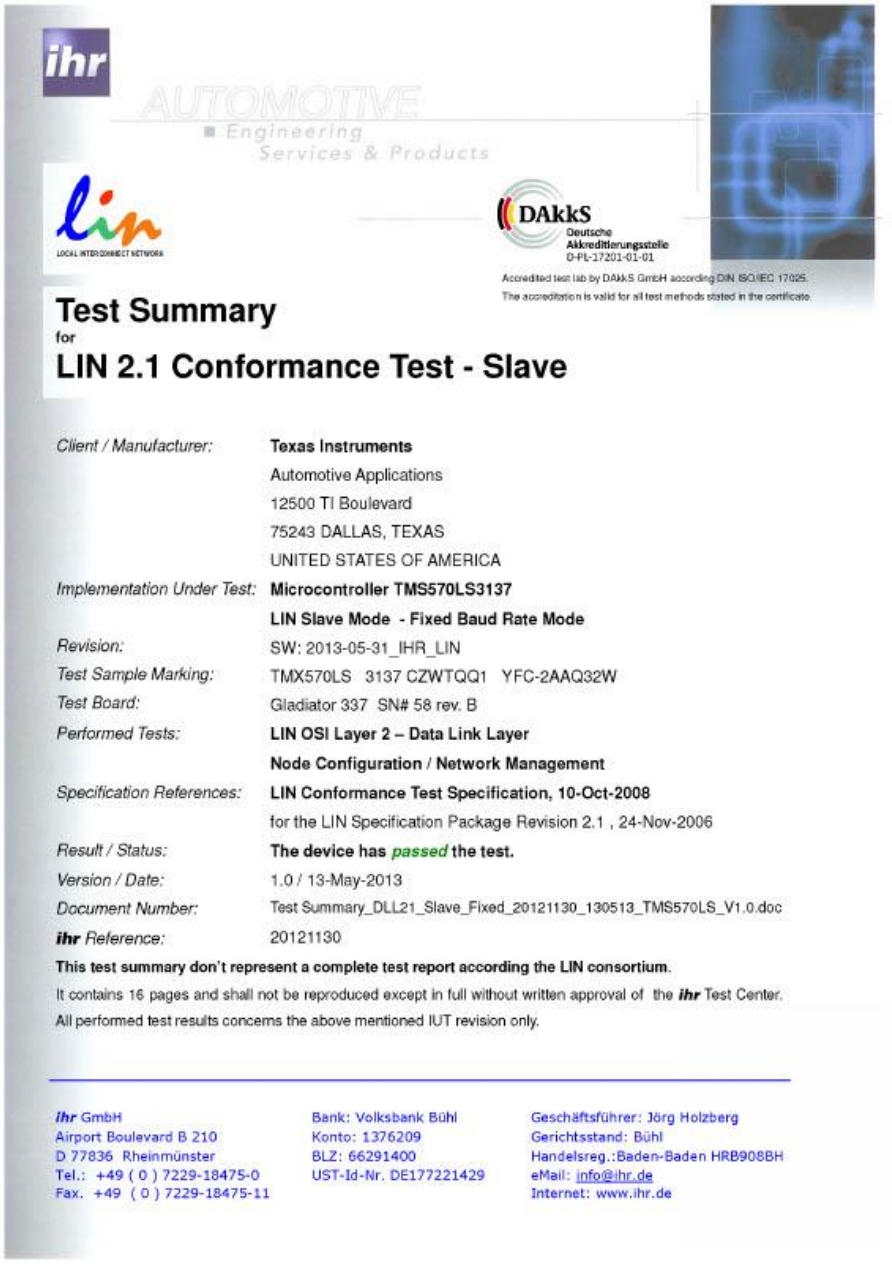
ihr GmbH
Airport Boulevard B 210
D 77836 Rheinmünster
Tel.: +49 (0) 7229-18475-0
Fax. +49 (0) 7229-18475-11

Bank: Volksbank Bühl
Konto: 1376209
BLZ: 66291400
UST-Id-Nr. DE177221429

Geschäftsführer: Jörg Holzberg
Gerichtsstand: Bühl
Handelsreg.:Baden-Baden HRB908BH
eMail: info@ihr.de
Internet: www.ihr.de

Figure 8-4. LIN Certification - Master Mode

8.9.2.2 LIN Slave Mode - Fixed Baud Rate



ihr

AUTOMOTIVE
Engineering
Services & Products

LOCAL INTERCONNECT NETWORK

DAkkS
Deutsche
Akkreditierungsstelle
D-PL-17201-01-01

Accredited test lab by DAkkS GmbH according DIN ISO/IEC 17025.
The accreditation is valid for all test methods stated in the certificate.

Test Summary
for
LIN 2.1 Conformance Test - Slave

Client / Manufacturer:	**Texas Instruments**
	Automotive Applications
	12500 TI Boulevard
	75243 DALLAS, TEXAS
	UNITED STATES OF AMERICA
Implementation Under Test:	**Microcontroller TMS570LS3137**
	LIN Slave Mode - Fixed Baud Rate Mode
Revision:	SW: 2013-05-31_IHR_LIN
Test Sample Marking:	TMX570LS 3137 CZWTQQ1 YFC-2AAQ32W
Test Board:	Gladiator 337 SN# 58 rev. B
Performed Tests:	**LIN OSI Layer 2 – Data Link Layer**
	Node Configuration / Network Management
Specification References:	**LIN Conformance Test Specification, 10-Oct-2008**
	for the LIN Specification Package Revision 2.1 , 24-Nov-2006
Result / Status:	**The device has *passed* the test.**
Version / Date:	1.0 / 13-May-2013
Document Number:	Test Summary_DLL21_Slave_Fixed_20121130_130513_TMS570LS_V1.0.doc
***ihr** Reference:*	20121130

This test summary don't represent a complete test report according the LIN consortium.
It contains 16 pages and shall not be reproduced except in full without written approval of the ***ihr*** Test Center.
All performed test results concerns the above mentioned IUT revision only.

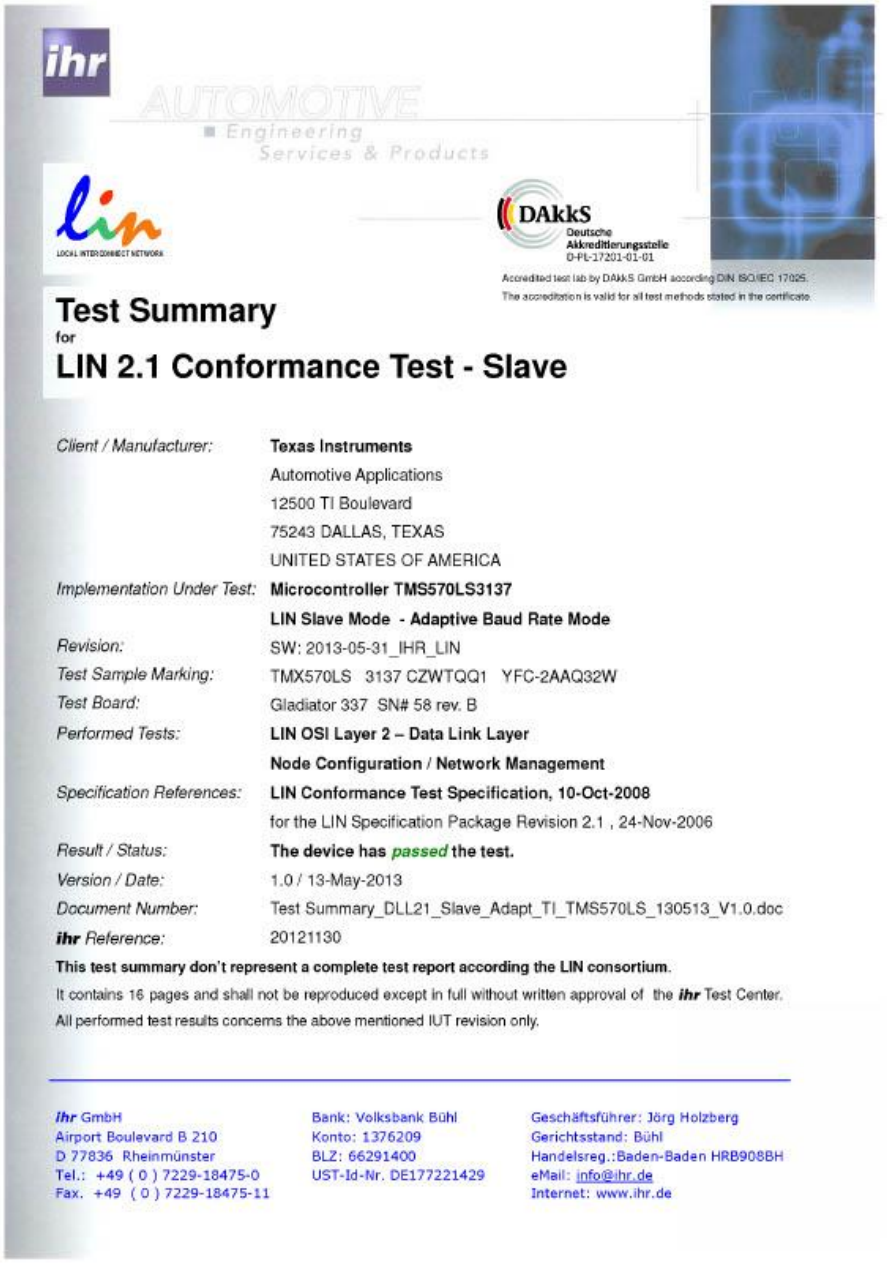
ihr GmbH
Airport Boulevard B 210
D 77836 Rheinmünster
Tel.: +49 (0) 7229-18475-0
Fax. +49 (0) 7229-18475-11

Bank: Volksbank Bühl
Konto: 1376209
BLZ: 66291400
UST-Id-Nr. DE177221429

Geschäftsführer: Jörg Holzberg
Gerichtsstand: Bühl
Handelsreg.:Baden-Baden HRB908BH
eMail: info@ihr.de
Internet: www.ihr.de

Figure 8-5. LIN Certification - Slave Mode - Fixed Baud Rate

8.9.2.3 LIN Slave Mode - Adaptive Baud Rate



ihr

AUTOMOTIVE
Engineering
Services & Products

LIN
LOCAL INTERCONNECT NETWORK

DAkkS
Deutsche
Akkreditierungsstelle
D-PL-17201-01-01

Accredited test lab by DAkkS GmbH according DIN ISO/IEC 17025.
The accreditation is valid for all test methods stated in the certificate.

Test Summary
for
LIN 2.1 Conformance Test - Slave

Client / Manufacturer:	**Texas Instruments** Automotive Applications 12500 TI Boulevard 75243 DALLAS, TEXAS UNITED STATES OF AMERICA
Implementation Under Test:	**Microcontroller TMS570LS3137** **LIN Slave Mode - Adaptive Baud Rate Mode**
Revision:	SW: 2013-05-31_IHR_LIN
Test Sample Marking:	TMX570LS 3137 CZWTQQ1 YFC-2AAQ32W
Test Board:	Gladiator 337 SN# 58 rev. B
Performed Tests:	**LIN OSI Layer 2 – Data Link Layer** **Node Configuration / Network Management**
Specification References:	**LIN Conformance Test Specification, 10-Oct-2008** for the LIN Specification Package Revision 2.1 , 24-Nov-2006
Result / Status:	**The device has *passed* the test.**
Version / Date:	1.0 / 13-May-2013
Document Number:	Test Summary_DLL21_Slave_Adapt_TI_TMS570LS_130513_V1.0.doc
***ihr** Reference:*	20121130

This test summary don't represent a complete test report according the LIN consortium.
It contains 16 pages and shall not be reproduced except in full without written approval of the ***ihr*** Test Center.
All performed test results concerns the above mentioned IUT revision only.

ihr GmbH
Airport Boulevard B 210
D 77836 Rheinmünster
Tel.: +49 (0) 7229-18475-0
Fax. +49 (0) 7229-18475-11

Bank: Volksbank Bühl
Konto: 1376209
BLZ: 66291400
UST-Id-Nr. DE177221429

Geschäftsführer: Jörg Holzberg
Gerichtsstand: Bühl
Handelsreg.:Baden-Baden HRB908BH
eMail: info@ihr.de
Internet: www.ihr.de

Figure 8-6. LIN Certification - Slave Mode - Adaptive Baud Rate

9 Mechanical Packaging and Orderable Addendum

9.1 Packaging Information

The following pages include mechanical packaging and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead finish/ Ball material (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
RM42L432BPZT	ACTIVE	LQFP	PZ	100	90	RoHS & Green	NIPDAU	Level-3-260C-168 HR	-40 to 105	RM42 L432BPZT	Samples

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
RoHS Exempt: TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
Green: TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

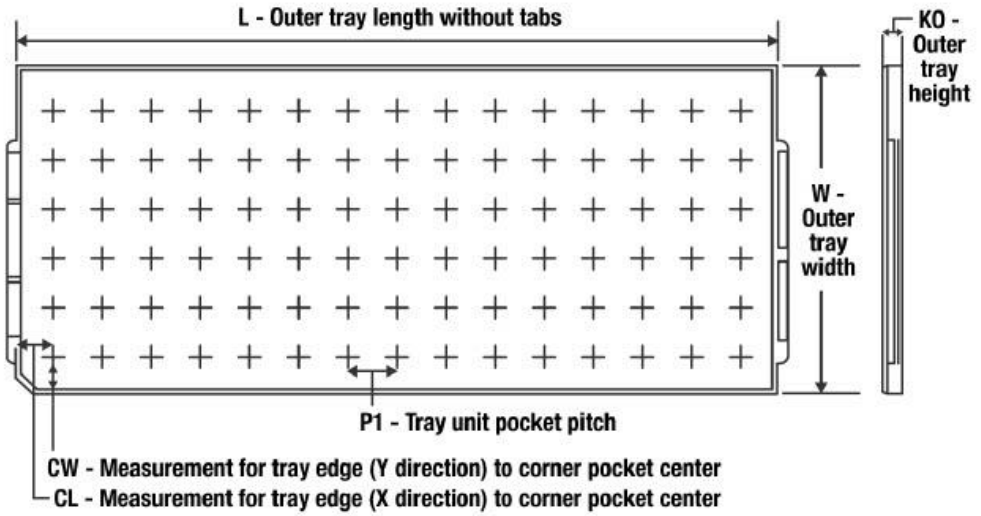
(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

TRAY



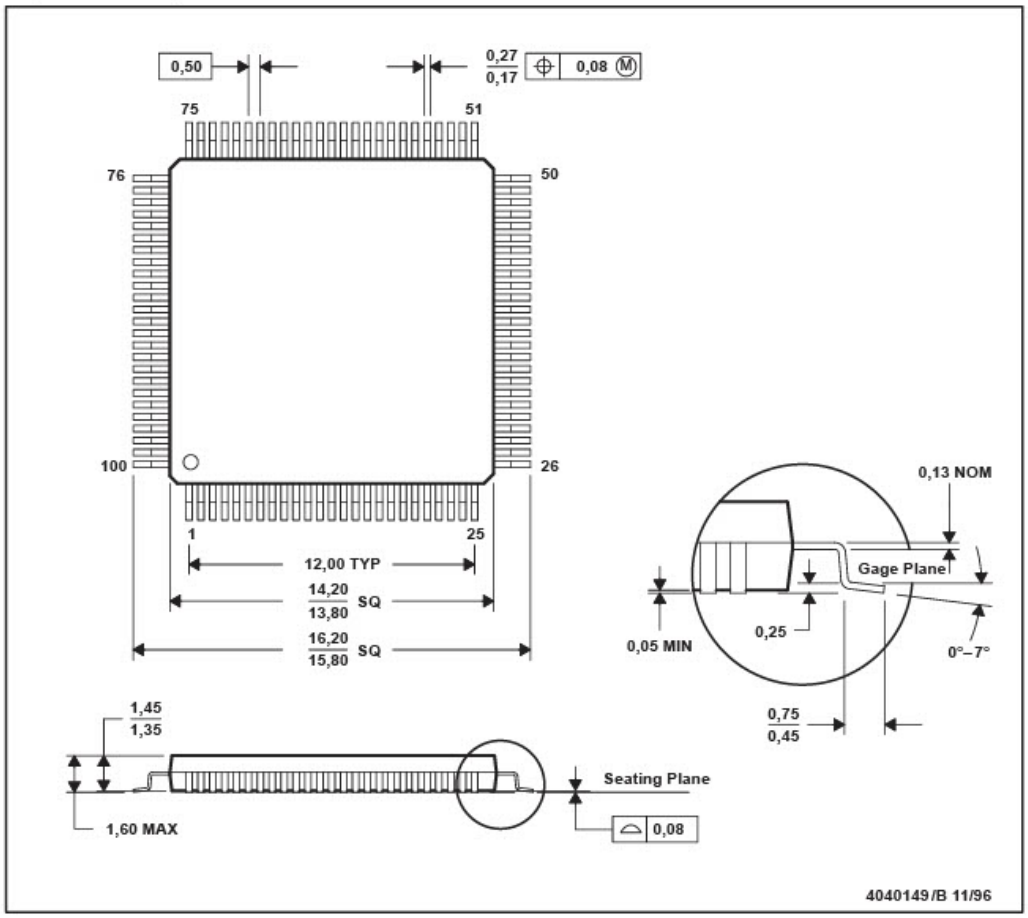
Chamfer on Tray corner indicates Pin 1 orientation of packed units.

*All dimensions are nominal

Device	Package Name	Package Type	Pins	SPQ	Unit array matrix	Max temperature (°C)	L (mm)	W (mm)	K0 (µm)	P1 (mm)	CL (mm)	CW (mm)
RM42L432BPZT	PZ	LQFP	100	90	6 x 15	150	315	135.9	7620	20.3	15.4	15.4

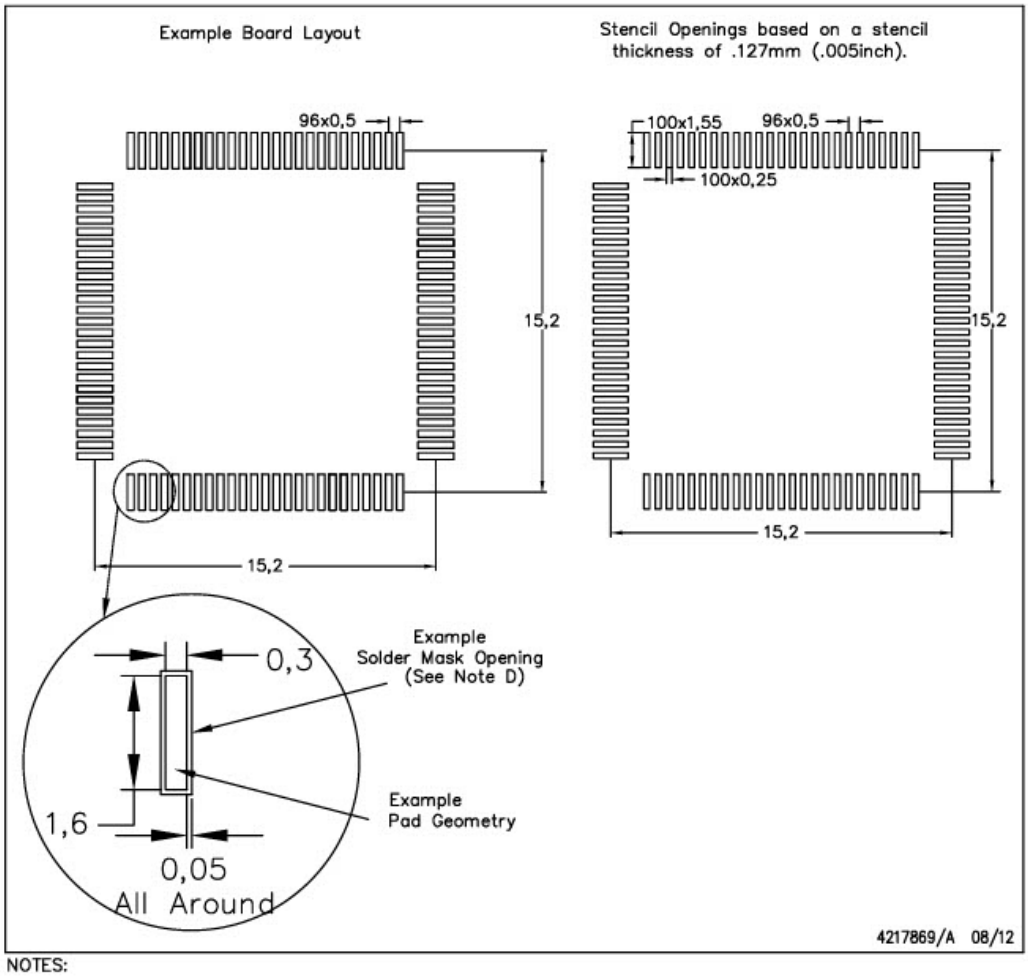
MTQF013A – OCTOBER 1994 – REVISED DECEMBER 1996

PZ (S-PQFP-G100) PLASTIC QUAD FLATPACK



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

PZ (S–PQFP–G100) PLASTIC QUAD FLAT PACK



NOTES:

A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Laser cutting apertures with trapezoidal walls and also rounding corners will offer better paste release. Customers should contact their board assembly site for stencil design recommendations. Example stencil design based on a 50% volumetric metal load solder paste. Refer to IPC–7525 for other stencil recommendations.
D. Customers should contact their board fabrication site for solder mask tolerances between and around signal pads.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

References

1. Augustine, G. J., Fitzpatrick, D., Hall, W. C., LaMantia, A.-S., Mooney, R. D., Platt, M. L., & White, L. E. (2018). *Neuroscience* (D. Purves, Ed.; 6th ed.). Sinauer Associates Is an Imprint of Oxford University Press.

2. Hedges, V. (2022). *Introduction to neuroscience*. Michigan State University.

3. Shih, J. J., Krusienski, D. J., & Wolpaw, J. R. (2012). Brain-computer interfaces in medicine. *Mayo Clinic Proceedings*, *87*(3), 268–279. https://doi.org/10.1016/j.mayocp.2011.12.008

4. John, C. E., & Jones, S. R. (2006). Chapter 4 - Fast Scan Cyclic Voltammetry of Dopamine and Serotonin in Mouse Brain Slices. In *Electrochemical Methods for Neuroscience.* (Ser. Frontiers in Neuroengineering, pp. 49–62). essay, CRC Press.

5. VB. (n.d.). *Nervous System Neuron*. Visible Body. Retrieved from https://www.visiblebody.com/learn/nervous/glossary.

6. Juárez Olguín, H., Calderón Guzmán, D., Hernández García, E., & Barragán Mejía, G. (2016). The role of dopamine and its dysfunction as a consequence of oxidative stress. *Oxidative Medicine and Cellular Longevity*, *2016*, 1–13. https://doi.org/10.1155/2016/9730467

7. Gepshtein, S., Li, X., Snider, J., Plank, M., Lee, D., & Poizner, H. (2014). Dopamine function and the efficiency of human movement. *Journal of Cognitive Neuroscience*, *26*(3), 645–657. https://doi.org/10.1162/jocn_a_00503

8. Arbuthnott, G. W., & Wickens, J. (2007). Space, time and dopamine. *Trends in Neurosciences*, *30*(2), 62–69. https://doi.org/10.1016/j.tins.2006.12.003

9. Wise, R. A. (2004). Dopamine, learning and motivation. *Nature Reviews Neuroscience*, *5*(6), 483–494. https://doi.org/10.1038/nrn1406

10. Bogacz, R. (2020). Dopamine role in learning and action inference. *eLife*, *9*. https://doi.org/10.7554/elife.53262

11. Titulaer, J., Björkholm, C., Feltmann, K., Malmlöf, T., Mishra, D., Bengtsson Gonzales, C., Schilström, B., & Konradsson-Geuken, Å. (2021). The importance of ventral hippocampal dopamine and norepinephrine in recognition memory. *Frontiers in Behavioral Neuroscience*, *15*, 1–6. https://doi.org/10.3389/fnbeh.2021.667244

12. Castillo Díaz, F., Caffino, L., & Fumagalli, F. (2021). Bidirectional role of dopamine in learning and memory-active forgetting. *Neuroscience & Biobehavioral Reviews*, *131*, 953–963. https://doi.org/10.1016/j.neubiorev.2021.10.011

13. Marsden, C. A. (2006). Dopamine: The rewarding years. *British Journal of Pharmacology*, *147*(S1), 136–144. https://doi.org/10.1038/sj.bjp.0706473

14. Chefer, V. I., Thompson, A. C., Zapata, A., & Shippenberg, T. S. (2009). Overview of brain microdialysis. *Current Protocols in Neuroscience*, *47*(1). https://doi.org/10.1002/0471142301.ns0701s47

15. Bungay, P. M., Newton-Vinson, P., Isele, W., Garris, P. A., & Justice, J. B. (2003). Microdialysis of dopamine interpreted with quantitative model incorporating probe implantation trauma. *Journal of Neurochemistry*, *86*(4), 932–946. https://doi.org/10.1046/j.1471-4159.2003.01904.x

16. Volkow, N. D., Fowler, J. S., & Gately, S. J. (1996). PET Evaluation of the Dopamine System of the Human Brain. *The Journal of Nuclear Medicine*, *37*(7), 1242–1256. https://doi.org/PMID: 8965206

17. Reneman, L., van der Pluijm, M., Schrantee, A., & van de Giessen, E. (2021). Imaging of the dopamine system with focus on pharmacological MRI and neuromelanin imaging. *European Journal of Radiology*, *140*. https://doi.org/10.1016/j.ejrad.2021.109752

18. Cid, E., & de la Prida, L. M. (2019). Methods for single-cell recording and labeling in vivo. *Journal of Neuroscience Methods*, *325*. https://doi.org/10.1016/j.jneumeth.2019.108354

19. UCLA, Dept. of P. (n.d.). *In Vivo Fast-Scan Cyclic Voltammetry*. Wassum Lab. Retrieved from http://wassumlab.psych.ucla.edu/research/in-vivo-fast-scan-cyclic-voltammetry/.

20. Foster, J. (2014). *Design of a Portable Fast Scan Cyclic Voltammetry System for Measuring Neurotransmitter Levels* (thesis). *ScholarWorks@GVSU*. Retrieved from https://scholarworks.gvsu.edu/cgi/viewcontent.cgi?article=1708&context=theses.