# DLP® NIRscan™ Nano EVM User's Guide

# User's Guide

Literature Number: DLPU030G
June 2015–Revised August 2017

# Contents

# List of Figures

## List of Tables

## 0.1 Trademarks

NIRscan, Tiva, TivaWare, SimpleLink, Code Composer Studio are trademarks of Texas Instruments.
DLP is a registered trademark of Texas Instruments.
ARM is a registered trademark of ARM Limited.
Apple, iPhone, iPad are registered trademarks of Apple Inc.
Bluetooth is a registered trademark of Bluetooth SIG.
Windows is a registered trademark of Microsoft Corporation.
All other trademarks are the property of their respective owners.

# Read This First

## About This Guide

The DLP® NIRscan™ Nano EVM is a third-party implementation of the next generation DLP reference design to enable faster development cycles for mobile spectrometer applications.

This guide is an introductory document for the DLP NIRscan Nano EVM that provides an overview of the system and the system software.

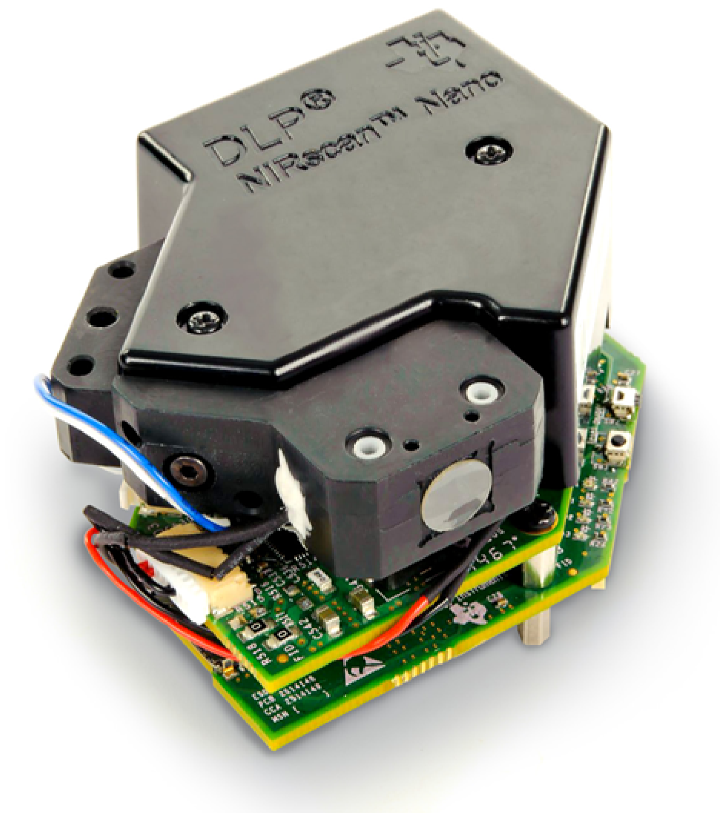


**Figure 1. DLP NIRscan Nano Evaluation Module**

## Related Documentation from TI

- DLP NIRscan Nano EVM: Getting Started Out of the Box video
- DLP2010NIR data sheet: *DLP 0.2 WVGA Near-Infrared DMD*, DLPS059
- DLPC150 data sheet: *DLPC150 DLP Digital Controller for Advanced Light Control*, DLPS048
- DLPC150 programmer's guide: *DLPC150 Programmer's Guide User's Guide*, DLPU031
- DLP design guide: *DLP Spectrometer Design Considerations*, DLPA049
- DLP Application Report: *DLP Spectrometer Optical Design Considerations*, DLPA062
- DLP Application Report: *Flexible Trade-offs in Maximizing SNR and Resolution in TI DLP® Technology-Based Spectrometer Systems*, DLPA066
- DLP Application Report: *Signal Chain Performance Optimizations in the TI DLP® Technology-Based Spectrometer*, DLPA072
- Tiva™ TM4C1297 data sheet: *Tiva TM4C1297NCZAD Microcontroller Data Sheet* SPMS435
- TivaWare™ USB library: *TivaWare USB Library User's Guide*, SPMU297
- TivaWare™ peripheral driver library: *TivaWare Peripheral Driver Library User's Guide*, SPMU298
- TI-RTOS 2.10: *TI-RTOS 2.10 User's Guide*, SPRUHD4
- CC2564MODN data sheet:*CC2564MODN Bluetooth® Host Controller Interface Module*, SWRS160
- ADS1255 data sheet:*Very Low Noise, 24-Bit Analog-to-Digital Converter Data Sheet*, SBAS288

## If You Need Assistance

Search the DLP NIRscan Nano E2E Community Support forum.

Search the TM4C Microcontrollers TI E2E Community Support forums.

Search the Bluetooth® CC256x TI E2E Community Support forums.

Search the SimpleLink™ Bluetooth® CC256x Wiki.

# DLP NIRscan Nano Overview

## 1.1 Introduction

Spectroscopy is a powerful technique for recognizing and characterizing physical materials through the variations in absorption or emission of different wavelengths of light by a sample. Spectrometers measure the variation of light absorption of materials. The DLP® NIRscan Nano™ EVM is a complete evaluation module to design a high performance, affordable near-infrared portable spectrometer. This flexible tool contains everything a designer needs to start developing a DLP-based spectrometer right out of the box. DLP technology enables handheld spectral analyzers for use in the food, pharmaceutical, oil and gas, medical, security, and other emerging industries to deliver lab performance levels in the field. The EVM contains the DLP2010NIR digital micromirror device, DLPC150 digital controller, and DLPA2005 integrated power management components. This technology brings together a set of components providing an efficient and compelling spectroscopy system solution for:

- Portable process analyzers
- Ultra-mobile spectrometer

The new DLP2010NIR DMD is optimized for operation at wavelengths between 700 and 2500 nm. The DLP NIRscan Nano EVM is one possible implementation of this new DLP technology, operating from 900 to 1700 nm.

DLP based spectrometers replace the traditional linear array detector with a DMD for wavelength selection and a single point detector as shown in Figure 1-1. By sequentially scanning through the columns (turning on specific columns of pixels) of the DMD, a particular wavelength of light is directed to the detector and captured. Refer to DLP Spectrometer Design Consideration for details.

DLP technology in Near-Infrared (NIR) spectroscopy provides the following advantages:

- Higher performance through the use of a larger single point 1-mm detector in comparison to a linear array with very small pixels.
- Lower cost system through the use of single element detectors and low cost optics. The high resolution DMD allows custom patterns to compensate for the optical distortion of each individual system.
- Greater signal captured not only because of the larger entendue of the DMD compared to traditional technologies, but also through the use of fast, flexible, and programmable patterns and spectral filters.
- With programmable patterns, a DLP spectrometer can [1]:
  - Vary the intensity of light to the detector by controlling the number of pixels in a column.
  - Vary the resolution of the system by controlling the width of the columns.
  - Use a set of Hadamard patterns that capture multiple wavelengths of light per pattern. Individual wavelengths are then retrieved through a decoding procedure. Each pattern turns on 50% of the DMD pixel at a time, directing much greater signal into the detector than the column scan shown in Figure 1-1.
  - Use custom spectral filters to select specific wavelengths of interest.

(1) Some of these (intensity variation, custom spectral filters) are not enabled in DLP NIRscan Nano software.

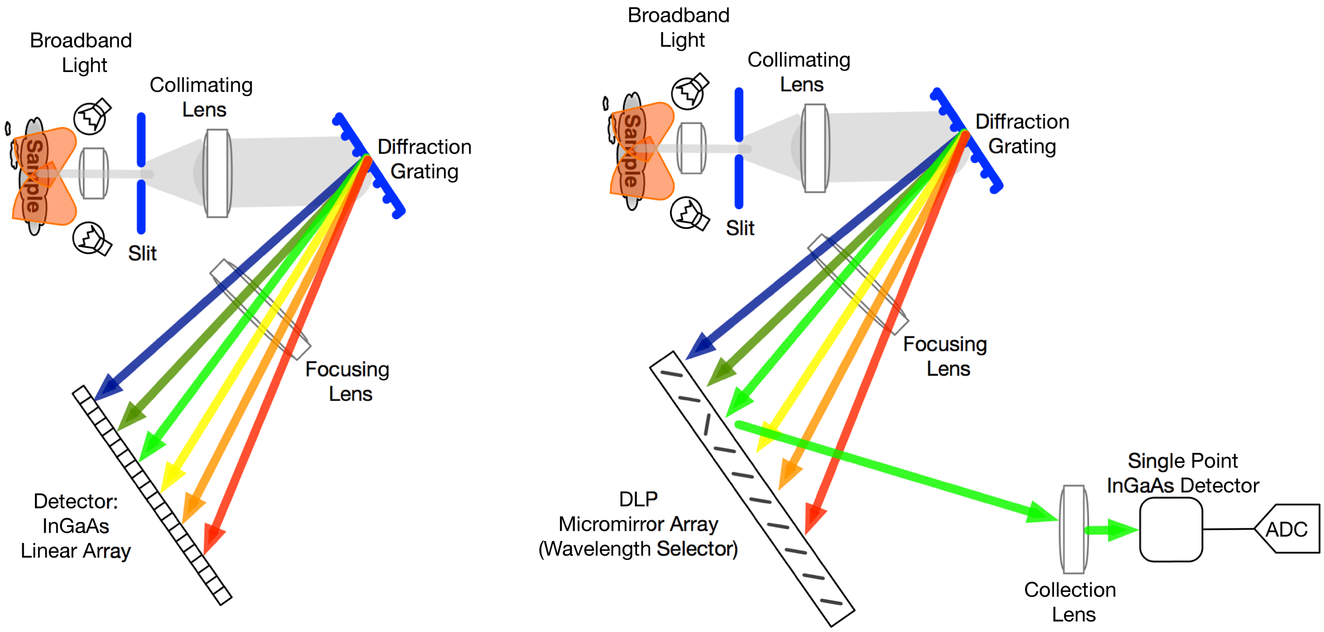


**Figure 1-1. Traditional Versus DLP-based Spectrometer**

## 1.2 What is the DLP NIRscan Nano EVM?

The DLP NIRscan Nano EVM is a complete NIR spectrometer EVM using DLP technology. The EVM package includes:

- Near-infrared optomechanical spectrometer engine optimized for 900 to 1700 nm wavelength range:
  - Reflective illumination module with two integrated infrared lamps
  - 1.8-mm × 0.025-mm input slit. To allow for mechanical tolerances, the slit is overfilled in the orthogonal axis. Therefore, 1.69-mm × 0.025-mm of the slit is imaged onto the DMD.
  - Collimating lenses
  - 885-nm long wavepass filter
  - Reflective diffraction grating
  - Focusing lenses
  - DLP2010NIR DMD (0.2-inch WVGA, 854 × 480 orthogonal pixel, NIR optimized)
  - Collection optics
  - 1-mm single-pixel InGaAs non-cooled detector
- Electronics subsystem with the electronics consisting of four boards:
  - Microcontroller board
    - Tiva TM4C1297 microprocessor for system control operating at 120 MHz
    - 32MB SDRAM for pattern storage
    - Power management with Lithium-polymer or Lithium-ion battery charging circuits using bq24250
    - CC2564MODN Bluetooth Low Energy module for Bluetooth 4.0 connectivity
    - USB micro connector for USB connectivity
    - microSD card slot for external data storage
    - HDC1000 humidity and temperature sensor

- DLP controller board
  - DLPC150 DLP controller
  - DLPA2005 integrated power management circuit for DMD and DLP controller supplies
  - Constant current lamp driver based on OPA567 and monitored by INA213
- Detector board
  - Low-noise differential amplifier circuit
  - ADS1255 30 kSPS analog-to-digital converter (ADC) with SPI
  - TMP006 thermopile sensor for detector and ambient temperature measurement
  - 1-mm non-cooled Hamamatsu G12180-010A InGaAs photodiode
- DMD board
  - DLP2010NIR near-infrared digital micromirror device

Figure 1-2 shows the NIRscan Nano hardware block diagram.

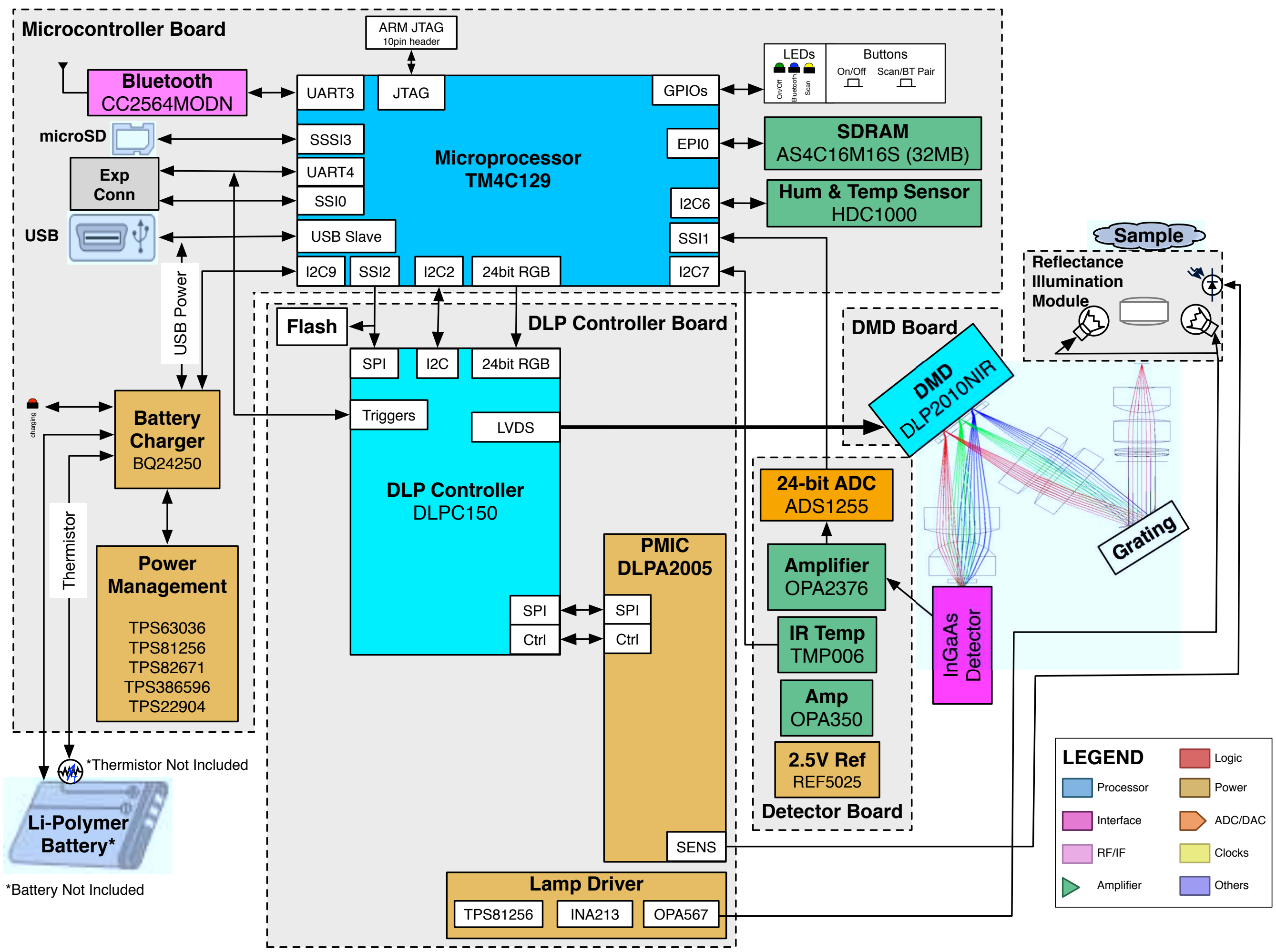


**Figure 1-2. DLP NIRscan Nano Block Diagram**

### 1.2.1 Optical Engine

The DLP NIRscan Nano EVM spectrometer optical engine is mounted on top of the electronics subsystem. The configuration is a post-dispersive architecture with a removable reflectance sample module. The reflectance module includes two lens-end broadband tungsten filament lamps. In this specific implementation, depicted in Figure 1-3, a sample is placed against the sapphire front window of the reflectance head. During a scan, the sample absorbs a specific amount of NIR light and diffusely reflects the non-absorbed light into the system. The amount of light absorbed at each wavelength is dependent on the molecular makeup of the material, and is specific to that material, a *chemical fingerprint*. The light diffusely reflected from the sample is gathered by the collection lens and focused into the optical engine through the input slit. The slit size is chosen to balance wavelength resolution with SNR of the spectrometer. This spectrometer uses a 25 µm wide by 1.8 mm tall slit. The light that passes through the slit is collimated by the first set of lenses, passes through an 885 nm long wavepass filter, and then strikes a reflective grating. This grating, in combination with the focusing lens, disperses the light into its constituent wavelengths. The focusing lenses form an image of the slit at the DLP2010NIR DMD. Different wavelengths of this slit image are spread horizontally across the DLP2010NIR DMD. The optical system images 900-nm wavelengths to one end of the DMD and 1700-nm to the other end, with all other wavelengths sequentially dispersed in between. When specific DMD columns are selected as *on*, or tilted to the +17° position, the energy reflected by the selected columns is directed through the collection optics to the single pixel InGaAs detector. All other DMD columns selected as *off*, or tilted to the –17° position, diverts the unselected wavelengths to the bottom of the light engine and away from the detector optical path so as not to interfere with the selected wavelength measurement. To allow for mechanical tolerances in slit position, grating angle and DMD position, the DLP NIRscan Nano slit image on the DMD is underfilled in the dispersion axis by 10% on each end and overfilled in the orthogonal axis. This results in about (1700 - 900 nm) / ( 854 * 0.8 pixels) = 1.17 nm per pixel on the DMD. During manufacturing a calibration is performed between wavelengths and their column position on the DMD. Since the number of DMD columns is typically not evenly divisible by the number of wavelength groups desired, the DLP NIRscan Nano maintains the column width constant during the scan, but steps through the DMD array by an amount different than the column width. This step amount depends on the desired width and number of patterns (wavelength points). Refer to DLP Spectrometer Design Consideration for implementation details.

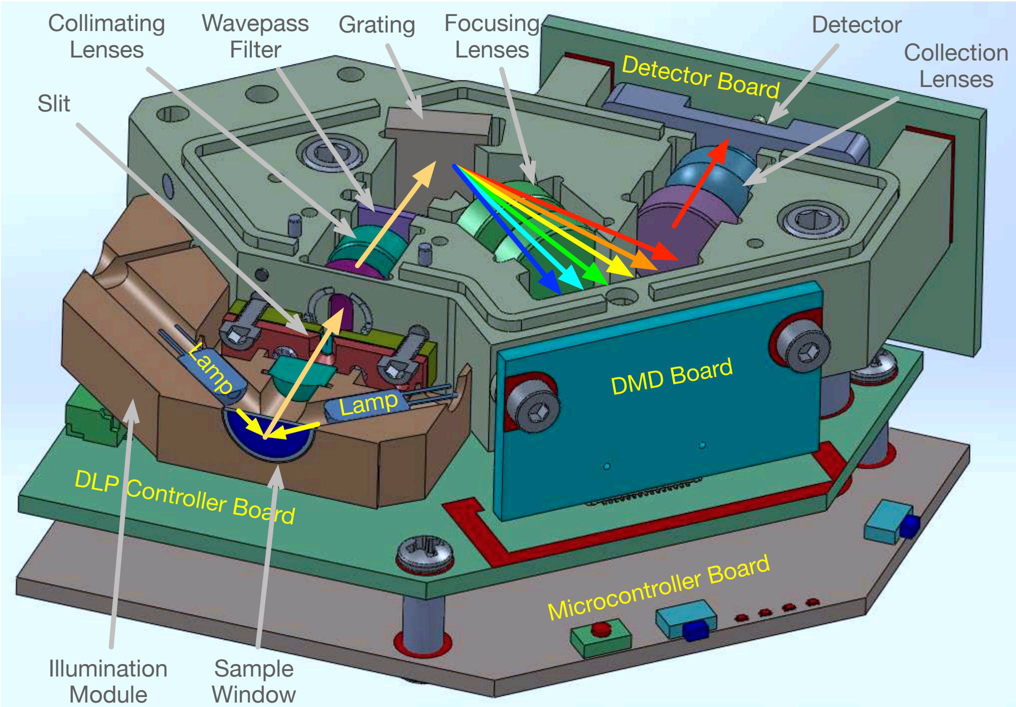


**Figure 1-3. DLP NIRscan Nano Optical Engine**

DLP NIRscan Nano reported wavelengths corresponds to the quantized DMD pixel of the column width center. Since calibration parameters are unique to each DLP NIRscan Nano, reported wavelengths between units will differ. To compare data with other DLP NIRscan Nano that have different reported wavelengths, interpolate different datasets to a common wavelength location vector before comparing them. Typically, the interpolation is accomplished with a spline-interpolation algorithm by oversampling the incoming wavelength vector. A smoothing algorithm like Svitsky-Golay is also commonly applied after the interpolation. Once all the datasets are interpolated to a common wavelength vector, comparison between multiple units is straightforward.

The DLP NIRscan Nano reflectance module operates by illuminating the sample under test at an angle so that specular reflections are not collected, while gathering and focusing diffuse reflections to the slit. The illuminating lamps are designated as *lens-end* lamps because the front end of the glass bulb is formed into a lens that directs more light from the filament to the sample test region. Figure 1-4 illustrates a top view of the illumination module. The top of the illustration depicts the cavity for the slit. The bottom of the illustration depicts the sapphire sample window. The green rectangles represent the lens-end lamps. The dark yellow cones are the lights outputted by the lamps. Each lamp produces a beam of light at 40 degree angles that intersect past the sapphire window at about 0.75 mm. There is about ± 0.25 mm tolerance to the beam intersection due to the mechanical tolerances of the chassis, the variations of lens-end from lamp to lamp, the variations of lamp shape, and the placement of the lamps. The lens-end lamps focus the light beam at about 3 mm away from the lamps and create a spot size that covers the sapphire sample window.

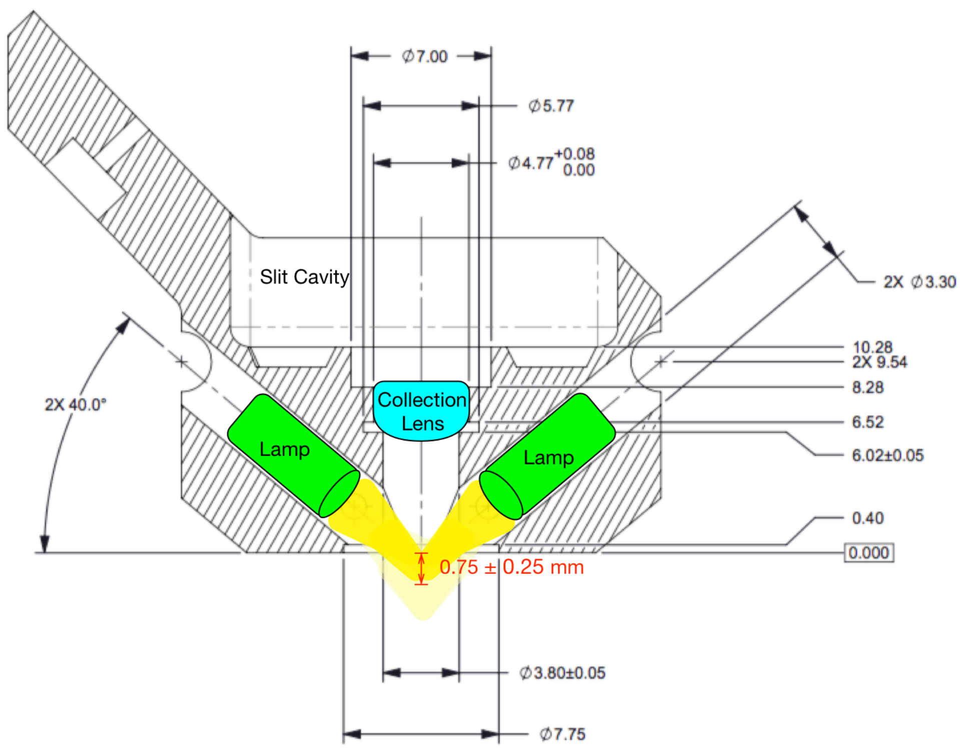


**Figure 1-4. Top View for Illumination Module**

The collection lens gathers light from a 2.5 mm diameter region at the sample window. The size of the collection region was matched to the nominal illumination spot size created by the lens-end lamps. **This requires that the sample be placed directly against the sapphire window, where the two angled light source paths intersect the collection vision cone of the lens. If the sample is shifted farther away from the window, the sample may not receive enough illumination for the system to perform an accurate scan.**

**NOTE:** For best performance, place sample against sapphire window during a scan. Note that the sample window does not have a watertight seal.

**WARNING**

**Opening or disassembling the optical engine voids the warranty on the NIRscan Nano system. Removing the cover on the optical engine allows dust and smudges to collect on the optics affecting its performance. Moreover, removing the cover might move the optics, slit, and detector out of alignment requiring factory realignment and recalibration. Removing the slit, InGaAs detector and DLP2010NIR will require the system to be realigned and recalibrated at the factory.**

The optical engine footprint drives the size of the DLP NIRscan Nano EVM. The NIRscan Nano EVM measures approximately 62-mm long, 58-mm wide, and 36-mm tall as shown in Figure 1-5.

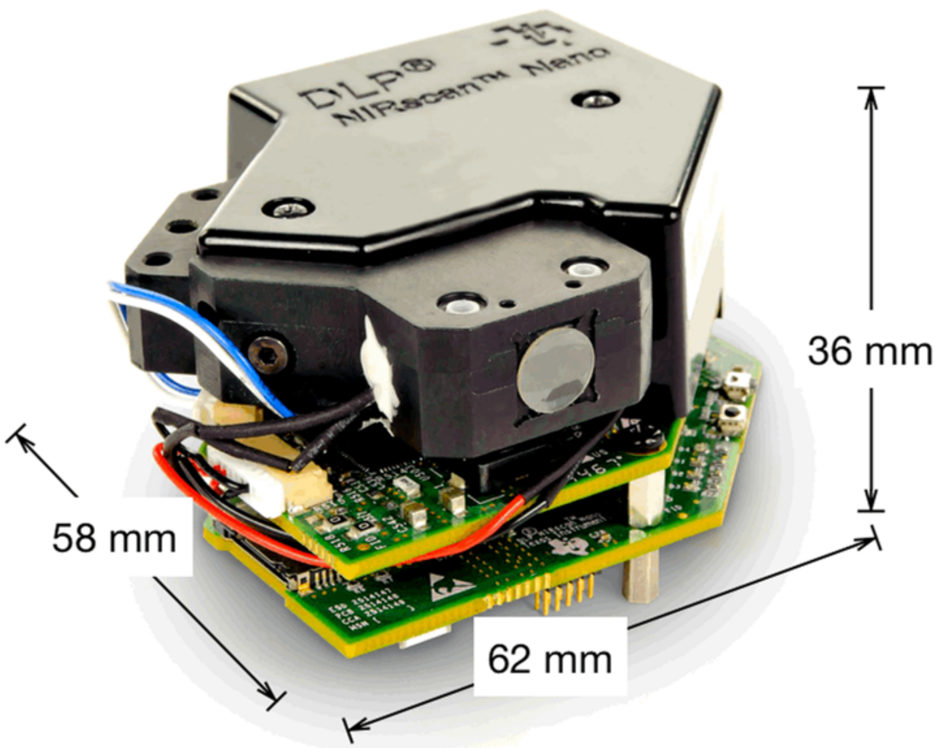


**Figure 1-5. DLP NIRscan Nano Dimensions**

Table 1-1 lists the specifications of the DLP NIRscan Nano EVM.

**Table 1-1. DLP NIRscan Nano EVM Specifications**

| PARAMETER | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|
| Supported wavelengths | 900 | | 1700 | nm |
| Optical resolution | | 10 | 12 | nm |
| Lamp power | | 1.4 | | W |
| Temperature(1) | 0 | 25 | 50 | °C |

(1) Tested temperature range. Many of the individual components of DLP NIRscan Nano exceed this temperature range. Refer to the data sheet of all individual components for their rated temperature ranges.

### 1.2.2 DLP NIRscan Nano Electronics

The DLP NIRscan Nano EVM contains the following four boards:

- **Microcontroller board:** The Microcontroller board is the largest board in the DLP NIRscan Nano EVM. This board includes the following:
  - Tiva TM4C1297 microcontroller: The Tiva processor controls the whole system. The Tiva runs the TI realtime operating system (RTOS), the Bluetopia stack, and the spectroscopy software. When it receives a scan command through USB, Bluetooth, or through pressing the scan button, the Tiva streams through its LCD interface a set of unique wavelength specific patterns to the DLPC150 for display on the DMD while synchronizing the sampling of the spectrometer's ADC. An external 32MB SDRAM allows for additional code storage and stores the pattern buffer streamed to the DLPC150.
  - External interfaces: The Microcontroller board provides two main interfaces to the outside world: USB and Bluetooth Low Energy with the CC2465MODN and on-board antenna. To leverage the DLP NIRscan EVM platform for new product development using the Tiva processor, the microcontroller board also contains a Tiva debug JTAG port, which can be used with Code Composer Studio™ emulation software and XDS100, XDS200, or XDS560 emulators. The Microcontroller board also includes and expansion connector with SPI, UART, and GPIO capability for connection to external systems.
  - bq24250 Battery charger: An optional 3.7-V Lithium-Ion or Lithium-Polymer can be added to power the system. The on-board power management circuits of the bq24250 device take power from USB and simultaneously charge the battery if its voltage is below 4.2 V at up to 1-A charge current. The bq24250 also monitors an optional thermistor for battery temperature monitoring during charge.
  - microSD card connector: The microSD card connector allows additional storage for scan data when the system is not connected to a PC nor iOS device.
  - HDC1000 humidity and temperature sensor: Measures the humidity and temperature of the system. These values are captured with each scan.
- **DLP controller board:** The DLPC150 controller board is the second largest board in the DLP NIRscan Nano EVM. This board includes the following:
  - DLPC150 controller: The DLPC150 receives the pattern data from the Tiva TM4C1297 processor over a 24-bit RGB bus. The DLPC150 decodes the pattern information and converts the information into the correct format for the DLP2010NIR DMD. The DLPC150 controls and synchronizes all the DMD signals, thereby directing each individual mirror to its desired state.
  - DLPA2005 PMIC: The DLPA2005 is a power management IC that controls all the supplies to the DLP2010NIR DMD and the DMD interface portion of the DLPC150 supplies.
  - Lamp driver circuit: To provide constant current to the near-infrared lamps, a OPA567 based power amplifier circuit regulates the current to the lamps to 280 mA at 5 V based on the voltage across a sense resistor monitored by the INA213 current shunt monitor.
- **Detector board:** The detector board includes the following:
  - OPA2376 Transimpedance low-noise amplifier: Amplifies the signal from the InGaAs detector to the ADC.
  - ADS1255 ADC: Converts the amplified signal of the InGaAs detector into a 24-bit value for Tiva processing.
  - TMP006 thermopile sensor: Measures the InGaAs detector temperature and ambient temperature of the system. These values are captured with each scan.
  - REF5025 low-noise, precision voltage reference: Provides the 2.5-V reference voltage for the transimpedance amplifier.
  - OPA350 high-speed operation amplifier: Buffers the 2.5-V reference voltage of the transimpedance amplifier.
  - 1-mm non-cooled Hamamatsu G12180-010A InGaAs photodiode
- **DMD board:** The DMD board includes the DLP near-infrared digital micromirror DLP2010NIR.

The DLP NIRscan Nano electronics contain many devices manufactured by Texas Instruments. Table 1-2 lists the main parts and their functions.

### Table 1-2. DLP NIRscan Nano Electronics

| Major Devices | Electronic Subsystem | Electronics Type | Description |
|---|---|---|---|
| TM4C1297 | Microcontroller board | Microprocessor | Cortex-M4 microprocessor operating at 120 MHz with integrated 1MB flash, 256K SRAM, and USB 2.0 interface. |
| bq24250 | | Battery charger | Single cell Lithium-Ion or Lithium-Polymer battery charger with up to 1-A charge current from USB. Battery is charged in four phases: trickle charge, precharge, constant current, and constant voltage. In all charge phases, an optional battery pack thermistor monitors the battery temperature for safe charging. |
| CC2465MODN | | Bluetooth Low Energy host controller interface module | Single chip Bluetooth 4.1 Low Energy subsystem module with on-board antenna. |
| HDC1000 | | Sensor | Low power, high accuracy temperature and humidity sensor with 14-bit resolution. |
| TPS63036 | | Power management | High-efficiency buck-boost converter in wafer chip scale package supplies 3.3 V. |
| TPS81256 | | | High-efficiency step-up converter in microSIP package supplies 5.0 V for analog circuits. |
| TPS82671 | | | High-efficiency step-down converter in microSIP package supplies 1.8 V. |
| TPS386596 | | | Quad reset supervisor |
| TPS22904 | | | Load switch supplies 1.8 V for Bluetooth circuits. |
| DLPC150 | DLP controller board | DLP Controller | DLP digital controller for advanced light control. The Tiva microprocessor in conjunction with the DLPC150 controls individual DLP2010NIR micromirrors to reflect specific wavelengths of light to a single point InGaAs detector. |
| DLPA2005 | | DLP power management | DLP power management integrated circuit that powers the DLP 1.8-V, 10-V, 18-V, and –14-V supplies. |
| TPS81256 | | Power management | High efficiency step-up converter in microSIP package that supplies the 5 V for the lamp driver |
| OPA567 | | Power amplifier | 2-A power amplifier that supplies 280-mA lamp current. |
| INA213 | | Analog monitor | Voltage output, current-shunt monitor that monitors lamp current. |
| ADS1255 | Detector board | Analog | Very-low-noise 24-bit analog-to-digital converter. Converts the analog output of the InGaAs detector into a 24-bit digital value. |
| REF5025 | | Power management | Low-noise, very-low-drift, precision voltage reference that provides the 2.5-V reference for the transimpedance amplifier. |
| OPA2376 | | Precision amplifier | Low-noise precision operational amplifier. Used as a transimpedance amplifier for the InGaAs detector. |
| OPA350 | | | High-speed operation amplifier that buffers the 2.5-V reference voltage of the transimpedance amplifier. |
| TMP006 | | Sensor | Infrared thermopile sensor that measures ambient and detector temperature |
| DLP2010NIR | DMD board | DLP Micromirror Device | DLP near-infrared digital micromirror |

### 1.2.3 Connections

Table 1-3 lists the DLP NIRscan Nano connectors with its locations shown in Figure 1-6 and Figure 1-7.

**Table 1-3. DLP NIRscan Nano Connectors**

| Board | Schematic Label | Description |
|---|---|---|
| Microcontroller board | J1 | Micro-USB connector: Provides power and PC connectivity with HID commands |
| | J2 | Detector board interface: Provides Tiva's SSI1 connection to ADS1255 and Tiva's I2C7 to TMP006 |
| | J3 | Expansion connector: Provides Tiva's UART4 or SSI0 interface to external device. UART4 is used as Tiva's console output for debugging information |
| | J4 | JTAG connector: ARM Cortex 10-pin emulation (XDS100, XDS200, or XDS560) connection |
| | J6 | Lithium-Ion or Lithium-Polymer battery connection |
| | J7 | Battery thermistor connection |
| DLP controller board | J500 | Trigger connector. This connector is covered by the top cover. Access to this connector requires to removal of the Microcontroller and DLP controller boards from the optical engine |
| | J501 | Lamp photodetector connector |
| | J503 | Lamp power connector |

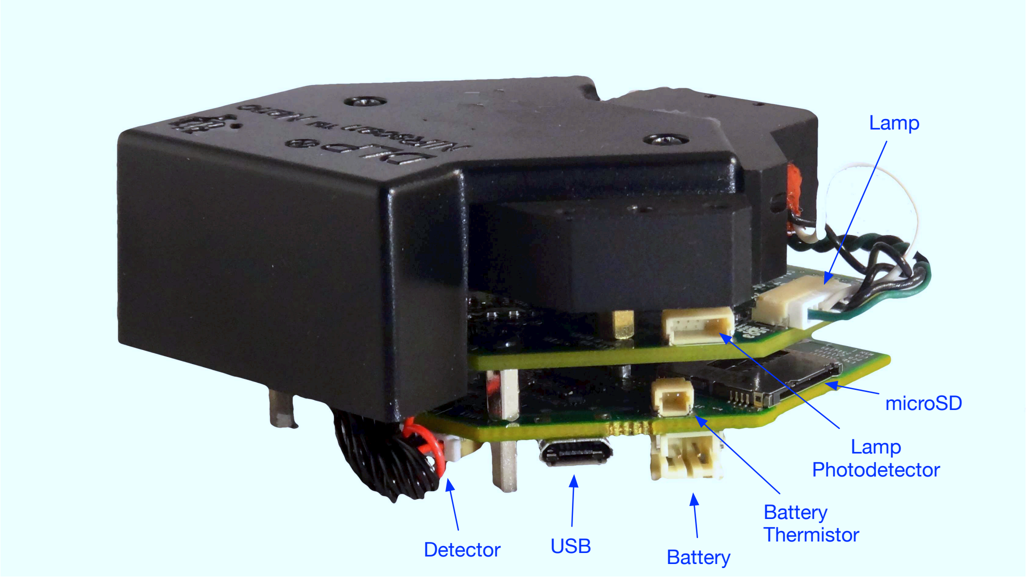


**Figure 1-6. DLP NIRscan Nano Connectors (Rear View)**

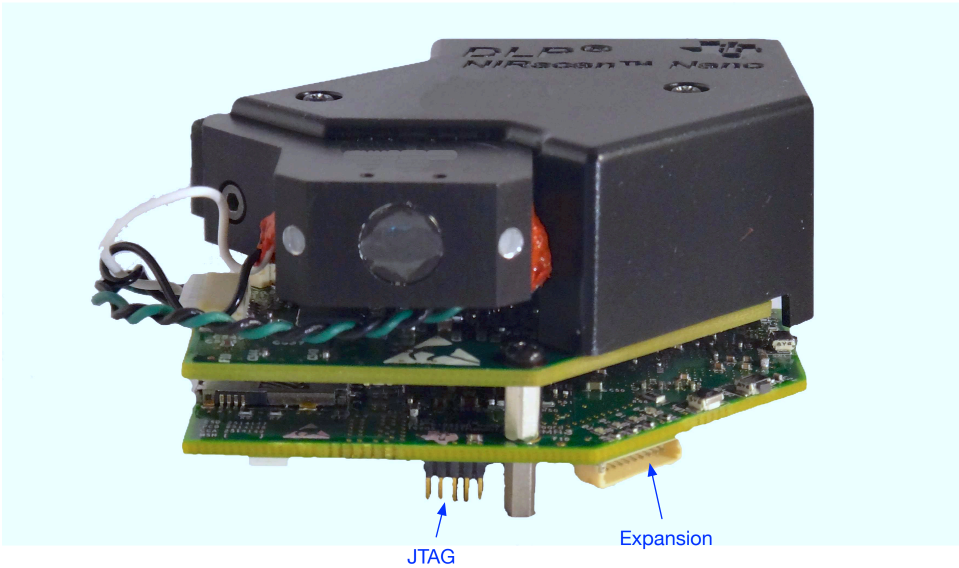


**Figure 1-7. DLP NIRscan Nano Connectors (Front View)**

#### 1.2.3.1 Buttons

The DLP NIRscan Nano EVM includes three buttons:

- Wake button:
  – When the system is in standby, pressing the Wake button will wake the system from hibernation mode.
  – Upon wake up, the green LED will pulse on and off.
- Scan/Bluetooth button:
  – When pressed and released, the system performs a scan. During a scan, the yellow LED is illuminated and the lamps will turn on for the duration of the scan.
  – When pressed, held for more than 3 seconds, and then released, the Bluetooth subsystem powers up and advertises a connection. While a Bluetooth Low Energy connection is advertised, the blue LED will turn on. When a Bluetooth Low Energy connection is active, the blue LED will pulse off and on. The pulsing may coincide with the green LED or may pulse opposite to the green LED pulses.
- Reset button:
  – Pressing the Reset button will initiate a hardware reset of the NIRscan Nano system.

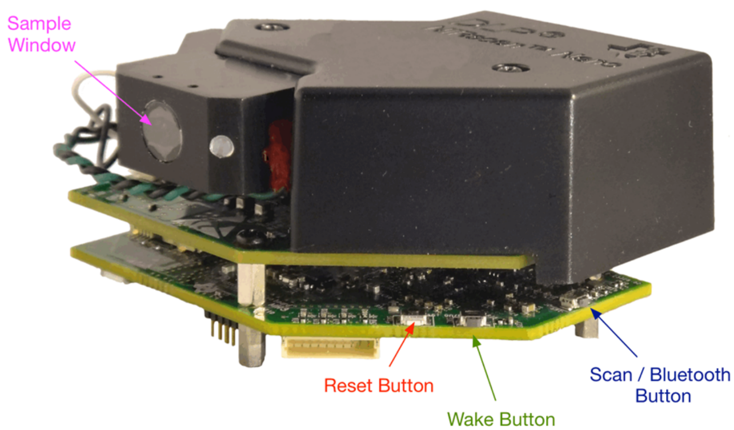


**Figure 1-8. DLP NIRscan Nano Button Locations**

#### *1.2.3.1.1 LEDs*

The DLP NIRscan Nano EVM includes four LEDs to indicate activity as shown in Table 1-4.

**Table 1-4. DLP NIRscan Nano LED Indicators**

| LED | Condition | Description |
|---|---|---|
| Green | Pulse on and off, once per second | Indicates system is powered and active |
| | Pulse on and off, twice per second | Indicates any of the following errors occurred:<br>• SD Card access error<br>• ADC data access error<br>• Tiva EEPROM access error<br>• Spectrum data calculation error<br>• Hardware error<br>• TMP006 access error<br>• HDC1000 access error<br>• Insufficient memory error<br>• Battery depleted error |
| Blue | ON | Bluetooth circuits are active and advertizing |
| | Pulse on and off, once per second | Bluetooth Low Energy connection has been established |
| | Pulse on and off, twice per second | Bluetooth communications error |
| Yellow | ON | Scan is being performed |
| | Pulse on and off, twice per second | Scan error |
| Red | ON | System is charging a battery |
| | Pulse on and off, every 256 microsecond | Battery Manager fault |

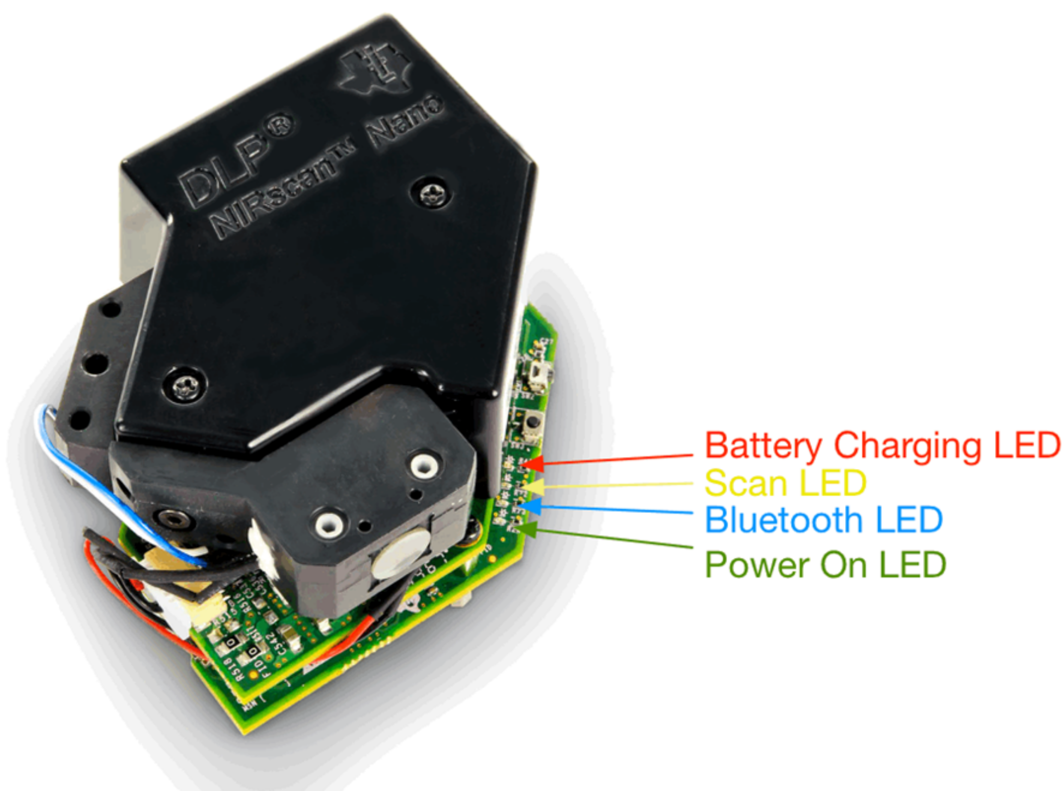


**Figure 1-9. DLP NIRscan Nano LED Locations**

# Getting Started

## 2.1 Download Software

The latest NIRscan Nano software can be downloaded at the bottom of the TIDA-00554: DLP ultra-mobile NIR Spectrometer for Portable Chemical analysis with Bluetooth Connectivity TI design page. The NIRscan Nano software is divided into four components:

- DLPC150PROM: the DLPC150 firmware.
- DLP NIRscan Nano GUI: the sources and binaries for the PC Graphical User Interface (GUI).
- DLP NIRscan Nano Reference Software: the sources and binaries for the Tiva firmware.
- DLP Spectrum Library Installer: the sources for the DLP Spectrum Library that is used by PC GUI, Tiva firmware, and iOS App to interpret data acquired by the system. This library handles the DLP routines needed to generate and transform raw scan data to a wavelength spectrum. The DLP library routines perform a scan, manage scan configurations, download patterns to the NIRscan Nano system, convert intensity information on a DMD set of pixels to wavelength intensity, reflectance, and absorbance based on factory calibrated and reference data stored on the NIRscan Nano.

## 2.2 Install Software

After downloading the four software components from the TI-DA00554 design page, install them by executing the files. Follow the on-screen instructions:

- Accept the license
- Choose installation directories. The default installation directories are:
  - DLPC150PROM: *C:\ti\DLPR150PROM_2.0.0*
  - DLP NIRscan Nano GUI Sources: *C:\ti\DLPNIRscanNanoGUI_2.1.0*
  - DLP NIRscan Nano GUI Executable: *C:\Program Files\Texas Instruments\DLPNIRscanNanoGUI_2.1.0*
  - DLP NIRscan Nano Software: *C:\ti\DLPNIRscanNanoSoftware_2.1.0*
  - DLP Spectrum Library: *C:\ti\DLPSpectrumLibrary_2.0.3*

During installation, the NIRscan Nano GUI installer will ask to set the environment library for the pre-compiled Spectrum Library version, answer yes. During installation, the NIRscan Nano GUI installer will prompt to install the Tiva Device Firmware Upgrade driver. This driver allows the Tiva to connect as Device Firmware Update (DFU) mode in order to download firmware to the Tiva.

During installation the NIRscan Nano Software will prompt to install the TI-RTOS 2.10.01.38. This package is needed if you have not installed this particular version of TI-RTOS with Code Composer installation and upgrades. The Tiva version 1.x firmware utilizes TI-RTOS version 2.10.01.38. Newer versions of TI-RTOS are not compatible with this version of Tiva firmware.

## 2.3 Upgrading Firmware

After executing the NIRscan Nano GUI software and connecting the NIRscan Nano to the PC via USB connection, a dialog window may indicate one or more of the following errors:

- Incompatible Tiva SW version detected.
- Incompatible DLPC FW version detected.
- Incompatible Spectrum library version detected.

These errors indicate the Tiva firmware, with its corresponding DLP Spectrum Library, and/or the DLPC150 firmware need to be updated on the NIRscan Nano. To upgrade:

- DLPC150 firmware - Download the DLPC150PROM and install it on the PCs. Then run the GUI, click on the Utilities Button. On the Utilities screen under DLPC150 Firmware Update (see Figure 3-11), click the corresponding Browse button to search for the DLPC150 firmware file (for example, C:\ti\DLPR150PROM_2.0.0.img). Then click on the corresponding Update button. Wait for the firmware to download.
- Tiva firmware and DLP Spectrum Library - Download the DLP NIRscan Nano Reference Software and install it on the PC. Then run the GUI, click on the Utilities Button. On the Utilities screen under Tiva Firmware Update (see Figure 3-11), click the corresponding Browse button to search for the Tiva firmware file (for example, C:\ti\DLPNIRscanNanoSoftware_2.0.1\Binaries\NIRscanNano.bin). Then, click on the corresponding Update button. Wait for the firmware to download. Once the new firmware is downloaded, the NIRscan Nano will reset and reconnect to the PC. This will update both the Tiva firmware and DLP Spectrum Library.

The following errors might display when using GUI version 2.X with a NIRscan Nano system with Tiva firmware version 1.X:

- Tiva firmware prior to version 1.0.X did not support the control of the hibernation state. Version 1.1.8 and later versions allow disabling the default hibernation after 5 minutes of inactivity.
- If you encounter the dialog error: "EEPROM cal data does not match with supported version. Wipe EEPROM data?" Do not wipe EEPROM data, click on "No." Erasing EEPROM data deletes all calibration data and the NIRscan Nano needs to be recalibrated.

## 2.4 Doxygen Documentation

The DLP NIRscan Nano GUI, Tiva, and DLP Spectrum Library Software includes doxygen documentation. To view the doxygen documentation for each software component, open the file \doc/html/index.html in the install directory in any browser. There is also a shortcut file in two places:

- Root of the source installation folder under the file Source Documentation.
- Start menu product folder (Start/Texas Instruments/productname/Source Documentation).

## 2.5 Operating Modes

The DLP NIRscan Nano supports the following modes of operation:

- USB connection: A Windows® application with a graphical user interface (GUI), running on a PC with the Windows 7 or 8 operating system, controls the system. Control includes scan initiation, parameter settings, and data download. The PC GUI displays the intensity or absorbance of the scan. The PC powers the NIRscan Nano through the USB cable.
- Bluetooth connection: An iOS app (available from KS Technologies through the Apple® App Store℠) running on an iPhone® or iPad® with iOS 7.1 or later operating system controls the system. Control includes scan initiation, parameter settings, and downloading data. The iOS app displays the intensity, reflectance, or absorbance of the scan. A USB cable or optional battery powers the NIRscan Nano.
- Standalone: The NIRscan Nano can be preconfigured through the PC GUI for a set of scan configurations. The selected default scan configuration is invoked through the Scan button and data is stored on the on-board microSD Card. The stored scan data can be later downloaded to a host PC through USB or Bluetooth connection. For more information on how data is stored on the microSD Card, refer to Section 3.1.7.

### *2.5.1 USB Connection*

When a USB cable is inserted into the DLP NIRscan Nano J1 micro-USB connector (see Figure 1-6), the system powers up from the PC's USB VBUS 5-V supply, and the power-on LED pulses to indicate the system is operational and ready for a command. The PC GUI will show as connected after the DLP NIRscan Nano enumerates through USB.

#### 2.5.1.1 NIRscan Nano GUI

The DLP NIRscan Nano software includes a QT-based PC GUI called NIRscanNanoGUI.exe. This GUI requires the following dynamic link libraries (DLLs) to reside in the same directory as the executable file. The installer copies these required DLLs in the same folder as the application. Please do not remove these DLLs from the application folder:

- hidapi.dll — USB human interface device (HID) class communication driver
- libEGL.dll — Almost Native Graphics Layer Engine (ANGLE) library. Default graphics library for QT 5.X
- libgcc_sdw2-1.dll — GCC library
- libGLESV2.dll — Almost Native Graphics Layer Engine (ANGLE) library. Default graphics library for QT 5.X
- libstdc++6.dll — Standard C++ library
- libwinpthread-1.dll — Pthreads for Windows library
- Qt5Core.dll — Qt Core class library
- Qt5Gui.dll — Qt Graphical User Interface class library
- Qt5Svg.dll — Qt Scalable vector graphics class library
- Qt5Widgets.dll — Qt Widgets class library
- msvcr100.dll — Microsoft Visual C++ Studio 10.0 library
- platforms\qwindows.dll — Platform plugin for Windows applications
- lmdfu.dll — Tiva USB device firmware upgrade
- lmusb.dll — Tiva USB driver

The Qt windeployqt executable will list all the DLLs necessary by a Qt application.

### 2.5.2 Bluetooth Connection

To connect to the DLP NIRscan Nano, the Bluetooth circuits must first be powered. The following steps activate the Bluetooth circuits:

1. Press the Scan/Bluetooth button, hold the button pressed for more than 3 seconds, and then release the button. Once released, the Bluetooth is powered and enabled.
2. After the Bluetooth circuits are powered and active, the blue LED turns on and the DLP NIRscan Nano advertises its presence through Bluetooth.
3. Run the iOS App and click the Scan button at the top-right of the screen. This will establish a connection with the DLP NIRscan Nano. The Bluetooth icon on the top-right of the screen will flash.
4. After the DLP NIRscan Nano establishes connection, the blue LED will pulse to indicate that the connection was successful.

# Operating the DLP NIRscan Nano EVM

## 3.1 NIRscan Nano GUI

Upon launching the NIRscanNanoGUI, the application checks for the DLP NIRscan Nano EVM enumerating through USB and displays the information screen shown in Figure 3-1. The GUI is divided into three sections:

- The top section displays the connected state of the DLP NIRscan Nano EVM on the top-right side. It has three buttons and one indicator:
  - Information Button: Changes the middle portion of the GUI to display version information, and links to online resources.
  - Scan Button: Changes the middle portion of the GUI to display spectrum plots and controls for scan configurations and parameters.
  - Utilities Button: Changes the middle portion of the GUI to display sensor information, synchronizes data and time with PC, provides firmware upgrades, and batch processor to convert scan data (*.dat) files from SD Card into comma separated values files (*.csv).
  - Connected Status Indicator: Once a DLP NIRscan Nano enumerates, the icon in the connected status changes from a gray indicator light with a "Not Connected" message to a green indicator light with a "Connected" message. Disconnecting the DLP NIRscan Nano, powering down the device, or resetting the DLP NIRscan Nano will toggle the state of this indicator light.
- The middle section displays information related to the three main operational modes: Information, Scan, and Utilities.
- The lower sections has two rows:
  - Device Status: indicator lights up to show the current status of the system
    - Tiva Status indicator lights up when the Tiva is active.
    - Scan status indicator lights up when a scan is in progress.
    - SD Card status indicator lights up when an SD Card is inserted and detected when the system is powered up.
    - Card I/O status indicator lights up when accessing the SD Card.
    - BT Ready status indicator lights up when DLP NIRscan Nano Bluetooth is powered up and advertising.
    - BT Connected status indicator lights up when the DLP NIRscan Nano is connected to another device through Bluetooth Low Energy.
  - Error Status: indicator lights up when an error is reported. These indicator light stays on until cleared by pressing the "Error Status Press to clear button."
    - Scan error indicator lights up when the system reports a scan error. Clicking on the scan error light displays a dialog window with more information on the specific scan error.
    - ADC error indicator lights up when the system reports an error reading ADC data. Clicking on the ADC error light displays a dialog window with more information on the specific ADC error.
    - SD Card indicator lights up when the system reports an error reading SD Card data. Clicking on the SD Card error light displays a dialog window with more information on the specific SD Card error.
    - EEPROM indicator lights up when the system reports an error accessing the Tiva EEPROM. Clicking on the EEPROM error light displays a dialog window with more information on the specific EEPROM error.
    - BLE indicator lights up when the system reports an error with Bluetooth communications.

Clicking on the BLE error light displays a dialog window with more information on the specific BLE error.

- Spectrum indicator lights up when the system reports an error while calculating spectrum data from the system. Clicking on the spectrum error light displays a dialog window with more information on the specific spectrum calculation error.
- Hardware indicator lights up when the system reports a hardware error. Clicking on the hardware error light displays a dialog window with more information on the specific hardware error.
- Temperature Sensor indicator lights up when the system reports an error communicating with the TMP006 device. Clicking on the temperature error light displays a dialog window with more information on the specific TMP006 error.
- Humidity Sensor indicator lights up when the system reports an error communicating with the HDC1000 device. Clicking on the humidity error light displays a dialog window with more information on the specific HDC1000 error.
- Battery indicator lights up when the system reports an error with the battery. Clicking on the battery error light displays a dialog window with more information on the specific BQ24250 error.
- Memory indicator lights up when the system reports an error with SDRAM memory. Clicking on the memory error light displays a dialog window with more information on the specific SDRAM error.

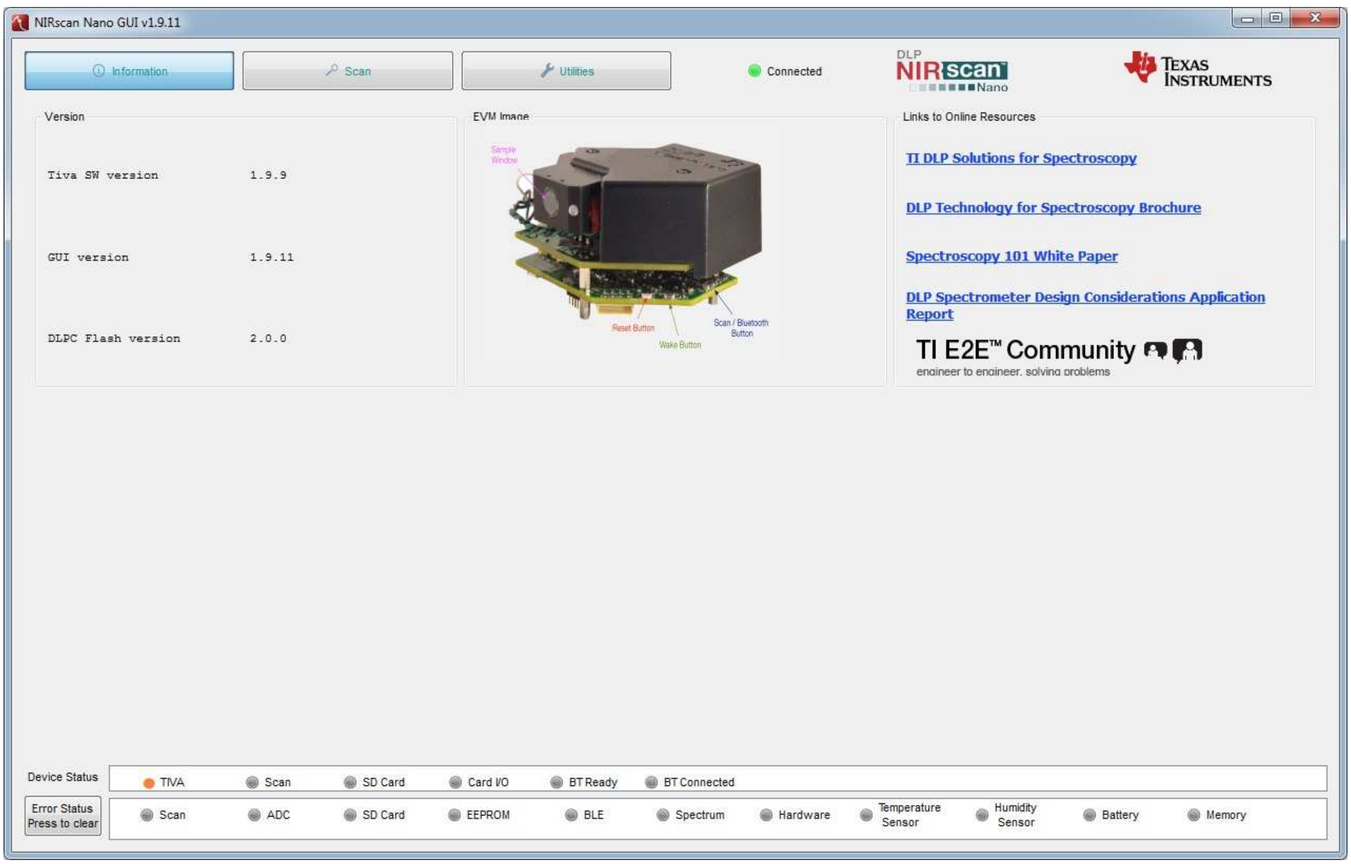


**Figure 3-1. DLP NIRscanNano GUI Information Screen**

The information screen displays:

- Version information, including the version number of the Tiva and DLPC150 (DLPC Flash) firmware, as well as the GUI software version number. If there is a mismatch between the version of the GUI, Tiva, DLPC150, and DLP Spectrum Library, the GUI will inform the user to please update the appropriate firmware.
- EVM image, which displays the locations of the buttons and their functionality.
- Links to online resources, including Texas Instruments DLP brochures, white papers, and application notes on spectroscopy with DLP technology. For support, users can search TI's E2E Community. The TI E2E icon includes a link to direct users to the Texas Instruments DLP E2E forums.

### *3.1.1 Scanning a Sample*

The Scan button at the top of the NIRscan Nano GUI displays the spectrum plots and the controls for scan configurations and parameters, as shown in Figure 3-2.


**Figure 3-2. DLP NIRscanNano GUI Scan Screen**

A scan configuration must be created to scan a sample. (See Figure 3-3 for the Scan Configuration dialog screen.) The DLP NIRscan Nano comes pre-loaded with two scan configurations from the factory: "Column 1" and "Hadamard 1" scan configurations. A **scan configuration** specifies the following parameters of a scan:

- **Name**: This labels the scan configuration. This name is used to select the previously define scan configurations.
- **Number of scans to average**: This is the repeated back-to-back scans that are averaged together. Averaging each wavelenght point across multiple scans results in lower noise while increasing the total scan time.

- **No. of Sections**: This is the number of sections of a scan. A scan can be broken up into 1-5 sections. Each section can have individual set of the following parameters:
  – **Method**: This controls the method of the scan. Two options are offered: Column or Hadamard. Column scan selects one wavelength at a time. Hadamard scan creates a set with several wavelengths multiplexed at a time and then decodes the individual wavelengths. The Hadamard scan collects much more light and offers greater SNR than column scan.
  – **Wavelength range:** Start and End wavelengths (in nm) or spectral range of interest for the scan. The minimum wavelength is 900 nm and the maximum wavelength is 1700 nm.
  – **Width in nm**: This number selects the width of the groups of pixels in the generated Column or Hadamard patterns. The options displayed correspond to the width of the dispersed spectrum in nm across the quantized pixel width.
  – **Digital Resolution**: This number defines how many wavelength points are captured across the defined spectral range. This corresponds to the number of patterns displayed on the DMD during the scan. Increasing the digital resolution leads to an oversampling of the spectrum. In general, set this resolution to oversample at least twice of the desired full width half maximum (FWHM) desired. For example, for a 15 nm FWHM between 900 and 1700 nm, use 2 * (1700 - 900) / 15 ≥ 107 wavelength points. Depending on the previous setting, the GUI computes the maximum number of wavelength points and indicates them as the "Max patterns used" in the bottom left corner of the scan configuration window. The total maximum number of patterns for all sections of a scan is 624. Each wavelength point corresponds to a pattern that is displayed on the DMD.
  – **Exposure Time**: For scan configurations with one sections, the exposure time is set to 0.635 ms. For scan configurations with more than one sections, the exposure time can be individually set for each section in the range of 0.635 to 60.960 ms.

Typical scan configuration parameters for four type of scans that resolve wavelength content in 20-nm, 15-nm, 10-nm, and 8-nm, are shown in Table 3-1.

**Table 3-1. Typical Scan Configuration Parameters**

| Scan Configuration Parameters | 20-nm Content | 15-nm Content | 10-nm Content | 8-nm Content | |
|---|---|---|---|---|---|
| Wavelength range | 900 to 1700 nm | 900 to 1700 nm | 900 to 1700 nm | 900 to 1700 nm | 900 to 1700 nm |
| Width in nm | 20 | 15 | 10 | 8 | 8 |
| Digital Resolution | 80 | 108 | 160 | 225 | 248 |
| Oversampling | 2 | 2 | 2 | 2.25 | 2.48 |
| Number of scans to average | 18 | 12 | 8 to 9 | 6 | 5 |

The following steps create a scan configuration:

1. Click the "New/Edit/Export/Import" button in the Scan control box to invoke the Scan Configuration dialog box.
2. The Scan Configuration dialog box shown in Figure 3-3 has three sections:
   - The top-left section displays previous scan configurations saved to the PC.
   - The top-right section displays the scan configurations saved on the DLP NIRscan Nano EVM.
   - The bottom section displays the scan configuration parameters of the selected PC or DLP NIRscan Nano EVM stored scan configuration.
3. Click the New button in the top-left or top-right section of the Scan Configuration dialog, depending if the scan configuration will be stored on the PC (top-left) or the NIRscan Nano (top-right). Then, type the name of the scan configuration in the Name box.
4. Enter the number of scans to average for corresponding amount of back-to-back scans averaged together. Increasing this amount results in longer scan with the noise averaged out through several back-to-back scans.
5. Enter the number of sections. Use one section for a scan with the same width and digital resolution the wavelength range. Enter more than one section to create a fast scan with less resolution on wavelengths with little information and higher resolution on wavelengths with areas of interest. Sections can overlapp in start and end wavelengths.

6. For each section enter:
   (a) Desired spectral range between 900 and 1700 nm.
   (b) Width in nm that corresponds to the smallest wavelength content that you want to resolve.
   (c) Desired digital resolution. The digital resolution is the number of wavelength points captured across the spectral range.
   (d) Desired exposure time. The exposure times can differ for each section.
7. Save to the NIRscan Nano or to the PC (locally).
8. Close the Scan Configuration dialog by clicking OK.

The Copy and Move buttons allow copying or moving scan configurations stored on the PC to the DLP NIRscan Nano or from the DLP NIRscan Nano to the PC.

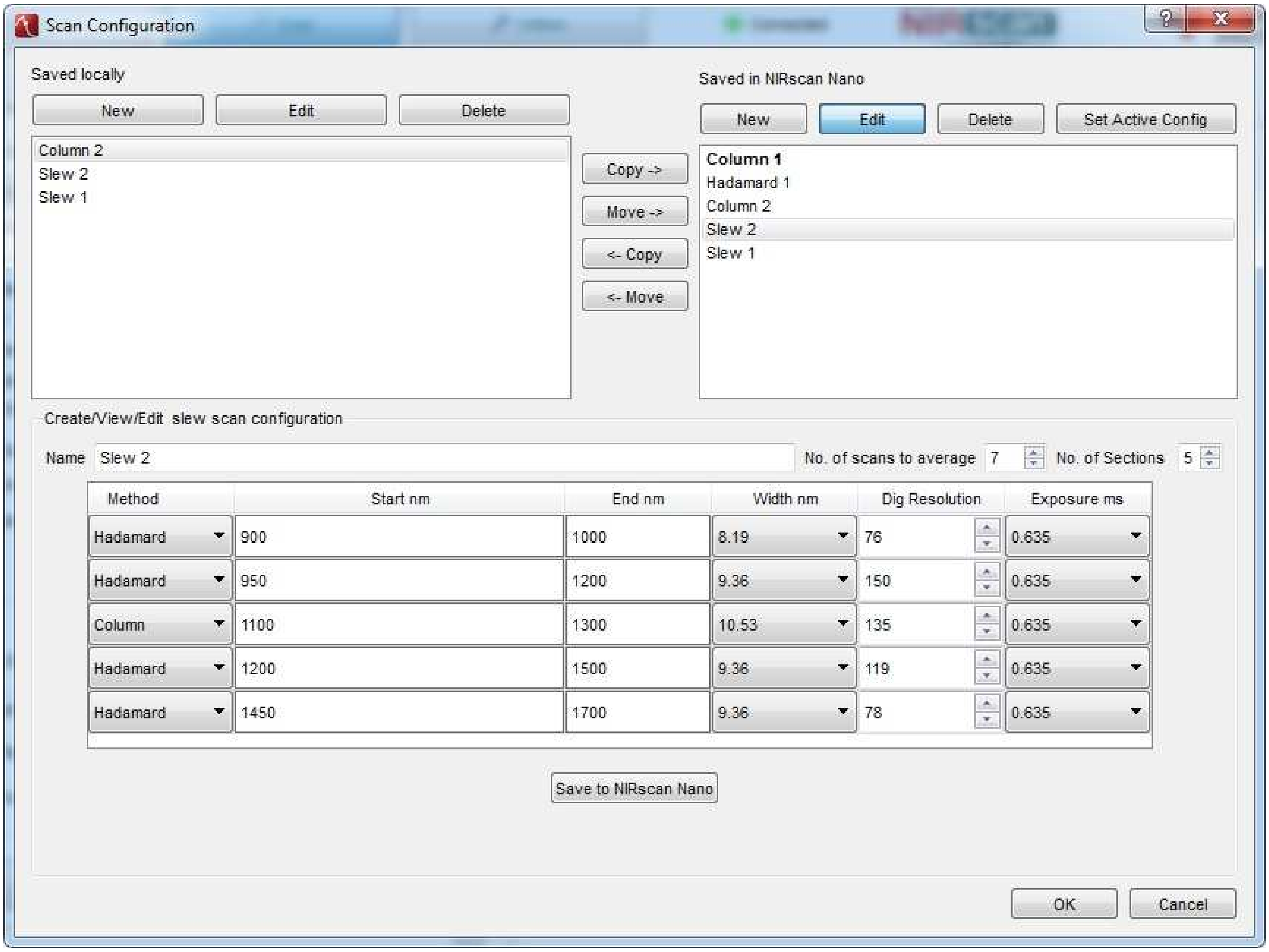


**Figure 3-3. DLP NIRscanNano GUI Scan Configuration Dialog**

After a scan configuration is defined, it appears under the Select Scan drop-down menu, as shown in Figure 3-4. The selected scan configuration will be used for the next scan. When displaying Absorbance or Reflectance, a reference is needed. The "Scan Reference Select" allows the user to choose the reference for the absorbance or reflectance graph. The reference options include:

- **Factory:** Interpolates the reference stored on Tiva EEPROM at the factory to match the current scan configuration parameters.
- **Previous:** Choose the reference from the previous use of the "New" option.
- **New:** Place a highly reflective material like a metal coated with Spectralon® on the sample window and perform a scan. This new scan is stored on the PC and can then be selected with the "Previous"

reference radio button.

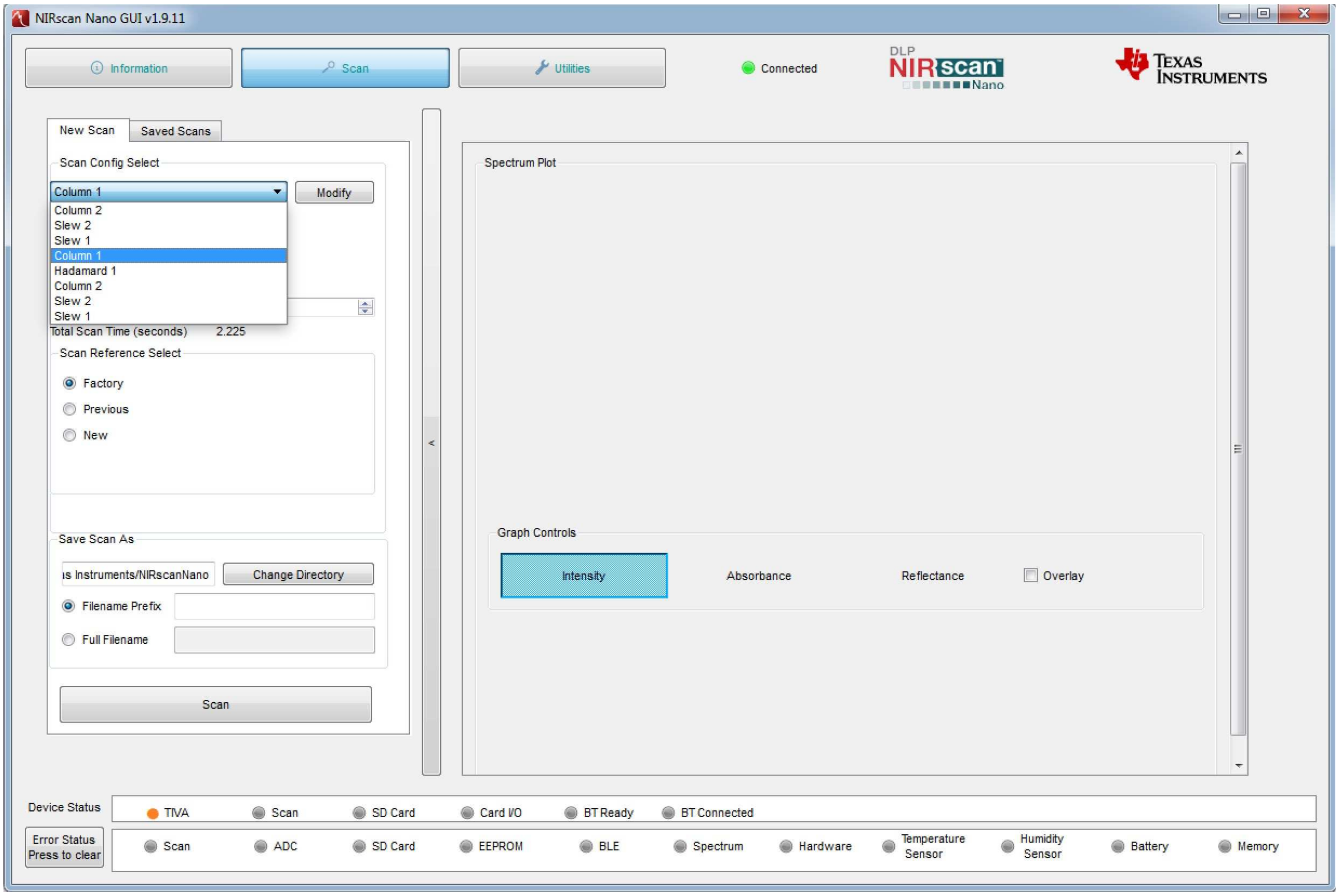


**Figure 3-4. DLP NIRscan Nano GUI Scan Select Menu**

### *3.1.2 Saving Scan Data*

The location where the scan data is saved is under the "Save Scan As" section of Figure 3-4. Clicking on the "Change Directory" button launches a directory browser window. This window allows the user to select a different directory folder to store future scans in the PC. A scan is typically labeled with the Scan configuration name followed by a unique number, and then followed by the date and time. The "Filename Prefix" option, will append the characters type into this input box before the scan configuration name. The "Full Filename" option will replace this default name (scanconfiguration_number_datetime) with the characters typed in this input box.

The following file format options are available to save scan data into a file on the PC:

- Binary file: A binary file that is interpreted by the DLP Spectrum C Library with filename scanconfiguration_number_datetime.dat.
- CSV file: A comma separated value file that can be imported into a spreadsheet application. The CSV file has two options:
  – One file with absorbance, intensity, and reflectance values with filename scanconfiguration_number_datetime.csv
  – Three separate files:
    - Absorbance file with filename scanconfiguration_number_datetime_a.csv
    - Intensity file with filename scanconfiguration_number_datetime_i.csv
    - Reflectance file with filename scanconfiguration_number_datetime_r.csv
- JCAMP file: A standard file format from the Joint Committee on Atomic and Molecular Physical data to

store spectral data with filename scanconfiguration_number_datetime.jdx

These options are available by clicking the "File Settings" button in the "Save Scan As" section shown in Figure 3-5.

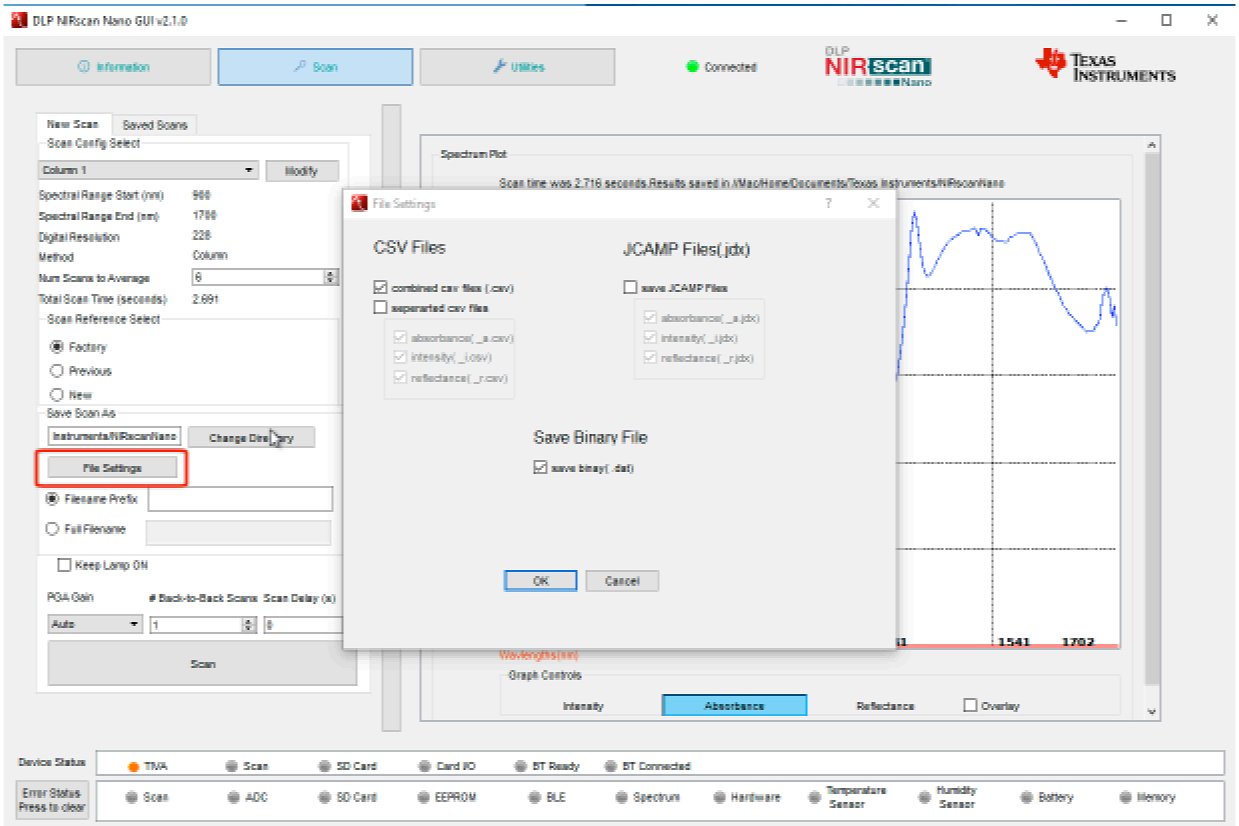


**Figure 3-5. DLP NIRscan Nano GUI Saving Scan Data**

### 3.1.3 Keep Lamp On

Version 2.1 of the GUI introduces "Keep Lamp ON" checkbox under the "Save Scan As" section, as shown in Figure 3-6. When the "Keep Lamp On" is checked, the Lamps are enabled and stay on until this checkbox is unchecked. This allows the user to avoid any lamp stability issues and reduce lamp wear caused by turning the lamps on and off, as well as the additional time needed to wait for the lamps to stabilize before executing a scan.

When Keep Lamp On is selected, the ADC PGA gain setting is *not* automatically computed before each scan and the PGA gain defaults to 64. The user can change the default PGA setting as explained in Section 3.1.4. Note that the automatic PGA gain setting feature is not supported while “Keep Lamp On” is active.

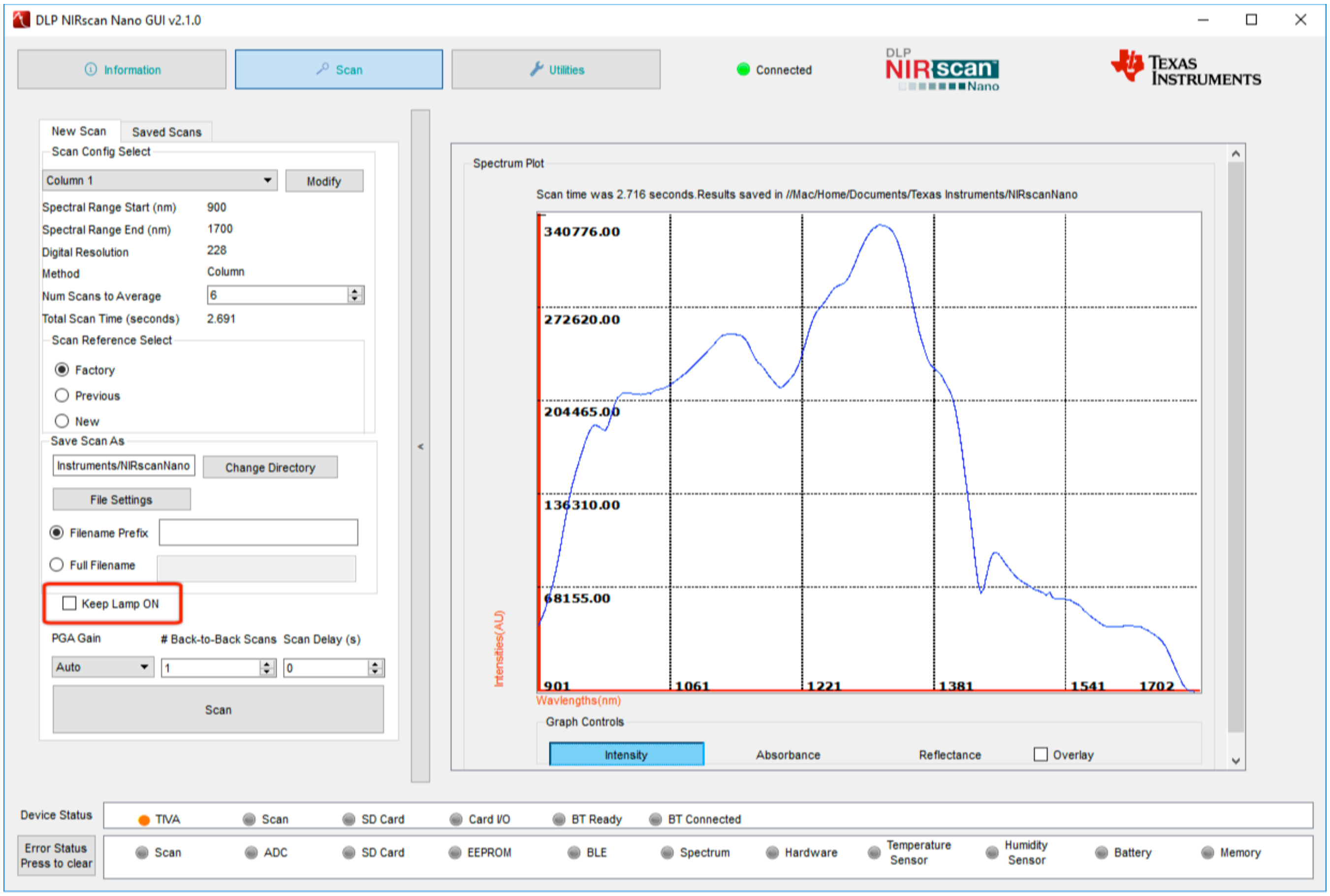


**Figure 3-6. DLP NIRscan Nano Keep Lamp On Checkbox**

## 3.1.4 Fixed PGA Gain

Version 2.1 of the GUI introduces a "PGA Gain" section under the "Keep Lamp ON" checkbox, as shown in Figure K-1. When set to Auto, each scan performs a quick scan to the set the ADS1255 programmable gain amplifier (PGA) to the highest possible amount that does not cause overflow. When set to a specific number, the PGA is set to that value for all scans. This avoids the quick scan. Care must be taken to not set the PGA to a value that causes overflow. An overflow would show a flat scan with the highest possible intensity value. This is useful when scanning a material with an intensity that lies at the border of two PGA settings where some scans might select a PGA setting while other scans select the next higher PGA setting.

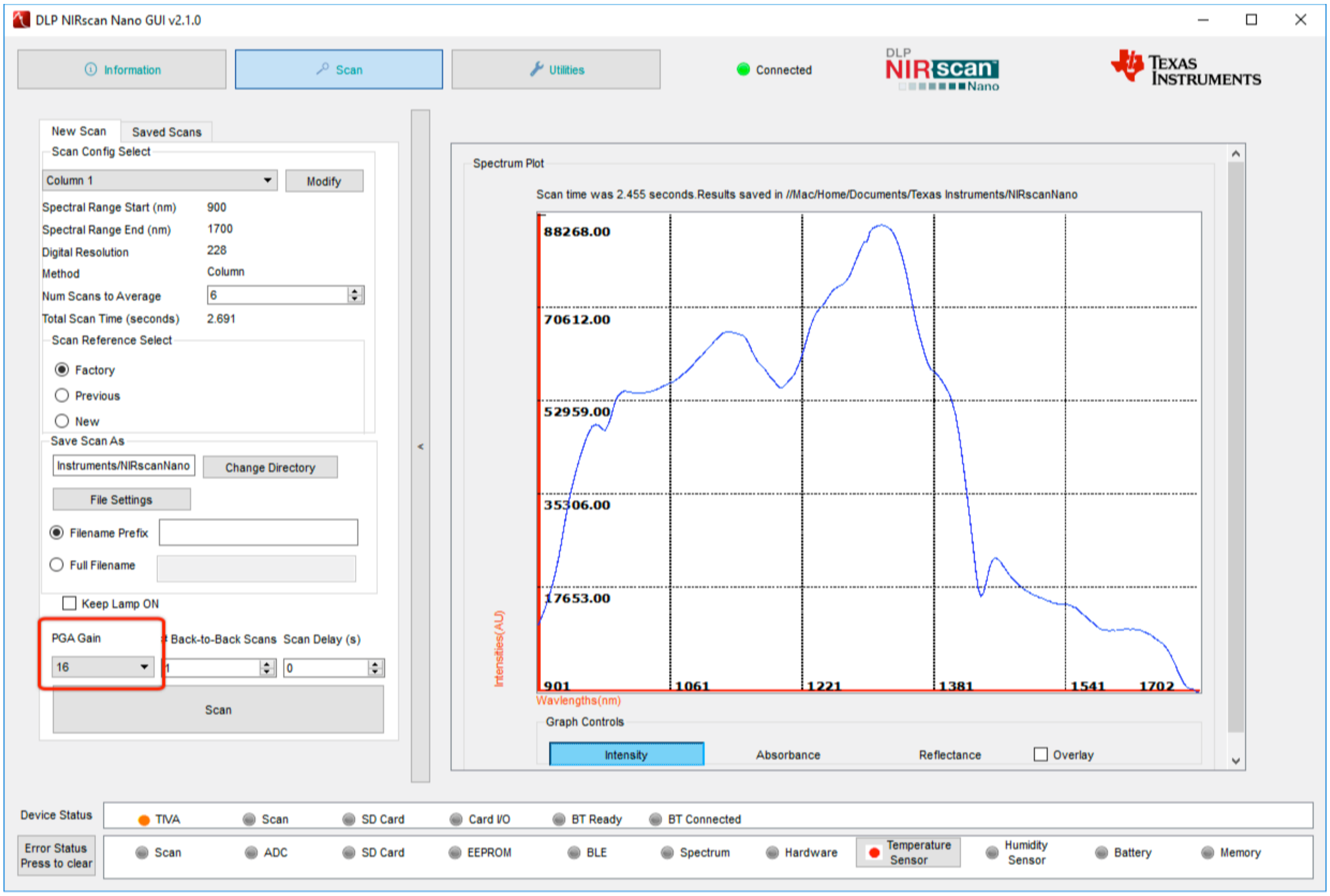


**Figure 3-7. DLP NIRscan Nano Fixed PGA Gain**

### 3.1.5 Back-to-Back Scans

Version 2.1 of the GUI introduces the "Back-to-Back Scans" control, as shown in Figure 3-8. The field under "Back-to-Back Scans" indicates the number of back-to-back scans to be completed once the Scan button is pressed. The field under "Scan Delay (s)" indicates the amount of seconds to wait between consecutive scans. IF no delay is desired, enter 0 for the scan delay. If the "Keep Lamp ON" is not checked, the lamps will turn on and off with each scan. If the "Keep Lamp ON" is checked, the lamps will turn on as soon as the checkbox is checked and remain on until turned off. The "Back-to-Back" field will count down after a scan is completed.

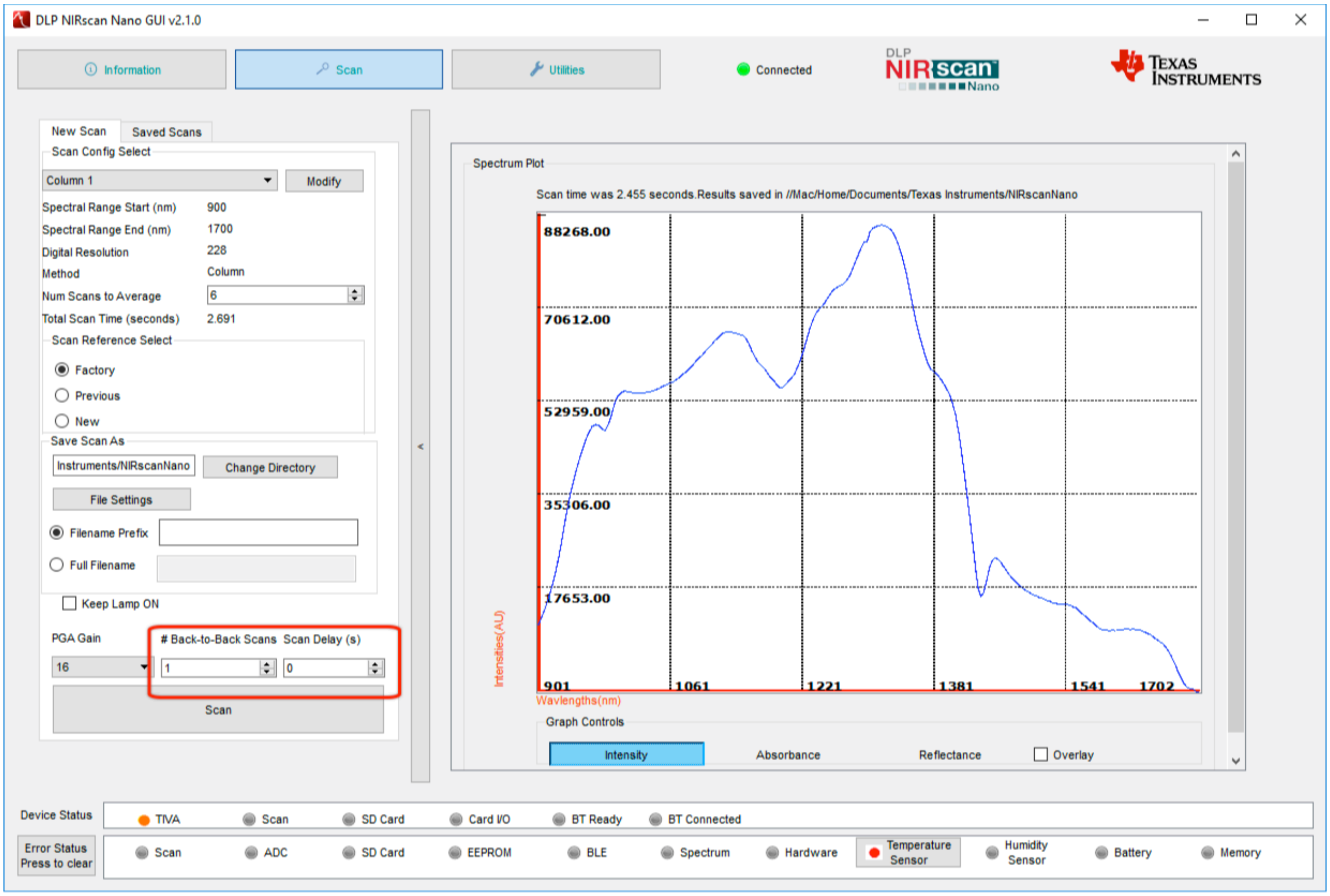

Figure 3-8. DLP NIRscan Nano Fixed PGA Gain

### 3.1.6 Displaying Previous Scans

To display previous scans, click the "Scans" button then the "Save Scans" tab on the left side of the window. A subwindow displays the previous scans stored in the PC. The files are stored with the name of the scan configuration appended with the date and time of the scan. Scans with multiple sections are denoted with "Method Slew." To plot a file as shown in Figure 3-9, select one of the files and click the Display Spectrum button. To hide this subwindow, click on the separating bar with "<." To show this subwindow, click on the ">."

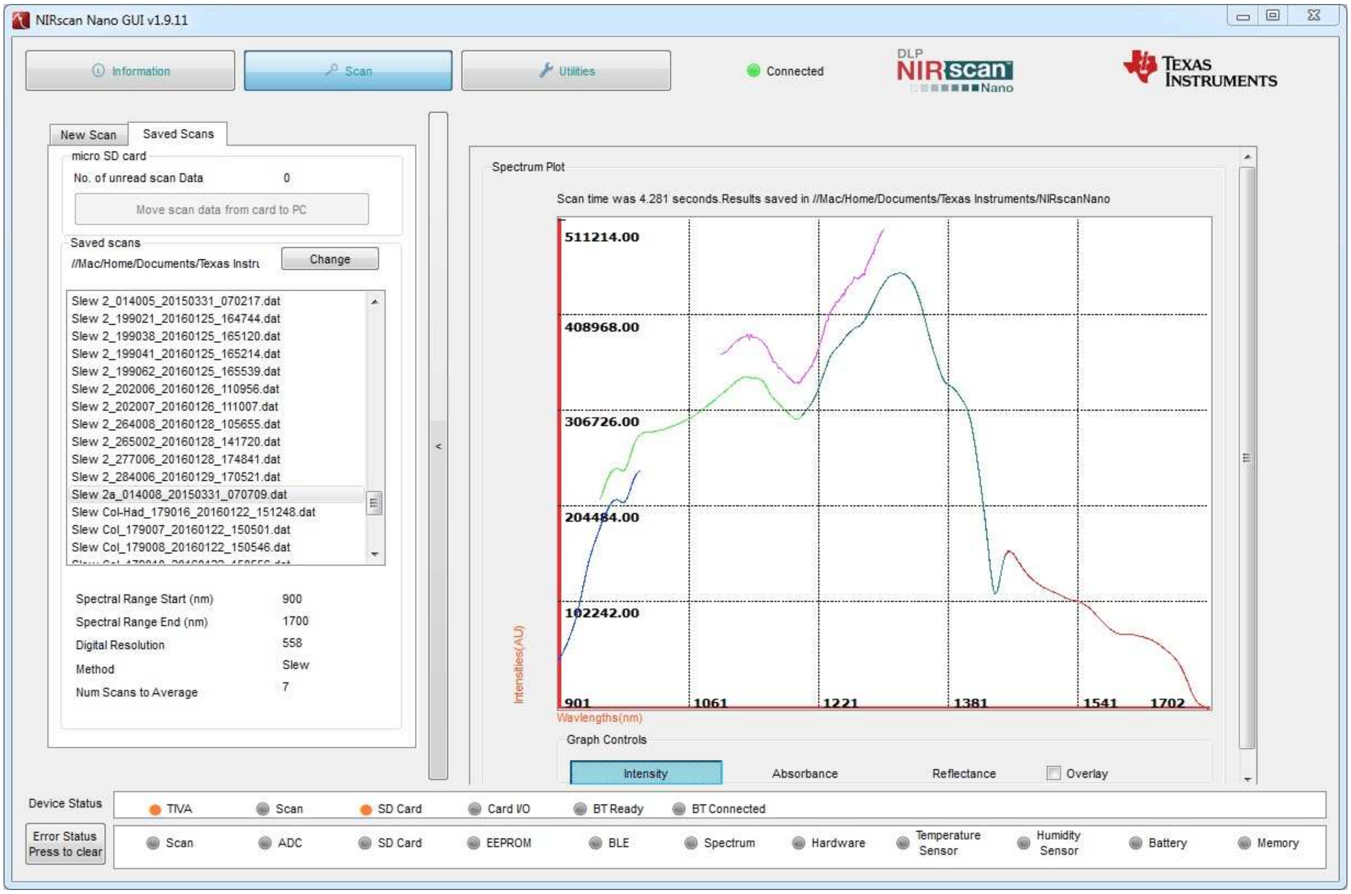


**Figure 3-9. Displaying Previous Scans**

### 3.1.7 Transferring Scans Stored on a MicroSD Card

Whenever scans are taken by pressing the Scan button on the microprocessor board, the scans are stored on the microSD Card, if one is present. To transfer the stored scan on microSD:

- With the DLP NIRscan Nano connected through USB:
  - On the Scan Tab, select the Save Scan tab.
  - Under the microSD card, "No. of unread scan Data" indicates the number of scans stored on the microSD Card. If an "er" is reported instead of a number, there was an error reading the SD Card or no SD Card is present.
  - Click the "Move scan data from card to PC" button. This will read all the scans stored on the microSD card, and will import the scans stored on the SD Card.
- With just the microSD Card inserted on the computer:
  - Batch convert all the files in the microSD Card by selecting ".dat to .csv conversion" on the Utilities tab and clicking on "Browse."
  - Search for the mounted microSD Card drive and the folder with the serial device name of the DLP NIRscan Nano.
  - Click on "Convert to CSV" button to start the batch conversion process from binary to CSV format. See Figure 3-11.

SD Card data is stored as binary data with filename extension ".dat". The scan data is stored under the root folder under a folder named according to the serial device name of the DLP NIRscan Nano. To batch convert all the binary data in a microSD Card to comma separated values ("*.csv"), select ".dat to .csv conversion" on the Utilities tab. See Figure 3-11.

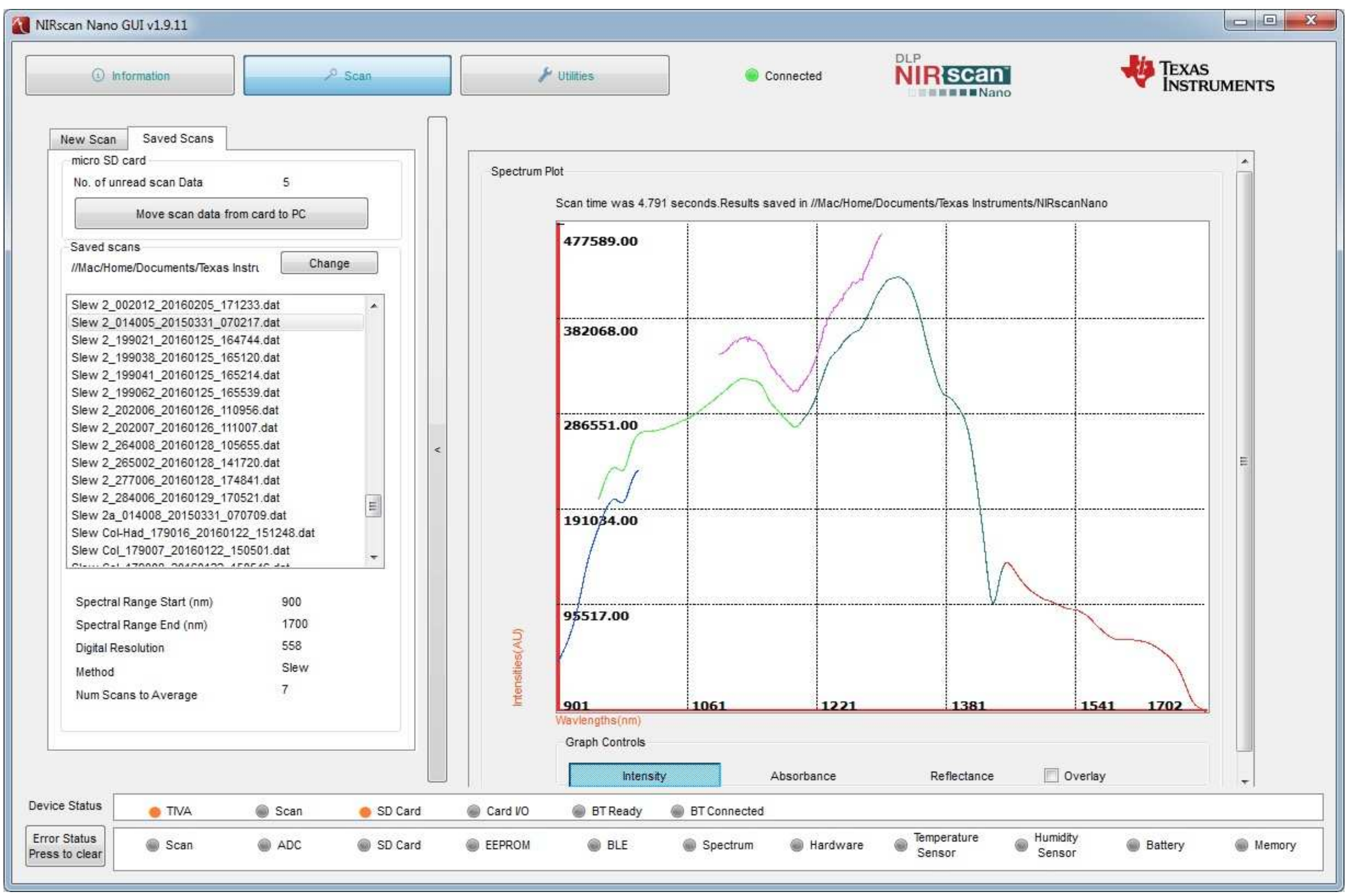


**Figure 3-10. Data Imported From microSD Card**

### 3.1.8 Utilities

The DLP NIRscan Nano GUI includes a Utilities screen, as shown in Figure 3-11, that displays:

- Sensor data:
  - Battery voltage, if a Lithium-Ion or Lithium polymer single cell battery is connected to the J6 connector.
  - Ambient temperature read by the TMP006 in the Detector Board.
  - Detector temperature read by the TMP006 in the Detector Board.
  - Ambient humidity read by the HDC1000 in the Microcontroller Board.
  - HDC temperature read by the HDC1000 in the Microcontroller Board.
  - Tiva internal temperature read by the Tiva internal sensor in the microcontroller board.
  - Phototodetector value of the lamp output.
- Tiva's hibernation module date and time. Pressing the "Sync Data/Time" button will read the PC's date and time and store it in the Tiva Real Time Clock module's date and time registers. When the DLP NIRscan Nano is connected to a PC with the GUI, the date and time of the Tiva Real Time Clock is updated and synchronized with the PC date and time automatically.
- DLPC150 Firmware update tool.
  - To update the DLPC150 firmware, click the Browse button to search for the DLPC150 firmware file (for example, *C:\ti\DLPR150PROM_1.1.0.img*).
  - Then, click the Update DLPC150 Firmware button. The firmware will be flashed to the board while the progress bar indicates the update process.
- Tiva Firmware update tool.
  - To update the Tiva firmware, click the Browse button to search for the Tiva firmware file (for example, *C:\ti\DLPNIRscanNanoSoftware_1.1.8\Binaries\NIRscanNano.bin*).
  - Then, click the Update TIVA Firmware button. The firmware will be flashed on the Tiva internal Flash while the progress bar indicates the update process. The "tiva Flash is empty/erased" check box needs to be enabled if no firmware was previously stored on the system or if the Tiva Flash was erased. The checkbox next to "Don't verify signature" removes the GUI check to determine if the binary file being downloaded to Tiva is a valid NIRscan Nano binary file. Loading a binary file not intended for the NIRscan Nano can lead to the inability to boot the system, so it is recommended to leave this option checked.
- To reset the DLP NIRscan Nano system, press the Reset button under the Reset NIRscan Nano section.
- To place the DLP NIRscan Nano in low power mode, press the Set button under the Hibernate Mode section. A check next to "Allow NIRscan Nano to hibernate after being inactive" will hibernate the NIRscan Nano after 5 minutes of inactivity. Clicking on this check will keep the NIRscan Nano active and will only power down when the "Hibernate Now" button is pressed.
- To replace the currently stored Reference data, follow these steps:
  - Place a highly reflective material on contact and fully covering the window. A 99% reflective material can be created by coating a metal with Spectralon®. A similarly reflective material can be created by coating a piece of metal with Wite-Out or liquid paper correction fluid.
  - Click on the scan button under the Replace Reference Calibration section and click on the dialog indicating that the reflective material is covering the sample window and ready for a scan.
  - Wait for the scan to complete.
  - Indicate if the reference scan is using a Wite-out or Spectralon material.
  - The new reference data is now stored on the Tiva EEPROM for future use in absorbance plots.

Replacing the reference data is useful as the lamps age or if the environmental conditions, like humidity, significantly change since these can affect the absorption plots that are referred from the stored reference data.

To batch convert all the binary data in a microSD Card to comma separated values ("*.csv"), use the ".dat to .csv conversion." Click on Browse, select the SD Card, and shift + click or drag to select all the .dat files for conversion from the SD Card.

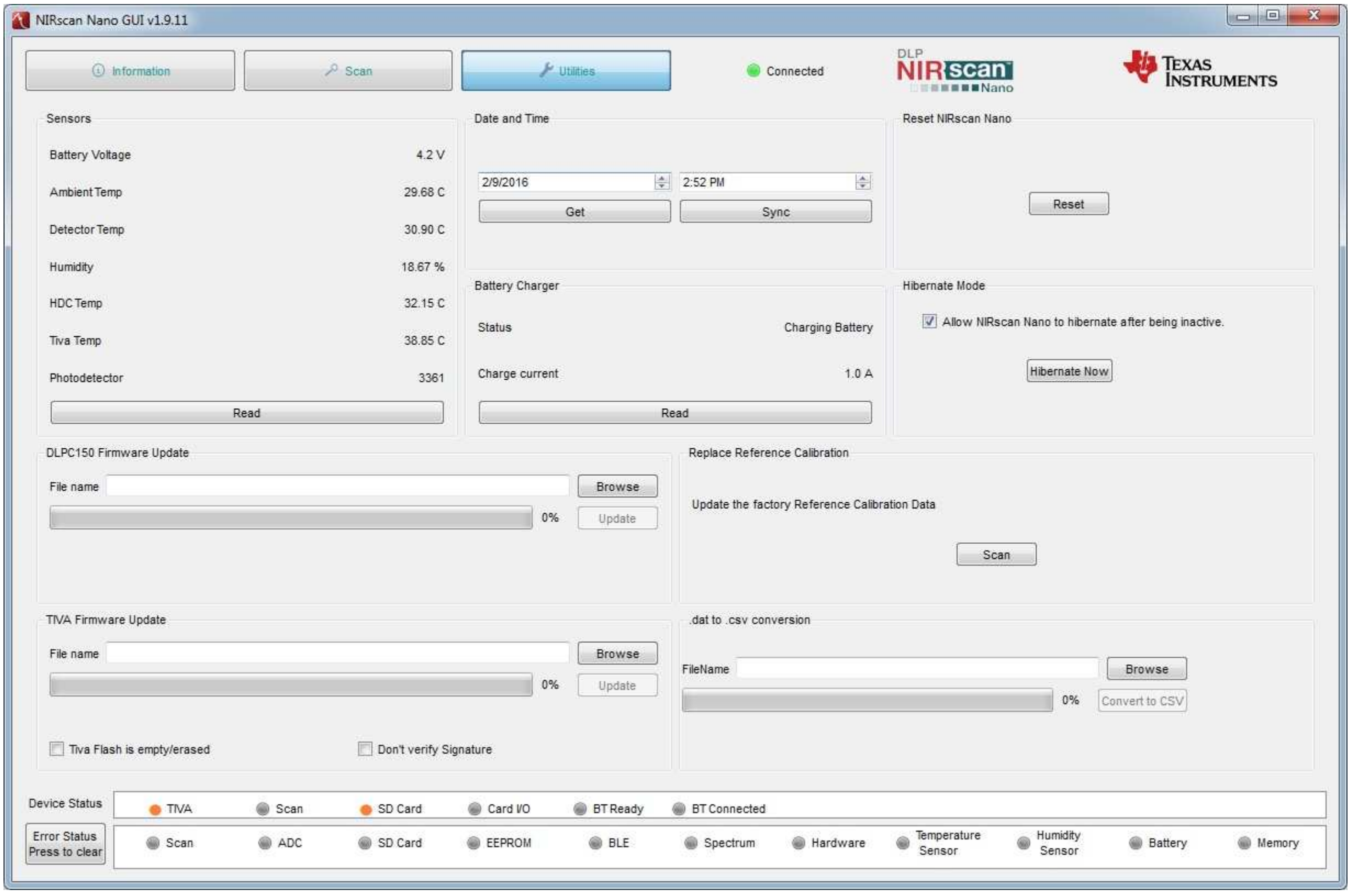


**Figure 3-11. DLP NIRscan Nano GUI Utilities Screen**

# DLP NIRscan Nano Hardware

## 4.1 External Power Supply Requirements

The DLP NIRscan Nano is powered from either a battery or a USB cable. The power requirements are:

- **USB Cable: (Not Included)**
  - Voltage: 4.75 to 5.25 V
  - Current Maximum: 560 mA when operating and 1 A when charging
  - Cable: 3 ft, USB A male to micro-USB B male (cable not included)
  - Digi-Key Part Number: Q853-ND
  - Manufacturer: Qualtek
  - Part Number: 3025010-03
- **Battery: (Not Included)**
  - Single-Cell Lithium-Polymer UL certified battery
    - Voltage: 3.7 V
    - Capacity: 1700 mA
    - Manufacturer: Tenergy
    - Part Number: 103450

**NOTE:** Only connect Tenergy 103450 Lithium Polymer UL certified battery or equivalent UL certified battery that meets: maximum charge current of 1 A or more, maximum charging voltage of 4.23 V or higher, battery over voltage protection at 4.305 V or higher, and battery under voltage lockout at 2.5 V or less.

If a battery is connected to the NIRscan Nano, a thermistor is required to safely charge the battery and monitor its temperature. The battery thermistor requirements are:

- **Battery Thermistor: (Not Included)**
  - 10-kΩ NTC thermistor
    - Manufacturer: Murata
    - Part number: NXRT15XH103FA1B040
    - Digi-Key part number: 490-7167-ND

Figure 4-1 shows a block diagram of the power circuits. The main power input is the external battery and USB connector. The bq24250 includes a single-cell battery charger and a highly efficient DC-DC converter to regulate the system voltage at 3.52 V. With an optional thermistor, the bq24250 monitors the temperature of the battery during charging. Note that a thermistor is required to charge the battery. The battery charger is set to supply up to a 1-A current during charging.

The rest of the devices regulate power to the subsystem as follows:

- The DLPA2005 in the DLP controller board regulates the power to the DLP2010NIR and DLPC150.
- The TPS82671 in the microcontroller board regulates the 1.8-V supply used by the Bluetooth subsystem CC2564MODN. To conserve power, a TPS22904 load switch turns off the 1.8-V supply to the Bluetooth subsystem when not in used.
- The TPS630636 in the microcontroller board supplies the main 3.3 V for the microprocessor and interface inputs and outputs to DLPC150, CC2564MODN, and Tiva microprocessor.

- The TPS81256 in the microcontroller board regulates the 5-V supply of the analog-to-digital converter (ADS1255), transimpedance amplifiers circuits (OPA350 and OPA2276), and 2.5-V reference voltage (REF5025) used in the detector board.
- A second TPS81256 in the DLP controller board regulates the 5-V supply for the lamp driver (OPA567 and INA213). The lamp driver drives two parallel lamps at 5 V and 280 mA. Each lamp is rated to a maximum 140 mA at 5 V.
- The TPS386596 serves as reset supervisor to hold the system in reset while all the supplies reach operational conditions. An external reset button issues a reset when the system has reached operational conditions.

For detailed connections of these devices, refer to the DLP NIRscan Nano schematics.

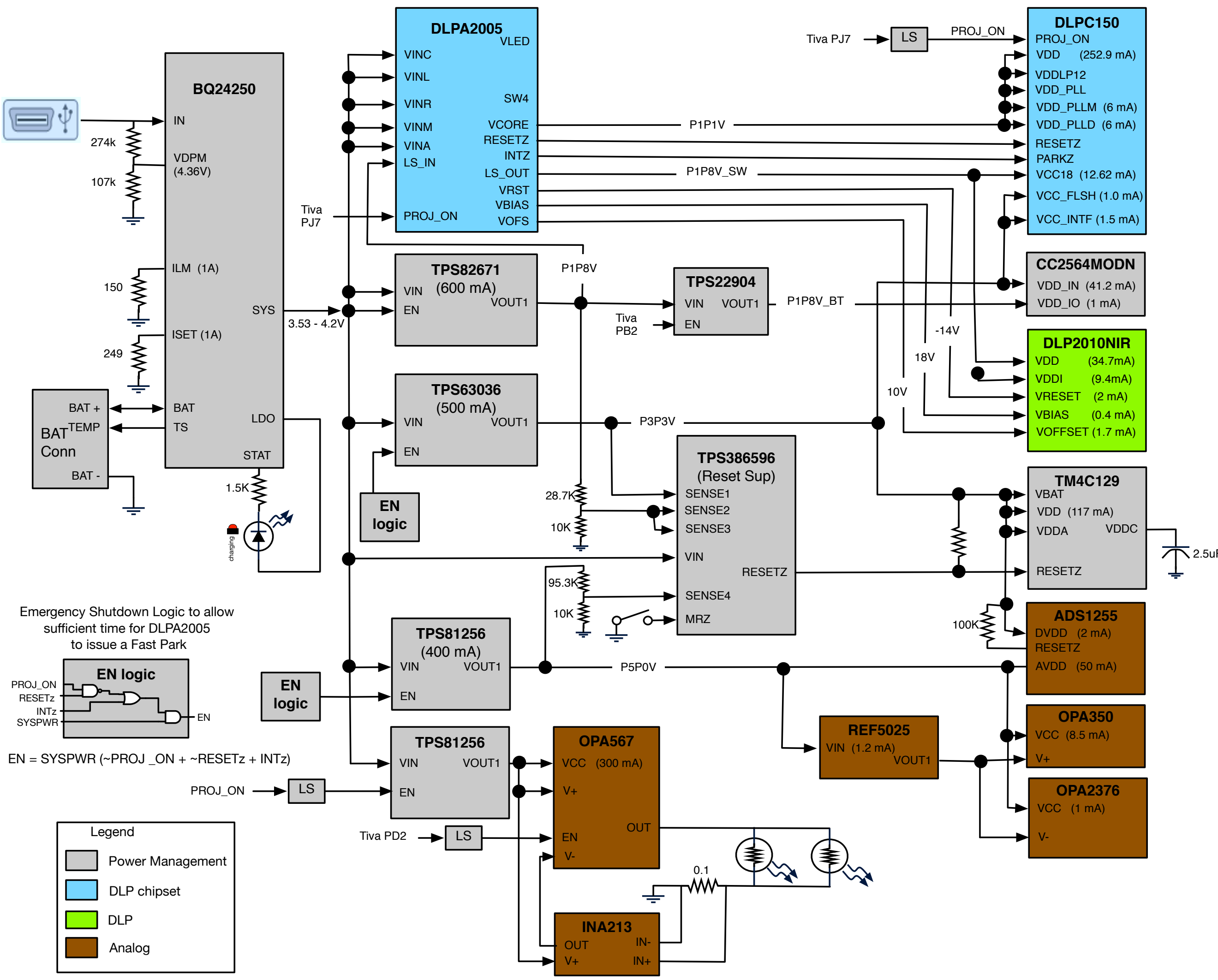


**Figure 4-1. DLP NIRscan Nano Power Block Diagram**

Figure 4-2 shows the Tiva connections to the components on the microprocessor board and detector board. The Tiva uses a 16-MHz external crystal as input to its on-board PLL to run the Tiva system at 120 MHz. A 32-kHz crystal supplies the clock to the Tiva's hibernation module and Bluetooth circuits. An external 32MB of SDRAM stores the patterns that are streamed to the DLPC150 through the Tiva's LCD interface. Tiva communicates to the HDC1000 and TMP006 sensors through its I2C6 and I2C7 peripherals. Both sensors generate a DRDY signal when a new value is available. This DRDY signals interrupt Tiva when a new value is available through PP7 for HDC1000 and PP6 for TMP006. Tiva's UART3 communicates with the CC2564MODN for Bluetooth transfers. The UART3 defaults to a 115200 baud transfer rate. Tiva's PH5 enables the Bluetooth circuits. Tiva interfaces to the microSD card through SSI3 in SPI mode. Tiva's SSI1 interfaces to the ADS1255. The ADS1255 also generates a DRDY signal when conversion is completed interrupting the Tiva processor through PP2. Tiva generates a synchronization signal to the ADS1255 through PH6 to start an ADC conversion when a pattern is displayed by the DLP2010NIR. To monitor the battery charger bq24250, PQ5 serves as a Tiva interrupt and commands are sent through I2C9. PQ0, PQ1, and PQ2 allow Tiva to override default bq24250 parameters. The Wake and Scan buttons are connected to Tiva's WAKE and PQ3 pins. To measure battery voltage, Tiva enables an analog MOSFET switch with PD5 to connect the battery to Tiva's ADC7 and perform a voltage measurement. An expansion headers supports a combination of Tiva SSI0 and UART4 pins. PF5, PH7, PL4 controls the green, blue, and yellow LED, respectively.

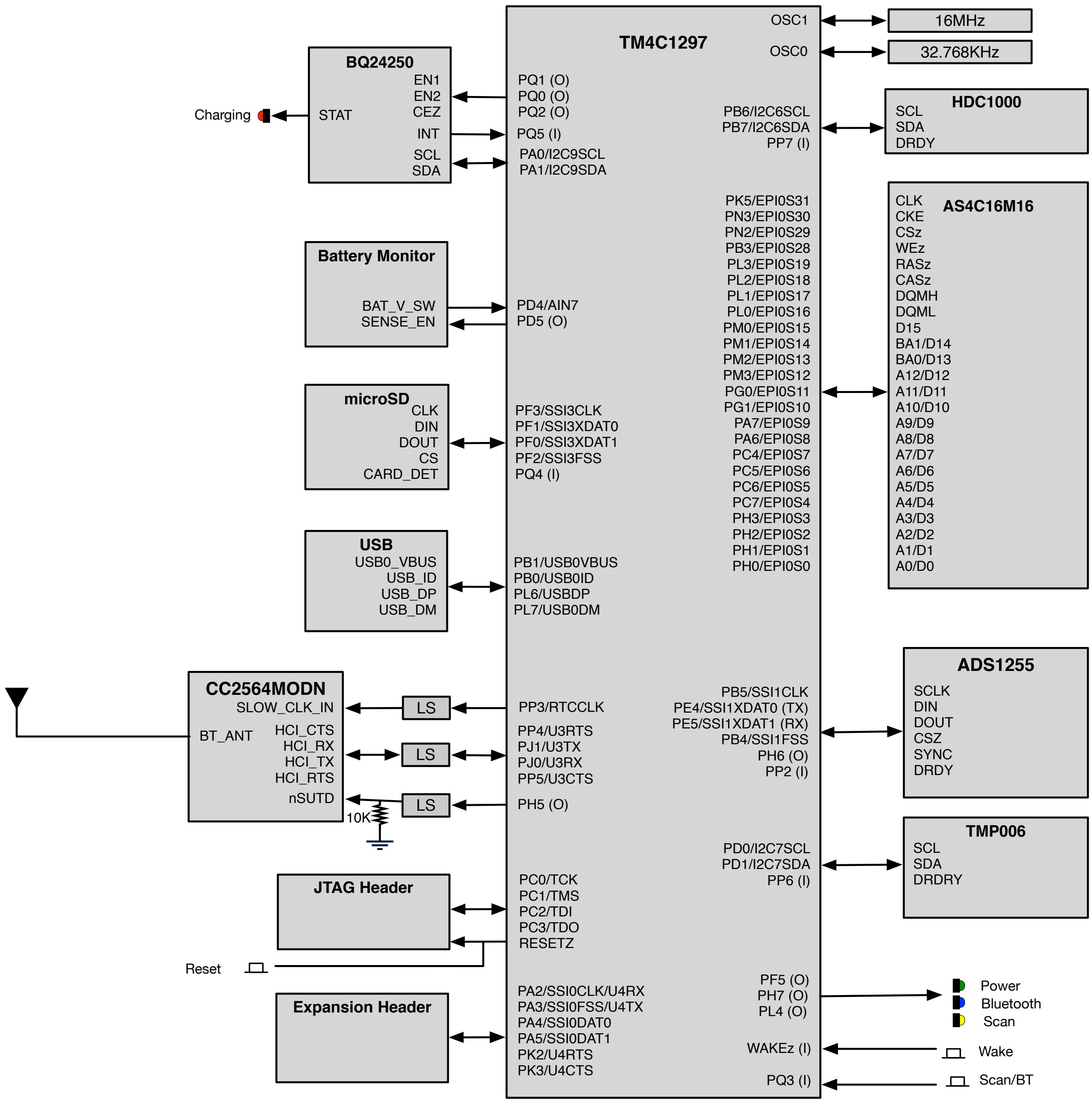


**Figure 4-2. DLP NIRscan Nano Tiva Connections**

Figure 4-3 shows the Tiva TM4C1297 connections to the DLPC150 controller board. Tiva powers up the DLP subsystem through PJ7. The Tiva's LCD interface is connected to the DLPC150 Parallel Port interface. Through this interface 24 patterns are transmitted per frame. DLPC150 sends two interrupts to the Tiva to indicate when a pattern is exposed (TRIG_OUT_2) and when a new frame begins (TRIG_OUT_1). For DLPC150 firmware updates, the Tiva writes to the DLPC150 serial flash through its SSI2 peripheral when the DLP subsystem is powered down. Tiva's PD2 controls the lamp. During each scan, a photodiode measures the light intensity. The intensity value is transmitted by the DLPA2005 to the DLPC150 and then to the Tiva.

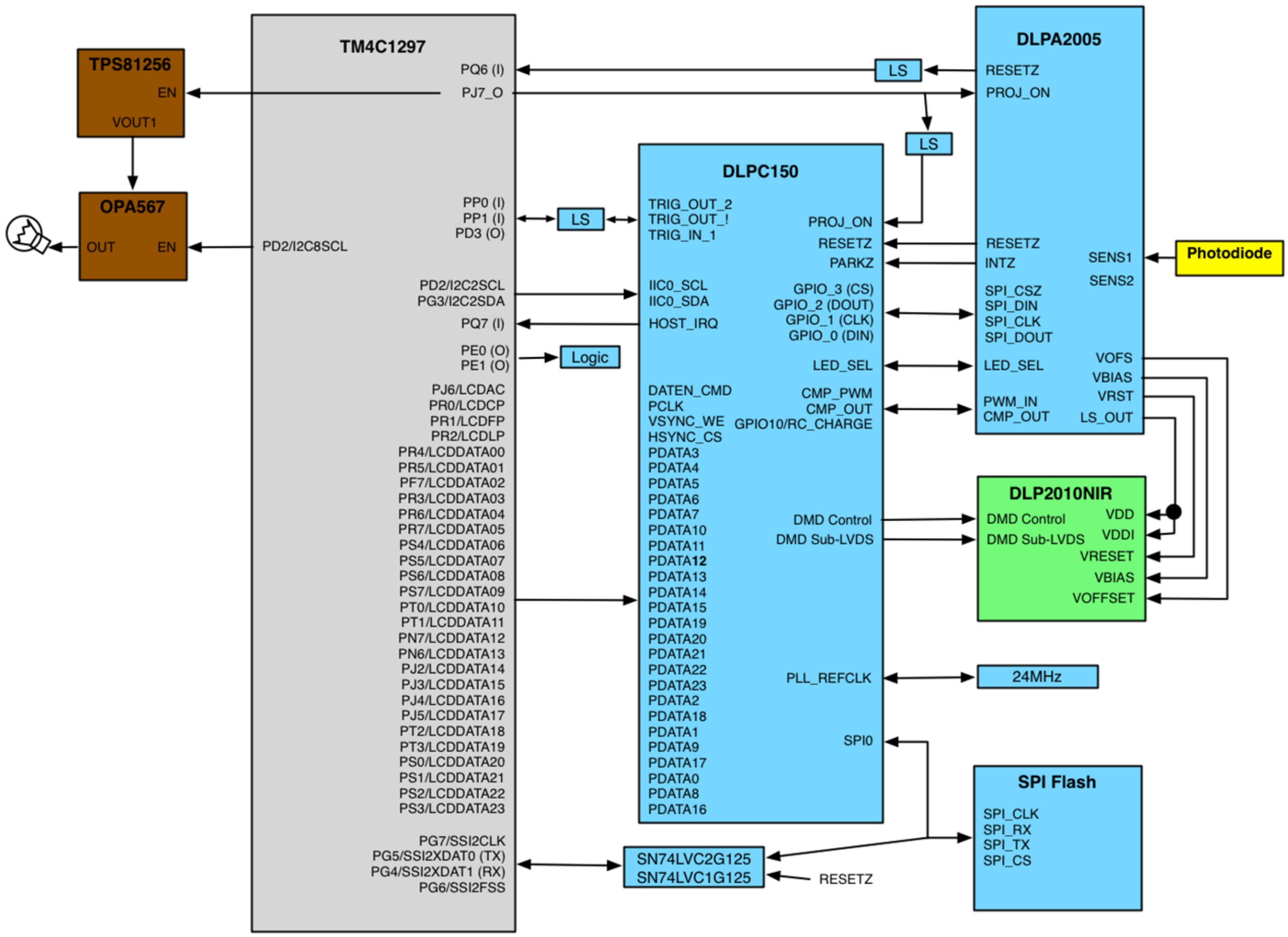


**Figure 4-3. DLP NIRscan Nano Tiva Connections to DLPC150 Controller Board**

# DLP NIRscan Nano Software

## 5.1 Overview

The DLP NIRscan Nano's Tiva microprocessor is the system's main control processor. The Tiva handles button presses, commands and data transfers over USB or Bluetooth, controls the DLP subsystem, streams the patterns to select specific wavelengths, captures data from InGaAs detector, activates lamps, and stores data in the microSD card. Due to the realtime nature of the system, the Tiva software includes TI-RTOS that coordinates tasks while handling realtime interrupts and semaphores. Low-level drivers for Tiva's USB, GPIO, EPI, $I^2C$, LCD, SPI, and UART peripherals are handled by TivaWare libraries and routines. The Bluetopia Stack handles Bluetooth communications. DLP Spectrum Library handles pattern generation and data transformation from raw scan data to a wavelength spectrum. A command handler interprets commands from USB or Bluetooth and starts the tasks needed to execute the commands. The main application initializes the system and waits for commands from USB and Bluetooth. The overall software architecture depicting these components is shown in Figure 5-1.

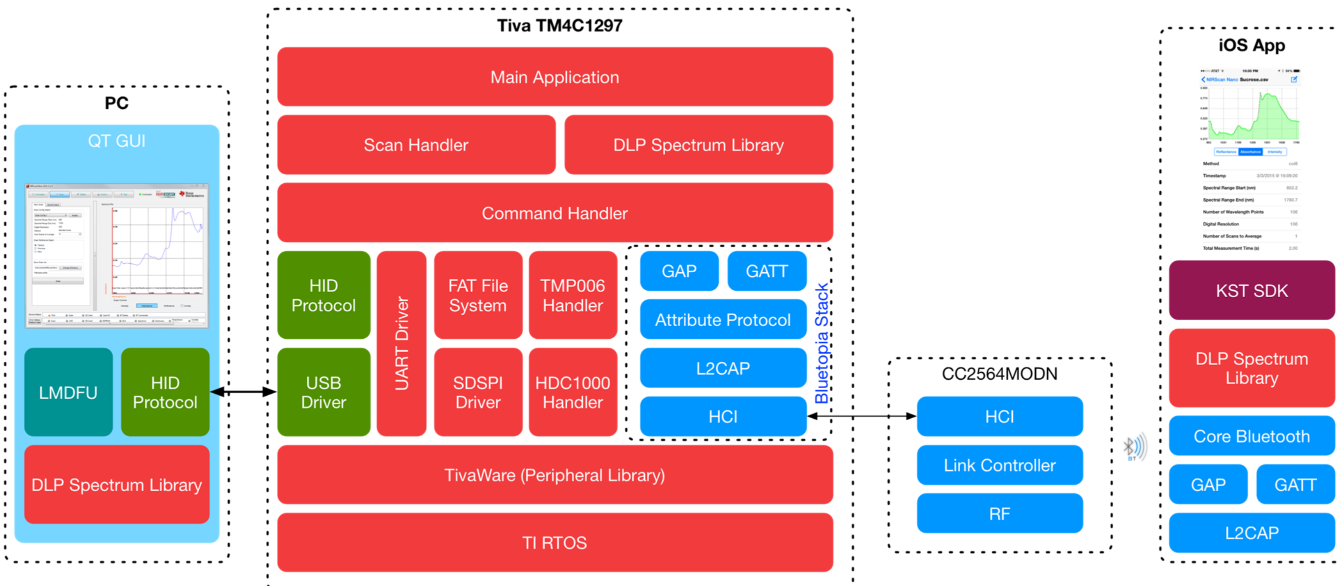


**Figure 5-1. DLP NIRscan Nano Software Architecture**

### 5.1.1 TI RTOS

TI RTOS is a scalable, real-time operating system that handles scheduling and synchronization of tasks, interrupts, includes a limited set of drivers, and provides hardware abstraction layer to ease application software development. The TI RTOS also includes the FAT file system (FATFs) module to store data in the microSD card. The TI RTOS drivers used by DLP NIRscan Nano are:

- $I^2C$: driver used for Tiva communication with DLPC150, TMP006, and HDC100.
- SDSPI: SPI driver for Tiva communication with SD Card.
- USBMSCHFatFS: driver for USB mass storage class.

### 5.1.2 TivaWare

TivaWare is a set of drivers for accessing the Tiva peripherals. DLP NIRscan Nano uses the following TivaWare drivers:

- UART: Driver for Tiva interface with CC2564MODN
- USB: Driver for HID transfers between Tiva and PC. The USB drivers handles Tiva's USB interrupts
- SPI: Driver for Tiva interface with ADS1255
- ADC: Driver to control Tiva ADC peripheral
- GPIO: Driver to control Tiva GPIO pins
- LCD: Driver to interface Tiva with DLPC150 parallel port

### 5.1.3 USB Driver

USB Communication to the DLP NIRscan Nano uses the HID class. Tiva enumerates as a slave USB 2.0 high power device. Appendix H lists the packet structure and commands supported through USB.

### 5.1.4 UART Driver

Universal Asynchronous Receive and Transmit (UART) Communication to the DLP NIRscan Nano. Appendix I lists the commands supported through UART.

### 5.1.5 SDSPI Driver

To store data on the microSD card, Tiva's SSI3 peripheral communicates with the microSD card using SPI mode (SDSPI). Tiva stores data on the microSD card using the file allocation table (FAT) file system.

### 5.1.6 Bluetopia Stack

The DLP NIRscan Nano wirelessly communicates using Bluetooth Low Energy (BLE) version 4.0. The Bluetooth communication is handled by the TI Bluetopia stack and the CC2564MODN. The TI Bluetopia stack and CC2564MODN implement a fully certified Bluetooth 4.0 specification. The BLE wireless communication uses two main profiles for discovery and communication with a remote host:

- GAP: Generic Access Profile for basic discovery and establishing connections.
- GATT: Generic Attribute Profile for commands and data transfer.

The DLP NIRscan Nano supports Bluetooth version 4.0 specification. When Bluetooth subsystem is activated, the DLP NIRscan Nano broadcasts its availability while a smartphone, tablet or PC acts as an observer. Once connected, the DLP NIRscan Nano acts as a server for the GATT profile while the smartphone, tablet, or PC acts as a client.

The DLP NIRscan Nano Bluetooth GATT Profile supports the following services:

- Battery Service (BAS) to provide battery charge capacity.
- Device Information Service (DIS) to provide manufacturer name, model number, serial number, hardware revision, spectrum library revision, and Tiva software revision.
- GATT General Information Service to provide temperature, humidity, status, hours of use, lamp hours, and battery recharge cycles.
- GATT Date and Time Service to synchronize date and time information between smartphone, tablet, or PC to the Tiva's realtime clock.
- GATT Calibration Service to provide calibration coefficients.
- GATT Scan Configuration Service to provide stored configurations and scan configuration data.
- GATT Scan Data Service to initiate scan, clear scan data, and return stored scan data.

The Tiva processor handles these profiles and uses the logical link control and adaptation protocol (L2CAP) to pass packets through a host controller interface (HCI). The Tiva's UART3 peripheral communicates with the CC2564MODN HCI module. The CC2564MODN transmits these packets to the client device.

### 5.1.7 DLP Spectrum Library

The DLP spectrum library is a collection of C-language routines that provide the fundamental pieces to use a DLP system in a spectroscopy application. The DLP spectrum library is resolution and host processor independent, allowing the routines to be used with different DMD resolutions and processor systems. The routine sources are shared by the Tiva code, the GUI code, and the iOS App. The DLP Spectrum Library are classified into three main categories:

- **Scan:** Performing a column or Hadamard scan by:
  - Generating the appropriate full-frame DMD patterns based on a specific scan configuration.
  - Computing reflectance and absorbance data form the intensity data during a scan.
  - Handling serialization and deserialization of scan configuration and scan data.
- **Calibration:** Calibrating a system at the factory by:
  - Finding peaks from a scan of a calibrated lamp.
  - Finding the full width half maximum of specific peaks data of a calibration scan.
  - Computing the calibration coefficients for a system.
- **Utilities:** Utilities to handle:
  - Conversion between DMD mirror column position to a calibrated wavelength or wave number, and vice versa.
  - Spectrum data calculations, such as: absorbance, reflectance, and spectrum comparisons.
  - Matrix operations.
  - Binary pattern packing.

The DLP NIRscan Nano utilizes a previously-created scan configuration (through the GUI or stored on the NIRscan Nano) to perform a scan. This scan configuration is created on the NIRscanNanoGUI and transferred to the system in serialized fashion. The system's Tiva processor deserializes this data and generates a set of full-frame DMD patterns based on the scan configuration and the factory-stored calibration data. Then, the Tiva turns on the lamps and streams the full-frame DMD patterns to perform a scan. Tiva collects several data points for each pattern from the detector's ADC conversion. This data is stored in a structure, and is then serialized and transferred to the PC through USB or Bluetooth. The NIRscanNanoGUI or the iOS App deserializes this scan data, interprets it using the DLP Spectrum Library, and plots the resulting spectrum.

### 5.1.8 DLP Spectrum Library Workflow

The following sections show the use of the DLP Spectrum Library workflow to read scan configuration information, decode scan data, and compute reference, absorbance, and reflectance.

#### 5.1.8.1 Scan Configuration Workflow

The DLP Spectrum Library routine to interpret scan configuration information is **dlpspec_scan_read_configuration()**. This routine takes as input the serialized scan configuration transferred through USB or Bluetooth and deserializes to extract all the scan configuration information. Figure 5-2 shows the typical workflow to view configuration information. The white input box is an argument generated by the client based on the size of the configuration serialized data. The blue input box denotes Bluetooth, USB, or UART scan configuration data.

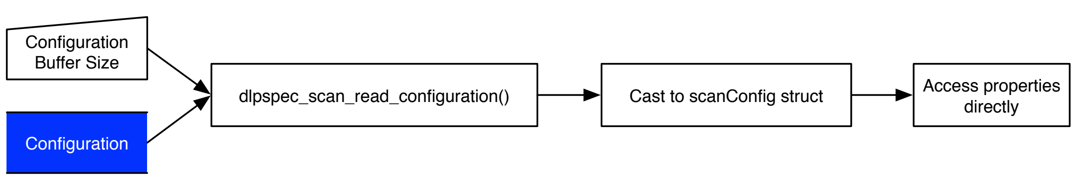


**Figure 5-2. DLP Spectrum Library View Configuration Information Workflow**

### 5.1.8.2 Decode Scan Workflow

The DLP Spectrum Library routine to interpret scan data is **dlp_scan_interpret()**. This routine takes as input the serialized scan data transferred through USB or Bluetooth and deserializes and extracts the intensities of each wavelength in a scan. Figure 5-3 shows the typical workflow to decode scan data. Scan Data Buffer Size is computed size of the data blob. White trapezoid is a pointer to a scanResults struct which the client app has allocated. The blue input box denotes Bluetooth, USB, or UART scan data.

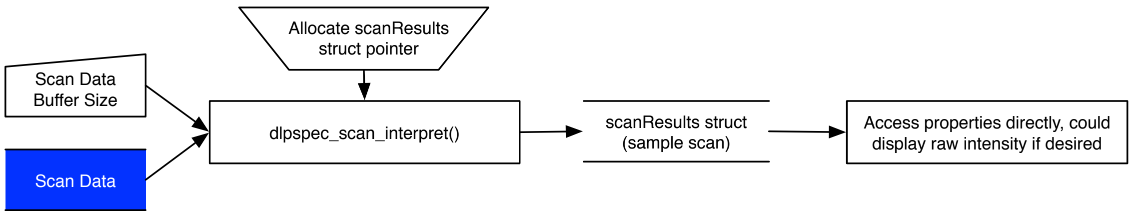


**Figure 5-3. DLP Spectrum Library Decode Scan Results Workflow**

### 5.1.8.3 Compute Reference Workflow

The DLP Spectrum Library routine to compute reference is **dlp_scan_interpReference()**. This routine takes as input the serialized reference calibration data stored on the DLP NIRscan Nano and transmitted through Bluetooth, USB, or UART and computes the reference intensities for each wavelength. Figure 5-4 shows the typical workflow to compute the reference. Sample scan struct is interpreted scanResults from a sample scan with an arbitrary scan configuration. White trapezoid is a pointer to a scanResults struct which the client app has allocated. The blue input box denotes Bluetooth, USB, or UART reference calibration data.

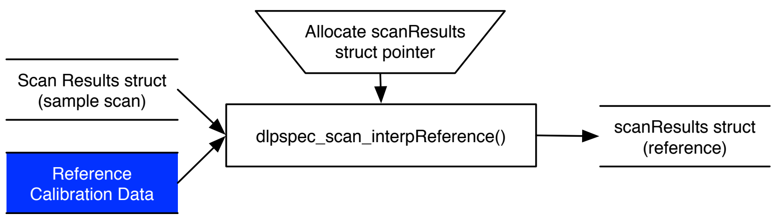


**Figure 5-4. DLP Spectrum Library Compute Reference Workflow**

### 5.1.8.4 Compute and Display Reflectance Workflow

Reflectance can be computed by dividing the sample scan intensities by the reference scan intensities. Figure 5-5 shows the typical workflow to compute the reflectance.

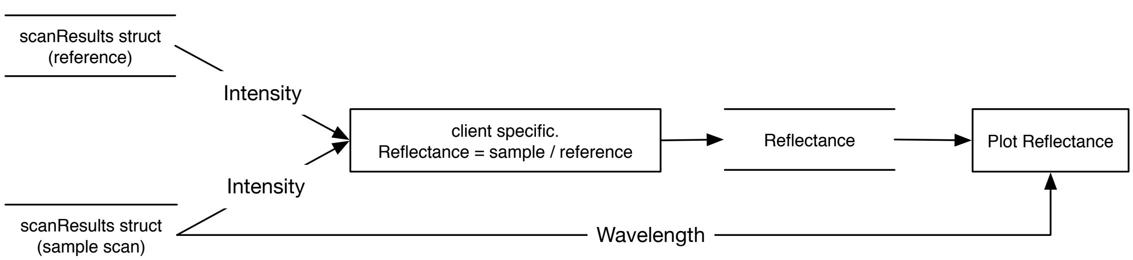


**Figure 5-5. DLP Spectrum Library Compute and Display Reflectance Workflow**

#### 5.1.8.5 Compute and Display Absorbance Workflow

Absorbance can be computed using the negative of the logarithm (based 10) of the reflectance. Figure 5-6 shows the typical workflow to compute the reflectance.

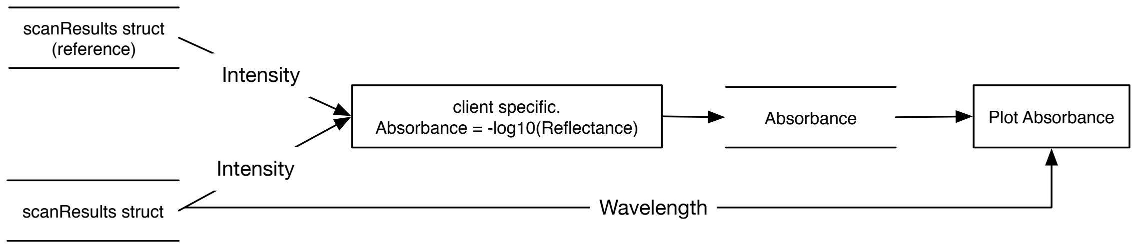


**Figure 5-6. DLP Spectrum Library Compute and Display Absorbance**

## 5.2 Software System Overview

The DLP NIRscan Nano software uses a sets of tasks, hardware interrupts, and semaphores to coordinate the efforts needed to interpret USB or Bluetooth commands, respond to button presses, scan an object, and capture the InGaAs detector values. Figure 5-7 shows a high-level block diagram of the software elements of the system. The DLP NIRscan Nano system includes the following hardware interrupt handlers to respond to hardware events:

- Button Interrupt Handler: Interrupt handler that responds to Scan/Bluetooth button presses and Wake button presses.
- Trigger Interrupt Handler: Interrupt handler that synchronizes and keeps track of the pattern displayed during a scan. Two interrupts from the DLPC150 (ihFrameTrigger and ihPatternTrigger) indicate when a pattern has been displayed, when 24 patterns have completed the display, and when the DLPC150 buffer needs to be reloaded with new patterns. This handler also captures data from the ADS1255 when a pattern is displayed. It uses the ADS1255 DRDY signal to trigger the read of the just-converted value.
- Display Interrupt Handler: Interrupt handler that controls the Tiva LCD peripheral and its frame buffer that resides in external SDRAM. The frame buffer streams 24 patterns per frame to the DLPC150.
- UARTStdioIntHandler: TivaWare UARTStdin driver that handles Tiva UART3 or UART4 transactions. Only one UART mode is allowed: console or UART commands.
  – The data from UART3 transactions is the console output for Bluetooth logs.
  – The data from UART4 transactions is passed to the Command Handler task to interpret and respond to a specific set of UART commands described in Table I-1.
- USB DriverLib: TivaWare USB driver that handles all USB HID transactions. The data from USB transactions is passed to the Command Handler task to interpret and respond to a specific set of USB commands described in Table H-2.
- TMP I2C Driver: TI-RTOS driver that manages the temperature readings from the TMP006.
- HDC I2C Driver: TI-RTOS driver that manages humidity and temperature readings form the HDC1000.
- microSD Driver: TI-RTOS driver that controls reading and writing data to the microSD card through a FAT32 file system.
- NanoTimer2: TI-RTOS timer that controls the pulsing of the LEDs depending on the error status.

The following tasks handle specific portions of the system:

- Bluetooth Stack: This task handles Bluetooth communication with a mobile application through the implementation of GATT profile in the Bluetopia Stack. Several semaphores control the operation:
  - BLEStartSem: Power-up the Bluetooth circuits and initializes the Bluetooth Stack.
  - BLEEndSem: Powers-down the Bluetooth circuits and gracefully closes the Bluetooth Stack.
  - BLECmdRecd: Coordinates the reception and processing of GATT profiles and notifies Command Handler of a new command to be processed.
  - BLECmdCom: Handles the response to a BLE Client's read and notify requests.
  - BLENotifySem: Handles the asynchronous notifications to a BLE Client's notification subscriptions.
- Command Handler: Interfaces the USB and Bluetooth tasks, interprets commands, and starts the sensor read task and the scanning tasks.
- Scan Handler: Controls power to the subsystems necessary to perform a scan (ADS1255, Lamp, DLPC150), manages the pattern streamed to the DLPC150, and reads sensor information during a scan. Two semaphores control the start of the scan (scanSem) and the end of a scan (endscanSem).

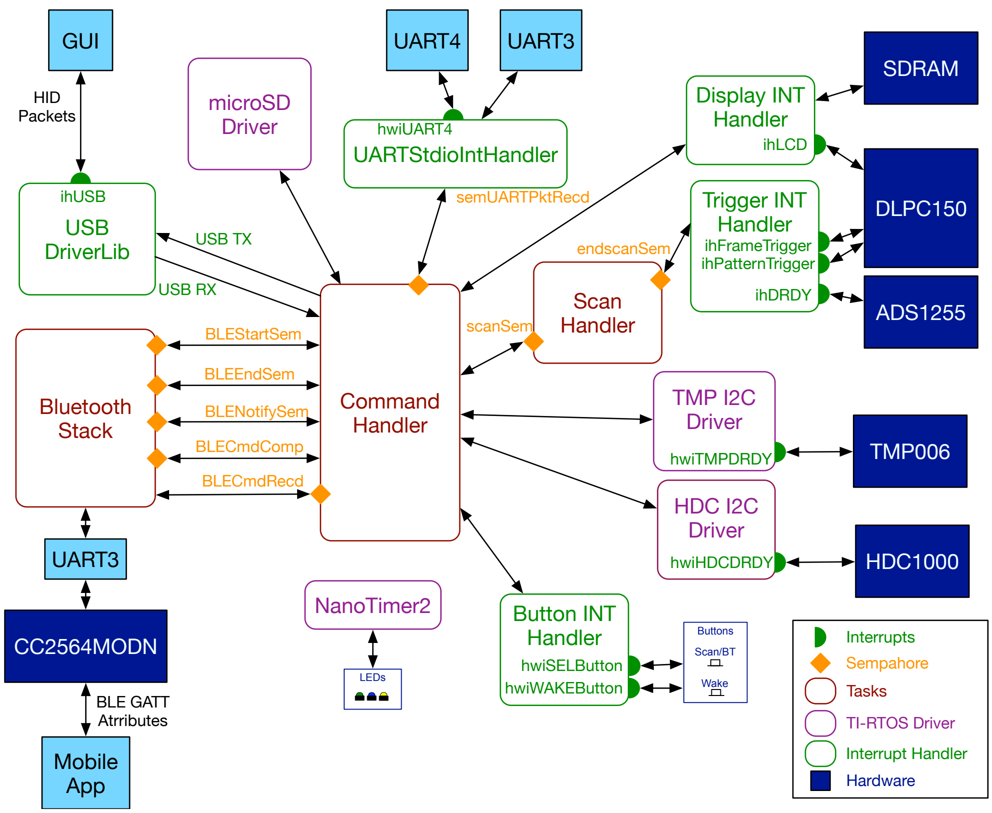


**Figure 5-7. DLP NIRscan Nano Software Block Diagram**

### *5.2.1 Scan Workflow*

To perform a successful scan, a set of commands must be issued to select a scan configuration, perform a scan, acquire the scan data, and interpret the scan:

1. Scan Configuration: A scan configuration must be selected before issuing a scan command. To select a scan, perform one of the following two steps:
   (a) Create and define a scan configuration (scanConfig) structure. This structure is defined in dlpspec_scan.h, Once the structure is created, serialize it using the API dlpspec_scan_write_configuration() and send it to the NIRscan Nano using the command NNO_CMD_SCAN_CFG_APPLY.
   (b) Choose an already defined scan configuration stored in the EEPROM of the NIRscan Nano. To apply this already defined scan configuration, use NNO_CMD_SCAN_SET_ACT_CFG command.

   After a scan configuration is created or selected, the NIRscanNano application running in Tiva generates the necessary patterns for the scan configuration and stores them on the external SDRAM. The 32MB size of the external SDRAM allows the storage of up to 624 patterns {32M / (864*480*3) = 26 groups of 24 binary images = 624 binary patterns}.
2. Perform Scan: After selecting a scan configuration, perform a scan. To perform a scan, follow these steps:
   (a) Optionally send NNO_CMD_SCAN_NUM_REPEATS command to override the number of times to repeat the scan, if a different amount from the scan configuration structure is desired.
   (b) Send NNO_CMD_READ_SCAN_TIME command to determine the amount of time necessary for the scan to complete. This number is used as a timeout to poll for scan completion in case of errors.
   (c) Send NNO_CMD_PERFORM_SCAN command to start the scan. A one byte parameter to this command indicates whether the scan result needs to be stored in the SD card (0x1) for future retrieval or not to store in SD card (0x0). This command performs the following actions:
      (i) Power up the DLP Subsystem by driving the PROJ_ON signal of DLPA2005 to high.
      (ii) Power up the lamps by enabling the OPA567. Read timer to mark start time.
      (iii) Wake up the ADS1255 by sending a wake up command through the serial port.
      (iv) Read humidity and temperature data from HDC1000.
      (v) Wait 100 msec after powering up the lamps so as not to power the rest of the system while the initial lamp power-on surge occurs. This wait is based on the timer difference from the timer value read on the second step.
      (vi) Enable Tiva LCD peripheral.
      (vii) Setup DLPC150 to take data from 24-bit RGB interface.
      (viii) Read temperature data from TMP006.
      (ix) Wait up to 625 ms after powering up the lamps for lamp stabilization. This wait is based on the timer difference from the timer value read on the second step.
      (x) Read photodetector.
      (xi) Program DLPC150 to take patterns from 24-bit RGB interface and display them on the DLP2010NIR.
      (xii) Perform a quick scan to set ADS1255's programmable gain amplifier (PGA) to the maximum value that does not saturate the ADC.
      (xiii) Tiva streams patterns to the DLPC150 while obtaining ADC data. Multiple ADC values are read per pattern and averaged. The pattern set is repeated depending on the number of times a scan is repeated.
      (xiv) When the scan is finished, read temperature and humidity from TMP006 and HDC1000.
      (xv) Power off the lamps by disabling OPA567.
      (xvi) Power off the DLP Subsystem by driving PROJ_ON signal of DLPA2005 to low.
      (xvii) Populate scan data header and values.
   (d) Send NNO_CMD_READ_DEVICE_STATUS command to query the EVM at regular intervals to determine if the scan has completed. After a scan has completed, bit 1 of the returned byte is

cleared. Bit 1 of the returned byte is defined as NNO_STATUS_SCAN_IN_PROGRESS in NNOStatusDefs.h. Bit 1 is set in deviceStatus when scan is in progress. Bit 1 is cleared in the deviceStatus when the scan has completed.

3. Acquire Scan Data: To retrieve the scan data, follow these steps:
   (a) Send NNO_CMD_FILE_GET_READSIZE command with NNO_FILE_SCAN_DATA as the parameter to request that the NIRscan Nano sends scan data when the subsequent NNO_CMD_FILE_GET_DATA command is sent. The Tiva application response to this command is the number of bytes to be read out with the NNO_CMD_FILE_GET_DATA command.
   (b) Send NNO_CMD_FILE_GET_DATA command to read out the serialized scan data in binary form.
4. Interpret Scan Data: Once the scan data has been retrieved, the dlpspec_scan_interpret() function converts the binary data into a scanResults structure. The scanResults structure will contain the vector of wavelength points (depending on the scanConfig selected), the corresponding intensity vector, as well as metadata that provides various other parameters such as, temperature, humidity, photodetector, etc. captured during the scan.

### 5.2.2 Total Scan Time

The total time taken for a given scan is the sum of the following:

1. 100 ms delay to avoid lamp power-on surge.
2. 625 ms for lamp output to stabilize.
3. Time taken for the quick scan to determine appropriate PGA Gain setting. This time is directly proportional to the “digital resolution” and “exposure time” settings in the selected scan configuration.
4. Time taken for the actual scan. This time is directly proportional to “digital resolution”, “exposure time” and “Number of scan to average” settings in selected scan configuration.
5. 100 ms for other overhead incurred while reading sensors and other data processing.

In v2.1 of the GUI (see Section 3.1.3), a new feature is provided to keep the lamp ON and fix the PGA setting, instead of the default behavior of shutting off the lamp after each scan. By enabling this feature, scan times can be substantially reduced. By keeping the lamp on between scans, the delays for lamp stabilization are avoided. Using a fixed PGA setting, the quick scan is bypassed. With the lamp on as well as a fixed PGA setting, the total scan time will be the sum of only Figure K-1 and Figure 3-8.

## 5.3 UART Command Processing Workflow

UART commands are processed by the workflow shown in Figure 5-8. The system waits for a packet received by UART4. If the packet matches the four Start ID bytes, then the next few bytes are stored in the received buffer. The next byte is the Checksum, followed by the Flags byte, Sequence byte, Length bytes, Command bytes, and Data bytes. After the Data bytes, four End ID bytes indicates the end of the packet. Once the end of the packet is received, the semaphore semUARTPktRecd is posted. The Command Processor task will wait for this semaphore. Once the semaphore is posted and no other commands are being processed, the Command Processor task processes the UART command and sends a respond through the UART.

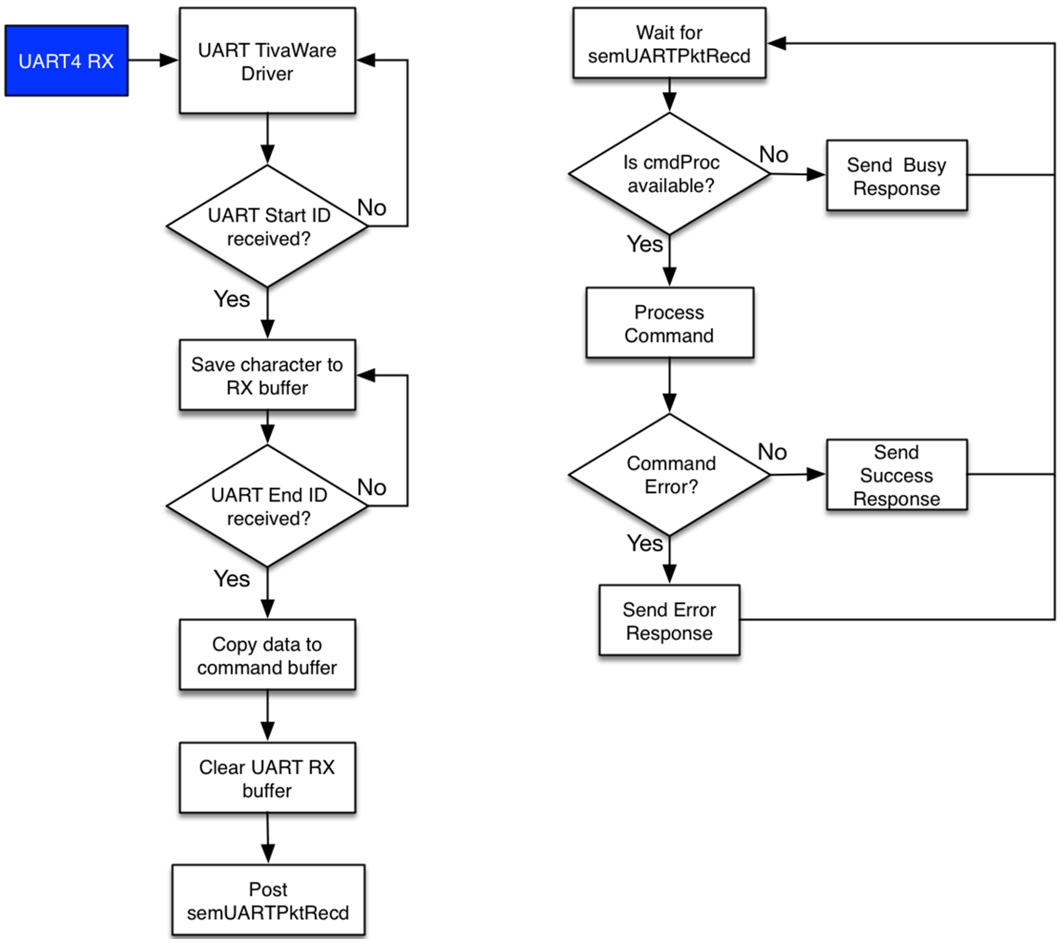


**Figure 5-8. UART Command Processing Workflow**

## 5.4 Bluetooth Client App Workflow

The following sections describe a suggested workflow for the Bluetooth Client to connect and transfer data.

### *5.4.1 Bluetooth Client Establishing a Connection*

The Bluetooth Client searches for the DLP NIRscan Nano using the GAP for discovery. Once the Bluetooth Client detects a DLP NIRscan Nano, the Bluetooth Client reads the DLP NIRscan Nano advertized packets, establishing a connection. Figure 5-9 describes this process.

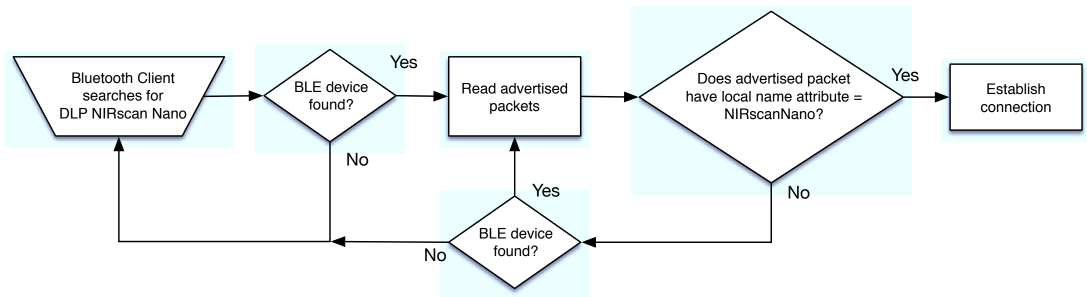


**Figure 5-9. Bluetooth Low Energy Connection Workflow**

### *5.4.2 Bluetooth Client GATT Profiles*

Once the Bluetooth Client establishes a connection with the DLP NIRscan Nano, the supported GATT profile is enumerated. DLP NIRscan Nano uses standard Bluetooth Low Energy services for Device Information (DIS) and Battery (BAS). The rest of the data transfer is through the custom GATT services and characteristics in the following sections. In the following workflow figures, blue denotes a GATT service characteristic, while red denotes a DLP Spectrum library routine.

#### 5.4.2.1 Bluetooth Client GATT General Information Service

Once the GATT General Information Service is enumerated, the Bluetooth Client can prompt DLP NIRscan Nano to read temperature and humidity values. The Bluetooth Client can set a threshold for temperature and humidity and then subscribe to the temperature and humidity threshold notification. The Bluetooth Client can also read a device and error status and then subscribe to the device and error status notifications.

#### 5.4.2.2 Bluetooth Client GATT Date and Time Service

Once the GATT Date and Time Service is enumerated, the Bluetooth Client can write the data and time values to the DLP NIRscan Nano. It is recommended that the Bluetooth Client sets the date and time every time a connection is established, so the scan data has the correct date and time stamp.

#### 5.4.2.3 Bluetooth Client GATT Calibration Service

Once the GATT Calibration Service is enumerated, the Bluetooth Client can prompt DLP NIRscan Nano to download spectrum calibration coefficients, reference calibration coefficients, and reference calibration matrix. These parameters are unique for each DLP NIRscan Nano and are required for spectrum intensity, reflectance, and absorbance plots. These parameters must to be downloaded whenever a new DLP NIRscan Nano is connected to a Bluetooth Client and before a scan is performed. To download these parameters, the Bluetooth Client must follow these steps:

- Subscribe to the notification of the characteristic UUID that returns the corresponding coefficient.
- Issue a request for the coefficient and wait for the notification to read the corresponding coefficient.
- Once a notification is received, then the Bluetooth Client reads the serialized multiple packets from DLP NIRscan Nano.

Figure 5-10 shows the workflow for this service.

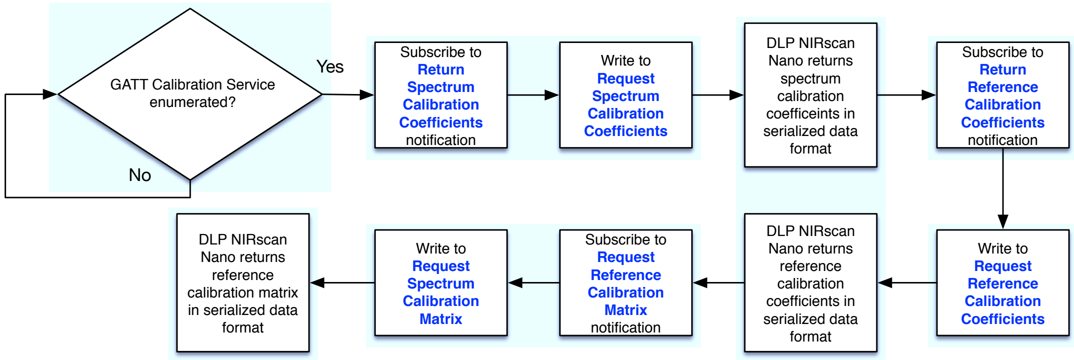


**Figure 5-10. GATT Calibration Service Workflow**

### 5.4.2.4 Bluetooth Client GATT Scan Configuration Service

Once the GATT Scan Configuration Service is enumerated, the Bluetooth Client can prompt DLP NIRscan Nano to download stored scan configurations. These parameters must to be downloaded whenever a new DLP NIRscan Nano is connected to a Bluetooth Client and before a scan is initiated. To read the stored scan configuration, the Bluetooth Client must perform the following steps:

- Read the number of stored configurations.
- Subscribe to the notification of the characteristic that returns stored configuration list.
- Issue a request for stored configuration list and wait for the notification to read the stored configuration list.
- Subscribe to the notification of the characteristic that returns scan configuration data.
- Read each scan configuration data by writing the scan configuration ID to the characteristic that returns the scan configuration data, wait for the notification, and then read the serialized scan configuration data returned. The DLP Spectrum Library provides a routine to interpret this serialized data: **dlpspec_deserialize**. Repeat this step for each stored scan configuration.
- Set the active scan configuration by writing to the scan configuration ID to the active scan configuration characteristic.

Figure 5-11 depicts the workflow for this service.

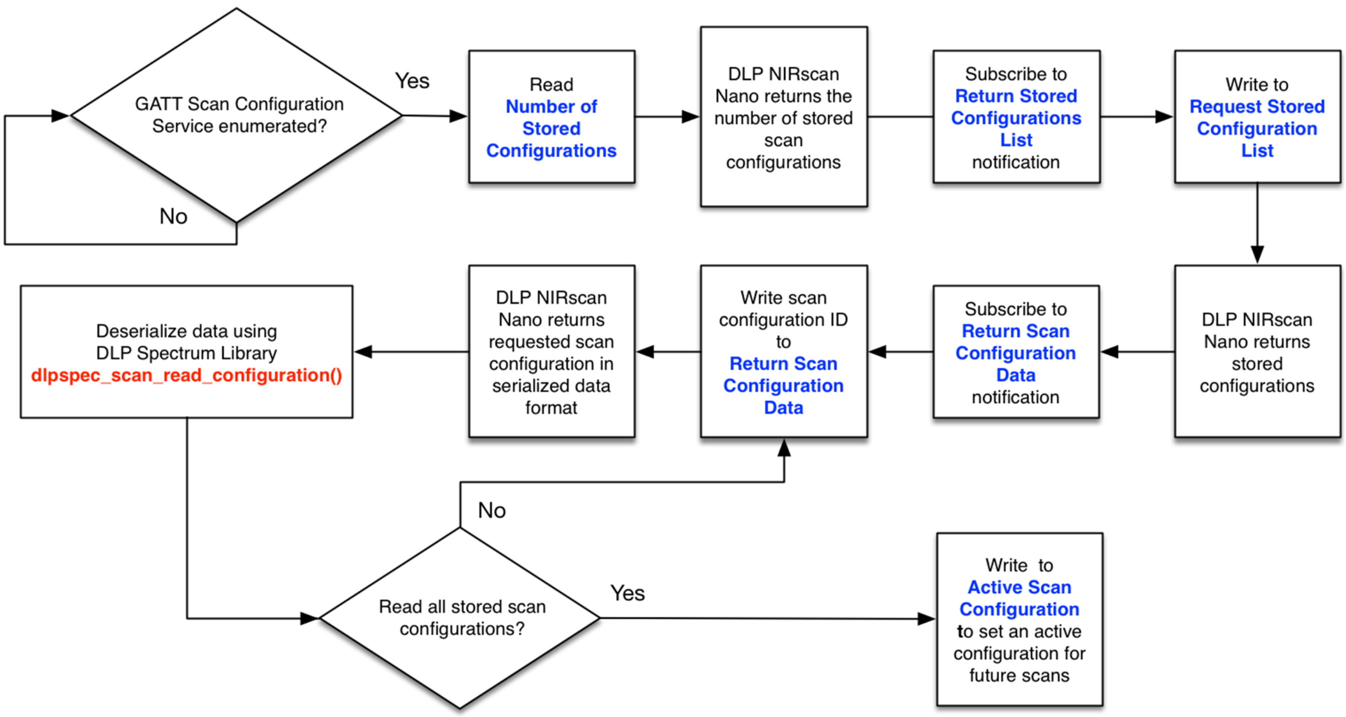


**Figure 5-11. GATT Scan Configuration Service Workflow**

### 5.4.2.5 Bluetooth Client GATT Scan Data Service

Once the GATT Scan Data Service is enumerated, the Bluetooth Client can prompt DLP NIRscan Nano to download stored scan data or perform a scan. To read the stored scan data, the Bluetooth Client must perform the following steps:

- Read the number of stored scans.
- Subscribe to the notification of the characteristic that returns stored scan indices list.
- Issue a request to read stored configuration list, wait for the notification to read the stored configuration list.
- Subscribe to the notifications of the characteristics to return scan name, scan type, and scan date/time, and to request packet format version.
- Issue requests for scan name, scan type, scan date/time, and packet format version. Then wait for the notifications to read scan name, scan type, scan date/time, and packet format version. Repeat the last two steps for each stored scan.

To perform a scan, the Bluetooth Client must perform the following steps:

- To display an existing scan:
  – If the scan information is not available, subscribe to the notification of the characteristic to return serialized scan data structure.
  – Issue a request to read serialized scan data structure and wait for the notification to read the scan data structure. The DLP Spectrum Library provides a routine to interpret this serialized data: **dlpspec_scan_interpret**.

- To initiate a scan:
  – Subscribe to the notification of the characteristics to start scan.
  – Issue a request to start scan and wait for the notification that indicates the scan completed.
  – Subscribe to the notifications of the characteristics to return scan name, scan type, and scan date/time, and to request packet format version.
  – Issue requests for scan name, scan type, scan date/time, and packet format version. Then wait for the notifications to read scan name, scan type, scan date/time, and packet format version.
  – Subscribe to the notification of the characteristic to return serialized scan data structure.
  – Issue a request to read serialized scan data structure and wait for the notification to read the scan data structure. The DLP Spectrum Library provides a routine to interpret this serialized data: **dlpspec_scan_interpret**.
- To delete stored scan data:
  – Subscribe to the notification of the characteristic to clear scan.
  – Issue a request to clear scan and wait for the notification that indicates the clear scan completed.

Figure 5-12 and Figure 5-13 show the workflow for this service.

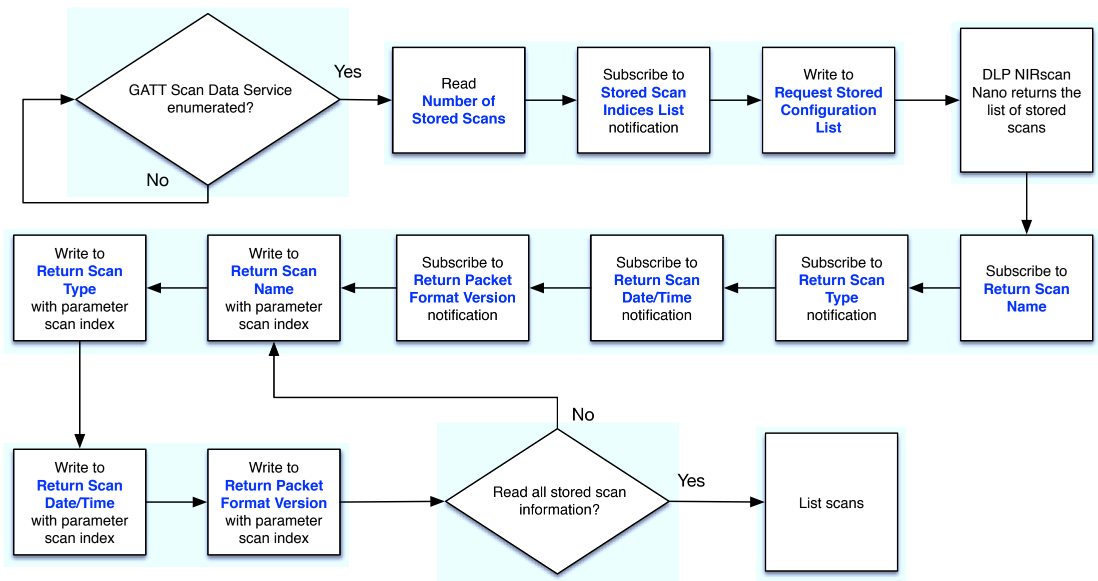


**Figure 5-12. GATT Scan Data Service Workflow**

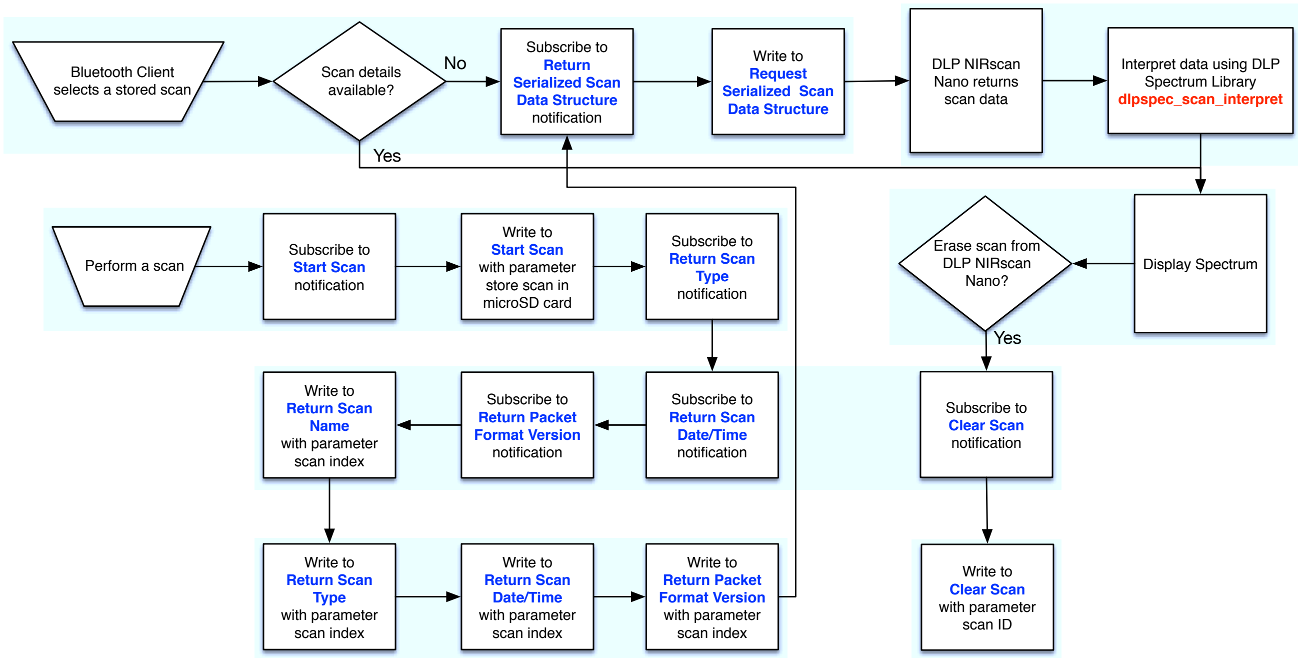


**Figure 5-13. GATT Scan Data Service Workflow to Display an Existing Scan or Performing a New Scan**

# iOS and Android App

## 6.1 NanoScan iOS App

KS Technologies has developed an example iOS and Android app that controls the DLP NIRscan Nano EVM. The NIRscan Nano iOS app is available for download through the Apple App Store free of charge. This app supports BLE iOS devices: iPhone 4S or later and iPad 3 or later with iOS 7.1 or later. The NIRscan Nano Android app is available for download through the Google Play Store free of charge. The next few paragraphs show how to use the iOS app. The ANdroid app has a very similar interface.

After running the NanoScan iOS App, the main screen shown in Figure 6-1 lists the previous scans performed with the NIRscan Nano EVM. First, power the DLP NIRscan Nano by conecting a micro USB cable or an optional battery and wait for the green LED to pulse. Second, enable the Bluetooth by pressing and holding the scan button on the EVM for more than three seconds. The blue LED will light up to indicate that the Bluetooth circuits are powered and actively scanning. If the blue LED does not turn on, but the lamp and yellow LED turned on, the scan button was not held long enough. Once the blue LED is on, press Scan on the top-right corner of the iOS App to initiate a connection. The Bluetooth icon in the top-right corner of the iPhone screen will blink as the connection is established. The NIRscan Nano EVM blue LED will pulse to indicate that a BLE connection was established. Then, the reference and calibration data is downloaded from the EVM. When that is completed, the Start Scan button will be activated.

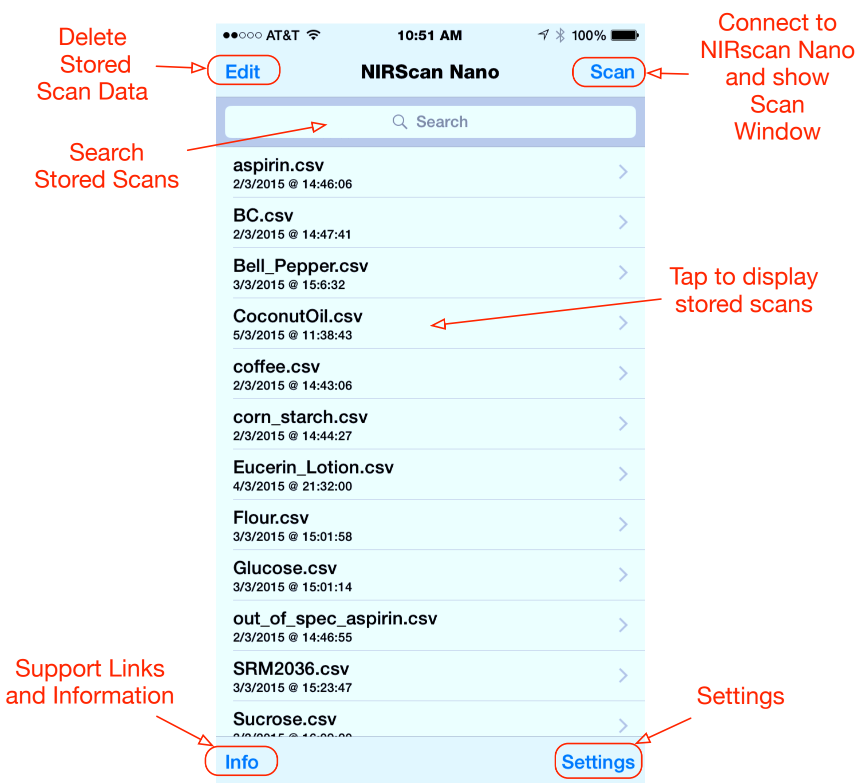


**Figure 6-1. NanoScan iOS App Main Screen**

Pressing the Configure button in the top-right corner of the NanoScan main screen opens the Configuration screen. The Configuration screen displays four buttons:

- Device Information: This button uses the following service:
  - GATT General Information Service to request Manufacture, Model Number, Serial Number, Hardware revision, Tiva firmware version, and Spectrum Library revision.
- Device Status: This button uses the following services:
  - Battery Service to prompt for the Battery voltage. The battery is reported in percentage capacity with 0%, 5%, 20%, 40%, 60%, and 80%.
  - GATT General Information Service to request temperature, humidity, device and error status, and also set threshold for temperature and humidity notifications.
  - GATT General Information Service to report device and error status.
- Scan Configurations: This button uses the following service:
  - GATT Scan Configuration Information Service to display stored scan configurations on the DLP NIRscan Nano. The user can select which stored scan configuration will be used in future scans.
- Stored Scan Data: This button uses the following service:
  - GATT Scan Data Information Service to retrieve stored scan data from the microSD card.

Before starting a scan, the user can set a filename prefix, can elect to also save the scan data on the microSD card, and can choose a default for future scans from the stored scan configurations, as shown in Figure 6-2. Pressing the Start Scan button will start a scan with the selected scan configuration. Once the scan completes, the scan data is transmitted from the NIRScan Nano and plotted. The user can choose to plot absorbance, reflectance, or raw intensity values by tapping on the corresponding button under the plot area. Pinching with two fingers on the graph are allows the user to zoom in the graph.

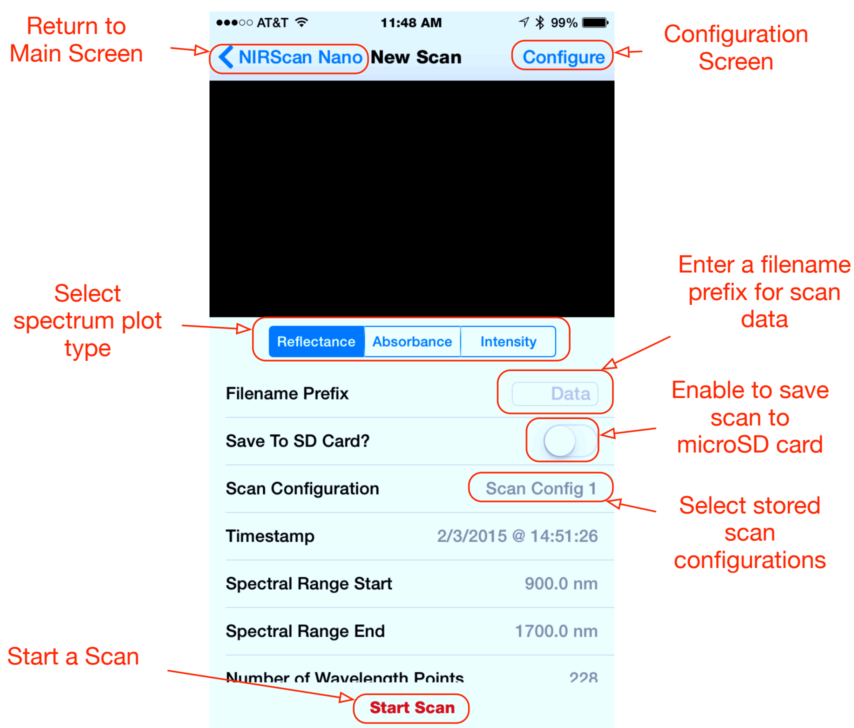


**Figure 6-2. NanoScan iOS App Scan Screen**

Figure 6-3 shows an example plot of absorbance for sugar.


**Figure 6-3. NanoScan iOS App Scan Plot Screen**

## 6.2 DLP Design House Partners with Spectroscopy Options

The following DLP Design House Partners offer spectroscopy options for the DLP NIRscan Nano EVM:

- Young Green Energy Company (YGE) is the third party that manufactures the NIRscan Nano. YGE also has a production modules which are tailored for various sampling techniques. For more information, see this website for details on transmissive sample, fiber coupled input, and fully encased units. If using a fiber couple input, ensure that the slit (1.69 mm by 0.025 mm) is fully illuminated with a round-to-linear fiber bundle.
- Opteks has designed a transmissive sample head for the NIRscan Nano. For more information, see this website for details on transmissive sample head for DLP NIRscan Nano EVM.

For a full list of DLP Design Partners that can implement various instances of DLP-based spectrometers with core expertise in DLP software, electronics, and optical options refer to this page.

# Installing the DLP NIRscan Nano Software

## A.1 DLP NIRscan Nano Software Installation

The NIRscan Nano software is broken into several packages:

- **DLP NIRscan Nano GUI installer**
  - This is the PC program that communicates with the DLP NIRscan Nano through USB.
  - This program installs the PC GUI NIRscanNanoGUI.exe under the default directory: *C:\Program Files\Texas Instruments\NIRscanNanoGUI_X.X.X\Binaries*
  - It also install the sources under the default directory: *C:\ti\NIRscanNanoGUI\Sources*
- **DLP NIRscan Nano Software installer**
  - This program installs the Tiva binary firmware file under the default directory: *C:\ti\DLPNIRscanNanoSoftware_X.X.X\Binaries\NIRscanNano.bin*
  - It also installs the sources under the default directory: *C:\ti\DLPNIRscanNanoSoftware_X.X.X\Sources*
  - This installer includes the TI-RTOS 2.10.1.38 and the Device Firmware Update (DFU) drivers for Tiva. The DFU drivers allow updating the Tiva firmware through USB. If your CCS installation already has TI-RTOS 2.10.1.38, you can skip installing this TI-RTOS package.
- **DLP Spectrum Library installer**
  - This program installs the DLP Spectrum Library. This library code is shared with the PC GUI, Tiva software, and iOS App. The source files are under the default directory: *C:\ti\DLPSpectrumLibrary_X.X.X\src*.
- **DLPR150 PROM installer**
  - This program installs the DLPC150 firmware binary (DLPR150PROM_1.1.0.img) under the default directory: *C:\ti\DLPR150PROM_1.1.0*

To install the software, run these installer executables and follow the on-screen prompts.

# Required Tools to Compile Tiva Software

## B.1 Tiva Tools Installation

To compile the DLP NIRscan Nano Tiva code, the following tools and software packages are required:

- Code Composer Studio (CCS) Integrated Development Environment (IDE) version 6.0.1
- TI-RTOS version 2.10.1.38
- TI ARM Compiler version 5.2.4

### *B.1.1 Code Composer Studio Installation*

To install, follow these steps:

1. Download the latest Windows or Linux version of the CCS IDE for TM4x ARM MCU.
2. Execute the download file: c_setup_win32.exe
3. Accept the License Agreement. Click Next.
4. Select the installation location and click Next. *C:\ti* is the default installation location.
5. If you are running anti-virus software, you will be warned that it is recommended to temporarily disable real-time scanning before proceeding with installation.
6. The installer will prompt for processor support. Ensure that under 32-bit ARM MCUs, Tiva C Series Support and TI ARM Compiler are checked. Click Next.
7. Select the appropriate Debug Probes or Emulators. Tiva C Series are supported by XDS100, XDS200, and XDS560 Debug Probes from TI, Spectrum Digital or Blackhawk.
   - For more information on supported Debug Probes, visit the Tiva Tools & Software page.
   - For emulation information visit the TI Emulation Wiki.
   - For more information on JTAG connectors and adapters visit the XDS Target Connection Guide.
8. If prompted for apps, there is no need to select any apps.
9. After the installation is complete, update CCS by following the steps in Section B.1.2.

### *B.1.2 Updating TI-RTOS*

The current Tiva Firmware build uses TI-RTOS for TivaC 2.10.1.38.

---

**NOTE:** Do not use TI-RTOS for TivaC greater than 2.10.1.38 since the TI-RTOS drivers are different.

---

If you want to install a newer version of TI-RTOS, update TI-RTOS with the following steps after launching Code Composer Studio:

1. Select CCS APP Center from the View Menu.
2. Type "Tiva" in the search box.
3. Update TI-RTOS for TivaC, if offered to update it. To update it, press the select button beneath the TI-RTOS for TivaC icon and then click on the install software icon to the left of the search box under the App Center logo.

TI-RTOS for TivaC 2.10.1.38 is found at this link.

#### B.1.2.1 Updating TI ARM Compiler

To update the TI ARM Compiler, follow these steps after launching Code Composer Studio:

1. Select CCS APP Center from the View Menu.
2. Type "compiler" in the search box.
3. Update TI ARM® Compiler, if offered to update it. To update it, press the select button beneath the TI ARM Compiler icon and then click on the install software icon to the left of the search box under the App Center logo.
4. Update TI ARM Compiler for TivaC.

# How to Compile Tiva Source Code

## C.1 Tiva Libraries Compilation

The DLP NIRscan Nano Tiva software uses two libraries from the TivaWare package: **Tiva driverlib** and **Tiva usblib**. It also uses a third library: **DLP Spectrum Library**. These libraries must be compiled under CCS before building the DLP NIRscan Nano software.

### *C.1.1 Tiva driverlib Compilation*

To compile the TivaWare driverlib library, follow these steps:

1. Import the driverlib library by selecting Import from the File Menu.
2. In the new Import dialog window, select CCS Projects under C/C++ folder and click Next.
3. Find the location of the TivaWare driverlib project by browsing to the directory *C:\ti\tirtos_tivac_2_10_01_38\products\TivaWare_C_Series-2.1.0.12573c\driverlib* and click OK.
4. Compile this newly added driverlib project by selecting Clean from the Project Menu. Ensure that driverlib project is checked and select "Build only the selected project" and then click OK.

### *C.1.2 Tiva usblib Library*

To compile the TivaWare usblib library, follow these steps:

1. Import the usblib library by selecting Import from the File Menu.
2. In the new Import dialog window, select CCS Projects under C/C++ folder and click next.
3. Find the location of the TivaWare usblib project by browsing to the directory *C:\ti\tirtos_tivac_2_10_01_38\products\TivaWare_C_Series-2.1.0.12573c\usblib* and then click OK.
4. Compile this newly added usblib project by select Clean from the Project Menu. Ensure that usblib project is checked and select "Build only the selected project" and then click OK.

### *C.1.3 DLP Spectrum Library*

The DLP Spectrum Library is a collection of C-language routines that provide the fundamental pieces to use a DLP system in a spectroscopy application. These routines are shared by the Tiva firmware, NIRscan Nano GUI, and iOS App. To compile the DLP Spectrum Library for Tiva firmware, follow these steps:

1. Import the DLP Spectrum Library by selecting Import from the File Menu.
2. In the new Import dialog window, select CCS Projects under C/C++ folder and click Next.
3. Find the location of the dlpspeclib project by browsing to the install directory: *C:/ti/DLPNIRNANO_SPECLIB*.
4. Compile this newly added dlpspeclib project by selecting Clean from the Project Menu. Ensure that dlpspeclib project is checked and select "Build only the selected project" and then click OK.

## C.2 Tiva Main Source

The Tiva main program sources are installed by the DLPNIRscanNanoSoftware package. This package installs the Code Composer Studio project and source at the default directory *C:\ti\DLPNIRscanNanoSoftware_vX.X.X*

1. Import the Mobile Spectroscopy Tiva EVM by pulling-down the File Menu and select Import.
2. In the new Import dialog window, select CCS Projects under C/C++ folder and click Next.
3. Find the location of the Mobile Spectroscopy Tiva EVM by browsing to the install directory of the sources: *C:\ti\DLPNIRscanNanoSoftware_X.X.X\Sources*.
4. Make sure all the other libraries are compiled and imported before this step. Then, compile this newly added Mobile Spectroscopy Tiva EVM project by selecting Clean from the Project Menu. Ensure that dlpspeclib project is checked and select "Build only the selected project" and then click OK.

## C.3 Project Settings

The compilation of the Tiva sources and libraries requires the following project settings by right clicking on:

- On Project Browser, select Mobile Spectroscopy Tiva EVM. Right-click and select "Show Build Settings..."
- Under CCS General, select the Main tab. Ensure that the Compiler version is TI v5.2.4 or later.
- Under CCS General, select the RTSC tab. Ensure that TI-RTOS for TivaC is set to 2.10.1.38 and XDCtools version is 3.30.4.52_core.
- Repeat this for all libraries.

# Required Tools to Compile NIRscan Nano GUI

## D.1 NIRscan Nano GUI

The NIRscan Nano GUI requires Qt Framework and tools 5.4.1 or later, MinGW compiler 4.9.1, and the DLP Spectrum Library. The Qt Framework and tools can be downloaded from the Qt website. The following Qt components are needed for the NIRscan Nano GUI compilation:

- Tools -> Qt Creator
- Tools -> MinGW 4.9.2
- Source Components -> Essentials

### *D.1.1 Compiling the DLP Spectrum Library*

The DLP Spectrum Library includes a batch file to compile for the PC. To compile the DLP Spectrum Library for the NIRscan Nano GUI, follow these steps:

1. Open MS-DOS window and change to the src directory of the DLP Spectrum Library: *C:/ti/DLPNIRNANO_SPECLIB/src*
2. Execute the build-lib.bat file. This batch file requires the prior installation of MinGW or a GCC toolchain in the Windows PC with their respective binaries added to the Windows PATH environment variable.

### *D.1.2 Compiling NIRscan Nano GUI*

After compiling the DLP Spectrum Library, compile the NIRscan Nano GUI with the following steps:

1. Run Qt Creator.
2. In Qt Creator, click on "Open Project" button. Navigate to the directory where the NIRscanNanoGUI sources were installed. The default installation directory is *C:\ti\DLPNIRscanNanoGUI_2.0.0\Sources*. Open the project file NirscanNanoGUI.pro
3. A dialog window will indicate that no user settings were found. Click the Yes button.
4. Click the Projects icon on the sidebar. Ensure that the Build settings are correct and that an existing Build directory is set.
5. From the Build menu, select Build All.

# Tiva EEPROM Contents

## E.1 Tiva EEPROM

The Tiva TM4C129XNCZAD microcontroller includes an EEPROM with 6kB of storage. The DLP NIRscan Nano uses this EEPROM to store the following factory information:

- DLP NIRscan Nano Serial Number: A seven digit number in the format YMMSSSS, where Y represents a one digit year of manufacturing number, MM represents a two digit month of manufacturing number, and SSSS represents a four digit serial number
- Scan Data Index Counter: Default scan
- Calibration Coefficients Data Structure Version Number
- Calibration Coefficients Data
- Reference Calibration Data Structure Version Number
- Reference Calibration Data
- Default Scan Name
- Default Scan Configuration
- Active Scan Configuration Number and Index
- Scan Configuration Data Structure Version
- Scan Configurations

See Table E-1 for the address, size, and content of these information in Tiva EEPROM.

**Table E-1. Tiva EEPROM**

| ADDRESS | SIZE (BYTES) | DESCRIPTION |
|---|---|---|
| 0x0000 | 8 | DLP NIRscan Nano Serial Number |
| 0x0008 | 4 | Scan Data Session Index |
| 0x000C | 4 | Scan Configuration Index Counter |
| 0x0010 | 4 | Calibration Coefficients Data Structure Version Number |
| 0x0014 | 50 | Calibration Coefficients Data |
| 0x0046 | 4 | Reference Calibration Data Structure Version Number |
| 0x004A | 3632 | Reference Calibration Data |
| 0x0E7A | 16 | Default Scan Name |
| 0x0E7E | 4 | Active Scan Configuration Number and Index |
| 0x0E82 | 4 | Scan Configuration Data Structure Version |
| 0x0E96 | 2154 | Scan Configurations (20 Scan Configuration entries, with 20 bytes per Scan Configuration section) |
| 0x1700 | 4 | Battery Calibration Data |
| 0x1704 | 16 | Model Name |
| 0x1714 | 108 | Reserved for Application Data Storage |
| 0x1780 | 128 | Reserved for Manufacturer Data Storage |

# DLP NIRscan Nano Connectors

## F.1 Battery Connector

The battery power (J6) connector of the microprocessor board requires the following 2-pin, 2-mm connector part numbers:

- JST part number: PHR-2
- Digi-Key part number: 455-1165-ND

The corresponding connector terminal (crimp) part numbers are:

- JST part number: SPH-002T-P0.5L
- Digi-Key part number: 455-2148-1-ND

**Table F-1. Battery Power Connector (Tiva J6)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Power positive | 1 | 4.2 V |
| Ground | 2 | Ground |

**NOTE:** The preferred connection to the battery may be a direct solder type or any low resistance contact connection.

## F.2 Battery Thermistor Connector

The battery thermistor connector (J7) of the microprocessor board requires the following 2-pin, 1-mm connector part numbers:

- JST part number: SHR-02V-S-B
- Digi-Key part number: 455-1377-ND

The corresponding connector terminal (crimp) part numbers are:

- JST part number: SSH-003T-P0.2
- Digi-Key part number: 455-1561-1-ND

**Table F-2. Battery Thermistor Connector (Tiva J7)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Power positive | 1 | 4.9 V |
| Ground | 2 | Ground |

## F.3 Expansion Connector

The expansion connector (J3) of the microprocessor board requires the following 10-pin, 1-mm connector part numbers:

- JST part number: SHR-10V-S-B
- Digi-Key part number: 455-1385-ND

The corresponding connector terminal (crimp) part numbers are:

- JST part number: SSH-003T-P0.2
- Digi-Key part number: 455-1561-1-ND

**Table F-3. Expansion Connector (Tiva J3)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Power | 1 | 3.3 V |
| Ground | 2 | Ground |
| Tiva PA2 (GPIO, UART4 receive or SSI0 clock) | 3 | 3.3 V |
| Tiva PA3 (GPIO, UART4 transmit or SSI0 frame sync) | 4 | 3.3 V |
| Tiva PA4 (GPIO or SSI0 Data0) | 5 | 3.3 V |
| Tiva PA5 (GPIO or SSI0 Data1) | 6 | 3.3 V |
| Tiva PK2 (GPIO or UART4RTS) | 7 | 3.3 V |
| Tiva PK3 (GPIO or UART4 CTS) | 8 | 3.3 V |
| Ground | 9 | Ground |
| Tiva wake | 10 | 3.3 V |

## F.4 JTAG Connector

The ARM Cortex 10-pin JTAG connector (J4) of the Microprocessor Board requires an adapter to interface to the standard TI 14-pin and 20-pin emulators. Refer to the TI JTAG Connector wiki page for more information.

**Table F-4. ARM Cortex 10-pin JTAG Connector (Tiva J4)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Power | 1 | 3.3 V |
| TMS | 2 | 3.3 V |
| Ground | 3 | Ground |
| TCK | 4 | 3.3 V |
| Ground | 5 | Ground |
| TDO | 6 | 3.3 V |
| Key (no connect) | 7 | — |
| TDI | 8 | 3.3 V |
| Ground | 9 | Ground |
| RESETz | 10 | 3.3 V |

## F.5 Trigger Connector

The trigger connector (J500) of the DLPC150 board requires the following 9-pin, 1-mm connector part numbers:

- JST part number: SHR-09V-S-B
- Digi-Key part number: SHR-09V-S-B

The corresponding connector terminal (crimp) part numbers are:

- JST part number: SSH-003T-P0.2
- Digi-Key part number: 455-1561-1-ND

Note that this connector is on the top side of the DLPC150 board covered by the plastic cover. To access this connector, the microprocessor and DLPC150 boards must be disassembled from the optical module. Do not remove the top cover because it protects the optical module from dust and keeps the lenses in place.

**Table F-5. Trigger Connector (DLPC150 J500)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Tiva PD2 (lamp control) | 1 | 3.3 V |
| TRIG_IN_1 | 2 | 3.3 V |
| Ground | 3 | Ground |
| DLPC150 GPIO_17 | 4 | 3.3 V |
| DLPC150 GPIO_18 | 5 | 3.3 V |
| Ground | 6 | Ground |
| TRIG_OUT_2 (frame trigger) | 7 | 3.3 V |
| TRIG_OUT_1 (pattern trigger) | 8 | 3.3 V |
| Ground | 9 | Ground |

## F.6 Lamp Connector

The lamp connector (J503) of the DLPC150 board provides power to the lamps. It requires the following 8-pin, 1-mm connector part numbers:

- JST part number: SHR-08V-S-B
- Digi-Key part number: SHR-08V-S-B

The corresponding connector terminal (crimp) part numbers are:

- JST part number: SSH-003T-P0.2
- Digi-Key part number: 455-1561-1-ND

**Table F-6. Lamp Connector (DLPC150 J503)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Lamp Power | 1 | 5.0 V |
| Lamp Power | 2 | 5.0 V |
| Lamp Power | 3 | 5.0 V |
| Lamp Power | 4 | 5.0 V |
| Ground | 5 | Ground |
| Ground | 6 | Ground |
| Ground | 7 | Ground |
| Ground | 8 | Ground |

## F.7 Lamp Photodetector Connector

The lamp photodetector connector (J501) of the DLPC150 board provides power and sense for lamp photodetector. Only one photodetector is installed and supported by the system. It requires the following 5-pin, 1-mm connector part numbers:

- JST part number: SHR-05V-S-B
- Digi-Key part number: SHR-05V-S-B

The corresponding connector terminal (crimp) part numbers are:

- JST part number: SSH-003T-P0.2
- Digi-Key part number: 455-1561-1-ND

**Table F-7. Lamp Photodetector Connector (DLPC150 J501)**

| DESCRIPTION | PIN | SUPPLY RANGE |
|---|---|---|
| Photodetector1 Power | 1 | 3.3 V |
| Photodetector2 Power | 2 | 3.3 V |
| Photodetector1 Sense | 3 | 3.3 V |
| Photodetector2 Sense | 4 | 3.3 V |
| Ground | 5 | Ground |

# DLP NIRscan Nano Command Description

## G.1 Command Structure

The DLP NIRscan Nano supports a set of commands through USB, UART, and Bluetooth interfaces. Each interface has a different command packet with the USB interface described in Appendix H, the UART interface described in Appendix I, and the Bluetooth interface described in Appendix J. This section list the individual commands supported by these interfaces.

## G.2 Interface Priority

If multiple interfaces are active, the priority between interfaces is set so that UART has highest priority, followed by Bluetooth, and then USB. Only commands from the highest priority interface will be interpreted. For example, activating and connecting a NIRscan Nano to an iOS device through Bluetooth when the system is connected to the GUI through USB, will interpret commands sent through Bluetooth. Once Bluetooth is disconnected, the NIRscan Nano will interpret commands from USB. If UART connected, all commands from UART will be interpreted while commands from Bluetooth and USB will be ignored.

An interface is considered active under the following conditions:

- USB Interface is active when a USB cable is connected and NIRscan Nano enumerates to a PC.
- Bluetooth is active when a BLE device connects to NIRscan Nano.
- UART is active when an external device sends a packet of data through the J3 Expansion Connector.

Interface priority is programmable and defined through the connection_type enumeration type in the cmdHandlerFmgr.h file.

## G.3 Command Handler Supported Commands

The DLP NIRscan Nano Command Handler supports a set of commands described in Table G-1. The DLP NIRscan Nano Tiva software defines these sets of commands and their enumerated parameters in the Common/include/NNOCommandDefs.h file.

**Table G-1. DLP NIRscan Nano Supported Commands Per Interface**

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Perform DLPC150 firmware file checksum | NNO_CMD_FLASH_GET_CHKSUM | 0 | | 4 | Checksum Value | ● | | |
| Write file command pair. Write the size, followed by the data write. | NNO_CMD_FILE_SET_WRITESIZE | 6 | File Size (4)<br>File Action (2):<br>NNO_FILE_DLPC_UPDATE<br>NNO_FILE_REFCAL_DATA | 0 | | ● | | |
| | NNO_CMD_FILE_WRITE_DATA | 0 | | 0 | | ● | | |
| Read list of scan configurations or scans stored in the SD card. | NNO_CMD_READ_FILE_LIST | 1 | NNO_FILE_SCAN_CONFIG_LIST | Variable | Requested data | ● | ● | |
| Read file list size | NNO_CMD_READ_FILE_LIST_SIZE | 1 | NNO_FILE_SCAN_LIST | 4 | Size in bytes of requested data | ● | ● | |
| Read file command pair. Read the file size, followed by the file data read. | NNO_CMD_FILE_GET_READSIZE | 1 | NNO_FILE_SCAN_DATA<br>NNO_FILE_SCAN_CONFIG<br>NNO_FILE_REF_CAL_COEFF<br>NNO_FILE_REF_CAL_MATRIX | 4 | Size in bytes of requested data | ● | ● | |
| | NNO_CMD_FILE_GET_DATA | 0 | | Variable | Requested data | ● | ● | |
| Switch Tiva to bootloader mode | NNO_CMD_GOTO_TIVA_BL | 0 | | 0 | | ● | | |
| EEPROM test | NNO_CMD_EEPROM_TEST | 0 | | 1 | Result | ● | ● | |
| Detector board test | NNO_CMD_ADC_TEST | 0 | | 1 | 0 = Pass | ● | ● | |
| Battery charger test | NNO_CMD_BQ_TEST | 0 | | 1 | -1 = Failure | ● | ● | |
| SDRAM test | NNO_CMD_SDRAM_TEST | 0 | | 1 | | ● | ● | |
| DLP Controller and Lamp power up | NNO_CMD_DLPC_ENABLE | 2 | Start Byte:<br>0 = Disable DLP Subsystem<br>1 = Enable DLP Subsystem<br>Lamp Enable Byte:<br>0 = Power off Lamps<br>1 = Power on Lamps | 0 | | ● | ● | |
| Temperature sensor test | NNO_CMD_TMP_TEST | 0 | | 1 | Result: | ● | ● | |
| Humidity sensor test | NNO_CMD_HDC_TEST | 0 | | 1 | | ● | ● | |
| Bluetooth test | NNO_CMD_BT_TEST | 1 | Start | 1 | 0 = Pass | ● | ● | |
| microSD Card test | NNO_CMD_SDC_TEST | 1 | 0 = End Test | 1 | | ● | ● | |

**Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)**

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| LED test | NNO_CMD_LED_TEST | 1 | 1 = Start Test | 1 | -1 = Failure | • | • | |
| Read button pressed during button test | NNO_CMD_BUTTON_TEST_RD | 1 | Button to test:<br>0 = No Button<br>1 = Scan Button<br>2 = On/Off Button | 1 | | • | • | |
| Set button to test | NNO_CMD_BUTTON_TEST_WR | 1 | | 1 | | • | • | |
| FACTORY USE ONLY: Write EEPROM Calibration Coefficient, Scan Configuration, and Reference Calibration Coefficients Versions (1) | NNO_CMD_EEPROM_CAL_TEST | 0 | | 1 | Result:<br>0 = Pass<br>1 = Failure | • | • | |
| Read Tiva version information | NNO_CMD_TIVA_VER | 0 | | 28 | Tiva SW Version (4)<br>DLPC SW Version (4)<br>DLPC Flash Version (4)<br>DLP Spectrum Library Version (4)<br>EEPROM Calibration Version (4)<br>EEPROM Reference Version (4)<br>EEPROM Scan Configuration Version(4) | • | • | DIS(2) |
| Write patterns in SDRAM for testing | NNO_CMD_STORE_PTN_SDRAM | 12 | Size in Bytes (4) | 0 | | • | • | |
| Start scan | NNO_CMD_PERFORM_SCAN | 1 | Store in SD:<br>0 = Do not store scan in SD Card<br>1 = Store scan in SD Card | 0 | | • | • | GSDIS(8) |
| Scan status | NNO_CMD_SCAN_GET_STATUS | 0 | | 1 | Scan Status:<br>0 = Scan in progress<br>1 = Scan complete | • | • | GGIS(4) |
| Reset Tiva | NNO_CMD_TIVA_RESET | 0 | | 0 | | • | • | GCS(9) |
| Write ADC PGA gain | NNO_CMD_SET_PGA | 1 (unsigned) | PGA Value:<br>PGA = 1X<br>PGA = 2X<br>PGA = 4X<br>PGA = 8X<br>PGA = 16X<br>PGA = 32X<br>PGA = 64X | 0 | | • | • | GCS(9) |

**Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)**

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| FACTORY USE ONLY: Write DLPC150 Register (1) | NNO_CMD_SET_DLPC_REG | 8 | Address (4)<br>Value (4) | 0 | | • | • | |
| Read DLPC150 register | NNO_CMD_GET_DLPC_REG | 4 | Address | 4 | Value | • | • | GCS(9) |
| Send scan configuration and generate patterns | NNO_CMD_SCAN_CFG_APPLY | Variable | Serialized Scan Configuration generated by dlpspec_scan_write_configuration() | 4 | Number of patterns | • | • | |
| Save given scan configuration in EEPROM | NNO_CMD_SCAN_CFG_SAVE | Variable | Index (1)<br>Buffer Size (1)<br>Serialized Scan Configuration (124) | 0 | | • | • | |
| Read stored scan configurations | NNO_CMD_SCAN_CFG_READ | 2 | Scan Configuration Index (1)<br>Buffer Size (1) | 124 | Returns Scan Configuration in serialized form indicated by the Index passed on this command | • | • | GSCIS(7) |
| Erase all stored scan configurations | NNO_CMD_SCAN_CFG_ERASEALL | 0 | | 0 | | • | • | GCS(9) |
| Read number of stored scan configurations | NNO_CMD_SCAN_CFG_NUM | 0 | | 1 | Number of Scan Configurations | • | • | GSCIS(7) |
| Read active scan configuration | NNO_CMD_SCAN_GET_ACT_CFG | 0 | | 1 | Active Scan Configuration Index | • | • | |
| Set active scan configuration and generate parameters | NNO_CMD_SCAN_SET_ACT_CFG | 1 | Scan Configuration Index | 0 | | • | • | |
| DLPC150 and lamp power control after each scan command | NNO_CMD_SET_DLPC_ONOFF_CTRL | 1 | ON/OFF Control:<br>0 = DLPC150 and lamp turn off after each scan<br>1 = DLPC150 and lamp stay on after each scan | 0 | | • | • | |
| Set scan subimage | NNO_CMD_SET_SCAN_SUBIMAGE | 4 | Start Y (2)<br>Height (2) | 0 | | • | • | GCS(9) |
| FACTORY USE ONLY: Erase Calibration Coefficients, Reference Calibration Coefficients, and Scan Configurations stored in EEPROM (1) | NNO_CMD_EEPROM_WIPE | 3 | Erase EEPROM Calibration (1)<br>0 = Do not erase<br>1 = Erase<br>Erase EEPROM Reference (1)<br>0 = Do not erase<br>1 = Erase<br>Erase EEPROM Scan Configuration (1)<br>0 = Do not erase<br>1 = Erase | 0 | | • | • | |

### Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Read PGA setting | NNO_CMD_GET_PGA | 0 | | 1 (unsigned) | PGA Value:<br>PGA = 1X<br>PGA = 2X<br>PGA = 4X<br>PGA = 8X<br>PGA = 16X<br>PGA = 32X<br>PGA = 64X | • | • | GCS(9) |
| FACTORY USE ONLY: Write spectrum calibration coefficients (1) | NNO_CMD_CALIB_STRUCT_SAVE | 144 | Calibration Coefficients | 0 | | • | • | |
| Read spectrum calibration coefficients | NNO_CMD_CALIB_STRUCT_READ | 0 | | 144 | Calibration Coefficients | • | • | GCIS(6) |
| Prepare DLP NIRscan Nano for Column SNR data capture during next scan. Scan configuration for next scan must have SNR_PATTERNS number of patterns, as defined in NNOSNRDefs.h | NNO_CMD_START_SNRSCAN | 0 | | 0 | | • | • | GCS(9) |
| Read SNR calculation results | NNO_CMD_SAVE_SNRDATA | 0 | | 20 | Returns SNR Results concatenated in the snrData and snrDataHard structures defined in NNOSNRDefs.h<br>SNR for Column Scan 17 ms<br>SNR for Column Scan 100 ms<br>SNR for Column Scan 500 ms<br>SNR for Hadamard Scan 120 ms<br>SNR for Hadamard Scan 1 s | • | • | GCS(9) |
| Generate calibration pattern | NNO_CMD_CALIB_GEN_PTNS | 1 | Scan Type argument from SCAN_TYPE enum in dlpspec_scan.h<br>0 = COLUMN_TYPE<br>1 = HADAMARD_TYPE<br>2 = SLEW TYPE | 4 | Number of patterns | • | • | GCS(9) |
| Write number of scan repeats | NNO_CMD_SCAN_NUM_REPEATS | 2 (unsigned) | Number of Repeats | 0 | | • | • | GCS(9) |

**Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)**

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Prepare DLP NIRscan Nano for Hadamard SNR data capture during next scan. Scan configuration for next scan must have SNR_HAD_PATTERNS number of patterns, as defined in NNOSNRDefs.h | NNO_CMD_START_HADSNRSCAN | 0 | | 0 | | • | • | GCS(9) |
| Store last scan data in EEPROM as reference calibration data | NNO_CMD_REFCAL_PERFORM | 0 | | 0 | | • | • | GCS(9) |
| FACTORY USE ONLY: Write Device Serial Number and write Column and Hadamard Default Scan configurations (1) | NNO_CMD_SERIAL_NUMBER_WRITE | 8 | Serial Number | 0 | | • | • | |
| Read device serial number | NNO_CMD_SERIAL_NUMBER_READ | 0 | | 8 | Serial Number | • | • | DIS (2) |
| Set scan name tag | NNO_CMD_WRITE_SCAN_NAME_TAG | Variable | Length (1)<br>Name Tag (Length) | 0 | | • | • | GSDIS(8) |
| Erase specified scan data from microSD card | NNO_CMD_DEL_SCAN_FILE_SD | 4 | Index of Scan | 0 | | • | • | GSDIS(8) |
| FACTORY USE ONLY: Mass erase Tiva EEPROM (1) | NNO_CMD_EEPROM_MASS_ERASE | 0 | | 0 | | • | • | |
| Read estimated scan time of current scan configuration | NNO_CMD_READ_SCAN_TIME | 0 | | 4 | Scan Time in milliseconds | • | • | Scan Time is sent as part of Scan Start Notification |
| Delete last Scan from SD Card | NNO_CMD_DEL_SCAN_FILE_SD | 0 | | 0 | | • | • | |
| Start Tiva scan interpretation process | NNO_CMD_START_SCAN_INTERPRET | 0 | | 0 | | • | • | |
| Tiva scan interpretation status | NNO_CMD_SCAN_INTERPRET_GET_STATUS | 0 | | 1 | 0 = Scan interpretation in progress<br>1 = Scan interpretation completed | • | • | |

## Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Read temperature (TMP006) sensor | NNO_CMD_READ_TEMP | 0 | | 8 | Ambient Temperature °C in hundredths(12) (4)<br>Detector Temperature °C in hundredths(12) (4) | • | • | GGIS(4) |
| Read humidity (HDC1000) sensor | NNO_CMD_READ_HUM | 0 | | 8 | HDC Temperature °C in hunderdths(12) (4)<br>HDC Humidity % in hunderedths(12)(4) | • | • | GGIS(4) |
| Set time and date | NNO_CMD_SET_DATE_TIME | 7 | Year, Range 0-99 (1)<br>Month, Range 0-11 (1)<br>Date, 1-31 (1)<br>Day of Week, 0-6 (1)<br>Hour, 0-23 (1)<br>Minute, 0-59 (1)<br>Second, 0-59 (1) | 0 | | • | • | GDTS(5) |
| Read battery voltage | NNO_CMD_READ_BATT_VOLT | 0 | | 4 | Battery voltage in hundredths of volts (4-byte integer) | • | • | |
| | | | | 1 | Battery voltage in percentage (1-byte integer) | | | BAS(3) |
| Read Tiva internal temperature | NNO_CMD_READ_TIVA_TEMP | 0 | | 4 | Floating point temperature in °C | • | • | GCS(9) |
| Get time and date | NNO_CMD_GET_DATE_TIME | 0 | | 7 | Year (1)<br>Month (1)<br>Date (1)<br>Day of Week (1)<br>Hour (1)<br>Minute (1)<br>Second (1) | • | • | GDTS(5) |
| Set Tiva in low power mode | NNO_CMD_HIBERNATE_MODE | 0 | | 0 | | • | • | GCS(9) |
| Set Hibernate Mode flag(10) | NNO_CMD_SET_HIBERNATE | 1 | If set, Tiva will hibernate after period of inactivity | 0 | | • | | GCS(9) |
| Get Hibernate Mode flag(10) | NNO_CMD_GET_HIBERNATE | 0 | | 1 | 0 = Hibernation mode flag off<br>1 = Hibernation mode flag on | • | | GCS(9) |
| Read number of scan files stored in microSD Card | NNO_CMD_GET_NUM_SCAN_FILES_SD | 0 | | 4 | Number of Scan Files on SD Card | • | • | GSDIS(8) |

## Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Read Photodetector | NNO_CMD_READ_PHOTODETECTOR | 0 | | 12 | DLPA2005 Red Photodetector (4)<br>DLPA2005 Green Photodetector (4)<br>DLPA2005 Blue Photodetector (4) | • | • | GCS(9) |
| Read Device Status | NNO_CMD_READ_DEVICE_STATUS | 0 | | 4 | Device Status:<br>0x01 = Tiva Active<br>0x02 = Scan in Progress<br>0x04 = SD Card Present<br>0x08 = SD Card I/O Access in Progress<br>0x10 = Bluetooth Active<br>0x20 = Blutooth Connected<br>0x40 = Scan Interpretation in Progress | • | • | |

## Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Read Error Status | NNO_CMD_READ_ERROR_STATUS(11) | 0 | | 20 | Error Status (4)<br>00x01 = Scan Error<br>0x002 = ADC Error<br>0x004 = SD Card Error<br>0x008 = EEPROM Error<br>0x010 = Bluetooth Error<br>0x020 = Spectrum Library Error<br>0x040 = Hardware Error<br>0x080 = TMP006 Error<br>0x100 = HDC1000 Error<br>0x200 = Battery Discharged<br>0x400 = Memory Error<br>0x800 = UART Error | • | • | |
| | | | | | Scan Error Description (1) | | | |
| | | | | | ADC Error Description (1) | | | |
| | | | | | SD Card Error Description (1) | | | |
| | | | | | EEPROM Error Description (1) | | | |
| | | | | | Bluetooth Error Description (2) | | | |
| | | | | | Spectrum Library Error Description (1) | | | |
| | | | | | Hardware Error Description (1) | | | |
| | | | | | TMP006 Error Description (1) | | | |
| | | | | | HDC Error Description (1) | | | |
| | | | | | Battery Error Description (1) | | | |
| | | | | | Memory Error Description (1) | | | |
| | | | | | UART Error Description (1) | | | |
| Clear Error Status | NNO_CMD_RESET_ERROR_STATUS | 0 | | 0 | | • | • | |
| Read Specific Error Status | NNO_CMD_GET_SPECIFIC_ERR_STATUS | 4 | Error Status Field | 0 | | • | • | |
| Read Specific Error Code | NNO_CMD_GET_SPECIFIC_ERR_CODE | 2 | Error Code | 0 | | • | • | |
| Clear Specific Error | NNO_CMD_CLEAR_SPECIFIC_ERR | 4 | Error Status Field | 0 | | • | • | |

**Table G-1. DLP NIRscan Nano Supported Commands Per Interface (continued)**

| Description | Command ID | Data Payload Input | | Data Payload Output | | Communications Interface | | |
|---|---|---|---|---|---|---|---|---|
| | | Length (Bytes) | Parameters (Bytes) | Length (Bytes) | Parameters (Bytes) | USB | UART | Bluetooth |
| Modify current scan data on DLP NIRscan Nano by dividing by whiteout reflectance. Used prior to saving reference calibration data when taking a scan with whiteout instead of spectralon reflective material. | NNO_CMD_UPDATE_REFCALDATA_WOREFL | 0 | | 0 | | • | | GCS(9) |
| Erase DLPC150 Serial Flash | NNO_CMD_ERASE_DLPC_FLASH | 0 | | 0 | | • | | |
| Write device model name | NNO_CMD_MODEL_NAME_WRITE | 0 | | 16 | Model Name | • | • | |
| Read device model name | NNO_CMD_MODEL_NAME_READ | 16 | Model Name | 0 | | • | • | DIS (2) |

(1) Commands highlighted in red are for factory calibration. Erasing the factory calibration data will render a unit non-functional and will need to be recalibrated.

(2) Refer to Table J-1.

(3) Refer to Table J-2.

(4) Refer to Table J-3.

(5) Refer to Table J-4.

(6) Refer to Table J-5.

(7) Refer to Table J-6.

(8) Refer to Table J-7.

(9) Refer to Table J-8.

(10) Command supported in Tiva Firmware version 1.2 or later.

(11) Refer to NNOStatusDef.h in the Common/include directory for detail description of the Eror Codes

(12) Temperature and Humidity values are reported as 4-byte integers in hundredths. For example, a temperature of 25.75 °C is reported as 2575. Thus, divide the reported value by 100.

# DLP NIRscan Nano USB Communications

## H.1 USB Communications

The DLP NIRscan Nano communicates using USB 1.1 human interface device (HID) to exchange commands and data with a host processor or PC. The USB commands are variable length data packets that are sent with the least significant byte first. The maximum HID packet length is 64 bytes.

The DLP NIRscan Nano enumerates as a Texas Instruments HID device with vendor ID = 0x0451 and product ID = 0x4200.

### *H.1.1 USB Transaction Sequence*

The USB 1.1 HID protocol has the structure shown in Table H-1:

**Table H-1. USB HID Protocol Transaction Sequence**

| Byte | Type | Field | Description |
|---|---|---|---|
| 0 | USB Header | ID | Byte set to 0 |
| 1 | Data Header | Flags | R/W (bit 7)<br>0 = Write<br>1 = Read |
| | | | Ready (bit 6)<br>0 = Busy<br>1 = Ready |
| | | | Error (bit 5:4)<br>0 = Success<br>1 = Error |
| | | | 2 = Busy |
| | | | Reserved (bits 3:0) |
| 2 | | Sequence | Number of command packet |
| 3 | | Length | Length LSB of this command packet |
| 4 | | | Length MSB of this command packet |
| 5 | Data Payload | Command | Command Byte |
| 6 | | | Group Byte |
| 7 | | Data | Data Byte 0 |
| ... | | | ... |
| N + 7 | | | Data Byte 1 |

#### H.1.1.1 USB Read Transaction Sequence

To issue a read command, the host must perform the following steps:

1. Host sends the ID byte, which is set to 0.
2. Host sends the Flags byte, where:
   - Bit 6 is set to 0x1 to indicate the host wants a reply from the device.
   - Bit 7 is set to 0x1 to indicate a read transaction.

3. Host sends the Sequence byte. When a single command is more than 64 bytes, it is sent as multiple USB or UART packets and the sequence byte is used to number the packets so the device can assemble them in the right sequence. Otherwise, this value is irrelevant and generally set to 0.
4. Host sends two bytes with the length of the data packet. This length denotes the number of data bytes in the packet and excludes the number of bytes in steps 1 through 4. It denotes the total number of bytes sent in steps 5 (command bytes) and 6 (data bytes).
5. Host sends two Command bytes: Command Byte 0 and Command Byte 1.
6. Host sends data appropriate to command.
7. After completion of this command, DLP NIRscan Nano responds with a packet that includes:
   (a) Byte with the command requested by the host (the matching Sequence byte)
   (b) Length of the data packet
   (c) Data requested

#### H.1.1.2 USB Write Transaction Sequence

To issue a write command, the host must perform the following steps:

1. Host sends the ID byte, which is set to 0.
2. Host sends the Flags byte, where
   - Bit 6 is set to 0x1 to indicate the host wants a reply from the device. The device responds with the NACK bit set if an unknown command was received or there was any error in processing the command due to invalid parameters or other reasons..
   - Bit 7 is set to 0x1 to indicate a read transaction
3. Host sends the Sequence byte. When a single command is more than 64 bytes, it is sent as multiple USB packets and the sequence byte is used to number the packets so the device can assemble them in the right sequence. In other cases, this value is irrelevant and generally set to 0.
4. Host sends two bytes with the length of the data packet. This length denotes the number of data bytes in the packet and excludes the number of bytes in steps 1 through 4. It denotes the total number of bytes sent in steps 5 (command bytes) and 6 (data bytes).
5. Host sends three Command bytes: Command Byte 0 and Command Byte 1.
6. Host sends data appropriate to command.
7. After completion of this command, DLP NIRscan Nano responds with a packet that includes a byte with the command requested by the host. This occurs only if bit 6 was set in the Flags byte.

### H.1.2 USB Commands

Table H-2 lists all the DLP NIRscan Nano supported USB commands.

**Table H-2. DLP NIRscan Nano USB Commands**

| Description | Command ID | USB Communications Interface | | | | |
|---|---|---|---|---|---|---|
| | | R / W | Command | | Data Payload[2] | |
| | | | Group Byte | Command Byte | Input (Bytes) | Output (Bytes) |
| Perform file checksum | NNO_CMD_FLASH_GET_CHKSUM | READ | 0x00 | 0x15 | 0 | 4 |
| Write file data | NNO_CMD_FILE_WRITE_DATA | WRITE | 0x00 | 0x25 | 0 | 0 |
| Write file payload information | NNO_CMD_FILE_SET_WRITESIZE | WRITE | 0x00 | 0x2A | 6 | 0 |
| Read file list size | NNO_CMD_READ_FILE_LIST_SIZE | READ | 0x00 | 0x2B | 1 | 4 |
| Read file list | NNO_CMD_READ_FILE_LIST | READ | 0x00 | 0x2C | 1 | Variable |
| Read file size | NNO_CMD_FILE_GET_READSIZE | READ | 0x00 | 0x2D | 1 | 4 |
| Read file data | NNO_CMD_FILE_GET_DATA | READ | 0x00 | 0x2E | 0 | Variable |
| Update Tiva firmware | NNO_CMD_GOTO_TIVA_BL | WRITE | 0x00 | 0x2F | 0 | 0 |
| EEPROM test | NNO_CMD_EEPROM_TEST | READ | 0x01 | 0x01 | 0 | 1 |
| Detector board test | NNO_CMD_ADC_TEST | READ | 0x01 | 0x02 | 0 | 1 |
| Battery charger test | NNO_CMD_BQ_TEST | READ | 0x01 | 0x03 | 0 | 1 |
| SDRAM test | NNO_CMD_SDRAM_TEST | READ | 0x01 | 0x04 | 0 | 1 |
| DLP Controller power up | NNO_CMD_DLPC_ENABLE | WRITE | 0x01 | 0x05 | 2 | 0 |
| Temperature sensor test | NNO_CMD_TMP_TEST | READ | 0x01 | 0x06 | 0 | 1 |
| Humidity sensor test | NNO_CMD_HDC_TEST | READ | 0x01 | 0x07 | 0 | 1 |
| Bluetooth test | NNO_CMD_BT_TEST | WRITE | 0x01 | 0x08 | 1 | 1 |
| microSD Card test | NNO_CMD_SDC_TEST | READ | 0x01 | 0x09 | 1 | 1 |
| LED test | NNO_CMD_LED_TEST | READ | 0x01 | 0x0B | 1 | 1 |
| Read button test | NNO_CMD_BUTTON_TEST_RD | READ | 0x01 | 0x0C | 1 | 1 |
| Write button test | NNO_CMD_BUTTON_TEST_WR | WRITE | 0x01 | 0x0D | 1 | 1 |
| FACTORY USE ONLY: Write EEPROM Calibration Coefficient, Scan Configuration, and Reference Calibration Coefficients Versions [1] | NNO_CMD_EEPROM_CAL_TEST | WRITE | 0x01 | 0x0E | 0 | 1 |
| Read Tiva version information | NNO_CMD_TIVA_VER | READ | 0x02 | 0x16 | 0 | 28 |
| Write patterns in SDRAM | NNO_CMD_STORE_PTN_SDRAM | WRITE | 0x02 | 0x17 | 12 | 0 |
| Start scan | NNO_CMD_PERFORM_SCAN | WRITE | 0x02 | 0x18 | 1 | 0 |
| Scan status | NNO_CMD_SCAN_GET_STATUS | READ | 0x02 | 0x19 | 0 | 1 |
| Reset Tiva | NNO_CMD_TIVA_RESET | WRITE | 0x02 | 0x1A | 0 | 0 |
| Write ADC PGA gain | NNO_CMD_SET_PGA | WRITE | 0x02 | 0x1B | 1 | 0 |
| FACTORY USE ONLY: Write DLPC150 Register [1] | NNO_CMD_SET_DLPC_REG | WRITE | 0x02 | 0x1C | 8 | 0 |
| Read DLPC150 register | NNO_CMD_GET_DLPC_REG | READ | 0x02 | 0x1D | 4 | 4 |
| Generate patterns | NNO_CMD_SCAN_CFG_APPLY | WRITE | 0x02 | 0x1E | Variable | 4 |
| Save given scan configuration in EEPROM | NNO_CMD_SCAN_CFG_SAVE | WRITE | 0x02 | 0x1F | Variable | 0 |
| Read stored scan configurations | NNO_CMD_SCAN_CFG_READ | READ | 0x02 | 0x20 | 2 | Variable |
| Erase all stored scan configurations | NNO_CMD_SCAN_CFG_ERASEALL | WRITE | 0x02 | 0x21 | 0 | 0 |
| Read number of stored scan configurations | NNO_CMD_SCAN_CFG_NUM | READ | 0x02 | 0x22 | 0 | 1 |
| Read active scan configuration | NNO_CMD_SCAN_GET_ACT_CFG | READ | 0x02 | 0x23 | 0 | 1 |
| Set active scan configuration | NNO_CMD_SCAN_SET_ACT_CFG | WRITE | 0x02 | 0x24 | 1 | 0 |
| DLPC150 power control | NNO_CMD_SET_DLPC_ONOFF_CTRL | WRITE | 0x02 | 0x25 | 1 | 0 |
| Set scan subimage | NNO_CMD_SET_SCAN_SUBIMAGE | WRITE | 0x02 | 0x26 | 4 | 0 |

## Table H-2. DLP NIRscan Nano USB Commands (continued)

| Description | Command ID | USB Communications Interface | | | | |
|---|---|---|---|---|---|---|
| | | R / W | Command | | Data Payload(2) | |
| | | | Group Byte | Command Byte | Input (Bytes) | Output (Bytes) |
| FACTORY USE ONLY: Erase Calibration Coefficients, Reference Calibration Coefficients, and Scan Configuration stored in EEPROM (1) | NNO_CMD_EEPROM_WIPE | WRITE | 0x02 | 0x27 | 3 | 0 |
| Read PGA setting | NNO_CMD_GET_PGA | READ | 0x02 | 0x28 | 0 | 1 |
| FACTORY USE ONLY: Write spectrum calibration coefficients (1) | NNO_CMD_CALIB_STRUCT_SAVE | WRITE | 0x02 | 0x29 | 144 | 0 |
| Read spectrum calibration coefficients | NNO_CMD_CALIB_STRUCT_READ | READ | 0x02 | 0x2A | 0 | 144 |
| Compute SNR | NNO_CMD_START_SNRSCAN | WRITE | 0x02 | 0x2B | 0 | 0 |
| Capture data for SNR computation. This command precedes the command that starts the scan for SNR computation. | NNO_CMD_SAVE_SNRDATA | READ | 0x02 | 0x2C | 0 | 20 |
| Generate calibration pattern | NNO_CMD_CALIB_GEN_PTNS | WRITE | 0x02 | 0x2D | 1 | 4 |
| Write number of scan repeats | NNO_CMD_SCAN_NUM_REPEATS | WRITE | 0x02 | 0x2E | 2 | 0 |
| Capture data for Hadamard SNR computation. This command precedes the command that starts the scan for SNR computation. | NNO_CMD_START_HADSNRSCAN | WRITE | 0x02 | 0x2F | 0 | 0 |
| FACTORY USE ONLY: Reference Calibration in EEPROM (1) | NNO_CMD_REFCAL_PERFORM | WRITE | 0x02 | 0x30 | 0 | 0 |
| Perform Scan with Patterns Stored in Flash | NNO_CMD_PERFORM_SCAN_FLASH_PTNS | WRITE | 0x02 | 0x31 | 1 | 0 |
| FACTORY USE ONLY: Write Device Serial Number (1) | NNO_CMD_SERIAL_NUMBER_WRITE | WRITE | 0x02 | 0x32 | 8 | 0 |
| Read device serial number | NNO_CMD_SERIAL_NUMBER_READ | READ | 0x02 | 0x33 | 0 | 8 |
| Set scan name tag | NNO_CMD_WRITE_SCAN_NAME_TAG | WRITE | 0x02 | 0x34 | Variable | 0 |
| Erase specified scan from microSD card | NNO_CMD_DEL_LAST_SCAN_FILE_SD | WRITE | 0x02 | 0x35 | 4 | 0 |
| FACTORY USE ONLY: Mass erase Tiva EEPROM (1) | NNO_CMD_EEPROM_MASS_ERASE | WRITE | 0x02 | 0x36 | 0 | 0 |
| Read estimated scan time for the last set scan configuration | NNO_CMD_READ_SCAN_TIME | READ | 0x02 | 0x37 | 0 | 4 |
| Delete last Scan from SD Card | NNO_CMD_DEL_LAST_SCAN_FILE_SD | WRITE | 0x02 | 0x38 | 0 | 0 |
| Start Tiva scan interpretation process | NNO_CMD_START_SCAN_INTERPRET | WRITE | 0x02 | 0x39 | 0 | 0 |
| Tiva scan interpretation status | NNO_CMD_SCAN_INTERPRET_GET_STATUS | READ | 0x02 | 0x3A | 0 | 1 |
| Write device model name | NNO_CMD_MODEL_NAME_WRITE | WRITE | 0x02 | 0x3B | 0 | 16 |
| Read device model name | NNO_CMD_MODEL_NAME_READ | READ | 0x02 | 0x3C | 16 | 0 |
| Read temperature (TMP006) sensor | NNO_CMD_READ_TEMP | READ | 0x03 | 0x00 | 0 | 8 |
| Read humidity (HDC1000) sensor | NNO_CMD_READ_HUM | READ | 0x03 | 0x02 | 0 | 8 |
| Set time and date | NNO_CMD_SET_DATE_TIME | WRITE | 0x03 | 0x09 | 7 | 0 |
| Read battery voltage | NNO_CMD_READ_BATT_VOLT | READ | 0x03 | 0x0A | 0 | 1 |
| Read Tiva internal temperature | NNO_CMD_READ_TIVA_TEMP | READ | 0x03 | 0x0B | 0 | 4 |
| Get time and date | NNO_CMD_GET_DATE_TIME | READ | 0x03 | 0x0C | 0 | 7 |
| Set Tiva in low power mode | NNO_CMD_HIBERNATE_MODE | WRITE | 0x03 | 0x0D | 0 | 0 |
| Set Hibernate Mode(3) | NNO_CMD_SET_HIBERNATE | WRITE | 0x03 | 0x0E | 1 | 0 |
| Get Hibernate Mode(3) | NNO_CMD_GET_HIBERNATE | READ | 0x03 | 0x0F | 0 | 1 |
| Read number of scan files stored in microSD card | NNO_CMD_GET_NUM_SCAN_FILES_SD | READ | 0x04 | 0x00 | 0 | 4 |

**Table H-2. DLP NIRscan Nano USB Commands (continued)**

| Description | Command ID | USB Communications Interface | | | | |
|---|---|---|---|---|---|---|
| | | R / W | Command | | Data Payload(2) | |
| | | | Group Byte | Command Byte | Input (Bytes) | Output (Bytes) |
| Read Photodetector | NNO_CMD_READ_PHOTODETECTOR | READ | 0x04 | 0x02 | 0 | 12 |
| Read Device Status | NNO_CMD_READ_DEVICE_STATUS | READ | 0x04 | 0x03 | 0 | 4 |
| Read Error Status | NNO_CMD_READ_ERROR_STATUS | READ | 0x04 | 0x04 | 0 | 24 |
| Clear Error Status | NNO_CMD_RESET_ERROR_STATUS | WRITE | 0x04 | 0x05 | 0 | 0 |
| Read Specific Error Status | NNO_CMD_GET_SPECIFIC_ERR_STATUS | READ | 0x04 | 0x06 | 4 | 0 |
| Read Specific Error Code | NNO_CMD_GET_SPECIFIC_ERR_CODE | READ | 0x04 | 0x07 | 2 | 0 |
| Clear Specific Error | NNO_CMD_CLEAR_SPECIFIC_ERR | WRITE | 0x04 | 0x08 | 4 | 0 |
| Overwrite Reference Data with Last Scan Data | NNO_CMD_UPDATE_REFCALDATA_WOREFL | WRITE | 0x04 | 0x0A | 0 | 0 |
| Erase DLPC150 Serial Flash | NNO_CMD_ERASE_DLPC_FLASH | WRITE | 0x04 | 0x0B | 0 | 0 |

(1) Commands highlighted in red are for factory calibration. Erasing the factory calibration data will render a unit non-functional and will need to be recalibrated.

(2) Refer toTable G-1 for description of data payload parameters.

(3) Command supported in Tiva Firmware version 1.2 or later.

# DLP NIRscan Nano UART Communications

## I.1 UART Communications

The DLP NIRscan Nano communicates using UART (Universal Asynchronous Receive and Transmit) through pin 3 and pin 4 of the J3 Expansion connector on the microprocessor board to exchange commands and data with a host processor or PC. The UART commands are variable length data packets that are sent with the least significant byte first. The maximum UART packet length is 64 bytes. The UART packets use the following serial configuration:

- Bits per second: 115200
- Data Bits: 8
- Parity: None
- Stops bits: 1
- Flow Control: None

The UART communications has a timeout feature. If more than 1 msec elapses between individual bytes in a transaction, Tiva will reset the receive buffer and return an error. Therefore, bytes transferred in a UART transaction sequence must be transmitted less than 1 msec apart from each other.

### I.1.1 UART Transaction Sequence

The UART transaction sequence has the structure shown in Table I-1:

**Table I-1. UART Transaction Sequence**

| Byte | Type | Field | Description | |
|---|---|---|---|---|
| 0 | UART Header | Start ID | Start ID Byte 0 = 0x41 | |
| 1 | | | Start ID Byte 1 = 0x42 | |
| 2 | | | Start ID Byte 2 = 0x43 | |
| 3 | | | Start ID Byte 3 = 0x44 | |
| 4 | Data Header | Checksum | Checksum Byte 0 | Checksum is a 32-bit unsigned value that sums all the bytes in Data Header and Data Payload. Checksum does not include Start ID, Checksum, and End ID Bytes. Checksum overflows are truncated. |
| 5 | | | Checksum Byte 1 | |
| 6 | | | Checksum Byte 2 | |
| 7 | | | Checksum Byte 3 | |
| 8 | | Flags | R/W (bit 7)<br>0 = Write<br>1 = Read | |
| | | | Reply (bit 6)<br>0 = No reply requested<br>1 = Reply requested | |
| | | | Response (bits 5:4)<br>0 = Success<br>1 = Error<br>2 = Busy | |
| | | | Reserved (bits 3:2) | |
| | | | Destination (bits 1:0)<br>0 = UART Command expansion connector<br>1 = Debug UART console connector | |
| 9 | Data Payload | Sequence | Number of command packet | |
| 10 | | Length | Length LSB of this command packet | |
| 11 | | | Length MSB of this command packet | |
| 12 | | Command | Command Byte | |
| 13 | | | Group Byte | |
| 14 | | Data | Data Byte 0 | |
| ... | | | ... | |
| N + 14 | | | Data Byte N | |
| N + 15 | UART Trailer | End ID | End ID Byte 0 = 0x44 | |
| N + 16 | | | End ID Byte 1 = 0x43 | |
| N + 17 | | | End ID Byte 2 = 0x42 | |
| N + 18 | | | End ID Byte 3 = 0x41 | |

#### I.1.1.1 UART Read Transaction Sequence

To issue a read command, the host must perform the following steps:

1. Host sends four Start bytes
2. Host sends the Checksum byte.
3. Host sends the Flags byte, where:
   - Bit 6 is set to 0x1 to indicate the host wants a reply from the device.
   - Bit 7 is set to 0x1 to indicate a read transaction.
4. Host sends the Sequence byte. When a single command is more than 64 bytes, it is sent as multiple USB packets and the sequence byte is used to number the packets so the device can assemble them in the right sequence. Otherwise, this value is irrelevant and generally set to 0.
5. Host sends two bytes with the length of the data packet. This length denotes the number of data bytes in the packet and excludes the number of bytes in steps 1 through 4. It denotes the total number of bytes sent in steps 5 (command bytes) and 6 (data bytes).
6. Host sends two Command bytes: Command Byte 0 and Command Byte 1.
7. Host sends data appropriate to command.
8. Host sends four Trailer bytes.
9. After completion of this command, DLP NIRscan Nano responds with a packet that includes:
   (a) Byte with the command requested by the host (the matching Sequence byte)
   (b) Length of the data packet
   (c) Data requested

#### I.1.1.2 UART Write Transaction Sequence

To issue a write command, the host must perform the following steps:

1. Host sends four Start bytes
2. Host sends the Checksum byte.
3. Host sends the Flags byte, where
   - Bit 6 is set to 0x1 to indicate the host wants a reply from the device. The device responds with the NACK bit set if an unknown command was received or there was any error in processing the command due to invalid parameters or other reasons..
   - Bit 7 is set to 0x1 to indicate a read transaction
4. Host sends the Sequence byte. When a single command is more than 64 bytes, it is sent as multiple USB packets and the sequence byte is used to number the packets so the device can assemble them in the right sequence. In other cases, this value is irrelevant and generally set to 0.
5. Host sends two bytes with the length of the data packet. This length denotes the number of data bytes in the packet and excludes the number of bytes in steps 1 through 4. It denotes the total number of bytes sent in steps 5 (command bytes) and 6 (data bytes).
6. Host sends two Command bytes: Command Byte 0 and Command Byte 1.
7. Host sends data appropriate to command
8. Host sends four Trailer bytes.
9. After completion of this command, DLP NIRscan Nano responds with a packet that includes a byte with the command requested by the host. This occurs only if bit 6 was set in the Flags byte.

### I.1.2 UART Error Packet

On Tiva Firmware version 1.2 or later, Tiva transmits the error packet shown in Table I-2 when Tiva detects an error in the command packet or does not receive a byte within 1 msec. transaction,:

**Table I-2. UART Error Packet**

| Byte | Type | Field | Description |
|---|---|---|---|
| 0 | UART Header | Start ID | Start ID Byte 0 = 0x41 |
| 1 | | | Start ID Byte 1 = 0x42 |
| 2 | | | Start ID Byte 2 = 0x43 |
| 3 | | | Start ID Byte 3 = 0x44 |
| 4 | Data Header | Checksum | Checksum Byte 0 = 0x10 |
| 5 | | | Checksum Byte 1 = 0x00 |
| 6 | | | Checksum Byte 2 = 0x00 |
| 7 | | | Checksum Byte 3 = 0x00 |
| 8 | | Flags | Flags = 0x10, Implies a NACK |
| 9 | Data Payload | Sequence | Sequence = 0x00 |
| 10 | | Length | Length LSB = 0x00 |
| 11 | | | Length MSB = 0x00 |
| 12 | UART Trailer | End ID | End ID Byte 0 = 0x44 |
| 13 | | | End ID Byte 1 = 0x43 |
| 14 | | | End ID Byte 2 = 0x42 |
| 15 | | | End ID Byte 3 = 0x41 |

### I.1.3 UART Commands

Table I-3 lists all the DLP NIRscan Nano supported UART commands.

**Table I-3. DLP NIRscan Nano UART Commands**

| Description | Command ID | UART Communications Interface | | | | |
|---|---|---|---|---|---|---|
| | | R / W | Command | | Data Payload[2] | |
| | | | Group Byte | Command Byte | Input (Bytes) | Output (Bytes) |
| Write file payload information | NNO_CMD_FILE_SET_WRITESIZE | WRITE | 0x00 | 0x2A | 6 | 0 |
| Read file list size | NNO_CMD_READ_FILE_LIST_SIZE | READ | 0x00 | 0x2B | 1 | 4 |
| Read file list | NNO_CMD_READ_FILE_LIST | READ | 0x00 | 0x2C | 1 | Variable |
| Read file size | NNO_CMD_FILE_GET_READSIZE | READ | 0x00 | 0x2D | 1 | 4 |
| Read file data | NNO_CMD_FILE_GET_DATA | READ | 0x00 | 0x2E | 0 | Variable |
| EEPROM test | NNO_CMD_EEPROM_TEST | READ | 0x01 | 0x01 | 0 | 1 |
| Detector board test | NNO_CMD_ADC_TEST | READ | 0x01 | 0x02 | 0 | 1 |
| Battery charger test | NNO_CMD_BQ_TEST | READ | 0x01 | 0x03 | 0 | 1 |
| SDRAM test | NNO_CMD_SDRAM_TEST | READ | 0x01 | 0x04 | 0 | 1 |
| DLP controller power up | NNO_CMD_DLPC_ENABLE | WRITE | 0x01 | 0x05 | 2 | 0 |
| Temperature sensor test | NNO_CMD_TMP_TEST | READ | 0x01 | 0x06 | 0 | 1 |
| Humidity sensor test | NNO_CMD_HDC_TEST | READ | 0x01 | 0x07 | 0 | 1 |
| Bluetooth test | NNO_CMD_BT_TEST | WRITE | 0x01 | 0x08 | 1 | 1 |
| microSD Card test | NNO_CMD_SDC_TEST | READ | 0x01 | 0x09 | 1 | 1 |
| LED test | NNO_CMD_LED_TEST | READ | 0x01 | 0x0B | 1 | 1 |
| Read button test | NNO_CMD_BUTTON_TEST_RD | READ | 0x01 | 0x0C | 1 | 1 |
| Write button test | NNO_CMD_BUTTON_TEST_WR | WRITE | 0x01 | 0x0D | 1 | 1 |

**Table I-3. DLP NIRscan Nano UART Commands (continued)**

| Description | Command ID | UART Communications Interface | | | | |
|---|---|---|---|---|---|---|
| | | R / W | Command | | Data Payload[(2)] | |
| | | | Group Byte | Command Byte | Input (Bytes) | Output (Bytes) |
| FACTORY USE ONLY: Write EEPROM Calibration Coefficient, Scan Configuration, and Reference Calibration Coefficients Versions [(1)] | NNO_CMD_EEPROM_CAL_TEST | WRITE | 0x01 | 0x0E | 0 | 1 |
| Read Tiva version information | NNO_CMD_TIVA_VER | READ | 0x02 | 0x16 | 0 | 28 |
| Write patterns in SDRAM | NNO_CMD_STORE_PTN_SDRAM | WRITE | 0x02 | 0x17 | 12 | 0 |
| Start scan | NNO_CMD_PERFORM_SCAN | WRITE | 0x02 | 0x18 | 1 | 0 |
| Scan status | NNO_CMD_SCAN_GET_STATUS | READ | 0x02 | 0x19 | 0 | 1 |
| Reset Tiva | NNO_CMD_TIVA_RESET | WRITE | 0x02 | 0x1A | 0 | 0 |
| Write ADC PGA gain | NNO_CMD_SET_PGA | WRITE | 0x02 | 0x1B | 1 | 0 |
| FACTORY USE ONLY: Write DLPC150 Register [(1)] | NNO_CMD_SET_DLPC_REG | WRITE | 0x02 | 0x1C | 8 | 0 |
| Read DLPC150 register | NNO_CMD_GET_DLPC_REG | READ | 0x02 | 0x1D | 4 | 4 |
| Generate patterns | NNO_CMD_SCAN_CFG_APPLY | WRITE | 0x02 | 0x1E | Variable | 4 |
| Save given scan configuration in EEPROM | NNO_CMD_SCAN_CFG_SAVE | WRITE | 0x02 | 0x1F | Variable | 0 |
| Read stored scan configurations | NNO_CMD_SCAN_CFG_READ | READ | 0x02 | 0x20 | 2 | Variable |
| Erase all stored scan configurations | NNO_CMD_SCAN_CFG_ERASEALL | WRITE | 0x02 | 0x21 | 0 | 0 |
| Read number of stored scan configurations | NNO_CMD_SCAN_CFG_NUM | READ | 0x02 | 0x22 | 0 | 1 |
| Read active scan configuration | NNO_CMD_SCAN_GET_ACT_CFG | READ | 0x02 | 0x23 | 0 | 1 |
| Set active scan configuration | NNO_CMD_SCAN_SET_ACT_CFG | WRITE | 0x02 | 0x24 | 1 | 0 |
| DLPC150 power control | NNO_CMD_SET_DLPC_ONOFF_CTRL | WRITE | 0x02 | 0x25 | 1 | 0 |
| Set scan subimage | NNO_CMD_SET_SCAN_SUBIMAGE | WRITE | 0x02 | 0x26 | 4 | 0 |
| FACTORY USE ONLY: Erase Calibration Coefficients, Reference Calibration Coefficients, and Scan Configuration stored in EEPROM [(1)] | NNO_CMD_EEPROM_WIPE | WRITE | 0x02 | 0x27 | 3 | 0 |
| Read PGA setting | NNO_CMD_GET_PGA | READ | 0x02 | 0x28 | 0 | 1 |
| FACTORY USE ONLY: Write spectrum calibration coefficients [(1)] | NNO_CMD_CALIB_STRUCT_SAVE | WRITE | 0x02 | 0x29 | 144 | 0 |
| Read spectrum calibration coefficients | NNO_CMD_CALIB_STRUCT_READ | READ | 0x02 | 0x2A | 0 | 144 |
| Compute SNR | NNO_CMD_START_SNRSCAN | WRITE | 0x02 | 0x2B | 0 | 0 |
| Read SNR calculation results | NNO_CMD_SAVE_SNRDATA | READ | 0x02 | 0x2C | 0 | 20 |
| Generate calibration pattern | NNO_CMD_CALIB_GEN_PTNS | WRITE | 0x02 | 0x2D | 1 | 4 |
| Write number of scan repeats | NNO_CMD_SCAN_NUM_REPEATS | WRITE | 0x02 | 0x2E | 2 | 0 |
| Capture data for Hadamard SNR computation. This command precedes the command that starts the scan for SNR computation. | NNO_CMD_START_HADSNRSCAN | WRITE | 0x02 | 0x2F | 0 | 0 |
| FACTORY USE ONLY: Reference Calibration in EEPROM [(1)] | NNO_CMD_REFCAL_PERFORM | WRITE | 0x02 | 0x30 | 0 | 0 |
| Perform Scan with Patterns Stored in Flash | NNO_CMD_PERFORM_SCAN_FLASH_PTNS | WRITE | 0x02 | 0x31 | 1 | 0 |
| FACTORY USE ONLY: Write Device Serial Number [(1)] | NNO_CMD_SERIAL_NUMBER_WRITE | WRITE | 0x02 | 0x32 | 8 | 0 |
| Read device serial number | NNO_CMD_SERIAL_NUMBER_READ | READ | 0x02 | 0x33 | 0 | 8 |
| Set scan name tag | NNO_CMD_WRITE_SCAN_NAME_TAG | WRITE | 0x02 | 0x34 | Variable | 0 |

## Table I-3. DLP NIRscan Nano UART Commands (continued)

| Description | Command ID | UART Communications Interface | | | | |
|---|---|---|---|---|---|---|
| | | R / W | Command | | Data Payload(2) | |
| | | | Group Byte | Command Byte | Input (Bytes) | Output (Bytes) |
| Erase specified scan from microSD card | NNO_CMD_DEL_LAST_SCAN_FILE_SD | WRITE | 0x02 | 0x35 | 4 | 0 |
| FACTORY USE ONLY: Mass erase Tiva EEPROM (1) | NNO_CMD_EEPROM_MASS_ERASE | WRITE | 0x02 | 0x36 | 0 | 0 |
| Read scan time | NNO_CMD_READ_SCAN_TIME | READ | 0x02 | 0x37 | 0 | 4 |
| Delete last Scan from SD Card | NNO_CMD_DEL_LAST_SCAN_FILE_SD | WRITE | 0x02 | 0x38 | 0 | 0 |
| Start Tiva scan interpretation process | NNO_CMD_START_SCAN_INTERPRET | WRITE | 0x02 | 0x39 | 0 | 0 |
| Tiva scan interpretation status | NNO_CMD_SCAN_INTERPRET_STATUS | READ | 0x02 | 0x3A | 0 | 1 |
| Write device model name | NNO_CMD_MODEL_NAME_WRITE | WRITE | 0x02 | 0x3B | 0 | 16 |
| Read device model name | NNO_CMD_MODEL_NAME_READ | READ | 0x02 | 0x3C | 16 | 0 |
| Read temperature (TMP006) sensor | NNO_CMD_READ_TEMP | READ | 0x03 | 0x00 | 0 | 8 |
| Read humidity (HDC1000) sensor | NNO_CMD_READ_HUM | READ | 0x03 | 0x02 | 0 | 8 |
| Set time and date | NNO_CMD_SET_DATE_TIME | WRITE | 0x03 | 0x09 | 7 | 0 |
| Read battery voltage | NNO_CMD_READ_BATT_VOLT | READ | 0x03 | 0x0A | 0 | 1 |
| Read Tiva internal temperature | NNO_CMD_READ_TIVA_TEMP | READ | 0x03 | 0x0B | 0 | 4 |
| Get time and date | NNO_CMD_GET_DATE_TIME | READ | 0x03 | 0x0C | 0 | 7 |
| Set Tiva in low power mode | NNO_CMD_HIBERNATE_MODE | WRITE | 0x03 | 0x0D | 0 | 0 |
| Set Hibernate Mode(3) | NNO_CMD_SET_HIBERNATE | WRITE | 0x03 | 0x0E | 1 | 0 |
| Get Hibernate Mode(3) | NNO_CMD_GET_HIBERNATE | READ | 0x03 | 0x0F | 0 | 1 |
| Read number of scan files stored in microSD Card | NNO_CMD_GET_NUM_SCAN_FILES_SD | READ | 0x04 | 0x00 | 0 | 0 |
| Read Photodetector | NNO_CMD_READ_PHOTODETECTOR | READ | 0x04 | 0x02 | 0 | 12 |
| Read Device Status | NNO_CMD_READ_DEVICE_STATUS | READ | 0x04 | 0x03 | 0 | 4 |
| Read Error Status(3) | NNO_CMD_READ_ERROR_STATUS | READ | 0x04 | 0x04 | 0 | 24 |
| Clear Error Status(3) | NNO_CMD_RESET_ERROR_STATUS | WRITE | 0x04 | 0x05 | 0 | 0 |
| Read Specific Error Status | NNO_CMD_GET_SPECIFIC_ERR_STATUS | READ | 0x04 | 0x06 | 4 | 0 |
| Read Specific Error Code | NNO_CMD_GET_SPECIFIC_ERR_CODE | READ | 0x04 | 0x07 | 2 | 0 |
| Clear Specific Error | NNO_CMD_CLEAR_SPECIFIC_ERR | WRITE | 0x04 | 0x08 | 4 | 0 |
| Overwrite Reference Data with Last Scan Data | NNO_CMD_UPDATE_REFCALDATA_WOREFL | WRITE | 0x04 | 0x0A | 0 | 0 |
| Erase DLPC150 Serial Flash | NNO_CMD_ERASE_DLPC_FLASH | WRITE | 0x04 | 0x0B | 0 | 0 |

(1) Commands highlighted in red are for factory calibration. Erasing the factory calibration data will render a unit non-functional and will need to be recalibrated.

(2) Refer to Table G-1 for description of data payload parameters.

(3) Command supported in Tiva Firmware version 1.2 or later.

# DLP NIRscan Nano Bluetooth Communications

## J.1 Bluetooth Communications

The DLP NIRscan Nano wirelessly communicates using Bluetooth Low Energy version 4.0 This wireless communication uses two main profiles for discovery and communication with a remote host:

- GAP: Generic access profile for basic discovery and establishing connections.
- GATT: Generic attribute profile for commands and data transfer.

The DLP NIRscan Nano supports Bluetooth version 4.0 specification. When the Bluetooth sub-system is activated, the DLP NIRscan Nano broadcasts its availability while a smartphone, tablet or PC acts as an observer. Once connected, the DLP NIRscan Nano acts as a server for the GATT profile while the smartphone, tablet, or PC acts as a client.

### *J.1.1 GATT Supported Services*

The DLP NIRscan Nano Bluetooth GATT Profile supports the following services:

- Battery Service (BAS) to provide battery charge capacity.
- Device Information Service (DIS) to provide manufacturer Name, model number, serial number, hardware revision, spectrum library revision, and Tiva software revision.
- GATT General Information Service to provide temperature, humidity, status, hours of use, lamp hours, and battery recharge cycles.
- GATT Date and Time Service to synchronize date and time information between smartphone, tablet, or PC to the Tiva's realtime clock.
- GATT Calibration Information Service to provide calibration coefficients
- GATT Scan Configuration Information Service to provide stored configurations and scan configuration data.
- GATT Scan Data Information Service to initiate scan, clear scan data, and return stored scan data.
- GATT Command Service to send generic commands.

A GATT service has a universally unique identifier (UUID) used to identify every service. A UUID is a 128-bit value. However, common or frequently used services that are included in the BLE specifications and/or certified by Bluetooth.org are shortened to 16-bit UUID to improve efficiency.

Each service is composed of a set of characteristics. Each characteristic contains a value with properties for how the value is accessed and information on how the value is displayed or represented. The properties are:

- R = Read.
- W = Write.
- WWoR = Write without response. Not used in DLP NIRscan Nano Bluetooth implementation.
- S = Signed write. Not used in DLP NIRscan Nano Bluetooth implementation.
- N = Notify.
- I = Indicate.
- WA = Writable auxiliaries. Not used in DLP NIRscan Nano Bluetooth implementation.
- B = Broadcast. Not used in DLP NIRscan Nano Bluetooth implementation.
- EP = Extended properties. Not used in DLP NIRscan Nano Bluetooth implementation.

An "X" in a supported property indicates the properties supported by a characteristic. Empty columns indicate properties not supported by the characteristic.

A data size entry with an MP value represents multiple packets. All data is transmitted little-endian.

### Table J-1. Device Information Service (DIS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0x180A | Device Information Service (DIS) | | | | | | | |
| Characteristic UUID | Description | Data Format | Data Size (Bytes) | Supported Properties | | | | Notes |
| | | | | R | W | N | I | |
| 0x2A29 | Manufacturer name string | string | 1 | X | | | | org.bluetooth.characteristic.manufacturer_name_string |
| 0x2A24 | Model number string | string | 1 | X | | | | org.bluetooth.characteristic.model_number_string |
| 0x2A25 | Serial number string | string | 1 | X | | | | org.bluetooth.characteristic.serial_number_string |
| 0x2A27 | Hardware revision string | string | 1 | X | | | | org.bluetooth.characteristic.hardware_revision_string |
| 0x2A26 | Tiva firmware revision string | string | 1 | X | | | | org.bluetooth.characteristic.firmware_revision_string |
| 0x2A28 | Spectrum library revision string | unsigned integer | 2 | X | | | | org.bluetooth.characteristic.software_revision_string |

### Table J-2. Battery Service (BAS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0x180F | Battery Service (BAS) | | | | | | | |
| Characteristic UUID | Description | Data Format | Data Size (Bytes) | Supported Properties | | | | Notes |
| | | | | R | W | N | I | |
| 0x2A19 | Battery level | unsigned integer | 1 | X | | | | org.bluetooth.service.battery_service. Value reported in the range of 0-100. |

### Table J-3. GATT General Information Service (GGIS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0x53455201-444C-5020-4E49-52204E616E6F | GATT General Information Service | | | | | | | |
| Characteristic UUID | Description | Data Format | Data Size (Bytes) | Supported Properties | | | | Notes |
| | | | | R | W | N | I | |
| 0x43484101-444C-5020-4E49-52204E616E6F | Temperature measurement | integer | 2 | X | | X | | Integer value returned in hundredths. Divide by 100 to get the actual floating point number. |
| 0x43484102-444C-5020-4E49-52204E616E6F | Humidity measurement | unsigned integer | 2 | X | | X | | |
| 0x43484103-444C-5020-4E49-52204E616E6F | Device status (Reserved for future support) | unsigned integer | 2 | X | | X | | |
| 0x43484104-444C-5020-4E49-52204E616E6F | Error status (Reserved for future support) | unsigned integer | 2 | X | | X | | |
| 0x43484105-444C-5020-4E49-52204E616E6F | Temperature threshold | integer | 2 | | X | | | Value set in hundredths. Input truncated integer of actual value multiplied by 100. |
| 0x43484106-444C-5020-4E49-52204E616E6F | Humidity threshold | unsigned integer | 2 | | X | | | |
| 0x43484107-444C-5020-4E49-52204E616E6F | Number of hours of use (Reserved for future support) | unsigned integer | 2 | X | | | | |

### Table J-3. GATT General Information Service (GGIS) (continued)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0x53455201-444C-5020-4E49-52204E616E6F** | **GATT General Information Service** | | | | | | | |
| **Characteristic UUID** | **Description** | **Data Format** | **Data Size (Bytes)** | **Supported Properties** | | | | **Notes** |
| | | | | **R** | **W** | **N** | **I** | |
| 0x43484108-444C-5020-4E49-52204E616E6F | Number of battery recharge cycles (Reserved for future support) | unsigned integer | 2 | X | | | | |
| 0x43484109-444C-5020-4E49-52204E616E6F | Total lamp hours (Reserved for future support) | unsigned integer | 2 | X | | | | |
| 0x4348410A-444C-5020-4E49-52204E616E6F | Error log (Reserved for future support) | string | 1 | X | | | | |

### Table J-4. GATT Date and Time Service (GDTS)

| Service UUID | Description | | | | | | |
|---|---|---|---|---|---|---|---|
| **0x53455203-444C-5020-4E49-52204E616E6F** | **GATT Current Date and Time** | | | | | | |
| **Characteristic UUID** | **Description** | **Data Format** | **Data Size (Bytes)** | **Supported Properties** | | | |
| | | | | **R** | **W** | **N** | **I** |
| | Current year (0-99: starting in year 2000) | unsigned integer | 1 | | | | |
| | Current month (1-12) | unsigned integer | 1 | | | | |
| | Current day (1-31) | unsigned integer | 1 | | | | |
| 0x4348410C-444C-5020-4E49-52204E616E6F | Current day of the week (0-6) | unsigned integer | 1 | | X | | |
| | Current hour (0-23) | unsigned integer | 1 | | | | |
| | Current minute (0-59) | unsigned integer | 1 | | | | |
| | Current second (0-59) | unsigned integer | 1 | | | | |

### Table J-5. GATT Calibration Information Service (GCIS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0x53455204-444C-5020-4E49-52204E616E6F** | **GATT Calibration Information Service** | | | | | | | |
| **Characteristic UUID** | **Description** | **Data Format** | **Data Size (Bytes)** | **Supported Properties** | | | | **Notes** |
| | | | | **R** | **W** | **N** | **I** | |
| 0x4348410D-444C-5020-4E49-52204E616E6F | Request Spectrum Calibration Coefficients | unsigned integer | 1 | | X | | | Indicate intent to read. No data transferred. |
| 0x4348410E-444C-5020-4E49-52204E616E6F | Return Spectrum Calibration Coefficients | | MP(1) | | | X | | Send 6 coefficients. Each coefficient is a double data-type of 8 bytes. The data is sent in serialized manner. |
| 0x4348410F-444C-5020-4E49-52204E616E6F | Request Reference Calibration Coefficients | unsigned integer | 1 | | X | | | Indicate intent to read. No data transferred. |
| 0x43484110-444C-5020-4E49-52204E616E6F | Return Reference Calibration Coefficients | | MP(1) | | | X | | Serialized data; refer to spectrum C library for data structure. |
| 0x43484111-444C-5020-4E49-52204E616E6F | Request Reference Calibration Matrix | unsigned integer | 1 | | X | | | Indicate intent to read. No data transferred. |
| 0x43484112-444C-5020-4E49-52204E616E6F | Return Reference Calibration Matrix | | MP(1) | | | X | | Serialized data; refer to spectrum C library for data structure. |

(1) Refer to Table J-9

### Table J-6. GATT Scan Configuration Information Service (GSCIS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0x53455205-444C-5020-4E49-52204E616E6F** | **GATT Scan Configuration Information Service** | | | | | | | |
| **Characteristic UUID** | **Description** | **Data Format** | **Data Size (Bytes)** | **Supported Properties** | | | | **Notes** |
| | | | | **R** | **W** | **N** | **I** | |
| 0x43484113-444C-5020-4E49-52204E616E6F | Number of stored configurations | unsigned integer | 2 | X | | | | |
| 0x43484114-444C-5020-4E49-52204E616E6F | Request stored configurations list | unsigned integer | 1 | | X | | | No data transmitted. |
| 0x43484115-444C-5020-4E49-52204E616E6F | Return stored configurations list | | MP(1) | | | X | | List of 2 byte indices. |
| 0x43484116-444C-5020-4E49-52204E616E6F | Request scan configuration data | unsigned integer | 2 | | X | | | Index to read. |
| 0x43484117-444C-5020-4E49-52204E616E6F | Return scan configuration data | | MP(1) | | | X | | Serialized data; refer to spectrum C library for data structure. |
| 0x43484118-444C-5020-4E49-52204E616E6F | Active scan configuration | | 2 | X | X | | Get/Set function. Parameter transmitted is a 2-byte index. | |

(1) Refer to Table J-9

## Table J-7. GATT Scan Data Information Service (GSDIS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0x53455206-444C-5020-4E49-52204E616E6F** | **GATT Scan Data Information Service** | | | | | | | |
| **Characteristic UUID** | **Description** | **Data Format** | **Data Size (Bytes)** | **Supported Properties** | | | | **Notes** |
| | | | | **R** | **W** | **N** | **I** | |
| 0x43484119-444C-5020-4E49-52204E616E6F | Number of SD Card stored scans | unsigned integer | 4 | X | | | | |
| 0x4348411A-444C-5020-4E49-52204E616E6F | SD Card stored scan indices list | unsigned integer | 4 | | X | | | No data transmitted. |
| 0x4348411B-444C-5020-4E49-52204E616E6F | SD Card stored scan indices list | | MP (1) | | | X | | Multiple packet transfer of five 4-byte indices per packet. |
| 0x4348411C-444C-5020-4E49-52204E616E6F | Set scan name stub | string | 2 | | X | | | Limited to 15 bytes. |
| 0x4348411D-444C-5020-4E49-52204E616E6F | Start scan | unsigned integer | 1 | | X | X | | Parameter value:<br>0x00 = do not store scan in microSD card<br>0x01 = store scan in microSD card.<br>Notification returned:<br>0xFF = scan completed<br>4 bytes with the scan index of the current scan |
| 0x4348411E-444C-5020-4E49-52204E616E6F | Clear scan | unsigned integer | 4 | | X | X | | Index of scan data to clear. Notification returned:<br>0x00 = success<br>non-zero return is an error |
| 0x4348411F-444C-5020-4E49-52204E616E6F | Request scan name | unsigned integer | 4 | | X | | | Index of scan data to read. |
| 0x43484120-444C-5020-4E49-52204E616E6F | Return scan name | string | 20 | | | X | | Scan name limited to 20 characters. |
| 0x43484121-444C-5020-4E49-52204E616E6F | Request scan type | unsigned integer | 4 | | X | | | Index of scan data to read. |
| 0x43484122-444C-5020-4E49-52204E616E6F | Return scan type | unsigned integer | 1 | | | X | | |
| 0x43484123-444C-5020-4E49-52204E616E6F | Request scan date/time | unsigned integer | 4 | | X | | | Index of scan data to read. |
| 0x43484124-444C-5020-4E49-52204E616E6F | Return scan date/time | unsigned integer | 7 | | | X | | Seven bytes returned. Same format as GATT Date and Time Service. |
| 0x43484125-444C-5020-4E49-52204E616E6F | Request packet format version | unsigned integer | 4 | | X | | | Index of data to read. |
| 0x43484126-444C-5020-4E49-52204E616E6F | Return packet format version | unsigned integer | 4 | | | X | | |
| 0x43484127-444C-5020-4E49-52204E616E6F | Request serialized scan data structure | unsigned integer | 4 | | X | | | Index of data to read |
| 0x43484128-444C-5020-4E49-52204E616E6F | Return serialized scan data structure | | MP (1) | | | X | | Serialized data; refer to spectrum C library for data structure. |

(1) Refer toTable J-9

## Table J-8. GATT Command Service (GCS)

| Service UUID | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0x53455202-444C-5020-4E49-52204E616E6F** | **GATT Command Service** | | | | | | | |
| **Characteristic UUID** | **Description** | **Data Format** | **Data Size (Bytes)** | **Supported Properties** | | | | **Notes** |
| | | | | **R** | **W** | **N** | **I** | |
| 0x4348410B-444C-5020-4E49-52204E616E6F | Command data packet | | < 20 | | X | | | Sends a command in one BLE packet. Packet Description:<br>Byte 0: Command Byte 0<br>Byte 1: Command Byte 1<br>Byte 2: Flag Byte<br>3 = Write<br>5 = Read<br>Byte 3: Length of Command Parameters Byte<br>Byte 4: Command Parameters<br>...<br>Byte 19: Command Parameters |
| | Response to command packet | unsigned integer | MP[1] | | | X | | NIRscan Nano may respond to the command with multiple packets. |

(1) Refer to Table J-9

## J.2 Bluetooth Packet

Bluetooth transmits in short packet sizes. The typical maximum transmission unit for an iOS App is 20 bytes. Multiple packets are needed to transfer the following information to DLP NIRscan Nano:

- Spectrum Calibration Coefficients
- Reference Calibration Coefficients
- Stored Configurations List
- Scan Configuration Data
- Stored Scan Indices
- Serialized Scan Data Structure
- Command Service Parameters

The previous tables label the data size as MP to denote that multiple packets are used during transfer. The packet structure is shown in Table J-9.

**Table J-9. Bluetooth Multiple Packet Structure**

| Packet | Byte | Type | Field | Description |
|---|---|---|---|---|
| 0 | 0 | Packet Header | Index | Number of packet: 0 |
| | 1 | Data Payload | Size | Length of data payload in bytes |
| | 2 | | | |
| | 3 | | | |
| | 4 | | | |
| 1 | 0 | Packet Header | Index | Number of packet: 1 |
| | 1 | Data Payload | Acknowledge | ACK/NACK (if required by command response) |
| | 2 | | Data Returned | Packet 1 Data byte 0 |
| | ... | | | ... |
| | 19 | | | Packet 1 Data byte 19 |
| 2 | 0 | Packet Header | Index | Number of packet: 2 |
| | 1 | Data Payload | Data Returned | Packet 2 Data byte 0 |
| | ... | | | ... |
| | 19 | | | Packet 2 Data byte 19 |
| ... | | | | |
| N | 0 | Packet Header | Index | Number of packet: N |
| | 1 | Data Payload | Data Returned | Packet N Data byte 0 |
| | ... | | | ... |
| | 19 | | | Packet N Data byte 19 |

# Troubleshooting

## K.1 Scan Error

A dialog box with a Scan Error might result from the following issues:

- **Power Issue:** If the lamps do not light up during a scan, there might a power issue. The DLP NIRscan Nano takes about 500 mA when the lamps turn on. In some systems, the current spike is more than an unpowered USB 2.0 hub or USB 2.0 USB PC port can output. Use a powered hub, or connect directly to a USB 3.0 PC port. Some very long and low quality cables can add enough resistance which decreases the power reaching the DLP NIRscan Nano. In this case, use a shorter cable or a known good quality cable. To narrow down power issues, connect the Nano to a USB power adapter capable of 1 A of current (like an iPhone adapter). Initiate a scan by pressing the scan button. If the lamps light up and the scan is completed, the green LED will flash at a higher frequency to indicate that it could not store the scan data. If the lamps light up and the scan completes, but then the green LED turns off for over a second and then starts blinking at the regular 1 blink per second, then the board resets. This is probably also a power issue.
- **Scan Configuration Issue:** If the lamps light up, but the yellow LED blinks after a scan, there might be a scan configuration issue. Ensure that a proper scan configuration is stored in the PC or the DLP NIRscan Nano. Also ensure that the latest Tiva and DLPC150 firmware and the corresponding GUI version are used. Refer to Section 2.3 on how to download new firmware to the DLP NIRscan Nano.

## K.2 Battery

Battery issues might result from:

- **Battery Not Charging:** To charge a battery, a battery thermistor is required to safely charge the battery and monitor its temperature. Refer to Section 4.1 for recommended battery and battery thermistor.

## K.3 System Not Responding

A system not responding to USB or Bluetooth commands might result from:

- **System in Hibernation Mode:** The default mode is system hibernation, which puts the system into a low power state after 5 minutes without a new command. During hibernation, the green LED will not blink. Refer to Section 3.1.8 to disable this default hibernation mode in the GUI. To exit hibernation, press the Wake or Reset button.
- **Invalid Scan Configuration Set as Active in EEPROM:** During power up initialization, the application software in Tiva, reads the EEPROM to find the active scan configuration set by the user. To set an active configuration with GUI, select one scan configuration that is stored in the NIRscan Nano and click the "Set Active Config" button in the Scan Configuration dialog window as shown in Figure K-1. If the active scan configuration has been corrupted or saved with invalid values in any of the entries, this can cause the Tiva firmware to hang on startup while applying the invalid scan configuration. If the Tiva firmware hangs, the Tiva may not respond to any USB commands. To recover from this situation, the user must instruct the Tiva application software to avoid applying the invalid active scan configuration from EEPROM. This is accomplished by placing a file named "SKIP_CFG" in the micro SD card under the folder with the NIRscan Nano EVM serial number. After successfully booting, connect the GUI and correct the invalid scan configuration. Then delete the "SKIP_CFG" file from the microSD card.

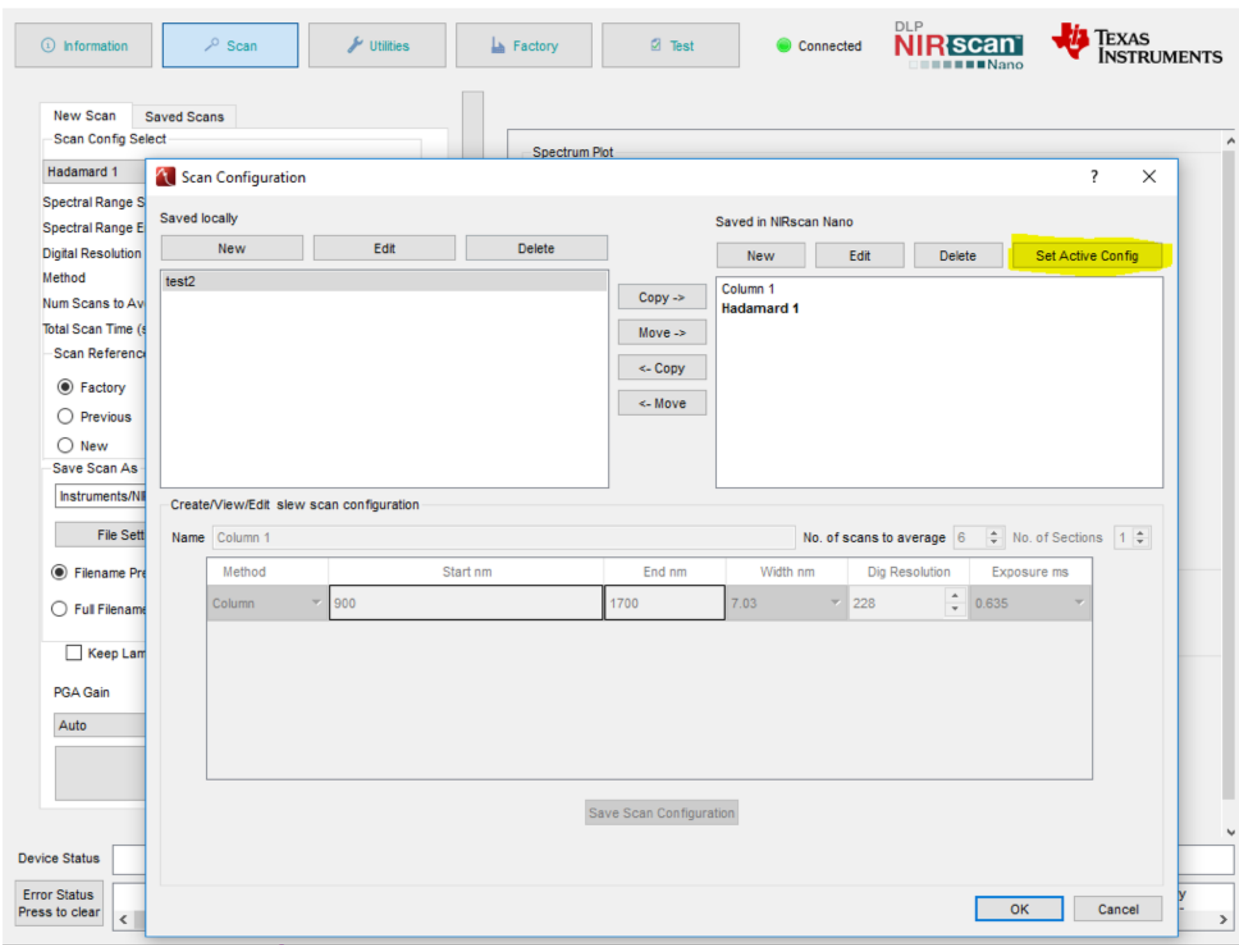

**Figure K-1. Setting Active Scan Configuration in NIRscan Nano EVM**

- **NIRscan Nano Updated with Untested/Faulty Tiva Firmware:**The user may modify the Tiva application source code, recompile the source code into binary, and load the binary into the EVM through Tiva firmware update procedure. IF there is a bug in the updated software, the application may become unresponsive to USB and/or Bluetooth commands. To work around this issue, Tiva firmware v2.1 or later includes a feature that will put the software in boot loader mode by pressing and holding the scan button on the EVM while the unit is powered up or reset. Once in bootloader mode, the GUI can be used to load the EVM with an updated version of the Tiva firmware. To upload new software, check the “Tiva Flash is empty/erased” checkbox shown in Figure K-2.

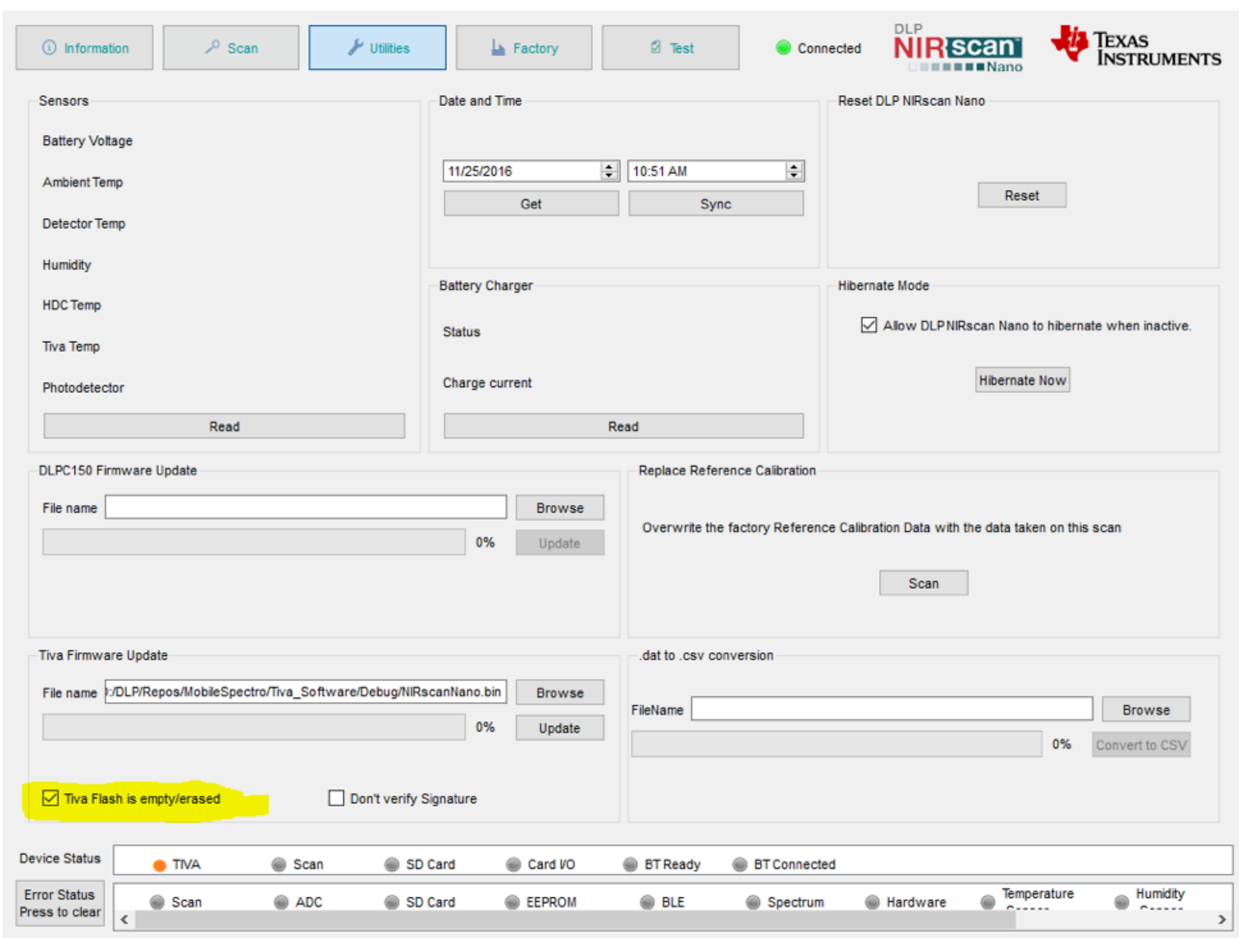


**Figure K-2. Uploading New Tiva Firmware**

## K.4 Intensity Variability

If the intensity varies between successive scans, ensure the lamp power connector is well-seated into its connector socket and the cables have not been pulled out of the socket.

Sometimes, the lamp might need to warm up to stabilize, so taking a couple of additional discardable scans will warm up the lamps for the actual scans.

## K.5 Version Mismatch

A dialog window indicating there is a version mismatch requires that the user upgrade the GUI software, Tiva firmware, and DLPC150 firmware to the latest versions. Refer to Section 2.3 for instructions describing the firmware upgrade process.

# Revision History

**Changes from F Revision (March 2016) to G Revision** **Page**

| Changes from E Revision (March 2016) to F Revision | Page |
| --- | --- |

- Added DLP NIRscan Nano EVM: Getting Started Out of the Box video .......... 9
- Changed the link of the E2E DLP forum to point to the NIRscan Nano E2E forum .......... 9
- Added portion of the slit that is imaged onto the DMD .......... 11
- Added links to software downloads .......... 22
- Added firmware update instructions .......... 22
- Added Fixed PGA Gain .......... 33
- Added Back-to-Back Scan .......... 34
- Added more scan workflow details and list of necessary commands .......... 50
- Added note about round to linear fiber bundles when using fiber input unit. .......... 60
- Corrected characteristic UUID number for Return Spectrum Calibration Coefficients .......... 97
- Added troubleshooting section .......... 101
- Added Intensity variability troubleshooting .......... 103

| Changes from D Revision (November 2015) to E Revision | Page |
|---|---|
| • Updated file path name from /doc/html/index.html to \doc/html/index.html in Section 2.4 | 23 |
| • Updated DLL name from platforms/qwindows.dll to platforms\qwindows.dll in Section 2.5.1.1 | 24 |
| • Updated Section 3.1, Section 3.1.1, Section 3.1.6, Section 3.1.7, and Section 3.1.8 to describe GUI version 2.0 changes | 26 |
| • Added new commands supported in version 2.0 release in Table G-1 | 74 |
| • Added new commands supported in version 2.0 release in Table H-2 | 85 |
| • Added new commands supported in version 2.0 release in Table I-3 | 91 |

| Changes from C Revision (October 2015) to D Revision | Page |
|---|---|
| • Added more detail on the illumination module | 14 |
| • Added command input and output parameters with their respective byte lengths. Added a table note to denote commands supported in Tiva firmware v1.2 or later. | 74 |
| • Added a table note to denote commands supported in Tiva firmware v1.2 or later. | 85 |
| • Added UART command timeout feature | 88 |
| • Added Checksum calculation | 89 |
| • Added UART error packet description | 91 |
| • Added a table note to denote commands supported in Tiva firmware v1.2 or later. | 91 |
| • Added GATT Command Service description | 99 |
| • Changed Bluetooth Multiple Packet description for greater clarity | 100 |

Changes from B Revision (June 2015) to C Revision **Page**

| Changes from A Revision (June 2015) to B Revision | Page |
|---|---|
| • Updated locations of firmware files in Section 3.1.8 | 38 |
| • Deleted OpCode Byte 3 | 74 |
| • Added USB Communications HID packet description | 83 |
| • Added UART packet description | 88 |

| Changes from Original (June 2014) to A Revision | Page |
|---|---|
| • Added missing thermistor from block diagram and moved red LED to bq24250 | 12 |
| • Changed battery requirements | 40 |
| • Changed *DLP Spectrum Library* description | 46 |
| • Added *DLP Spectrum Library Workflow* diagrams | 46 |
| • Added *Bluetooth Client App Workflow* | 53 |
| • Added steps to compile NIRscanNanoGUI and its DLP Spectrum Library under Windows | 66 |
| • Changed Command Handler Table to show the exact USB commands and Bluetooth services | 74 |
| • Added notes to the GATT services characteristics | 94 |

## IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources 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.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

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