

Application Note AN170701

Use of ISP1510 Development Kit



Introduction

Scope

This document gives details on hardware and software for using and testing Insight SiP Bluetooth Low Energy module ISP1510 (model ISP1510-UX).

Contents

1.	Recommended Documentation	2
2.	ISP1510 Dev Kit Hardware Content	4
3.	Software Installation	5
4. 4.1. 4.2. 4.3. 4.4. 4.5. 4.6.	Hardware Description ISP1510-UX Module ISP1510-UX-TB Test Board ISP130603 Interface Board ISP1510-AN Anchor Board ISP1510-TG Tag Board nRF5 series Development Dongle	7 7
5. 5.1. 5.2. 5.3. 5.4. 5.5. 5.6.	Basic Application using ISP1510-UX-TB Test Board Basic BLE Proximity Application Direct Test Mode (UART) UART Mode Example BLE UART Mode Example NFC Launch App Example Simple UWB transmissions using Decawave	11
6. 6.1. 6.2.	Basic Range Application with ISP1510 On Master Control Panel On Android Device	37 37 39

APPLICATION NOT ISP1510 DEV KI

1. Recommended Documentation

t's all in the package

The following documents and Dev Kits (software portion) are required to understand the complete setup and programming methods:

Nordic Semiconductor Documents:

- Insight SiP development kit User Guide (hardware section should be partially ignored Insight SiP development kit hardware replaces Nordic Semiconductor hardware).
- Index nRF52 Series Reference Manual.
- IRF52832 PS (data sheet).
- S132 nRF52832 SoftDevice Specification.
- InRF5 SDK (for software development on the nRF51 and nRF52 Series).

To access documentation, information, go to:

- http://www.nordicsemi.com (Official Nordic Semi website)
- http://infocenter.nordicsemi.com/index.jsp (The Nordic Semiconductor Infocenter is a "comprehensive library" containing technical documentation for current and legacy solutions and technologies)
- https://devzone.nordicsemi.com/questions (Ask any Nordic related question and get help)
- For any question, you can also open a case on the http://www.nordicsemi.com

Nordic Semiconductor Dev kits (software portion):

- \rm hRFgo Studio.
- nRF5 Software Development Kit (SDK):
 - Precompiled HEX files.
 - Source code.
- Keil ARM project files.
- S132 nRF52832 SoftDevice.
- Master Control Panel.

To access these files, go to <u>www.nordicsemi.com</u> and download the files. Instructions can be found in Chapter 3.

DecaWave Documents:

- DW1000 Data Sheet
- 🔸 DW1000 User Manual
- APS013: DW1000 and Two Way Ranging

To access documentation, information, go to:

<u>https://www.decawave.com/support</u> (Official DecaWave website, registration required)



Other ISP documents that complete the above:

- **4** AN1606 App Note this document.
- DS1510 module data sheet.
- ISP1510-UX-TB Test Board schematic "Schematic_ ISP1510-UX-TB".
- ISP130603 Interface Board schematic "SC130604".
- ISP1510 Tag Board schematic "Schematic_ ISP1510-UX-TG".
- ISP1510 Anchor Board schematic "Schematic_ ISP1510-UX-AN".



2. ISP1510 Dev Kit Hardware Content



Document Ref: isp_ble_AN170701_R0.docx

Insight SiP – Green Side – 400 avenue Roumanille – BP 309 – 06906 Sophia-Antipolis Cedex – France – www.insightsip.com The information contained in this document is the property of Insight SiP and should not be disclosed to any third party without written permission. Specification subject to change without notice.

Page 4/41

APPLICATION NOTE ISP1510 DEV KIT

3. Software Installation

lt's **all** in the **package**

This paragraph describes the steps to follow for software installation.

1. Download and install Keil MDK-ARM from https://www.keil.com/demo/eval/arm.htm to your hard drive. After installation, a Pack Installer window appears. Click on the "Packs" section and "Check for updates". After you can show in the Device section "Nordic Semiconductor" on the left side of the screen and different Packs available on the right side of the screen.

2 Device: Nordic Semiconductor - nRF528	832_xxAA			
4 Devices Boards		A Packs Examples		
Search: • ×		k	Action Description	
Device	/ Summary	Device Specific	18 Packs nRF52832_xxAA selected	
E 🏂 All Devices	3515 Devices	-NordicSemiconductor::nRF_ANT	Install ANT services and data modelling support modules.	
ABOV Semiconductor	5 Devices	NordicSemiconductor::nRF_BLE	💠 Up to date Bluetooth Low Energy (Bluetooth Smart) services and software modules for Nordic	: Semicond
+ 🔮 Ambia Micro	8 Devices	B -NordicSemiconductor::nRF_DeviceFamilyPack	Update Nordic Semiconductor nRF ARM devices Device Family Pack.	
Analog Devices	16 Devices	NordicSemiconductor::nRF_Drivers	Up to date Drivers for Nordic Semiconductor nRF family.	
	26 Devices	NordicSemiconductor::nRF_Drivers_External	Up to date Drivers for external hardware used by Nordic Semiconductor nRF family examples.	
+ 🔮 Atmel	260 Devices	NordicSemiconductor::nRF_Examples	Up to date Examples and BSP for Nordic Semiconductor nRF family.	
Cypress	381 Devices	NordicSemiconductor::nRF_Libraries	Up to date Software modules for Nordic Semiconductor nRF family.	
+ - · Freescale	241 Devices	NordicSemiconductor::nRF_NFC	Install NFC services and data modelling support modules.	
 GigaDevice 	40 Devices	NordicSemiconductor::nRF_Properitary_RF	Install Proprietary RF protocols for Nordic Semiconductor nRF family.	
Holtek	19 Devices	■ NordicSemiconductor::nRF_RTX	Install Port of the ARM CMSIS-RTOS based RTX for Nordic Semiconductor nRF family.	
🗉 - 🔗 Infineon	144 Devices	NordicSemiconductor::nRF_Serialization	Up to date Serialization for Nordic Semiconductor nRF family Bluetooth Low Energy (Bluetoot	ch Smart) Sc
🕂 🎐 Maxim	4 Devices	NordicSemiconductor::nRF_SoftDevice_Common	Up to date Common components for Nordic Semiconductor nRF family SoftDevices.	
Mediatek	2 Devices	IN NordicSemiconductor::nRF_SoftDevice_S110	🚸 Up to date Components for Bluetooth Low Energy (Bluetooth Smart) S110 SoftDevice for Nord	dic Semicor
🗉 🌳 Microsemi	6 Devices	B −NordicSemiconductor::nRF_SoftDevice_S120	Install Components for Bluetooth Low Energy (Bluetooth Smart) S120 SoftDevice for Nord	dic Semicor
MindMotion	2 Devices	NordicSemiconductor::nRF_SoftDevice_S130	🚸 Update 🔤 Components for Bluetooth Low Energy (Bluetooth Smart) S130 SoftDevice for Nord	dic Semicor
Nordic Semiconductor	8 Devices	NordicSemiconductor::nRF_SoftDevice_S132	Up to date Components for Bluetooth Low Energy (Bluetooth Smart) \$132 SoftDevice for Nord	dic Semicor
🗉 🔧 nRF51 Series	7 Devices	2.0.0-7.alpha (2015-12-18)	Remove Components for Bluetooth Low Energy (Bluetooth Smart) S132 SoftDevice for Nord	dic Semicor
RE52 Series	1 Device	Previous	NordicSemiconductor::nRF_SoftDevice_S132 - Previous Pack Versions	
nRF52832_xxAA	ARM Cortex+M4, 64 MHz, 64 kB RAM, 512 kB ROM		Install Components for Bluetooth Low Energy (Bluetooth Smart) S132 SoftDevice for Nord	dic Semicor
Nuvoton	433 Devices	NordicSemiconductor::nRF_SoftDevice_S210	Install Components for ANT/ANT+ S210 SoftDevice for Nordic Semiconductor nRF family	y.
- 🔶 NXP	275 Devices	NordicSemiconductor::nRF_SoftDevice_S310	Install Components for Bluetooth Low Energy (Bluetooth Smart) and ANT/ANT+ S310 So	/ftDevice fo
🗉 – 🌳 Renesas	2 Devices	Generic	18 Packs	
Silicon Labs	397 Devices	ARM::CMSIS	Up to date CMSIS (Cortex Microcontroller Software Interface Standard)	
🗉 🎐 SONEX	49 Devices	ARM::CMSIS-Driver_Validation	San Install CMSIS-Driver Validation	
STMicroelectronics	768 Devices	ARM::CMSIS-RTOS_Validation	Install CMSIS-RTOS Validation	
Texas Instruments	341 Devices	ARM::mbedClient	S Install ARM mbed Client for Cortex-M devices	
🛨 🎐 Toshiba	88 Devices	⊕-ARM::mbedTLS	Install ARM mbed Cryptographic and SSL/TLS library for Cortex-M devices	
		⊕ ARM::minar	Install mbed OS Scheduler for Cortex-M devices	
		Keil::ARM_Compiler	🚸 Update Keil ARM Compiler extensions	
		:e:−Keil::Jansson	Install Jansson is a C library for encoding, decoding and manipulating JSON data	
		Keil::MDK-Middleware	In the second se	
			Series Device Support, deprecated: Use "Infineon::XMC1000_DFF	P" instead
		E Keil::XMC4000_DFP	Infineon XMC4000 Series Device Support, deprecated: Use "Infineon::XMC4000_DFF	P" instead
		⊕-lwIP::lwIP	Up to date will be a light-weight implementation of the TCP/IP protocol suite	
		Micrium:RTOS	Install Micrium software components	
				•

On the "Packs" section, you can download and update Nordic example, nRF SoftDevice, nRF DeviceFamilyPack, nRF examples..., etc ...

All the Packs are installed on the following directory: C:\Keil_v5\ARM\Pack\NordicSemiconductor.

- 2. Download and run the J-Link Software and documentation pack for Windows from http://www.segger.com/jlink-software.html. The serial number from your SEGGER J-Link hardware is needed to identify your device and can be found printed on the chip on the J-Link Lite emulator board.
- 3. Go to www.nordicsemi.com and log in to your Nordic My Page account.

Page 5/41



- 4. Go to Products and click on Bluetooth Smart/Bluetooth Energy. You will have access to the different product :
 - a. <u>nRF52 Series</u> : Click on nRF52832 and on the download section you have access to the documentation, SoftDevice, Master control panel, nRFgo studio, SDK ... etc ...
 - b. <u>nRF52 Development Tools</u>: You can download the last nRF5 SDK.
- 5. You can also download the SDK in the following link: <u>https://developer.nordicsemi.com/</u>.
- 6. Download and install nRFgo Studio (Make sure to download the last version updated).
- 7. Download and install Master Control Panel (x86 is for 32 bits windows and x64 is for 64 bits windows).

APPLICATION NOTE ISP1510 DEV KIT

4. Hardware Description

lt's <mark>all</mark> in the <mark>package</mark>

4.1. ISP1510-UX Module

ISP1510-UX is a Bluetooth Low Energy and UWB module with integrated antennas.

Despite its small size of 9.0 x 16.0 x 1.9 mm, the ISP1510 module integrates the DecaWave DW1000 chipset which is an Ultra Wide Band transceiver dedicated to indoor localization applications. This chipset needs an external processor to operate and Insight SiP decided to integrate the Nordic Semi nRF52 SoC. In addition to the powerful ARM Cortex M4F MCU, this chip also provides a BLE and NFC connectivity for wireless set up and control of the UWB chip.



ISP1510 also integrates all passives, crystals and DC/DC converters to optimize the power consumption. And last but not least, it integrates 2 antennas in the same package. This is a new and unique concept developed by Insight SiP within a SIP package. For BLE function, we keep using our usual PCB trace antenna design. For UWB function, the second antenna is printed directly on top of the molding.

For more details, see Insight SiP module data sheet (document DS1510).

4.2. ISP1510-UX-TB Test Board

ISP1510-UX-TB is the basic application test board that has dimensions of 50 x 60 mm².



Page 7/41

Document Ref: isp_ble_AN170701_R0.docx



It encloses:

- ISP1510-UX UWB and BLE module
- **4** 3 x FPC connectors in order to access the nRF52832 GPIOs:
- 1 x 10 pin FPC connector on top side of the board.
- 1 x 14 pin FPC connector on top side of the board.
- 1 x 22 pin FPC connector on top side of the board.
- 1 x 10 pin header order to access the decaWave GPIOs

The ISP1510-UX-TB electrical schematic is presented in document Schematic_ ISP1510-UX-TB.

4.3. ISP130603 Interface Board

ISP130603 is the application type interface board that has dimensions of $100 \times 80 \text{ mm}^2$. The ISP130603 electrical schematic is presented in document SC130604.



July 5, 2017

Page 8/41

Document Ref: isp_ble_AN170701_R0.docx



4.4. ISP1510-AN Anchor Board

ISP1510 Anchor Board is one of two the application boards that is used for the ranging demonstration. It has dimensions of $62.5 \times 40 \text{ mm}^2$.

It encloses:

4 ISP1510-UX UWB and BLE module

- 4 2 x mini-LEDs
- 4 USB connector
- Removable 10 pin FPC connector for software loading
- \rm FTDI USB-to-Serial adapter
- UWB Antenna on PCB
- Ranging firmware

The ISP1510-AN Anchor Board electrical schematic is presented in the document Schematic_ISP1510-UX-AN.



4.5. ISP1510-TG Tag Board

ISP1510 Tag Board is one of two the application boards that is used for the ranging demonstration. It has dimensions of $31 \times 18 \text{ mm}^2$.

It encloses:

- ISP1510-UX UWB and BLE module
- 4 2 x mini-LEDs
- Battery holder
- Removable 10 pin FPC connector for software loading
- 4 Ranging firmware

The ISP1510-TG Tag Board electrical schematic is presented in the document Schematic_ISP1510-UX-TG.





4.6. nRF5 series Development Dongle

The reader should refer to the corresponding paragraph in nRF52832 Development Kit User Guide document.



5. Basic Application using ISP1510-UX-TB Test Board

5.1. Basic BLE Proximity Application

This paragraph shows you how to set up and program a BLE proximity application on top of a SoftDevice that will send data on a Bluetooth link from the ISP1510-UX-TB Test Board to the Master Emulator. In order to use Bluetooth Low Energy radio, the software is loaded in 2 parts:

- 4 S132 SoftDevice using nRFgo Studio (hex file, no source).
- Proximity Application using Keil uVision.

Then Master Emulator is connected and Proximity Application is launched.

S132 SoftDevice loading

- 1. Connect the provided USB cable from the Interface Board ISP130603 to your computer.
- 2. Connect the ISP1510-UX-TB Test Board to the ISP130603 Interface Board with the 10 pin, 14 pin and 22 pin FPC jumper cables (0.5 mm pitch, provided in the Development Kit).



- 3. Start nRFgo Studio.
- 4. Select nRF5x Programming.



5. Click Erase all.

 2.4 GHz Front-End Te 	SEGGER to use: 518004334 Refresh		
TX carrie RX const	nRF51822		
TX/RX c RX sensit	Region 1 (Application)	Program SoftParies Deserve Audiophias Deserve	Restlander
 Bluetooth nRF8001 Co 	Region 1 (appleador)	Programming of SoftDevice on nRF5x	device
Dispatcher Trace Trapel		File to program: rf51822 8.0.0 softdevice.hex	Browse
Direct Test		Lock SoftDevice from readback	k
nRF8002		SoftDevice size (HB): 0	
		Enable SoftDevice protection ((UICR.CLENRO)
	Size: 128 kB		
		Program Verify	Read
nRF24LU1+ Boo			
og			

6. Browse to SoftDevice hex file and click Program. The SoftDevice is available on the Nordic Website or by installing the SoftDevice S132 with the Pack Installer, for example : C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_SoftDevice_S132\2.0.0-7.alpha\hex

Page 12/41

Document Ref: isp_ble_AN170701_R0.docx



💦 nRFgo Studio					
File View nRE8001	Setup Help				
Features X					
 2.4 GHz Front-End Te TX carrie RX const 	SEGGER to use: 518004330 Refresh				
TX/RX c RX sensit	Region 1 (Application)	Program SoftDevice Program Application Program Bootloader			
 Bluetooth nRF8001 Co 		Programming of SoftDevice on nRF5x device			
Dispatcher Trace Transl	Size: 404 kB	File to program: 2.0.0-7.alpha_softdevice.hex Browse			
Direct Test nRF8002		SoftDevice size (kB): 108			
	Address: 0x1b000	Enable SoftDevice protection (UICR.CLENR0)			
	Region 0 (SoftDevice)				
	Size: 108 kB	Program Verify Read			
	Firmware: S132_nRF52_2.0.0-7.alpha (Id: 0x0079)				
Device Manager X Motherboards nRF5x Program nRF5x Bootloader nRF24LU1+ Boo	Erase all				
Log		×			
Unable to find or open Recover completed Erase completed	the JLinkARM dll	^ III			
Softdevice C:/Keil_v5/A	RM/Pack/NordicSemiconductor/nRF_SoftDevice_S132/2.0.0-7.alpha/hex/	s132_nrf52_2.0.0-7.alpha_softdevice.hex programmed successfully			

Page 13/41



Proximity Application loading

- 1. Start Keil uVision.
- 2. Select Project then Open Project in order to open Proximity app. Make sure it is the right file project. The project is locked, it is read only, if you want to modify it, you have to change the right in the file App directory properties.

(Ex: C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha\ble_peripheral\ble_app_proximity\pca10040\s132\arm5 \ ble_app_proximity_s132_pca10040.uvprojx).



If some Install Softwares Packs is missing, a window appears, and you can install it.

Page 14/41



3. Click Build Target and Load. If you have this message after click Load, it means that the power is not enough and you have to increase the voltage by adding jumpers on the interface board.



You can also load the hex file (generated after building target with keil uVision) by nRFgo studio in the program application.

🔀 nRFgo Studio		
Eile View nRF8001 S	etup <u>H</u> elp	
Features X 2.4 GHz Front-End Te TX carrie DV second	SEGGER to use: S18004330 • Refresh	
TXPR c Resent Buetoch nR9801 Co Dispatcher Trace Frand Direct Test nR98002	Region 1 (Application) Size: 40488 Address: 0x1b000 Region 0 (SoftDevice) Size: 108188 Firmware: S132_mRFS2_2.0.0-7.alpha (Id: 0x0079)	Program SoftDevice Program Application Program Bootbader Programming of Application on nRF5x device Bernse Bernse File to program: 5/j.budl/rrf52832.yxxa.g.132.hzr Brense Lock entire drip from readbadk Brense Bernse
Device Manager × Motherboards nPFS: Program nPFS: Bootbader nRF24UU+ Boo	Erse al	
100		
Erase completed Softdevice C:/Keil_v5/ARI Application C:/Keil_v5/AI	M/Pack/NordicSemiconductor/nRF_SoftDevice_S132/20.0-7.alpha/hev/ RM/Pack/NordicSemiconductor/nRF_Examples/11.00-2.alpha/ble_perip ====================================	s132_nrf52_20.0-7 alpha_softdevice.hex programmed successfully heral/ble_app_proximity/pcs10040/s132/arm5/_build/nrf52832_xxas_s132.hex

The file project is located for example:

C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha\ble_peripheral\ble_app_proximity\pca10040\s132\arm5_build _build\ nrf52832_xxaa_s132.hex).



Master Emulator and Proximity Application

- 1. Connect nRF51 Dongle (Master Emulator) into a USB port on your computer.
- 2. Start Master Control Panel. If you have no master emulator found, you have to flash the dongle in Flash programming section.



3. Click Start Discovery.

Master Control Panel	
File Help	
Master emulator	
COM113 - 680791841 - 680791841 connected	Reset
Scan for devices	
Stop discovery	
Discovered devices	
Select device	
Delete bond info	
Log [16:57:49 1] Beady	
[16:57:49.1] SERVER: Server has started	
[16:57:49.6] Device discovery started	ſ
[16:58:13.2] Device discovery stopped	
[16:58:13.9] Device discovery started	

 July 5, 2017
 Page 16/41
 Document Ref: isp_ble_AN170701_R0.docx

 Insight SiP – Green Side – 400 avenue Roumanille – BP 309 – 06906 Sophia-Antipolis Cedex – France – www.insightsip.com
 The information contained in this document is the property of Insight SiP and should not be disclosed to any third party without written permission. Specification subject to change without notice.



- 4. After starting discovery, if no device appearing, disconnect and connect again the interface board power supply.
- 5. If you have an error message as indicated in the photo :



It means that the nRF5 Dongle is not programmed. For your information, please refer to the nRF51822 Development Kit User Guide document as indicated in our application note ISP130301-DK1 part 4.5 on page 14-7. You can find this user guide on the Nordicsemi website. The procedure to program the nRF51 Dongle is described on page 13. I enclose a copy of the user guide for your convenience.

- 1) Open the Master Control Panel from the Start menu (Start > All Programs > Nordic Semiconductor > Master Control Panel).
- 2) Make sure the Development Dongle is detected. The Master Emulator item list should show COMnn-xxxxxxxx (nn gives the COM port number; xxxxxxxx is the SEGGER serial number printed on the dongle). Restart the application if it doesn't appear in the item list. Before continuing, make sure you have selected the correct device by verifying the serial number in the item list with the serial number printed on the Development Dongle.
- 3) When you use the Development Dongle for the first time, you must first program it with the Master Emulator Firmware.
 - a. In the Master Control Panel menu click File and select Flash Programming.
 - b. Click Browse. This opens a browser that automatically points to the location of the
 - c. mefw_nrf51822_<version>_firmware.hex (<version> will be replaced by a number
 - d. giving the version of the actual firmware).
 - e. The Master Control Panel Firmware file is located in:
 - f. C:\Program Files (x86) \ Nordic Semiconductor \ Master Control Panel \<version>\firmware\
 - g. pca10000/MEFW_nRF51822_<version>_firmware.hex.
 - h. Select the Master Emulator Firmware file and click Open.
 - *i.* Click Program to start programming the selected device.
 - j. When the programming is finished click Exit to go back to the main window.
- 6. Click Select Device.
- 7. On the following display, click successively on Bond, Discover Services and Enable Services.

PLICATION NOTE ISP1510 DEV KIT

🖸 Master Control Panel			
File Help			
Master emulator			
COM113 - 680791841			
Device info Device address: C9E7671F2E15 Bonded: True			
Actions			
Disconnect Bond Update Discover services Disable services DFU			
Service Discovery			
PrimaryService, Value: 02-18, Immediate Alert (0x1802) CharacteristicDeclaration, Value: 04-0E-00-06-2A, Properties: WriteWithoutResponse, Cha AlertLevel, (No values read) PrimaryService, Value: 03-18, Link Loss (0x1803) CharacteristicDeclaration, Value: 0A-11-00-06-2A, Properties: Read, Write, Characteristic L AlertLevel, (No values read) PrimaryService, Value: 0F-18, BatteryService (0x180F) CharacteristicDeclaration, Value: 12-14-00-19-2A, Properties: Read, Notify, Characteristic L Battery Level, Value: 64			
UUID (0x): Handle (0x): Display as UTF8 Read long Read			
Value: hex text Write long Write			
Back			
Log			
[17:00:21.7] ConnectionParameterUpdateResponse sent			
[17:00:22.0] Connection Parameters Updated. ConnInterval:500ms, SlaveLatency:0, Supervisio			
[17:00:22.9] Enable Services({0x0015:1,})			
[17:00:23.5] Updated handle 0015 with value [1, 0]			
[17:00:23:5] Successfully updated the store value of CCCD			

It's all in the package

- 8. You can note Battery voltage is sent by the ISP1510-UX-TB Test Board to the Master Emulator via the Bluetooth link. The application is written to send a value that changes cyclically.
- 9. You can also use the "nRF Master Control Panel" application which is available for iOS from Apple Store and for Android from Play Store. Download, install and run the application, click to SCAN and select the device: Nordic_PROX.

<u>Important notification</u>: Pay attention to the compatibility between the IC revision, the SoftDevice, the nRF5 SDK, etc, ... Please read the compatibility matrix available on the "infocenter" website from Nordic.

http://infocenter.nordicsemi.com/index.jsp



5.2. Direct Test Mode (UART)

This paragraph shows you how to set up and program the Direct Test Mode through the UART on ISP1510-UX-TB Test Board.

Direct Test Mode Set-up

- 1. Connect the USB cable from the Interface Board ISP130603 to your computer.
- 2. Connect the ISP1510-UX-TB Test Board to the ISP130603 Interface Board with the 10 pin, 14 pin and 22 pin FPC jumper cables (0.5 mm pitch, provided in the Development Kit)
- 3. On the ISP130603 Interface Board, connect the 2-lead patch cable in order to connect:
 - RXD to P0_08
 - TXD to P0_06

Make sure the RXD/TXD labels match for each wire. This matches the default setting if you are using the Nordic Board PCA10040 in the SDK project (be careful: depending on the Nordic Board version you are using, the ports used could be different, see next part: Direct Test Mode loading).

Direct Test Mode Loading

- 1. Start Keil uVision.
- Select Project then Open Project in order to open Direct Test Mode application: C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha\ dtm\direct_test_mode\ pca10040\blank\arm5\direct_test_mode_blank_pca10040.uvprojx.

<u>Warning</u>: Regarding the Nordic Board you are using, the RX and TX pin number could be different. In the following pictures, the Nordic board PCA10040 is used (you can modify the Nordic Board model in : Flash \rightarrow Configure Flash Tools \rightarrow C/C++, by writing the correct Board name in the "Define" area with the name indicated in the boards.h).



Configure Flash				
Contigure Flash	▲ 등 ♦ ♦ 🙆			
jeu	h Tools 📄 nrf6310.h 😴 main.c			
Project direct test_mode_blank_pca mf51422_xac ble_dtm.h core_crm0.h core_crm0.h core_crm0.h core_crm0.h core_crm0.h dif1.h mf51.h mf51_deprecated.h mf51_oh stdint.h stdint.h ble_dtm.h mf51_deprecated.h mf51_depre	<pre>1 = //* Copyright (c) 2014 Nordic Semicond 2 * The information contained herein i 4 * Terms and conditions of usage are 5 * SEMICONDUCTOR STANDARD SOFTWARE L1 6 * 7 * Licensees are granted free, non-tr. 8 * WARRANTY of ANY KIND is provided. 7 9 * the file. 1 * 2 = * ifndef BOARDS_H 1 = */ 2 = * ifndef BOARDS_H 1 = */ 1 = */ 2 = */ 1 = */ 2 = */</pre>	uctor. All Rights Reserved. s property of Nordic Semiconduc described in detail in NORDIC CENSE AGREEMENT. ansferable use of the informati This heading must NOT be remove Device Target Output Listing User Preprocessor Symbols Define: [BSP_DEFINES_ONL_BOA Ugdefine: Language / Code Generation □ Controls [Evel 3 (O3)] □ Optimize for Time □ Split Load and Store Multiple V One ELF Section per Function Include Paths Controls [-C99] Compiler Compiler Compiler Optimizetion: [Level 3 (O3)] □ Optimize for Time □ Split Load and Store Multiple V One ELF Section per Function Include Paths Controls [-C99] Controls [-C99] Compiler Optimize for Contex-MO - EVA Controls = [-Optimize for Contex-MO - EVA] Controls = [-Optimize for Contex-MO - EVA] Contex = [-Optimize for Contex + [-	Itor ASA.	Warnings: anspecified> Thung Mode No Auto Includes C 59 Mode int stt_sections 4c99 Help

In the following picture, the Nordic board PCA10040 is used by default and the RX and TX pin number are 08 and 06.



K: C\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha\dtm\direct_test_mode\pca10040\blank\arm5\direct_test_mode_blank_pca10040.uvprojx[Read Only]					
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help					
🗋 🔐 🛃 🐉 🐁 🛍 🔄 🗢 🏱 🖛 🔶 隆 🎕 聰 🎼 🎼 🎼 🎉 🦉 ADIO_TXPOWER_TXPOW 🕢 🗟 🎺 🔍 🌢 💿 🔗 🍓 💷 🔦					
😒 🗏 🕮 🥪 📇 🕎 nrf52832 xxaa 🔍 🔊 🕼 📥 📥 📥 🛸					
Project					
	<u>pca10040.h</u> * main.c				
Project: direct_test_mode_blank_pca	34 <pre>#define BSP_LED_2_MASK (1<<bsp_led_2)< pre=""></bsp_led_2)<></pre>				
🖃 泸 nrf52832_xxaa	35 #define BSP_LED_3_MASK (1< <bsp_led_3)< th=""></bsp_led_3)<>				
🖻 🗁 Application					
🖃 🛍 main.c	3/ #GETINE LEDS MASK (BSP_LEL_U MASK BSP_LEL_I MASK BSP_LEL_Z MASK BSP_LEL_Z MASK)				
ble dtm.h	30 /* all LEDS are lit when GFIO IS 100 */				
P hoards h					
Sources in the second s	41 #define BUTIONS NUMBER 4				
osp.n	42				
cmsis_armcc.h	43 #define BUTTON START 13				
compiler_abstraction	44 #define BUTTON_1 13				
🔤 🚰 core_cm4.h	45 #define BUTTON_2 14				
🔄 🔄 core_cmFunc.h	46 #define BUTTON_3 15				
Core cmInstr.h	47 #define BUTTON_4 16				
Core cmSimd b	48 #define BUTTON_STOP 16				
S and b	49 Factine Bullon_POLL NRF_GPIO_PIN_POLLOP				
nrt.n	50 Idefine PILTONS LIST (PILTON 1 PILTON 2 PILTON 2 PILTON 4)				
nrf51_to_nrf52.h	52				
	53 #define BSP BUTTON 0 BUTTON 1				
nrf52_bitfields.h	54 #define BSP BUTTON 1 BUTTON 2				
🔤 👕 nrf_gpio.h	55 #define BSP BUTTON 2 BUTTON 3				
[©] pca10040.h	56 #define BSP_BUTTON_3 BUTTON_4				
ttdbool b	57				
	58 #define BSP_BUTTON_0_MASK (1< <bsp_button_0)< th=""></bsp_button_0)<>				
Staint.n	59 #define BSP_BUTTON 1 MASK (1< <bsp_button 1)<="" th=""></bsp_button>				
system_nrt52.h	60 Factine BSP_BUTTON 2 MASK (1< <spe_button 2)<="" th=""></spe_button>				
Documentation	61 #define bar_bullow_3_mask (1< bullow_3)				
🗢 😌 Board Support	63 idefine BITTONS MASK 0x001E0000				
	64				
🗉 🐟 Device	65 #define RX PIN NUMBER 8				
nRE BLE	66 #define TX_PIN_NUMBER 6				
aPE Drivers	67 #define CTS_PIN_NUMBER 7				
MKr_Drivers	68 #define RTS_PIN_NUMBER 5				
	69 #define HWFC true				
	78				
	71 #define SPIS_MISO_PIN_28 // SPI_MISO_signal.				
	72 factine SPIS_CSN_FIN 12 // SPI CSN Signal.				
	74 idefine SPIS_SCK PIN 29 // SPI SCK signal.				
	76 #define SPIMO SCK PIN 29 // SPI clock GPIO pin number.				
	77 #define SPIMO MOSI PIN 25 // SPI Master Out Slave In GPIO pin number.				
	78 #define SPIMO_MISO_PIN 28 // SPI Master In Slave Out GPIO pin number.				
	79 #define SPIM0_SS_PIN 12 // SPI Slave Select GPIO pin number.				
	80				
	81 #define SPIM1_SCK_PIN 2 // SPI clock GPIO pin number.				
🔟 Project 🌀 Books 🕄 Func 🛛 🙀 Temp					

- 3. Click Build Target and Load.
- 4. If you have the next error message, it means that you have not enough RAM to flash algorithm.

Page 21/41



File Edit View Project Flash Debug Peri	pherals Tools	SVCS Window Help	
🗋 😂 🖬 🖉 🕹 🍇 🖎 🗠 🗠 🗠	→ P B B	& 🐘 津 洋 /// /// 🖄 baud 🕢 🖳 🖓 🍳 🔶 🔿 🔗 🔝 🔍	
🧇 🖾 🖽 🧼 🧮 🙀 nrf51822_xxab_s110_	spi 🖃 🌋 🛔		
Project 🛛 🗖 🔝	📩 main.c	LC	•
	1 📮	3/* Copyright (c) 2014 Nordic Semiconductor. All Rights Reserved.	
😑 🚎 Source Code	2	•	
i main.c	3	* The information contained herein is property of Nordic Semiconductor ASA.	
🖻 🔄 Libraries	5	Unk - Cortex Error CiKeil vS\ARM\RACK\NordicSemiconductor\nRE DeviceEa	
ble_advdata.c	6		
H app_timer.c	7	ormation. NO	
H app_outton.c	8	Cannot Load Flash Programming Algorithm ! removed from	
Sondevice_nancier.c	9		
	10		
ser codecs mw	12		
🗐 🧰 ser utils	13 E	OK	
🖬 🧰 ser hal	14		
ser_hal_serial	15	* @defgroup ble_sdk_app_dtm_main main.c	
CMSIS	10		
🖬 🚸 Device	18	v Vision file.	
	19		
	20	* 1 A activation example.	
	21	*/ Error: Flash Download failed - "Cortex-M0"	
	22 -		
	24	7 - 113 	
	25	#ing OK	
	26	#inq hereita	
	27	#ind	
	28	finclude "ble advdata.h"	
	30	finclude "borga,b"	
	31	#include "nordic common.h"	
	32	<pre>#include "softdevice_handler.h"</pre>	
	33		
	34	<pre>#define DTM_INIT_BUTTON_PIN_NO BUTTON_0 /**< Button to initializing DTM mode</pre>	on co
	36	#define READY LED PIN NO LED 0 /**< LED indicating that the example	is re
	37	#define DTM READY LED PIN NO LED 1 /**< LED indicating that the connect:	ivity
	38	<pre>#define ASSERT_LED_PIN_NO LED_7 /**< Is on when application has asset</pre>	rted.
	39		
Project Books {} Functions 0, Templates		III	
uild Output			ņ
Atchpoints: 2			
ING Speed: 2000 kHz			
nsufficient RAM for Flash Algorith	ams !		
Tase falled:			
rror: Flash Download failed - "C	Cortex-M0"		
	_	III	•

5. You can modify the size of the RAM Algorithm in : *Flash→ Configure Flash Tools…* and on the windows click on *Debug → Settings* Click on *Flash Download* and modify the size (put 0x02000).

Use <u>S</u> imulator Settings Umit Speed to Real-Time	Use: J-LINK / J-TRACE Cortex Settings	Download Function Down C Erase Full Chip Program C Erase Sectors P Verfar Sant Dr.20000000 Size Dr.20000
Load Application at Startup Run to main() nitialization File:	✓ Load Application at Startup ✓ Run to main() Initialization File:	Pogramming Algorithm Pogramming Algorithm
Edit	Edit	Description Device Size Device Type Address Range
Restore Debug Session Settings F Breakpoints Toolbox Watch Windows & Performance Analyzer Memory Display System Viewer	Restore Debug Session Settings IF Breakpoints IF Toolbox IF Watch Windows IF Memory Display IF System Viewer	nRF51xxx 2M On-chip Rash 00000000H - 001FFFFFH
PU DLL: Parameter:	Driver DLL: Parameter:	Start: Size:
SARMCM3.DLL	SARMCM3.DLL	Add Remove
Nalog DLL: Parameter: DARMCM1.DLL PCM0	Dialog DLL: Parameter: TARMCM1.DLL pCM0	
ОК Са	incel Defaults Help	

6. Click OK and load again the project.



Direct Test Mode Testing

- 1. Start nRFgo Studio.
- 2. Select Direct Test Mode.

🗙 nRFgo Studio – Direct Test Mode UART interface			
ile <u>V</u> iew <u>n</u> RF8001 S	Setup <u>H</u> elp		
eatures ×	Direct Test Mode	UART interface	
2.4 GHz Econt-End Te	Set up on	Program	
TX carrie			
RX const	Com port COM28	Refresh list of com ports	
TX/RX c	Mode		
Bluetooth	Transmit	Receive	
nRF8001 Co	Channel		
Dispatcher Trace Transl	Single	🔘 Sweep	
Direct Test			
nRF8002	Channel	19	
	Pavload model	Constant carrier	
	Payload Inodel		
	Payload length	I Dytes y	
	Packets received	N/A	
		Start test	
vice Manager X			
Motherboards			
nRF5x Bootloader			
nRF24LU1+ Boo			

For details on how to use the Direct Test Mode, press F1 to open the nRFgo Studio help. 3.

Important notification: Erase all before loading Direct Test Mode program. The SoftDevice must not be loaded, only the Program Application with uvision or with nRFgo studio in "Program Application" (load the .hex generated by uvision).



5.3. UART Mode Example

This paragraph shows you how to set up and program a communication by sending some characters through the UART interface on ISP1510-UX-TB Test Board. This example just echoes input characters from the PC terminal.

UART Mode Set-up

- 1. Connect the USB cable from the Interface Board ISP130603 to your computer.
- 2. Connect the ISP1510-UX-TB Test Board to the ISP130603 Interface Board with the 10 pin, 14 pin and 22 pin FPC jumper cables (0.5 mm pitch, provided in the Development Kit)
- 3. On the ISP130603 Interface Board, connect the 2-lead patch cable in order to connect:
 - RXD to P0_11
 - TXD to P0_06
 - CTS to P0_07
 - RTS to P0_05

Make sure the RXD/TXD and CTS/RTS labels match for each wire. This matches the default setting if you are using the Nordic Board pca10040 in the SDK project (be careful: depending on the Nordic Board version you are using, the ports used could be different, see next part: UART Mode loading). CTS and RTS are needed because in the UART process when the TX is ready to send (RTS), the RX needs to allow the TX send data (CTS) and vice versa.

UART Mode Loading

- 1. Start Keil uVision.
- 2. Select Project then Open Project in order to open UART_example application: C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha\peripheral\uart \pca10040\arm5\uart_pca10040.uvprojx.



C:\Keil_v5\ARM\Pack\NordicSemicondu	uctor\nRF_Exan	nples\11.0.0-2.alpha\peripheral\uart\pca10040\arm5\uart_pca10040.uvprojx - µVision	- 🗆 ×
File Edit View Project Flash Debug	g Peripherals	Tools SVCS Window Help	
📄 💕 🖬 🎒 🐰 🛍 🛍 🌱 🥲	← ⇒ १	🎙 隐 隐 健 律 /// //// //////////////////////	
🔅 🍱 🎬 🧼 🧮 🙀 nrf52832_xxa	a v	永 書 号 ◆ 今 曲	
Project 📮 🗵	🔬 mair	n.c] pca10040.h	▼ ×
🖃 🍄 Project: uart_pca10040	49	#define BUTTON_PULL NRF_GPIO_PIN_PULLUP	^
🖻 🚂 nrf52832_xxaa	50	Adding RITTONS LIST (RITTON & RITTON & RITTON & RITTON &)	
Application	52	suffice Bollows_Lisi (Bollow_1, Bollow_2, Bollow_3, Bollow_3)	
🖽 🍇 main.c	53	<pre>#define BSP_BUTTON_0 BUTTON_1</pre>	
Board Support	54	<pre>fdefine BSP_BUTTON_1 BUTTON_2 tdefine BSP_BUTTON_3</pre>	
	56	define BSF BUTTON 3 BUTTON 4	
	57		
🗉 🍓 nRF Drivers	58	<pre>fdefine BSP_BUTTON 0 MASK (1<<bsp_button (1<<bsp_button="" 0)="" 1="" 1)<="" bsp_button="" idefine="" mask="" pre=""></bsp_button></pre>	
🗈 🍖 nRF_Libraries	60	<pre>#define BSP_BUTTON 2 MASK (1<<bsp_button 2)<="" pre=""></bsp_button></pre>	
	61	<pre>#define BSP_BUTTON_3_MASK (1<<bsp_button_3)< pre=""></bsp_button_3)<></pre>	
	62	fdefine BUTTONS MASK 0x001E0000	
	64		
	65	#define RX PIN NUMBER 11 //8	
	67	foeline 12_PIN_NUMBER 6 idefine CTS PIN NUMBER 7	
	68	#define RTS_PIN_NUMBER 5	
	69	fdefine HWFC true	
	70	fdefine SPIS MISO FIN 28 // SPI MISO signal.	
	72	#define SPIS_CSN_PIN 12 // SPI CSN signal.	
	73	<pre>#define SPIS MOSI PIN 25 // SPI MOSI signal.</pre>	
	74	foerine SPIS_SCK_PIN 29 // SPI SCK signal.	
	76	<pre>#define SPIM0_SCK_PIN 29 // SPI clock GPIO pin number.</pre>	
	77	fdefine SPIMO MOSI PIN 25 // SPI Master Cut Slave In GPIO pin number.	
	79	<pre>fdefine SFIM0_SFIN 20 // SFI Master in Slave Out GFIO Din number. fdefine SFIM0_SF FIN 12 // SFI Slave Select GFIO pin number.</pre>	
	80		
	81	fdefine SPIMI SCK PIN 2 // SPI clock GPIO pin number.	
	83	define SFIM MIS FIN 5 // SFI Master Out Slave In GFIO DIN number.	
	84	<pre>#define SPIM1_SS_PIN 5 // SPI Slave Select GPIO pin number.</pre>	
	85	the fire SDIN2 SCY DIN 12 // SDI clock CDIO nin number	
	87	define SFIMZ_SCA_FIN 12 // SFI Clock GFIO pin number.	
	88	#define SPIM2_MISO_PIN 14 // SPI Master In Slave Out GPIO pin number.	
	89	<pre>#define SPIM2_SS_PIN 15 // SPI Slave Select GPIO pin number.</pre>	
	90	// serialization APPLICATION board - temp. setup for running serialized MEMU tests	
	92	<pre>#define SER_APP_RX_PIN 31 // UART RX pin number.</pre>	
	93	#define SER APP TX PIN 30 // UART TX pin number. tdefine SEP APP TS PIN 28 // UPT Clear To Send pin number.	
	95	facine SER APP RIS PIN 29 // URAT Request To Send pin number.	
	96		
	97	<pre>stafine SER APP SPIMO SCK PIN 2 // SPI clock GPIO pin number. idefine SFP APP SPIMO MOST PIN 4 // SPI Master Our Slave In GPIO pin number</pre>	
	99	tefine SR APP SPIMO MISO FIN 3 // SFI Master In Slave Out GPIO pin number	~
🔚 Project 🧒 Books {} Func 0→ Temp	<		>

3. Click Build Target and Load.

Document Ref: isp_ble_AN170701_R0.docx



UART Mode Testing

1. Download and install the program "Putty.exe" (or equivalent like TTERM for example) in order to configure the baudrate, the port COM, .. etc ...

Real PuTTY Configuration	States and the second	×				
Category:						
□ Session	Options controlling	local serial lines				
	Select a serial line					
- Terminal	Serial line to connect to	COM28				
Bell Features	Configure the serial line					
⊡ · Window	Speed (baud)	38400				
Appearance Behaviour	Data <u>b</u> its	8				
Translation	Stop bits	1				
Colours	<u>P</u> arity	None 👻				
Connection	Flow control	RTS/CTS ▼				
Data Proxy Telnet Rlogin ●- SSH Senal						
About	C)pen <u>C</u> ancel				

- 2. Click to Open.
- 3. You can write and/or delete some characters. Sometimes you have to reset the ISP130603 Interface Board in case if it is not working.

ළු COM28 - PuTTY	
	*
Start: Welcome to UART example program !!!!q	
Exit!	

4. Press 'q' to exit



5.4. BLE UART Mode Example

This paragraph shows you how to set up and program an example that emulates a serial port over BLE. In the example, Nordic Semiconductor's development board serves as a peer to the phone application "nRF UART", which is available for iOS from Apple Store and for Android from Play Store. In addition, the example demonstrates how to use a proprietary (vendor-specific) service and characteristics with the SoftDevice. In order to use Bluetooth Low Energy and UART interface, the software is loaded in 2 parts:

- The SoftDevice: **S132**.
- ble_app_uart using Keil uVision.

UART Mode Set-up

- 1. Connect the USB cable from the Interface Board ISP130603 to your computer.
- 2. Connect the ISP1510-UX-TB Test Board to the ISP130603 Interface Board with the 10 pin, 14 pin and 22 pin FPC jumper cables (0.5 mm pitch, provided in the Development Kit)
- 3. On the ISP130603 Interface Board, connect the 2-lead patch cable in order to connect:
 - RXD to P0_11
 - TXD to P0_06
 - CTS to P0_07
 - RTS to P0_05

Make sure the RXD/TXD and CTS/RTS labels match for each wire (be careful: depending on the Nordic Board version you are using, the ports used could be different, see next part: UART Mode loading).

CTS and RTS are needed because in the UART process when the TX is ready to send (RTS), the RX needs to allow the TX send data (CTS) and vice versa.

S132 SoftDevice loading

- 1. Start nRFgo Studio
- 2. Select nRF5x Programming
- 3. Click Erase all
- Browse to SoftDevice hex file and click Program. The SoftDevice is available on the Nordic Website or by installing the SoftDevice with the Pack Installer, for example: C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_SoftDevice_S132\2.0.0-7.alpha\hex



UART Mode Loading

- 1. Start Keil uVision.
- Select Project then Open Project in order to open ble_app_uart application: C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha\ble_peripheral\ ble_app_uart\pca10040\s132\arm5 \ble_app_uart_s132_pca10040.
- 3. Click Build Target and Load.

Insight SiP can provide the Hex files on demand at contact@insightsip.com



Document Ref: isp_ble_AN170701_R0.docx



BLE Mode Testing

- 1. Download and install "nRF UART" on your iOS or Android device, which is available for iOS from Apple Store and for Android from Play Store.
- 2. Run the App, click to Connect and select the device: Nordic_UART.

* ¥ © ‴্রা 24% এ 11:11 nRF UART v2.0	■ ★ ≤ ∅ 𝑘 🖬 24% ≜ 11:11 ■ nRF UART v2.0	■ * ¥ © ₩/ 24% = 11:11	nRF UAR	T v2.0	≹≮∅⊈""	24% 🗎	11:11			
Connect	nRE LIART v2 0 ^{00nnect}	Disconnect	Disconnect							
	Select a device Image: Construction of the second sec	[11:11:17] Connected to: Nordic_UART	[11:11:17] Connec [11:11:28] TX: allo	ted to: Nord	dic_UART					
	Cancel					\$	Send			
			Device: Nordic_U	ART - ready			-			
			LEAT	je	рс	our	>			
			1 2 3	4 5	6 7 8	9	0			
			a z e	r t y	y u i	0	р			
			q s d	f g l	h j k	1	m			
Send	Send	Send	↑ w	xc	v b n		×			
Device: <select a="" device=""></select>	Device: <select a="" device=""></select>	Device: Nordic_UART - ready	Sym ,*	 Français((FR) •	·	4			

3. You are now able to send and receive data through the BLE. Next step is to establish the communication with the UART interface.

UART Mode Testing

1. Download and install the program "RealTerm" (or equivalent like Putty, TTERM for example) in order to configure the baudrate, the port COM, .. etc ...

Display Port	Capture Pins Send Echo Port 12C	12C-2 12CMisc Misc	1	\n Clear Freeze ?
Display As ← Ascii ← Ansi ← Hextspace] ← Hextspace] ← Hext+Ascii ← Hext+Ascii ← Hext+Ascii ← Hext+Ascii ← HextSt ← Hoat4 ← HextSt ← H	Half Duplex newLine mode Invert ZBits Big Endian Data Frames Bytes 2 € Single Gulp Terminal Eont 16 € 80 € Scale	ollback		Status Disconnect RXD (2) TXD (3) CTS (8) DCD (1) DSR (6) Ring (9) BREAK Error
		Char Count:10	CPS:0	Port: 28 38400 8N1 RTS/ //



Display Port	Capture Pins Send Echo Port 12	C 12C-2	2 I2CMisc Misc	<u>\n</u>	Clear Freeze ?
Baud 38400 Parity	Port 28 Stop Bits G 8 bits G 7 bits G 6 bits G 5 bits G 5 bits DTR/DSR ⁺ HS485-rts	Open Sp Software □ Receir □ Transr	y Change Flow Control ve Xon Char: 17 mit Xoff Char: 19 Winsock is: C Raw € Telnet		Status Disconnect XXD (2) TXD (3) CTS (8) DCD (1) DSR (6) Ring (9) BREAK Error
You can use Act	iveX automation to control me!		Char Count:10	CPS:0	Port: 28 38400 8N1 RTS/

Display Port Capture Pins Send Echo Port 12C 12C-2 12CMisc Misc In C	lear Freeze ?
yes Send <u>N</u> umbers Send <u>A</u> SCI ♥ +CF Before ♥ +LF Alter	Status Disconnect RXD (2)
Send Numbers Send ASCII Send ASCII	TXD (3) CTS (8) DCD (1)
Dump File to Port	DSR (6)
You can use ActiveX automation to control me! Char Count:10 CPS:0 Po	rt: 28 38400 8N1 RTS/

2. Click to Send, write some characters and click to "Send ASCII".

RealTerm: Serial Capture Program 2.0.0.70	1					**	\$Ø ‼	^e .dí 2	4% 🗎	1:11
allo4 yesû		nRF	UA	RT v2	2.0					
				D	isco	nne	ct			
	(11:11 (11:11 (11:11	:17] C :28] T :50] R	Conne X: all X: ye	ected lo s	to: Ne	ordic,	UAR	r		
	.1								s	end
Display Port Capture Pins Send Echo Port 12C 12C-2 12CMisc Misc 1 Clear Freeze ?	Device	: No	ordic_	UART	- reac	ły				
	L	EAT	I		je		I.	ροι	ır	>
Pes Send ASUI V +UR Before RXD (2)	1	2	3	4	5	6	7	8	9	0
Send AgCli + 4LH TXD [3] TXD [3] TXD [3] TXD [3] CTS [8] D ^C LF Rgpeats 1	а	z	е	r	t	у	u	i	0	р
Dump File to Port Delays 0 \$10 \$10 \$10 \$10 \$10 \$10 \$10 \$10 \$10 \$	q	s	d	f	g	h	j	k	I	m
BREAK	+		w	x	c	v	b	n		×
You can use ActiveX automation to control me! Char Count:10 CPS:0 Port: 28 38400 8N1 RTS/	Sym	ı ,	*	4	França	iis(FR)	•		•	μ

3. The communication is established, you are now able to emulate a serial port over BLE.



5.5. NFC Launch App Example

The Launch App Example shows how to use the NFC tag to launch an app on a device that supports NFC and runs Android (4.0 or later) or Windows Phone (8.0 or later).

NFC Mode Set-up

- 1. Connect the USB cable from the Interface Board ISP130603 to your computer.
- 2. Connect the ISP1510-UX-TB Test Board to the ISP130603 Interface Board with the 10 pin, 14 pin and 22 pin FPC jumper cables (0.5 mm pitch, provided in the Development Kit)
- 3. On the ISP130603 Interface Board, connect the 2-lead patch cable in order to connect:
 - NFC_1 to P0_09
 - NFC_2 to P0_10



NFC Mode Loading

- 1. Start Keil uVision.
- 2. Select Project then Open Project in order to open UART_example application: C:\Keil_v5\ARM\Pack\NordicSemiconductor\nRF_Examples\11.0.0-2.alpha \nfc\experimental_record_launch_app\pca10040\arm5\ nfc_launchapp_record_pca10040.uvprojx.
- 3. Click Build Target and Load.



NFC Mode Testing

Test the Launch App Example with a smartphone or a tablet with NFC support by performing the following steps:

- 1. After programming the application, touch the NFC antenna with the smartphone or tablet
- 2. Observe that the smartphone/tablet tries to:
 - Launch the nRF Toolbox app if it is installed.
 - Download the nRF Toolbox app from the store if it is not installed.



5.6. Simple UWB transmissions using Decawave

DecaWave provides various example codes in C language: simple transmission, simple reception, ranging etc.

However, these examples are made to be used by the IDE CooCox with an STM32 target. This chapter will explain how to use decaWave driver and example code in an nRF52 Keil project.

- Download "DW1000 Application Programming Interface with application examples" from decawave website: <u>https://www.decawave.com/support/software</u>. This package contains drivers and example codes for the DW1000.
- 2. Create a Keil project. In this chapter we will start from the Nordic's template_project provided in their SDK. Open project in \pca10040\blank\arm5_no_packs.
- 3. Copy the decadriver directory of the decawave package into your project directory. Add deca_device.c and deca_param_init.c to your project. Add the decadriver directory to your Include Paths.



July 5, 2017

Page 33/41

Document Ref: isp_ble_AN170701_R0.docx



4. Copy the content of examples/ex_01a_simple_tx main.c of the decawave package into your main.c.

Modify the "config" static variable which holds the decawave's signal parameters. We will be using channel 5.

Remove the following lines of code we don't need:

- #include "lcd.h", #include "sleep.h" and #include "port.h"
- #define APP_NAME "SIMPLE TX v1.2"
- lcd_display_str(APP_NAME);
- reset_DW1000();
- lcd_display_str("INIT FAILED");

Building the project now will result in the following errors:

```
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol peripherals_init (referred from main.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol sleep_ms (referred from main.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol spi_set_rate_high (referred from main.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol spi_set_rate_low (referred from main.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol deca_sleep (referred from deca_device.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol decamutexoff (referred from deca_device.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol decamutexon (referred from deca_device.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol decamutexon (referred from deca_device.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol readfromspi (referred from deca_device.o).
.\_build\nrf52832_xxaa.axf: Error: L6218E: Undefined symbol readfromspi (referred from deca_device.o).
```

Now we need to implement these 9 missing functions.

5. Implement the peripherals_init function; this is where the SPI peripheral will be initialized. Don't forget to add "nrf_drv_spi.h" in your include list.

```
#define SPI_INSTANCE 0 /**< SPI instance index. */
static const nrf_drv_spi_t spi = NRF_DRV_SPI_INSTANCE(SPI_INSTANCE); /**< SPI instance. */
static void peripherals_init ()
        nrf_drv_spi_config_t spi_config = NRF_DRV_SPI_DEFAULT_CONFIG;
        spi_config.frequency =
                                  NRF_DRV_SPI_FREQ_1M;
        spi_config.mode =
                                  NRF_DRV_SPI_MODE_0;
        spi_config.bit_order =
                                  NRF_DRV_SPI_BIT_ORDER_MSB_FIRST;
        spi_config.miso_pin =
                                  24;
        spi_config.mosi_pin =
                                  17;
        spi_config.sck_pin =
                                  8:
        spi_config.orc
                                  0x00:
        APP_ERROR_CHECK(nrf_drv_spi_init(&spi, &spi_config, NULL));
        // Set chip select high
        nrf_gpio_cfg_output (23);
        nrf_gpio_pin_write(23, 1);
```

Page 34/41



6. Implement readfromspi function.

7. Implement writetospi function.

```
int writetospi(uint16 headerLength, const uint8 *headerBuffer, uint32 bodylength, const uint8 *bodyBuffer)
{
     // prepare TX buffer
     uint8_t txBuff[headerLength+bodylength];
     for(uint8_t i=0;i<headerLength;i++) txBuff[i] = headerBuffer[i];
     for(uint8_t i=0;i<bodylength;i++) txBuff[headerLength+i] = bodyBuffer[i];
     // Drive CS Low
     nrf_gpio_pin_write (23, 0);
     // send data
     nrf_drv_spi_transfer (&spi, txBuff, headerLength+bodylength, NULL, 0);
     // Drive CS High
     nrf_gpio_pin_write (23, 1);
     return 0;
}</pre>
```

8. Implement spi_set_rate_high and spi_set_rate_low functions.

```
static void spi_set_rate_low()
{
            NRF_SPIM_Type * p_spim = spi.p_registers;
            nrf_spim_frequency_set (p_spim, (nrf_spim_frequency_t)NRF_DRV_SPI_FREQ_1M);
}
static void spi_set_rate_high()
{
            NRF_SPIM_Type * p_spim = spi.p_registers;
            nrf_spim_frequency_set (p_spim, (nrf_spim_frequency_t)NRF_DRV_SPI_FREQ_8M);
}
```



9. Implement deca sleep and sleep ms functions.

void sleep_ms(unsigned int time_ms) { nrf_delay_ms (time_ms); } void deca_sleep(unsigned int time_ms) { nrf_delay_ms (time_ms); }

10. Implement decamutexon and decamutexoff functions.



11. Build the project (This should now succeed). Load the firmware to the ISP1510. The module is now emitting an UWB packet every second. The UWB signal can be observed using a Signal Analyzer at 6.5GHz.



July 5, 2017

Page 36/41

APPLICATION NOTI ISP1510 DEV KI

6. Basic Range Application with ISP1510

This paragraph shows you how to set up the Insight SiP's Range application between ISP1510 Anchor and Tag Board. The range is calculated using UWB and results are sent via the Bluetooth link to the Master Emulator or to a smartphone / tablet.

In this demo, the Anchor and the Tag operate as a pair. The Tag is the one initiating the ranging exchange and the "Anchor" is listening for the tag messages and responding to it. At the end of the ranging exchange the Anchor can then calculate the time of flight thus the range between the Tag and the Anchor.

Two types of demonstration are presented. The first one is directly executable with hardware and software provided in the Development Kit using Master Control Panel application. The second demonstration requires the use of an Android device. The Android application is available on the Google Play.

6.1. On Master Control Panel

- 1. Place the CR2032 lithium battery into the battery holder of the Tag Board.
- 2. Plug the Anchor Board into an USB port.

lt's **all** in the **package**

- 3. Connect nRF51 Dongle (Master Emulator) into a USB port on your computer.
- 4. Start Master Control Panel.

Click Start Discovery.	Master Control Panel	– 🗆 ×								
	File Help									
	Master emulator									
	COM7 - 680791841 V 680791841 connected	Reset								
	Scan for devices Start discovery									
	Discovered devices									
	Delete bond info									
	Log									
	[10:25:42.6] Device discovery started	^								
	[10:25:55 7] Device discovery stopped									
	[10:26:03.2] Device discovery started									
	[10:26:06.1] Device discovery stopped									
	[10:26:07.0] Device discovery started									
	[10:26:15.7] Device discovery stopped	×								

5.

Page 37/41

Document Ref: isp_ble_AN170701_R0.docx



- 6. Two devices should appear: UWB_TAG_SNXXX and UWB_ANC_SNXXX. Select UWB_ANC_SNXXX and click on "Select Device".
- 7. On the following display, click successively on Connect, Discover Services and Enable Services.

Master Control Panel	-		\times	
File Help				
Master emulator				
COM7 - 680791841 V 680791841 connected Rese	t			
- Device info Device address: CEFCF81AA38B Bonded: False				
Actions				
Disconnect Bond Update Discover services Disable services	DFU			
Service Discovery				
CharacteristicDeclaration, Value: 12-12-00-19-2A, Properties: Read, Notify, Ch Battery Level, Value: 00 ClientCharacteristicConfiguration, Value: 01-00, CharacteristicConfiguration	aracteristic Bits: Notific	UUID: 0	x2A 🔨	Range value
- PrimaryService, Value: 00-11-22-33-44-55-66-77-88-99-AA-BB-00-11-EE-FF, 0xFFE	E1100-BBA	A-9988-	776	
CharacteristicDeclaration, Value: 12-16-00-00-11-22-33-44-55-66-77-88-99-AA	BB-01-11-E	E-FF. P	ope	
UUID: FFEE1101-BBAA-9988-7766-554433221100, Value 68-5D-3A-72-	5-00-EA-3F		000	
ClientCharactensticConfiguration, Value: 01-00, CharactensticConfiguration CharacteristicDoolarstice, Value: 12.19.00.00.11.22.22.44.55.66.77.99.99.0.0	Bits: Notific	ation (Ux	000	
— CharacteristicDeclaration, value: 12-15-00-00-11-22-35-44-35-06-77-86-35-84 UUID: EEEE1102.BR&&:9988-7766-554433221100 Value: & B.77.72-78.	E-7C-ED-3	E-FF, FI :	ope	
ClientCharacteristicConfiguration, Value: 01-00, CharacteristicConfiguration	Bits: Notific	ation (0x	:000	
Characteristic Declaration Value: 0A-10-00-00-11-22-33-44-55-66-77-88-99-AA	-RR-03-11-F	F-FF P	mne Y	
<			>	
Attribute value	Dand Jana	P	and a	
UUID (0x): Handle (0x): Display as UTF8	neau iong		au	
Value: hex () text	Write long	W	rite	
Back				
Log				
[10:31:19.2] Received a HandleValueNotification on handle 0016 with value C9C4440	E5F1EC3F		^	
[10:31:19.7] Received a HandleValueNotification on handle 0016 with value 809F71E	BODAEC3F	:		
[10:31:19.7] Received a HandleValueNotification on handle 0019 with value AB77727	B2E7CED3			
[10:31:20.2] Received a HandleValueNotification on handle 0016 with value 93A56D8	826F4EB3F			
[10:31:20.7] Received a HandleValueNotification on handle 0016 with value 384EC45	OFF40EC3F		- 10	
[10:31:21.2] Received a HandleValueNotification on handle 0016 with value 685D3A7	2A500EA3F			

- You can see data that transit between the ISP1510 Anchor Board and the Master Emulator via the Bluetooth link.
 The distance range is located in the service 0xFFEE1100BBAA99887766554433221100. In this example data is 0x3FEA00A5723A5D68 which is the 64bit floating point representation of 0.81 m.
- 9. To switch off the ISP1510 Tag Board, remove battery from the Tag Board, unplug the Anchor Board from the USB port.



6.2. On Android Device

An App is also available for Android Devices. On Google Play, search "sip sensor" or "insight sip" and download the App. The android App is a demonstration App that is provided "as is" in order to demonstrate the Smart Bluetooth sensor node.

Make sure your Android device is compatible with Bluetooth 4.0 (Android v4.3 at least).

After the application is downloaded and installed you should see the application on your Android device. Then you will be able to set up the application demonstration as follows:

- (For Android v6 or older) In order to be able to scan for BLE devices, the application needs to have "Location" permission. You have to enable "Location" service in the settings and give permission to the Sensor App to use Location (Settings -> Application Manager -> Sensor -> Permission).
- 2. Power up both boards: Place the CR2032 lithium battery into the battery holder of the Tag Board and plug the Anchor Board into an USB port.
- 3. Start "ISP1510 UWB Demo" application on your Android, In the "Anchor part" click Connect and select your Anchor Board (UWB_ANC_SNxxx). Do the same for the Tag Board.



NB: Ranging data is only provided by the Anchor Board, thus only BLE connection to the Anchor Board is necessary.



4. Ranging should automatically starts at power up. Instant and average range should be updating.



5. By pressing the "Configure" buttons you have access to a number of settings of the Tag/Anchor.

교 🖬 🎽 🕨 🔺 💐 🗑 🍞 📶 100% 🖥 12:09	🖬 🖬 🖻 🕨 🔺 😻 🖄 🛱 🛜 📶 100% 🛢 12:10
\leftarrow 🙀 Anchor Settings :	\leftarrow Settings :
Anchor settings	Door closing Threshold (cm)
Sample size for Average value	120
9	UWB Signal Settings
Door opening Threshold (cm)	Pulse Repetition Frequency
100	64 MHz
Door closing Threshold (cm)	Data Rate
120	6.81Mbits/s
UWB Signal Settings	Preamble Length
Pulse Repetition Frequency	128
64 MHz	Preamble Code
Data Rate 6.81Mbits/s	Channel
Preamble Length	5 – 6489.6 MHz
128	BW 499.2 MHz
Preamble Code	Non-Standard SFD

Page 40/41

Document Ref: isp_ble_AN170701_R0.docx





The list of Anchor parameters is :

- Average : set the number of sample used to calculte the average range value
- Door opening Threshold : set distance which trigger the opening of the door in cm
- Door ClosingThreshold : set distance which trigger the the closing of the door in cm

The list of Tag parameters is:

• Delay : set the delay in ms between each range update in

The list of Decawave specific parameters is:

- Pulse Repetition Frequency : 16 MHz or 64 MHz
- Data Rate : 110 Kbit/s, 850 Kbits/s or 6.81 Mbits/s
- Preamble Length : 64, 128, 256, 512, 1024, 1536, 2048 or 4096 symbols
- Preamble Code : Preamble pattern selection
- Channel : Frequency channel selection
- Non-Standard SFD : Enable to use the DecaWave defined non-standard SFD

It is not recommanded to change this parameters.

Page 41/41



Insightsip.com 代理商联系方式:

样品,技术支持,参考设计,评估板,报价购买请联系

电话:0755-82565851

邮件:dwin100@dwintech.com

手机:156-2521-4151

网址: <u>http://www.dwintech.com/insight%20Pamphlet.html</u>

深圳市南频科技有限公司

D-Win Technology(HongKong) Co.,Ltd