ULE Starter Kit User's Manual



# ULE Starter Kit User's Manual Revision 1.0

**Document Objective**: Instruct IoT device designers how to leverage the ULE Starter Kit to empower their lowpower sensor and control applications with ULE wireless connectivity

> Document is subject to change without notice Copyright © 2015 DSP Group Ltd. All Rights Reserved

#### ULE Starter Kit User's Manual

# **Revision History**

Revision	Date	Author	Description
1.0	June 2015	Levi	Baseline Release

# **Table Of Contents**

1.	INT	RODUCTION	.4
	1.1	OTHER DOCUMENTS	4
	1.2	TARGET AUDIENCE	.4
	1.3	STARTER KIT BASICS: A BIT ABOUT DECT-ULE	.4
	1.4	STARTER KIT HARDWARE	. 5
	1.4.1	<i>DHX91 EVB</i> (= <i>ULE Node</i> )	5
	1.4.2	DCX81 CMBS UART (= ULE Controller)	6
	1.5	STARTER KIT SW TOOLS (DOWNLOADED FROM THE WEBSITE)	6
2.	GET	YOUR APPLICATION UP AND RUNNING WITH ULE	.7
	2.1	STEP 1: ULE NODE WITH THE DHX91 ULE EVB & VAR SIMULATION TOOL	.7
	2.2	STEP 2: ULE NODE WITH THE DHX91 ULE EVB WIRED TO YOUR MCU	.7
	2.3	STEP 3: ULE NODE WITH DHAN MODULE MOUNTED ON YOUR APPLICATION	. 8
3.	GET	TING THE ULE CONTROLLER UP AND RUNNING	9
4	DIN		10
4.	KUN	NING THE ULE NODE USING THE VAR APPLICATION 1	LU
4.	4.1	NING THE ULE NODE USING THE VAR APPLICATION	LU 10
4.	4.1 4.2	INING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1	10 10 10
4.	4.1 4.2 4.3	INING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1	10 10 10 10
4.	4.1 4.2 4.3 4.4	INING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         INSTALLING THE VAR SIMULATOR       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1	10 10 10 10
4.	4.1 4.2 4.3 4.4 4.5	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1	10 10 10 10 12 13
4.	4.1 4.2 4.3 4.4 4.5 4.6	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1         Place the ULE Node in Hibernate and Wake it Up       1	10 10 10 12 13
4.	4.1 4.2 4.3 4.4 4.5 4.6 4.7	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup.       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1         Place the ULE Node in Hibernate and Wake it Up       1         Changing the Frequency Band for your Region       1	10 10 10 12 13 13 14
4.	4.1 4.2 4.3 4.4 4.5 4.6 4.7 <i>4.7.1</i>	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1         Place the ULE Node in Hibernate and Wake it Up       1         Changing the Frequency Band for your Region       1         Offset 0x204 (Node) and Offset 0x20 (Controller) Values for changing Frequency Band       1	10 10 10 10 12 13 13 14 14
4.	4.1 4.2 4.3 4.4 4.5 4.6 4.7 <i>4.7.1</i> 4.8	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1         Place the ULE Node in Hibernate and Wake it Up       1         Changing the Frequency Band for your Region       1         Offset 0x204 (Node) and Offset 0x20 (Controller) Values for changing Frequency Band       1         Registering the ULE Node to the ULE Controller       1	10 10 10 10 12 13 13 14 14 15
4.	4.1 4.2 4.3 4.4 4.5 4.6 4.7 <i>4.7.1</i> 4.8 <b>INT</b>	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1         Place the ULE Node in Hibernate and Wake it Up       1         Changing the Frequency Band for your Region       1         Offset 0x204 (Node) and Offset 0x20 (Controller) Values for changing Frequency Band       1         Registering the ULE Node to the ULE Controller       1         Registering the ULE Node to the ULE Controller       1         Registering the ULE Node to the ULE Controller       1         Image: Control of the ULE Controller       1         Image: Control of the ULE Contro	10 10 10 10 12 13 13 14 14 15 16
<ol> <li>5.</li> </ol>	4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.7.1 4.8 <b>INT</b> 5.1	INING THE ULE NODE USING THE VAR APPLICATION       I         Hardware Setup       1         Installing the VAR Simulator       1         Running the VarCmndSimulator.exe       1         Messages from ULE Node to Controller       1         Messages from the ULE Controller to the ULE Node       1         Place the ULE Node in Hibernate and Wake it Up       1         Changing the Frequency Band for your Region       1         Offset 0x204 (Node) and Offset 0x20 (Controller) Values for changing Frequency Band       1         Registering the ULE Node to the ULE Controller       1         UART CONNECTION ON THE DHX91 EVB       1	10 10 10 10 12 13 13 14 14 15 <b>16</b>
4.	4.1         4.2         4.3         4.4         4.5         4.6         4.7         4.8         INT1         5.1         5.2	INING THE ULE NODE USING THE VAR APPLICATION       I         Hardware Setup       1         INSTALLING THE VAR SIMULATOR       1         RUNNING THE VAR CMNDSIMULATOR.EXE       1         MESSAGES FROM ULE NODE TO CONTROLLER       1         MESSAGES FROM THE ULE CONTROLLER TO THE ULE NODE       1         PLACE THE ULE NODE IN HIBERNATE AND WAKE IT UP       1         CHANGING THE FREQUENCY BAND FOR YOUR REGION       1         Offset 0x204 (Node) and Offset 0x20 (Controller) Values for changing Frequency Band       1         REGISTERING THE ULE NODE TO THE ULE CONTROLLER       1         UART CONNECTION ON THE DHX91 EVB       1         UART CONNECTION OF UART COMMUNICATION       1	10 10 10 10 12 13 13 14 14 15 16 16
<ol> <li>5.</li> </ol>	4.1         4.2         4.3         4.4         4.5         4.6         4.7         4.8         INT1         5.1         5.2         5.3	NING THE ULE NODE USING THE VAR APPLICATION       1         Hardware Setup       1         Installing the VAR Simulator       1         Running the VAR Conditional and the variable of the variable o	10 10 10 10 12 13 13 14 15 16 16 16

# 1. Introduction

#### **1.1 Other Documents**

- A. DHX91 DHAN Module Data Sheet (Downloadable from DSP Group Website)
- B. HAN-FUN Protocol (Downloadable from www.ulealliance.org)

#### **1.2 Target Audience**

The Starter Kit and this User's Manual are targeting those customers familiar with running IoT applications on a particular microcontroller (with a spare UART Interface!) and now are interested in adding ULE wireless connectivity capability via the DHAN module. That said, the DHAN module can be used for standalone (single-processor) applications and also can interface with an external microcontroller via I<sup>2</sup>C or SPI interfaces. If any of these use cases interest you, please don't hesitate to contact DSP Group at the link shown below. "Ditto" for those interested in developing new ULE Controller or Gateway devices. We can get you started with these applications as well!

http://www.dspg.com/dhan-module/

#### 1.3 Starter Kit Basics: A bit about DECT-ULE

DECT-ULE uses a star topology. Each DECT-ULE system contains only one device at the hub of the star. This hub is typically the gateway to the WAN. In DECT cordless this hub is called the Base Station. In the IoT context, the hub is called a ULE Controller. The points of the star are traditional DECT cordless handsets and ULE Nodes or Devices – they can be integrated into a single system. Thousands of ULE Devices can be associated with a single ULE Controller.

This Starter Kit incorporates a ULE Controller (BS Device) and a ULE Node (Device). The ULE Controller includes a graphical interface for control and status reporting. It does not provide a means to connect to the WAN. The Starter Kit promotes development of ULE Node applications per the ULE Standard. Once completed and certified, your ULE Node can operate in any setting within range of a certified ULE Controller. The ULE Controller is a common element in many gateways and over-the-top boxes that populate our homes and offices.



Figure 1-1: Star Topology

# **1.4 Starter Kit Hardware**

#### 1.4.1 DHX91 EVB (= ULE Node)

The DHX91 EVB includes a DHAN Module (a custom version with printed antenna), several push buttons, LEDs and access to vital IO pins that facilitate simple application emulation. The EVB can be powered using the AC transformer outputting 5V (supplied with the Kit) or using a CR123A battery. The EVB is delivered configured for this latter case. See the jumper configuration below for both cases. It is suggested to initially power up the EVB with a 3V power supply clipped to the battery holder to get feedback regarding the current drain.

The EVB can function as a standalone demo device (Step 1 below) or as a development tool to be interfaced via RS232 to a PC (Step 2 below) or via UART to an external controller (Step 3 below).



	CR123A	5V
JP3	ON	OFF
JP4	OFF	ON
JP5	ON	OFF
JP8	OFF	OFF
JP10	OFF	ON
JP11	ON	ON
JP12	OFF	OFF
JP13	OFF	OFF
JP18	OFF	ON
JP19	OFF	Left position
JP20	OFF	OFF

#### **1.4.2 DCX81 CMBS UART (= ULE Controller)**

This platform includes the DCX81 CMBS Module mounted on a ULE Controller motherboard. It is powered by the AC transformer supplied with the Kit. The Kit includes 2 antennas which are threaded on to the SMA connectors, and positioned as shown in the photo below. The connection between the Controller RS232 (UART I/F) connector and the PC USB port is also shown below.

The driver for the ST Lab adaptor (included in the Kit) usually loads on its own when it is plugged into the PC. If not, it can be copied from the CD included in the Kit. Once the driver is loaded you should be able to see a COM port opened in your PC Device Manager. You should note the COM port value as you will need it to run the DSPG Test Application SW which you have downloaded.



#### **1.5** Starter Kit SW Tools (Downloaded from the website)

- VarCmndSimulatorSetup.msi Installer for the GUI communicating (via USB/UART path) with the DHX91 ULE Node
- DSPG Test Application.exe GUI for communicating (via USB/UART path) with the ULE CMBS Controller

# 2. Get your application up and running with ULE

Working through the steps below will enable the User to get an ULE Node application with basic ULE connectivity up and running. This section provides an overview of the 3 steps, with following sections elaborating Steps 1 and 2.

# 2.1 Step 1: ULE Node with the DHX91 ULE EVB & VAR Simulation Tool



Here the ULE Node application is running on the PC using the VAR simulator you have downloaded. (Note: Both sides of the link can run on the same computer). In this setup the User will learn how to:

- Install and run VAR simulator on the Node side and the Controller GUI simulator on the Controller side
- Register the ULE Node, modify the operating frequency to comply with local regulations
- Use the VAR simulator to trigger events and send maintenance messages (=Keep Alive) that can be monitored at the Controller GUI
- Copy the basic transactions at the CMND I/F so that they can easily be transplanted from the simulator to the code running on the User's application MCU.

#### 2.2 Step 2: ULE Node with the DHX91 ULE EVB wired to your MCU



ULE transactions have been embedded into the Application SW and their effectiveness can now be validated. User will be able to register (=pair) his ULE Node Application with the ULE Controller, send maintenance (= Keep Alive) and simple Alarm/Event messages, turn actuators ON-OFF, send battery and RSSI status over to the Controller.



2.3 Step 3: ULE Node with DHAN Module mounted on your application

The User designs an application board which integrates a favorite MCU, sensors/switches, the DHAN module and a 2GHz antenna. Prior to designing this application board, the User should discuss application needs and constraints with DSP Group to insure that the application schematic is optimal as well as receiving the SW training and detailed documentation needed to create an full-fledged application.

# 3. Getting the ULE Controller Up and Running

After connecting the ULE Controller (= DCX81 CMBS UART) as depicted in a previous section (and confirming via your Device Manager that a new COM port is up and running!), you should run the DSPG Test Application executable you have downloaded. The window below will pop up and you should enter the appropriate value:

🕅 CMBS Start Menu 🗙
TDM is Master
Use Coma
DVF99 IP Address 192.168.1.3
Detect USB Com Port
Com Port:
Start
Failed Detection!
Continue

A Graphical User Interface will pop up as below and the log window should indicate that the initialization was successfully completed ("CMBS Init Done!).

CMBS Test Application 1.1J Build 30342		
Running!         Host Version 03.65 - Build 26         Target Version 0           TDM is Slave         RFP1 00feb04220         RXTUN a1           Start         Mode 00         RVBG	3.65 - Build 32	EEPROM Parameter Value Read Write Get Set
Registration         Open         Close         Close           Handsets         HS         Status         IPUI         HS Name         Caps         ^	Make Call         HS:         1         Line:         1           Make Call         CLIP:         11         CNIP:         D           Codec:         Narrow         V         V	SPG Release Answer Delete
1 Unregistered E 2 Unregistered E 4 Unregistered 4 5 Unregistered 4 5 Unregistered 7 Delete Delete All Refresh	Id Instance State DCM State	HS Line CLIP CNIP Codec
		Clear Log Screen Mg Log File Open Mg Log File 
Mes Int Lesson CoM 11 → Constant Com 11 ← Constant Com 11		Open HAN SUOTA Open HAN Bind
Exit Open JSystem Window Open Price Demo App	Open HAN Attributes window Open HAN Report Wind	Open HAN Test Window Open HAN Window

The log window shown above is an abbreviated version of the incoming/outgoing flow of information at the CMBS UART Interface. A more detailed (message) log is recorded in a file that can be opened from the button indicated above. This file is located in the same directory as the DSPG Test Application.exe file.

Click on the "Open HAN Window" button in the lower right and note that a "Smoke" device is registered to your Controller, as shown below. This is because the EVB ULE Node in your Starter Kit has already been registered to the ULE Controller and has been programmed as a Smoke device. A bit later you will delete this registration and can run through the pairing (=registration) exercise yourself!

Device Id         IPUI         Reg Status         Link Status         Unit Id         Unit Type         Keep Alives         Alerts         Tampers         Last Event         On/Off         Alert         Tamper           1         00FEB49600         Registered         Unknown         1         Smoke(0x204)         0         0         No Events         N/A         N/A	Initialized!	ъ	Request	Tx End	Get Link S	atus	Clear Unit Cou	inters	Clear	All Counte	ers	Delete Device	Delete al	Devices	Refresh Table
1 00FEB49600 Registered Unknown 1 Smoke(0x204) 0 0 0 No Events N/A N/A N/A	Device Id	IPUI	Reg Stat	us Link St	atus Unit	id Un	nit Type	Кеер	Alives	Alerts	Tampers	Last Event	On/Off	Alert	Tamper
	1	00FEB49600	Registere	ed Unkno	wn 1	Sm	noke(0x204)	0		0	0	No Events	N/A	N/A	N/A
	-														

# 4. Running the ULE Node using the VAR application

Time to get the ULE node up and running with the VAR simulator. This simulator converses with the UART Driver loaded in the DHAN module at an interface called the CMND API. The VAR simulates your MCU's communication with the CMND - so this is means to learn about this communication pipe as well as grasp the overall connectivity scheme of ULE.



#### 4.1 Hardware Setup

Power up the DHX91 ULE EVB. If you are using a Power Supply with metering, you should observe ~28mA. Now connect the EVB RS232 port to the PC USB port using the USB2UART Cable and UART2UART adaptor provided in the Starter Kit. As before, query your Windows Device Manager for the # of this added COM port!

#### 4.2 Installing the VAR Simulator

Run the VarCmndSimulatorSetup.msi and follow instructions to generate the VarCmndSimulator.exe file.

#### 4.3 Running the VarCmndSimulator.exe

Run the VarCmndSimulator.exe file and choose the VAR Mode.

Select Mode		×
Tools:		
VAR Mode	O Advanced Mode	Start

The window below will pop. Configure the COM port (upper right) with appropriate value and click "Open"

Starts Configuration Wizard Log file managemement Messages log COM port management																		
Configuration Wizard		Clear	Clear Log File Oper	n Log File				COM PC	rt COM6 💌	Close								
Messages Log	1 .	1							1									
Time TX/RX	Length	Service Id	Message Id	Unit Id	Cookie	IE	IE Parsed		Raw Message									
15:40:52 MCU> C	MND: 20	Production<02 0	prod_start<01>	00	00	000100	CMND TE DECRONCE (0:00)		dada00100000020	b01d5cccc0006								
15:40:51 CMND>	MCII: 38	Param<02.03>	naram set dir res<08>	00	00	00150200	CMND_IE_RESPONSE<00002	T<0v0C> len<21	dada00220000020	308f00c001502								
15:40:51 MCU> C	MND: 34	Param<02 03>	param set dir reg<07>	00	00	00150200	CMND IE PARAMETER DIREC	T<0x0C> len<21	dada001e0000020	307ea0c001502								
ie cont						000100	CMND_IE_RESPONSE<0x00>	len<1> val <succ< td=""><td></td><td></td></succ<>										
15:40:51 CMND>	MCU: 38	Param<02 03>	param_set_dir_res<08>	00	00	00150200	CMND_IE_PARAMETER_DIREC	T<0x0C> len<21	dada00220000020	308ab0c001502								
15:40:51 MCU> C	MND: 34	Param<02 03>	param_set_dir_req<07>	00	00	00150200	CMND_IE_PARAMETER_DIREC	T<0x0C> len<21	dada001e0000020	307a50c001502								
ie cont		Damar (02,02)				000100	CMND_IE_RESPONSE<0x00>	len<1> val <succ< td=""><td>d- d- 00110000000</td><td>0044205000400</td></succ<>	d- d- 00110000000	0044205000400								
▼ 15:40:51 CMND>	MCU: 21	Param<02.03>	param_set_res<04>	00	00	00040018	CMND_IE_PARAMETER<0x082	> len<4> val <ee< td=""><td>dada00110000020</td><td>304420b000400</td></ee<>	dada00110000020	304420b000400								
15.40.51 MC0 > C	-into. 10	Taramsoz 05×	burum set reukusz	00	00	00050010	CHIND IE TRICHMETERSOADDA	varsee	000000000000000000000000000000000000000	▶								
- Outgoing CMND Messages			Cana	otod (		band												
Message Type	Generated		Gener	aleu	.011111													
medolage Type	Generated																	
Alert On	DA DA 00 1	0 00 00 00 01 03 29 0	08 00 07 01 05 00 00 00 00	00														
Alert Off	Service: Dev	vice Managment<00 (	)1>	leads														
Tamper	Message:re	gister_dev_req<03>	-Parsed Pay	1080:														
Keep Alive	Unit ID: 00	· · · ·	RAW IE: 0	00701050	000000	000				<b>A</b>								
On/Off/Toggle	Cookie: 00	1	Parsed IE:	CMND_IE	E_REGIS	TRATION<0x0	8> len<7> val <numofunitent< td=""><td>ries:1,LengthOfUnit</td><td>intry:5,UnitId:0x00,Ur</td><td>hitType:</td></numofunitent<>	ries:1,LengthOfUnit	intry:5,UnitId:0x00,Ur	hitType:								
Rattery Request																		
▼ RSSI	Generate	Send								<b>v</b>								
Generate message Send message to device Parsed command																		
Message Type L	ist box									Message Type List box								

Go to the "Message Type" window, pull down "Get Version", hit "Generate Message" button then "Send". You should see the "req" and "resp" between the VAR (=MCU) and the DHAN (=CMND), as shown below.

12	incosages Lug											
Ш	Tin	ne	TX/RX	Length	Service Id	Message Id	Unit Id	Cookie	IE	IE Parsed	Raw Messag	
11	12	:18:18:023	CMND> MCU:	26	General<00 00>	get_version_res<0c>	00	00	000d0c312	CMND_IE_VERSION<0x09> len<13> val <len: 12;="" td="" valu<=""><td>dada00160.</td></len:>	dada00160.	
	12	:18:17:923	MCU> CMND:	14	General<00 00>	get_version_req<0b>	00	00	000101	CMND_IE_U8<0x1E> len<1> val<01>	dada000a0.	

NOTE: Please avoid running the Configuration Wizard (upper left). If this button is accidentally depressed, press Cancel and you will return to the Main Var window above.

#### 4.4 Messages from ULE Node to Controller

The Alert On, Alert Off, Tamper, Keep Alive, On/Off/Toggle, Raw Fun messages are all items that are sent over the link from the Node to the Controller. To send any of these messages you:

- 1) Select Message
- 2) Click Generate Button and note command generated
- 3) Press the Send button. This should generate an MCU < -- > CMND exchange in the VAR log window and an incoming message in the Controller GUI log window

Alert On example is shown below:

1) The command generated by the VAR

Senerated command:										
DA DA 00 0F 00 00 01 00 03 26 06 00 06 02 04 00 00 00 01										
Service: Alert<0100> Message: notify_status_req<03> Unit ID: 00 Cookie: 00	Parsed Payload: TAW IE: 0006/2040000001 Parsed IE: CMND_JE_ALERT<0x06> len<6> val<020400000001> 									
Generate Send										

2) The MCU < --- > CMND exchange on the Node Side. The CMND informs the MCU that it received an ACK from the ULE Controller

	messages Log -	zssages Log										
	Time	TX/RX	Length	Service Id	Message Id	Unit Id	Cookie	IE	IE Parsed	Raw Message		
	13:30:21:070	CMND> MCU:	14	General<00 00>	link_cfm<07>	00	00	000100	CMND_IE_RESPONSE<0x00> len<1> val <success></success>	dada000a0		
	13:30:20:970	MCU> CMND:	19	Alert<0100>	notify_status_req<	00	00	000602040	CMND_IE_ALERT<0x06> len<6> val<020400000001>	dada000f0		
1												

3) The incoming message as seen on the graphical Log on the ULE Controller side:

	Log	
	RX < D1:U0:D0:U2:T0:S3:M1:T0:I100:C1:L6:P020400000001	

4) The incoming message as seen parsed in the Log File on the ULE Controller side (This \*.txt file is located in the same directory as the DSPG Test Application.exe file):

```
28-05-2015 13:40:03:607 (1432809603607)

Target ---> Host: 29 00 00 00 16 30 21 00 00 30 17 00 01 00 00 00 00 02 00 00 03 01 00 00 01

01 06 00 02 04 00 00 00 01 00 0F 02 00 A5 FC

{CMBS_EV_DSR_HAN_MSG_RECV(12310)}

<CMBS_IE_HAN_MSG(12288)>:

Src: (type = 0, dev_id = 1, un_id = 0)

Dst: (type = 0, dev_id = 0, un_id = 2)

Transport : 0

Msg Sequence : 3, Message Type : 1

IF Type : 0, IF Id : 0x100, IF Member : 1

Data Len: 6

2 4 0 0 0 1
```

Alert Off, Tamper, Keep Alive, On/Off/Toggle work similarly to the example above.

#### 4.5 Messages from the ULE Controller to the ULE Node

To send a message from the Controller to the Node:

- 1) Press the Tx Request button on upper left in HAN Window of Controller GUI. You should observe Link Status as "Requesting".
- 2) Now you need to wake up the ULE Node and have it respond to this Link Request by triggering an event from the VAR simulator (eg Keep Alive, Tamper, Alert). You will see the Link Status on the ULE Controller change to "In Link". The ULE Node is now available for an incoming message from the Controller
- 3) Let's pretend that the ULE Node is an AC Outlet and send it an ON command from the Controller GUI, HAN Window (Lower Left as shown below)

🕅 HAN			
Initialized: Device Id IPUI 1 006655 00 Registered Press TX request	Tx End     Get Link Status     Clear Unit Counters       Link Sta     Unit Id     Unit Type     Keep Ali       In Link     1     Smoke(0x204)     0	Clear All Counters     Delete Device     De       Alerts     Tam     Last Event     On/Off     Alert       1     0     Alert     N/A     ON	Iete all Devices Refresh Table
Chevice 0. Information	Make sure Link status is In Link		
(0x01) HF Core Release (0x02) Profile Release (R) (0x03) Interface Release (R) (0x04) Paging Caps (R)	01 01 01 00 00	Device Info Device Info Get Pack ALL Get Pack Mandatory	Keepalive Get Attribute         Keepalive Set Attribute           Attribute Id:         1
(0x05) Min Sleep Time (R) (0x06) Actual Response Time (R) (0x07) Application Version (R)	00000000 1.1.J CMBS 3.65 Build 32	Device Info Get Attribute Id: 01	
(0x08) Hardware Version (R) (0x09) EMC (R) (0x0A) IPUI (R)	DCX81 0FEB 0011223340	Device Info Set Attribute	Get Device Subscription
(0x0B) Manufacture (0) (0x0C) Location (0x0D) Device E (0x0E) Friendly (0x0E) Pevice UB (N) (0x10) Serial (R)	ON request to evice	Clear All attributes	Status TPUI IPUI UAK
3 OFF Create 0x 80	Add Delete ON-OFF Get Attribute ON-OFF State Start	Device         Unit         Interval           0x         0001         0x         01         5	ring Interval Set Metering Interval

4) Now check the VAR Log window and you will notice the report of an incoming ON message:

 Image: Log
 Time
 TX\_RX
 Length
 Service Id
 Message Id
 Unit Id
 Cooke
 IE
 IE Parsed

 is cont
 is cont
 000100
 CMND\_IE\_OTA\_COOKIE<0x13> len<1> val<00>
 0003000002
 CMND\_IE\_UNIT\_ADDR<0x03> len<3> val<000002>

#### 4.6 Place the ULE Node in Hibernate and Wake it Up

The ULE Node has several options (configurable via EEPROM) for entering and exiting hibernation. The Node in your Starter Kit has been configured to enter hibernation upon command from the MCU (the SLEEP message in the VAR window) and exit via the Wakeup button (AMP\_OUT2 from the DHAN) on the DHX91 EVB (see the photo in the earlier section). Let's try it!

- a) Prior to entering Sleep/Hibernate you see ~28mA on your Node Power Supply
- b) Generate and Send the Sleep Command
- c) You should drop down to  $\sim 4\mu A$
- d) Now hit the Wakeup Button and note the return to 28mA!

#### 4.7 Changing the Frequency Band for your Region

Your Starter Kit has been shipped to you operating the European DECT Band (1880-1900MHz). In this section instructions are given to reconfigure your system to operate in the DECT band allocated by your local regulator:

Step 1: Steps 2-6 are for modifying the ULE Node Band setting. This setting will take effect upon powering down then up the Node.

Step 2: Go the ULE Controller GUI Main Window and Open HAN Test Window

nenl	Open HAN SUOTA Open HAN Bind	
DTLOG		
oshi	Exit Open JSystem Window Open Price Demo App Open HAN Attributes Window Open HAN Report Window Open HAN Test Window Open HAN Window	
en		J

Step 3: Click the Tx Request Button in the HAN Test Window – Link Status should show "requested"

Step 4: Go the VAR Window and Send Tamper – Link Status in HAN Test Window should now indicate "In Link" – as shown below

Step 5: Enter Value Length and Start address, Value for your Region (see table below, example given is for transitioning to US) and hit "Set" button. Confirm change with the Get button. Now power Down the Node

HAN Test	1 St. 10		M	e 4		×
Device Id	1	Tx Request	Tx End	Get Status	Link Status: In Link	
Remote Devic	EEPROM Access	Valua	Longth C		Value	
Set	0x DECT (0x0)	• 0x 1	0x	204	0x 01	

Step 6: Now for the ULE Controller Side. Close the HAN Test Window and return to the Main Window and Click on the EEPROM Read/Write button at upper right

📓 CMBS Test Ap	oplication 1.1J Build 30342	Due Blacker Kit Unier's Marrial			X
Running! TDM is Slave Start	Host Version 03.65 - Build 26 RFPI 00feb04220 RXTI PIN ffff0000 RVRI Mode 00 RVB	Target Version 03.65 - Build 32 N a1 F 1f	Read Write	Parameter RFPI • Get	Value 00feb04220 Set

Step 7: You will get the EU value of 0x00. Now go to Write section and enter 0x20 for Start Adress and Value of region -0x01 in case of the US. Hit the Write EEPROM and Confirm change with the Read EEPROM. Power Down and Up the Controller

EEPROM				×
Address 0x0020	Value 0x00	Read Start Address Length	0x 20	Read EEPROM
		Write Start Address Value	0x 0x	Write EEPROM
			EEPROM Reset	

Step 8: Power Up the Node and send a Tamper message to the Controller. Confirm that it succeeds!!

# **4.7.1** Offset 0x204 (Node) and Offset 0x20 (Controller) Values for changing Frequency Band

EU	Latin America	Korea	US (DECT6.0)	Thaiwan	Brazil	Japan	Argentina
0×0	0x7	0xB	0x1	0x9	0×E	0x12	0x8

Thiland				
0xF				

#### 4.8 Registering the ULE Node to the ULE Controller

Ready to take some risk and De-register the ULE Node from the Controller and Re-register it?

- a) Go to the ULE Controller HAN Window, select the Smoke Detector and hit the "Delete Device" button. The Smoke will now disappear from the list
- b) Power down the ULE Node (you can leave the VAR running!!). Power Back On!!
- c) Go the VAR simulator on the Node side, generate and send the Registration Command
- d) Go the Controller side and press Open in the Registration window on the upper left
- e) Note the 3-Stages of the registration process scroll in the Controller Log

🛱 CMBS Test Application 1.1J Build 30346		_ 🗆 ×
Runningt Host Version 03.65 - Build 30 Target Version 03.65 - Build 32	EEPROM Parameter	Value
TDM is Slave RFPI 0011223340 RXTUN 70	Read RFPI V	0011223340
Diant         PIN ffff0000         RVREF         1f           Mode 00         RVBG	Write Get	Set
Registration Make Call		
Open Close Close HS: 1 Line	e: 1 Release	Answer
Handsets Make Call CLIP: 11 CNI	P: DSPG D	elete
HS Status IPUI HS Name Caps		
1 Unregiste		
2 Unregiste Id Instance State DCM State	HS Line CLIP CNIP Code	:
4 Unregiste		
5 Unregiste		
Delete Delete All Refresh		
		_
Time Log		Clear Log
18:27:43: Got Device Table Read Response with 1 entries starting from index 0 18:27:43: <<<< Unhandled Event (Flash Stop Request)		Screen
18:27:43: Get Devices		Clear
18:27:43: Device Stage 3 registration OK (eviceId=1) 18:27:35: <<<<>> Linhandled Event (Each Start Request)		Msg Log
18:27:35: Device Stage 2 registration OK (deviceId=1)		File
18:27:33: <<<< Unhandled Event (Flash Stop Request)		Open Msg Log
18:27:29: Registration Closed!		File
18:27:29: Device Stage 1 registration Of deviceId=1 IPUI=I:0044558800		
18:25:35: Open Registration Succeeded- waiting for handset or device to register		Enable
18:25:34: Got Device Table Read Response with 0 entries starting from index 0		Log
18:25:34: Get Devices	to see device list	
	to see device list	Þ
	Onon HAN SUOT	Open HAN Bind
	Open HAN SOUTA	opennerbind
Exit Open JSystem Window Open Price Demo App Open HAN Attributes Window Open HAN Report	Window Open HAN Test Window Op	en HAN Window

To see list of devices press Open HAN window button.



# 5. Interfacing the DHX91 EVB ULE Node to your MCU

Now you are ready to drive the ULE Node with YOUR MCU. You will need to bypass the RS232 connector and IC on the EVB and configure your UART driver for appropriate protocol – both described below. You can also reconnect the AMP2\_OUT from the Wakeup Button and connect to an IO from your MCU for triggering wakeup of the ULE Node.



### 5.1 UART connection on the DHX91 EVB

Remove JP21,22 and connect RxD and TxD from external MCU to GPIO9 and GPIO10



# 5.2 Configuration of UART Communication

UART protocol settings are as follows:

- Baud rate: 115200K bps (configurable)
- Data: 8 bit
- Parity bit: None
- Stop bit: 1 bit
- Network byte order: Big Endian

Flow control: Enable/Disable (at compilation time, optional). Recovery from message loss should be done in higher layers.



## 5.3 IO, Push Button and LED Schematic of DHX91 EVB



# 5.4 Modifying the Application SW for ULE

By cutting and pasting the commands (into your App SW) from the VAR "Generate" window (shown once again below) you should be able to repeat all the connectivity exercises from the previous section, this time with your MCU in control!

Generated command:						
DA DA 00 0F 00 00 01 00 03 26 06 00 06 02 04 00 00 00 01						
Service: Alert<0100> Message:notify_status_req<03> Unit ID: 00 Cookie: 00 Generate Send	Parsed Payload: RAW IE: 0006020400000001 Parsed IE: CMMD_IE_ALERT<0x06> len<6> val<020400000001> 					