

Report No:

AN145

Title:

In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices

Author:	Date:	Version Number:
John Marriott	11th September 2015	0.80



Z-Wave® Next Generation Products

All rights are reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner. The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without prior notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent or other industrial or intellectual property rights





Contents

1.0 Introduction	7
1.1 Features	7
1.2 Programmers supporting Z-WAVE 500 series devices	9
1.3 Calibration overview	10
1.3.1 Overview	10
1.3.2 Crystal (XTAL) calibration	10
1.3.3 TX calibration	11
1.3.4 Calibration recovery after accidental NVR erasure / corruption	11
1.3.5 Programmer selection guide for Z-Wave calibration	12
1.4 Device Support	13
1.4.1 Overview	13
1.5 Upgrading your Equinox Programmer to support Sigma 500 series Z-WAVE device	
programming	14
1.5.1 Overview	14
1.5.2 Purchasing a Sigma Z-WAVE 500 series License	14
1.5.3 How do I enable the programmer for Z-WAVE programming?	14
1.5.4 Entering the License String to upgrade your programmer	15
1.6 Programmer firmware versions for Sigma 500 series support	16
2.0 Z-Wave - Programming Interfaces	17
2.1 Overview	17
2.2 SPI - Programming Interface - Features	17
2.3 Z-WAVE single-chip In-System Programming (ISP) Schematic	18
2.4 ISPnano programmer - Target ISP Port - SPI connections	19
2.4 Z-WAVE UART ISP Schematic	20
3.0 Creating an EDS (Development) Project	23
3.1 Overview	23
3.2 Information required to create an EDS Development project	23
3.3 Creating an EDS (Development project)	24
3.4 Selecting the correct Target Device	25
3.4.1 Device selection	25
3.4.2 Device Chip ID / Signature	26
3.5 Target System – Power Supply Settings	27
3.6 Erase options	28
3.7 Specifying the FLASH (Code) File	29
3.8 Launching EDS at the end of the EDS Wizard	31
4.0 Testing a Project in Development (EDS) Mode	33
4.1 Introduction to EDS	33
4.2 EDS - Default settings for SPI, statemachine etc	34
4.3 SPI - speed settings	35
4.4 Checking the Target Voltage	35
4.5 Testing SPI communication with the 500 series device	36
4.5.1 Overview	36
4.5.2 Diagnostic Info	36
4.5.3 Possible failure messages	37
4.6 NVR Area - reading / writing in EDS mode	38
4.6.1 Overview	38
4.6.2 NVR Area tab - explanation	38
4.6.3 Reading the 'NVR parameters' from a device	39
4.6.4 Exporting specified 'NVR parameters' to an NVR parameter file	40
4.6.5 Importing an 'NVR parameter file' back into EDS	41
Appendix 1 – Standalone Programming Mode setup	43
1.0 Overview	43
1.1 Information required to create a Standalone Programming Project	44



1.2 Software versions for Standalone Programming mode	
2.0 Explanation of 'standalone mode projects'	45
3.0 Running a 'Standalone Programming Project'	46
4.0 Creating a 'Standalone Programming Project'	
4.1 Getting started	
4.2 Header Information screen	
4.3 Programmer and Project Type options	48
4.4 Select the 'Target Device'	49
4.5 Select the 'SPI frequency' for programming	50
4.6 Sotting up the 'Chin Erase' / 'Amond NI/P parameters' options	
4.0 Setting up the Onlp Llase / America NVN parameters options	
4.7 FLASH Area - programming options	
4.0 Final Congratulations screen - saving your project	
4.9 Importing an INVR parameter file into a standalone project	
4.10 Compliing your standalone project	
4.11 Adding your standalone project to a Project Collection	
4.12 Uploading your standalone project to a programmer	
Appendix 2 - ISP-PRO script - how to set up the PRODUCTION standalone project(s)	
1.0 Overview	
2.0 Explanation of 'standalone projects'	
2.1 Opening the Sigma Project Collection	59
2.2 Changing the 'Product firmware' file	60
2.3 Uploading the new Project Collection to the programmer	
Appendix 3 - Setting up the 'Tx Power' parameters	67
1.0 Overview	67
1.1 Why do I get a 'TX Power parameter' error?	67
1.2 Where do I find the 'Tx Power' settings in Sigma's SDK	68
1.3 'Tx Power' parameters – overview of merging process	69
1.4 'Tx Power' parameters – merging into the FLASH hex file	70
A manual line A - A sufficient in a the 7 Marca I Frederica al Manual a title manual with MAN assessed	
Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter	ers 73
1.0 Overview	ers 73 73
1.0 Overview	ers 73 73 73
Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 73 74
 Appendix 4 - Configuring the 2-wave "External Non-volatile memory (NVM)" parameters 1.0 Overview	ers 73 73 73 74 74
 Appendix 4 - Configuring the 2-wave "External Non-volatile memory (NVM)" parameter 1.0 Overview	ers 73 73 73 74 74 74
 Appendix 4 - Configuring the 2-wave "External Non-volatile memory (NVM)" parameter 1.0 Overview	ers 73 73 74 74 74 75 values75
 Appendix 4 - Configuring the 2-wave "External Non-volatile memory (NVM)" parameter 1.0 Overview	ers73 73 74 74 74 75 values75 77
 Appendix 4 - Configuring the 2-wave "External Non-volatile memory (NVM)" parameter 1.0 Overview	ers73 73 74 74 74 75 values75 77 78
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers73 73 74 74 74 75 75 75 75 75 77 78 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers73 73 73 74 74 74 75 75 75 75 75 75 75 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-Volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 74 74 74 74 75 75 75 75 75 75 75 79
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 73 74 74 74 75 75 75 75 75 75 75 75 75 79 79 79 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 74 74 74 75 75 75 75 75 75 79 79 79 79 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-Volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 74 74 74 75 75 75 75 75 75 79 79 79 79 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-Volatile memory (NVM)' parameter 1.0 Overview. 1.1 NVM memory – configuration parameters overview. 2.0 No external NVM (memory) device fitted on target board. 3.0 Custom NVM (memory) device fitted on target board. 3.1 How to work out the NVM (memory) device parameter values. 3.2 Configuring the script file to program custom NVM (memory) device parameter values. 3.3 Configuring individual NVM (memory) device parameters. 3.4 Testing the custom NVM (memory) device parameters. Appendix 5 – Restore 'NVR Area' script. 1.0 Overview. 1.1 Modifying the default 'Restore NVR' script . 1.2 Testing the 'Restore NVR' script in ISP-PRO. 1.3 Checking the final 'NVR settings' are correct. Appendix 6 - ISP-PRO - Quick Start Guide. 1.0 Overview. 2.1 Overview. 2.2 Overview of files required for ISP-PRO software. 2.3 Setting up ISP-PRO to run the Programming Script(s). 2.4 Start the ISP-PRO application and log in 	ers 73 73 74 74 74 74 75 75 75 75 75 75 79 79 79 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-Volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 74 74 74 74 75 75 75 77 78 79 79 79 79 79 79 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-Volatile memory (NVM)' parameter 1.0 Overview	ers 73 73 74 74 74 75 75 75 75 75 77 78 79 79 79 79 79 79 79 79 79
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview. 1.1 NVM memory – configuration parameters overview. 2.0 No external NVM (memory) device fitted on target board. 3.0 Custom NVM (memory) device fitted on target board. 3.1 How to work out the NVM (memory) device parameter values. 3.2 Configuring the script file to program custom NVM (memory) device parameter values. 3.4 Testing the custom NVM (memory) device parameters. 3.4 Testing the custom NVM (memory) device parameters. Appendix 5 - Restore 'NVR Area' script. 1.0 Overview. 1.1 Modifying the default 'Restore NVR' script . 1.2 Testing the 'Restore NVR' script in ISP-PRO. 1.3 Checking the final 'NVR settings' are correct. Appendix 6 - ISP-PRO - Quick Start Guide. 1.0 Overview. 2.0 Installing the Sigma scripts and projects. 2.1 Overview. 2.2 Overview of files required for ISP-PRO software. 2.3 Setting up ISP-PRO to run the Programming Script(s). 2.4 Start the ISP-PRO application and log in. 2.5 Setting up the correct COM port. 2.6 Detecting the attached programmer(s). 2.7 Selecting a Script File to run 2.8 Userding up up Reviet Collection to the programmer(a). 	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-Volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73
 Appendix 4 - Configuring the 2-wave 'External Non-volatile memory (NVM)' parameter 1.0 Overview	ers 73



2 12 Programming sequence - FAII	95
3.0 Setting up an Incremental Serial Number	96
3 1 Overview	96
3.2 Importing a Serial Number from file	96
Annendix 7 - EL ASH Eile - CRC32 Checksum	aa
1.1 Using EOTools - EDS to undate the CBC32 checksum	100
1.2 Why do L got an invalid ELASH CPC32 checksum error?	100
1.2 Why do right an invalid r LASIT CICC32 Checksum enory	101
1.5 Loading Statuard Z-Wave Firmware releases from Signa	101
1.4 Loduling a custoffi Z-vvave Filliwale release	101
1.5 How to correct / add a valid CRC52 checksum to an input file	101
1.6 Opening the project in EDS (Development mode)	102
1.7 Calculating the CRC32 checksum of the input file	102
1.8 Saving the revised file with the CRC32 checksum	104
Appendix 8 - Sigma SD3502 Evaluation Module	105
1.0 Overview	105
1.1 Equipment required	105
1.2 Connecting the programmer to the Sigma eval module	106
Appendix 9 – Sigma connector definitions	107
1.0 Sigma - ISP Header Selection	107
Appendix 10 – Connecting the Z-Wave target board to an ISPnano programmer	109
1.1 IOMOD10 module - connections to Z-Wave target board (SPI interface)	109
1.2 IOMOD7 module - connections to Z-Wave target board (SPI interface)	111
1.3 IOMOD4 connector module - connections to Z-Wave target board (SPI interface)	112
1.4 IOMOD4 connector module + Clock Buffer - connections to Z-Wave target board (SPI	
interface)	113





1.0 Introduction

Equinox Technologies manufacture a comprehensive range of programmers suitable for high-speed In-System Programming (ISP) of *Sigma 500 series Z-WAVE devices*. This application note describes how to develop and implement *In-System Programming (ISP)* support for the Z-WAVE devices using the '*SPI Programming Interface*'. The document details how to make a '*Programming Project*' which will operate on any Equinox ISP programmer including a full description of how to implement *In-System Programming (ISP)* of Z-WAVE devices.

1.1 Features

The Equinox programming range includes solutions for development, low / mid / high volume production and field programming of *Sigma 500 series Z-WAVE SOC (System on Chip) devices*. and *Z-WAVE modules*.

General features.....

- High-speed In-System Programming (ISP) support of Sigma 500 series Z-WAVE SOC devices and modules
- Programming solutions for development, low / mid / high volume production and field programming of Z-WAVE devices
- Programs the on-chip FLASH Memory and NVR area of Z-WAVE devices
- Uses a high-speed 'SPI bus' port as the ISP interface
- Very high-speed programming due to fast SPI programming interface, local user data storage and optimised programming algorithms
- Programmers can be used in 'Standalone Mode' (no PC required)
- Supports high-speed program / verify of the on-chip FLASH in a single operation.
- Fully user-configurable pre-programming statemachine supports custom target reset circuits

In 'Development Mode'.....

- Powerful yet simple-to-use Development Suite called 'EDS'
- All aspects of programming the *Z*-*WAVE* device can be controlled from *EDS*
- Program and read back the Sigma device on-chip FLASH memory under PC control
- All projects can be developed and tested on a real device before uploading a 'Standalone Programming Project' to the programmer
- Tested '*Programming Projects*' can then be uploaded to the programmer for use in '*Standalone Mode*'

Production Programming solutions.....

- Programmers can be used in 'Standalone Mode' (no PC required)
- A single 'Standalone Programming Project' can Erase the device and program /verify the FLASH area in a single operation.
- Up to 64 x Z-WAVE '*Standalone Programming Projects*' can be stored inside the ISPnano programmer.



- Programmer can store multiple versions of firmware for different '*customer product versions*'.
- Support for programming unique data per device including serial numbers, MAC addresses, calibration data, barcode data etc.
- **ConsoleEDS** powerful '**console application**' allows the programmer to be controlled from any custom remote application.
- *ISP-PRO* powerful production control / sequencing utility supports controlling of up to 32 programmers from the same PC.
- **ISPnano-MUX** programmer family supports sequential programming of up to 8 x independent Target Boards (UUTs) on a '**PCB Panel**'
- **ISPnano-GANG** programmer family supports concurrent gang programming of up to 32 x independent Target Boards (UUTs) on a '**PCB Panel**'



1.2 Programmers supporting Z-WAVE 500 series devices

The 'Z-WAVE 500 series' devices are currently only supported by the Equinox 'ISPnano' family of production ISP Programmers. The 'ISPnano' programmers can be upgraded to support high-speed programming of via the 'SPI Programming Interface'.

The table below lists all the Equinox ISP programmers which are capable of programming '**Z**-WAVE **500 series'** devices....

Programmer	Sigma Z-wave Support	Requirements	Upgrade Order Code
 ISPnano Series III ISPnano Series III ATE ISPnano Series IV ATE ISPnano-MUX 	Upgrade	License upgrade	ISPnano-UPG35

Please note:

- A chargeable '*License Upgrade*' is required to enable the '*Z-WAVE 500 series*' device support on any of these programmers.
- The *programmer firmware* will probably also need to be upgraded in order to support 'Z-WAVE 500 series' device programming – see section 1.3.
- It is also recommended that EQTools version 4 build 3498 or above is used when programming 'Z-WAVE 500 series' devices.

1.3 Calibration overview

1.3.1 Overview

The Z-Wave 500 series SOC devices and modules must be calibrated at the customer production programming stage before they will operate correctly.

The calibration procedure(s) required depend on whether you are programming a Z-Wave module, SOC (just the bare IC) or 'Bare die' version of the Z-Wave product.

The table below details which calibration procedure(s) is / are required for the different Z-Wave product types.....

Sigma product family	Z-Wave Product type	Customer Crystal (XTAL) calibration required	Customer TX calibration required	Equinox IOMOD10 Calibration module required
ZM5101	SiP Module	NO (Sigma factory calibrated)	YES	NO
ZM5xxx ZDB5xxx (Except ZM5101)	Module	NO (Sigma factory calibrated)	NO (Sigma factory calibrated)	NO
SD35xx	SOC (System On chip - Bare IC)	YES	YES	YES
ZW05xx	Bare die	YES	YES	YES
ALL	Any product type where the 'NVR Area' has been erased or corrupted.	YES	YES	YES

1.3.2 Crystal (XTAL) calibration

The 'Crystal (XTAL) calibration' procedure tunes the TX- and RX radio frequency of the Z-Wave device so as to give the minimum frequency error. This calibration must be carried out on the final crystal which will be used with the Z-Wave device.

Important notes:

- The 'Crystal (XTAL) calibration' procedure is performed by Sigma at the factory for all Z-Wave modules as these modules have the final crystal already fitted to them.
- If you are programming SOCs (bare ICs) or 'Bare Die' products, then these devices are <u>NOT</u> pre-calibrated by Sigma at the factory. You will need to perform the 'Crystal (XTAL) calibration' procedure on these devices at the customer production programming stage.



1.3.3 TX calibration

The **'TX calibration'** procedure is required to tune the Z-Wave frequency separation during modulation to an optimum value.

Important notes:

- The 'TX calibration' procedure is currently NOT performed by Sigma at the factory.
- The customer must therefore perform the '*TX calibration*' procedure on ALL Z-Wave modules, SOCs and 'Bare die' devices at the '*customer production programming*' stage.#

1.3.4 Calibration recovery after accidental NVR erasure / corruption

If the 'NVR Area' of a 500 series Z-Wave device is accidentally erased or corrupted, then both the 'Crystal (XTAL)' and 'TX' calibration parameters may be invalid. This means that the Z-Wave device will no longer function properly.

If this happens, then it is necessary to fully re-calibrate the Z-Wave device. This recalibration process involves performing both the '*Crystal (XTAL) calibration*' and the '*TX calibration*' procedures and also programming some default 'factory settings' for the device or module back into the 'NVR Area' of the device.

This procedure requires the following equipment and other information:

- An ISPnano Series 4 or ISPnano-MUX programmer
- An Equinox 'IOMOD10 Sigma Calibration Module' plugged into the programmer
- A special 'Restore NVR Calibration' script
- A custom parameter file to restore the relevant 'factory parameters' to the device



1.3.5 Programmer selection guide for Z-Wave calibration

The table below details which Equinox programmers are capable of performing the 'Crystal (XTAL) calibration' and 'TX calibration' procedures.

Programmer name	Crystal (XTAL) calibration supported	TX calibration supported	IOMOD10 Calibration module required
ISP nano Series III	NO	YES	Not applicable
Series III - ATE	NO	YES	Not applicable
Series IV - ATE	YES	YES	YES 1 x IOMOD10 module for 'XTAL Calibration'
Multi-Channel Gang Production ISP Programming Systems	NO	YES	Not applicable
2, 4 or 8 Channels	YES	YES	YES Requires 2, 4 or 8 x IOMOD10 modules for 'XTAL Calibration'



1.4 Device Support

1.4.1 Overview

The Equinox ISPnano programmer range supports the following 'Z-WAVE 500 series' - SOC (System on Chip) devices and Z-WAVE modules...



Z-Wave Next Gen SoCs (500 series): SD3502, SD3503



Z-Wave Next Gen modules (500 series): ZM5101, ZM5202, ZM5304



1.5 Upgrading your Equinox Programmer to support Sigma 500 series Z-WAVE device programming

1.5.1 Overview

The Sigma '**Z**-WAVE 500 series' algorithms are not supported as standard on any Equinox programmers. It is necessary to purchase a '*License Upgrade*' for '**Z**-WAVE 500 series' support from Equinox. Equinox will then send you a '*Upgrade License String*' which will upgrade your programmer to support programming of this device family.

1.5.2 Purchasing a Sigma Z-WAVE 500 series License

All Equinox ISP programmers require the purchase of a '*License Upgrade*' to enable '*Z-WAVE 500 series*' programming support. Please see the table in section 1.2 for the relevant upgrade for your programmer.

1.5.3 How do I enable the programmer for Z-WAVE programming?

To enable your programmer to support **'Z-WAVE 500 series'** ISP programming, please purchase the relevant upgrade from Equinox or an Equinox distributor:

1. If you purchase the upgrade directly from Equinox

- Equinox will email you a 'JTAG License String'.
- This string can be entered directly into the *Enter License* screen in EQTools.
- 2. If you purchase the upgrade from a distributor
 - The distributor will send you the Upgrade Pack by courier.
 - Within the Upgrade Pack you will find an Upgrade Form with a Code String on it.
 - Email this Code String plus your programmer 'Serial Number' to support@equinox-tech.com
 - Equinox will then send you a '*License String*' which is keyed to your programmer Serial Number.
 - This string can be entered directly into the *<Enter License>* screen in EQTools.



1.5.4 Entering the License String to upgrade your programmer

Once you have received the License String from Equinox, please follow the steps below to apply the upgrade to your programmer:

- Launch EQTools → The EQTools 'Welcome Screen' is displayed.
- Close down the EQTools 'Welcome Screen'
- From the top menu bar, select <Programmer><Programmer Info>
- \rightarrow the Programmer Information screen is displayed
- Click the <*Enter License*> button
- → The <*Enter License Key*> screen is displayed.

Enter Licen	ce key 🔀
۴	It is possible to purchase License Upgrades from Equinox for this product which will enable certain utilities or Device Libraries. Please refer to the Equinox Website (http://www.equinox-tech.com) for a full list of upgrades for this product or e-mail support@equinox-tech.com.
	Please enter the 24 character Hexadecimal licence key provided by Equinox to enable certain options. Key EAF997545585EE8A5854AA50
	<u> </u>

Enter the License String you were sent by Equinox

- Click <OK>
- \rightarrow EQTools should acknowledge that the attached programmer has been upgraded.

Informa	tion		×
•	Operation: Result:	Update Programmer Licence information Pass	
	Press <ok> to</ok>	view the updated programmer information.	
		ОК	

- Click <OK>
- If you now check the Programmer Info screen, you should find that the entry for 'Sigma 500 Series devices' is now ENABLED.



1.6 Programmer firmware versions for Sigma 500 series support

Most Equinox ISP Programmers can be upgraded to support high-speed programming of 'Z-WAVE 500 series' microcontrollers via the 'SPI Programming Interface'. The table below lists all the Equinox ISP programmers which are capable of programming 'Z-WAVE 500 series'. A chargeable 'License Upgrade' is required to enable the 'Z-WAVE 500 series' support on any of these programmers.

Fig. 1.3 Programmer firmware versions for 'Z-WAVE 500 series' In-System Programming (ISP) Support

Programmer	'Z-WAVE 500 series' support
ISPnano Series III	6.28
ISPnano Series IV	6.28
ISPnano-MUX 2 / 4 / 8	6.28

Please note:

 Due to limited firmware storage space and the lack of required hardware on the EPSILON5-MK4 and FS2009 / FS2009USB and PPM4-MK1 programmers, these programmers cannot support the 'Z-WAVE 500 series' devices.



2.0 Z-Wave - Programming Interfaces

2.1 Overview

The 'Z-WAVE 500 series' devices can be programmed using three different physical 'programming interfaces' as detailed in the table below.

Interface		Comment
USB	USB Interface	Uses the USB port of the Z-Wave device to program the
		on-chip FLASH memory.
		This programming mode still requires an external device
		programmer to set the device into 'programming mode'
		via the SPI or UART interface before the UART
		programming interface can be used.
SPI	SPI Programming	Uses an SPI Port + RESET pin as an In-System
	Port	Programming (ISP) interface
UART	UART interface	Uses a 2-pin UART interface as an In-System
		Programming (ISP) interface

2.2 SPI - Programming Interface - Features

- Fast Programming speed (compared to UART interface)
- Simple 3-wire SPI bus connection + RESET_N signal





2.3 Z-WAVE single-chip In-System Programming (ISP) Schematic

The diagram below details the connections required to implement In-System Programming (ISP) of a single 'Z-WAVE 500 series' device via the 'SPI programming interface' using an Equinox ISP programmer.

Fig 2.3 - 'Z-WAVE 500 series' device - SPI Programming Interface connection



Fig 2.3.b – Sigma Z-wave 500 series device - SPI Signal names and directions

Programmer Signal Name	Signal description	Signal direction (from Programmer)	Connect to Z-Wave Pin	Signal direction (from Microcontroller)
PROG_MOSI	Master OUT, Slave In	Output	MOSI	Input
PROG_MISO	Master IN, Slave OUT	Input	MISO	Output
PROG_SCK	Serial Clock	Output	SCK	Input
PROG_RESET	RESET	Output	RESET_N	Input



2.4 ISPnano programmer - Target ISP Port - SPI connections

The table below details the connections for programming a Z-WAVE 500 series device via the 'SPI Interface' using the 'Target ISP Port' on the 'ISPnano Series 3', or 'ISPnano Series 4' programmer..



'Target ISP Connector' port

The connector is a 16-pin bump-polarised IDC connector with 0.1" pin spacing.

Pin 1 is the top right pin as shown in the diagram opposite.

Pin No	Programmer Pin name	Programmer Input / Output	Connect to pin on target Z-Wave device	Notes
1 + 2	TARGET_VCC	Ρ	TARGET_VCC	Target VCC
3 + 4	TARGET_EXT_VCC	Р	See notes.	Target External VCC
5 + 6	PROG_GND	Ρ	Signal GROUND (0V)	Signal Ground Connection
12	Programmer I/O3	I/O	SPI - SCK	SPI - Serial Clock Signal
13	Programmer I/O2	I/O	SPI - MISO	SPI – Master In Slave Out
14	Programmer I/O1	I/O	SPI - MOSI	SPI – Master Out Slave In
16	PROG_RESET	0	RESET_N	Target RESET control pin

O - Output from programmer to Target Device

I - Input to programmer from Target Device

P - Passive e.g. GROUND and power rails

N/C - Not connected



2.4 Z-WAVE UART ISP Schematic

The diagram below details the connections required to implement In-System Programming of a single 'Z-WAVE 500 series' device using an Equinox ISP programmer via the 'UART interface'.

Fig 2.3 - 'Z-WAVE 500 series' device - UART Programming Interface connections







UART Interface to a 3.3V UART Programming Equipment

F :-	. 0 1 6	Ciamaa	7	EOO agrica	daviaa		Cianal		d dive ations
r-IC	1/40-	Sioma	z-wave	SUU Series	CARVICA	- HARI	Sionai	names an	a airections
' 'S	2.1.0	oigina	2 11010	000 00//00	001100	0/ 11 (/	Cigiiai	numbe un	a an oonono

Programmer Signal Name	Signal description	Signal direction (from Programmer)	Connect to Z-Wave Pin	Signal direction (from Z-Wave device)
PROG_TXD	UART TRANSMIT	Output	UART0 Rx (P2.0)	Input
PROG_RXD	UART RECEIVE	Input	UART0 Tx (P2.1)	Output
PROG_RESET	RESET	Output	RESET_N	Input





3.0 Creating an EDS (Development) Project

3.1 Overview

This section describes how to make a 'Programming Project' for a 'Z-WAVE 500 series' device.

Please note:

The following versions of EQTools and firmware are required to support a '**Z**-WAVE 500 series' device programming:

- EQTools version 4 build 3490 or higher
- Firmware 6.11 please consult Equinox

3.2 Information required to create an EDS Development project

The following information is required about your Target Board / application in order to create a development project for a **'Z-WAVE 500 series'** device.

#	Information / data required	Example
1	Sigma Z-WAVE device part number	SD3502
2	Connector on Target board	10-way IDC connector (SPI version)
3	Programming interface (SPI or UART)	SPI
4	Target System Vcc voltage	e.g. 3.3V
5	Target System maximum current consumption	e.g. 100mA
6	FLASH area 'Program File'	Binary (*.bin) or Intel Hex (*.hex)
7	Reset circuit parameters	 e.g. Capacitor / Resistor circuit Watchdog supervisor circuit Voltage monitoring circuit
8	TX Power parameters	The 'TX Power parameters' should be obtained by from RF testing of your final product.



3.3 Creating an EDS (Development project)

The simplest way to create a Programming Project for a JTAG device is to use the **EDS** (*Development Mode*) Wizard.

The steps required to create a project are as follows:

- Click the 'New' icon on the task bar
- \rightarrow The 'New items' screen will be displayed...

Nev	w Items			X
N	lew	~	X]
	Development	Project Collection	Project Source	
	Project			
	1			
	Script Wizard	Script File		
			<u>o</u> k	<u>C</u> ancel

- Select < Development Project > and click < OK >
- \rightarrow The EDS (Development) Wizard will launch
- Select the relevant 'Programmer' and then click <Next>



3.4 Selecting the correct Target Device

It is important to select the correct '*Target Device*' when programming a '*Z-WAVE 500 series*' device. The part number of the device should be printed on the top of the chip e.g. '*SD3502*'.

3.4.1 Device selection

- Click $\langle Next \rangle \rightarrow$ the $\langle Select Target Device \rangle$ screen will be displayed.
- Type in the 'Device Part Number' e.g. 'SD3502' into the 'Search for Device' field
 → a list of all matching devices will be displayed in the box underneath.

Select Target Device	×
Search for Device by Name	Device Details Notes Timings
	Manufacturer:
Search by Signature	Sigma Designs
	Family:
Target Programming Interface	Z-Wave 500 series - Serial Interface SoC
	Device Code:
All Search Now	SD3502 (SPI)
▷ - 🕌 Holtek	Target Programming Interface
D - IC Microsystems	SPI 3-wire + RESET_N (Z-Wave)
▷ • 👜 ISSI ▷ • 🛄 Microchin	Elash Size:
▷ · · · NXP(Philips)	131072 (0x20000)
⊳ - 🏭 Philips	Elach Start Address
Amtron	
▷ · · · · · · · · · · · · · · · · · · ·	0 (0.0)
▷ - 📔 Seiko Instruments 🛛 🗉	EEPROM Size:
a 📲 Sigma Designs	256 (0x100)
Z-Wave Single Chips Z-Wave 500 arrive Sector Sector	Signature:
Z-wave Sou series - Serial Interface Soc Sp3502 (SPI)	0x7F1F0401
SD3502 (UART)	Algorithm Version:
Vinbond	0.20
Library: SD3502 (SPI).XMI Version: 0.	20 1135 devices loaded

- As the *Z-wave 500 series* devices can be programmed via different 'programming interfaces', the device list shows the available interfaces eg. SPI or UART interfaces for the SD3502 device.
- Select the required device / programming interface from the list e.g. 'SD3502 (SPI)' and then click <OK>
- → The SD3502 device is now selected and will be programmed via the 'SPI' interface..



3.4.2 Device Chip ID / Signature

 On the next screen, check that the device selection and all other device parameters are correct

	get device fro	m the list and devic	e signature(s) to be checked	d 💽
Selected Device	Courthan 1			
Sigma Designs	Z-Wave 50)0 series - Serial	Interface SoC	Select Device
Device Code: Device		Algorithm Version: Silicon Revision:		L
SD3502 (SPI)	0.20		?	
Flash Size:	Flash Page:	NVR Size:	NVR Page:	
131072 (0x20000)	256	256 (0x100)	1	•
Target Programming Inter	face	Polling I	Method	
SPI 3-wire + RESET N (Z-	-Wave)	Conver	ntional BYTE polling	

- The project is set to automatically read and validate the '*Device Signature*' of the Target Device by default.
- The actual '*Signature / Chip ID*' for the device being programmed can be found in the User Manual for the device. Alternatively, it can be read from the target device using EDS.



3.5 Target System – Power Supply Settings

This screen allows you to set up the 'Power Supply' characteristics of your Target System.

Equinox Development Suite(F Target System Power Select Target System Voltag	EDS) Wizard Untitled
Target Voltage Settings	Programmer Controlled Power Supply Programmer controlled Target Power Supply: ON
3.3 🐑 <u>Iolerance</u> (mV) 500 🐑 <u>S</u> tabilise Time (ms) 200 💭	Maximum Current (mA) Powerdown Time (ms) 200 1000 Current Settle Time (ms) PSU Out OK Delay (ms) 100 500 100 100 Power Status at end of project: Power Supply is left switched ON at end of project
	Target Discharge Circuit Target Discharge Circuit ON Powerdown Time (ms) Discharge Voltage 1000 (**********************************
	Enabled Enabled at end of project

i. Select the Target Voltage

- This should be the voltage at which the Target Z-Wave device itself is being powered at during the programming operation. This is usually 3.0 3.6V.
- If the target board is being externally powered, then the programmer voltage should be set so that it matches the I/O voltage of the Z-Wave device.
- Set the 'Voltage Tolerance' to be as wide as possible e.g. 500mV to allow for power supply variations. If the programmer is powering the Target System, this will also give a faster power-up time.
- It may also be possible to power the entire Target System by feeding in a higher voltage e.g.
 +5V into the power supply input on the Target System.

ii. Set up the Target Powering and current parameters

- This option is only available for the PPM3-MK2, PPM4-MK1 and ISPnano programmers.
- If the programmer is to power the Target System, select < Programmer controlled Target Power Supply: ON>
- Set the '*Maximum Current*' to the maximum possible current which the Target System could draw from the programmer.
- Leave all other settings as default.



3.6 Erase options

This screen allows you to set up the 'Erase options' for the target device....

🔐 Equinox Development Suite(EDS) Wizard Untitled	- • 💌
Erase Options Tick check box to Erase target device, and set post erase delay	**
Erase Device	
In Full Chip Erase - All FLASH and EEPROM (if present) areas is erased. Security Fuses are also erased.	ased
Program Memory Erase - FLASH Program Area only is erased'	
☑ Backup NVR Area (from 0x10 to 0xFF) before Erase	
Restore NVR Area backup after Erase	
Pause after Erase (ms)	

Backup NVR Area before Erase

If this option is selected, the programmer will automatically read back the 'NVR data area' from the target device before a 'Chip Erase' operation is performed.

Restore NVR Area after Erase

If this option is selected, the programmer will automatically restore the '*NVR data*' contents which were read back from the target device before the '*Chip Erase*' operation was performed.



3.7 Specifying the FLASH (Code) File

This screen allows you to specify the *Code (firmware*) file which is to be programmed into the FLASH area of the Target Device.

Equinox Development Suite(EDS) Wizard Untitled	
FLASH Area Programming Options Select the required programming options for the FLASH memory area	\$
Blank Check Flash	
Operation: O <u>N</u> one O <u>P</u> rogram/Verify <u>V</u> erify Only Flash File <u>Timings</u>	
Eile: C:\test\Sigma\serialapi_controller_static_ZW050x_EU_crc.hex Browse	661
Status: Loaded OK Type: Intel Hex (Generic)	-
Buffer Image: State of the	
Min Add. 0x0000 Max Add. 0x1FFFF Bytes: 0xDFB0 CRC 0x4F92	-
● Auto Range Custom: Write From 0x0000 To 0x1FFFE Bytes: 0x1FFFF	

This is an optional step – you can also specify the file once you are in the Development Suite (EDS).

Selecting the FLASH File

- Click the <Browse> button
- Browse to and select the file you wish to load and then select <OK>
- --> The file will be automatically loaded into the 'FLASH File Preview' window see below...

Flash File Preview			- • ×
🔲 View entire area	☑ Discard leading 0xFF	☑ Discard trailing 0xFF up to page boundary	
0x00000: 02 18 0x00010: 41 62 0x00020: 22 D3 0x00030: 22 FB 0x00040: B3 FB 0x00040: B3 FB 0x00050: 70 FB 0x00050: 70 FB 0x00060: E4 FB 0x00070: 13 22 0x00080: 98 FB	3 00 02 18 03 6C 88 2 2 02 18 13 03 47 3 22 02 18 23 A2 42 FF 02 18 33 A2 44 FF 02 18 43 02 2A FF 02 18 53 C2 21 FF 02 18 63 E4 FF FF 02 18 73 C2 D5 FF 02 18 73 C2 D5 FF 4 9A FA E4 99 F9	25 E8 22 02 18 0B 6D C31 %è"mÃ 03 5B FF 02 18 1B 7F 01 Ab"G.[ÿ]. 92 30 22 02 18 2B C2 36 "Ó"#¢B'0"+Å6 92 37 22 02 18 3B 02 21 "ÿÿ3¢D'7";.! D4 FF FF 02 18 4B E4 A1 °ÿÿC.*ÔÿÿKäi D2 1E 22 02 18 5B 02 61 pÿÿSÅ!Ò."[.a 02 0F 5B 02 18 6B E5 F7 äÿÿcäÿ[kå÷ E8 30 E7 0F B2 D5 E4 C3"ÿsÅÕè0ç.²ÕäÃ E4 98 F8 FC 30 E7 17 B2 10å!úå!úå!da.2	H T
Sigma 500 Checksum: Inp	out file CRC = 0xFFA83B10,	Calculated CRC = 0xFFA83B10	Ōĸ
Sigma 500 Checksum: Inp Min Addr. = 0x0000	Max Addr. = 0xFFA83B10,	Calculated CRC = 0xFFA83B10 Bytes Loaded = 57264 (0xDFB0) CRC = 0xFFA83B10	<u>Q</u> K File Format: Intel Hex (Generic)





- If the input file is a '**BINARY file'** then the wizard will load the data in from file starting at address 0x0000 and continuing contiguously to the end of the file.
- If the input file is an '**INTEL HEX**' or '**Motorola S-Record**' file, then the wizard will load in from file from the start address specified in the file to end address specified in the file.



3.8 Launching EDS at the end of the EDS Wizard

Once you reach the end of the EDS Wizard, click the *<Test>* button to launch the project in the Equinox Development Suite (EDS).



Enter a name for the EDS project e.g. **SD3502** and click the <**Test**> button \rightarrow Your project will now launch in EDS (Development) Mode.





4.0 Testing a Project in Development (EDS) Mode

4.1 Introduction to EDS

If you have clicked the **<Test>** button at the end of the EDS Wizard, then an EDS (Development Mode) session will now launch.

rogrammer prmation: / Modify Bas troject file to te this projec	Target Device e Programming I a new Project C	Target Oscillator Project	SPI Settings	Target Power Supply	Erase	Flash	NVR	Security	SRAM		
rogrammer ormation: / Modify Bas troject file to te this projec	Target Device	Target Oscilļator Project	SPI Settings	Target Power Supply	Erase	Flash	NVR	Security	SRAM		
rogrammer ormation: / Modify Bas troject file to te this projec	Target Device	Target Oscillator Project	SPI Settings	Target Power Supply	Erase	Flash	NVR	Security	SRAM		
ormation: / Modify Bas Project file to te this project	e Programming I a new Project C	Project									
/ Modify Bas Project file to te this projec	e Programming I a new Project C	Project									
/ Modify Bas roject file to te this projec	e Programming I a new Project C	Project			_						
roject file to te this projec	a new Project C	Copen / Modify Base Programming Project									
te this projec	a new Project C	🛱 Add Project file to a new Project Collection									
te this projec		olectori									
	t in an existing F	Project Collection									
gma\SD3502	-3.PPM										
met		Author									
inc.											
	Croption Do	ta									
rsion:											
	10/12/2013										
gs:											
		Value									
evice		SD3502									
ning Interfac	e	LV SPI 3-wire +	RESET								
oltage		3.3V +/- 500m	V								
ner Powers T	arget	No									
ramming Stat	e Machine	1 - Active LOW	RESET - RESE	T Driven HIGH (50ms) a	and LOW	(50ms)					
		C:\\Sigma\se	erialapi_contro	ller_static_ZW050x_EU	_crc.hex						
		None selected			_						
sPI:											
PI:		921.6 KHz									
		115.2 KHz									
	me: rsion: pte: To upl ng Project gs: evice ning Interfac oltage ner Powers T camming Stat ngs: e SPI: pI:	me: rsion: <u>C</u> reation Da 10/12/2013 pte: To upload this proje ng Project Collection or o gs: evice ning Interface oltage ner Powers Target :amming State Machine ngs: e SPI: PI:	me: <u>A</u> uthor: rsion: <u>Creation Date</u> 10/12/2013 ote: To upload this project to a program ng Project Collection or create a new Pro gs: <u>Value</u> evice SD3502 ning Interface LV SPI 3-wire 4 oltage 3.3V +/- 500m' ner Powers Target No ramming State Machine 1 - Active LOW C:\\Sigma\se None selected ngs: e SPI: PI: 921.6 KHz	me: <u>Q</u> reation Date rsion: <u>Q</u> reation Date 10/12/2013 T ote: To upload this project to a programmer, please ng Project Collection or create a new Project Collection gs: Value evice SD3502 ning Interface LV SPI 3-wire + RESET oltage 3.3V +/- 500mV ner Powers Target No ramming State Machine 1 - Active LOW RESET - RESE C:\\Sigma \serialapi_control None selected ngs: e SPI: PI: 921.6 KHz	me: <u>Q</u> reation Date rsion: <u>Q</u> reation Date 10/12/2013 T ote: To upload this project to a programmer, please either add the project generated of the project Collection. gs: Value evice SD3502 ning Interface LV SPI 3-wire + RESET oltage 3.3V +/- 500mV ner Powers Target No ramming State Machine 1 - Active LOW RESET - RESET Driven HIGH (50ms) a C:\\Sigma\serialapi_controller_static_ZW050x_EU None selected None selected	me:	me:Author:	me:Author:	me:Author: rsion:reation Date 10/12/2013 pte: To upload this project to a programmer, please either add the project to ng Project Collection or create a new Project Collection. gs: gs: vice SD3502 ning Interface LV SPI 3-wire + RESET Atage 3.3V +/- 500mV ner Powers Target No ramming State Machine 1 - Active LOW RESET - RESET Driven HIGH (50ms) and LOW (50ms) C:\\Sigma\serialapi_controller_static_ZW050x_EU_crc.hex None selected ngs: e SPI: PI: 921.6.KHz	me:Author:	



4.2 EDS - Default settings for SPI, statemachine etc

The following default settings will be used:

• 'Hardware SPI' interface

At this stage there are still a few parameters which may need to be set up / checked before the programmer will communicate with the Target Device on the Target Board.

Please follow the instructions in the next sections which explain how to set up the:

• Test the Target Voltage



4.3 SPI - speed settings

The 'SPI speed' should be set up before any programming operation can take place.

4.4 Checking the Target Voltage

It is a good idea to check that the target device is powered at the correct voltage before trying to program it. A Sigma Z-Wave device normally runs at between 3.0 and 3.6V. The programmer '*Target Vcc'* pin should be connected to the 3.3V rail on the Target System allowing the programmer to measure the Target Voltage (even if the programmer is not powering the Target System).

To check the Target Voltage using the programmer, please follow the instructions detailed below....

• Select the <Target Power Supply> tab

Overview Programmer	Target Device	Target Oscillator	JTAG Settings	Target Power Supply	Flash Fuses			
Target Voltage Settings Programmer Controlled Power Supply								
⊻oltage 3.3 🚔		Programmer control	r Supply: ON	*				
		Aaximum Current (m	A) <u>P</u> owerdo	Powerdown Time (ms)				
Tolerance (mV)		Surrent Settle Time (me) PSU Out	💌 OK Delau (ms) Voltage	Settle Time (mc)			
<u>500</u> Stabilise Time (ms)		500 🗲	800	501 5 01	D 🚖			
200 🚔		Power Status at end of project:						
		Power Supply is switched OFF at end of project						
		Power up	<u>M</u> easu	re V/I 🛛 🐺	Reset			

If the programmer is going to power the Target System.....

- Set up the voltage / current parameters accordingly (see programmer User Manual for detailed instructions)
- The '*Target Voltage*' should be set the actual voltage which the Z-Wave device is running at e.g. 3.3V.
- The programmer will then generate JTAG signals which swing between 0V and the '*Target Voltage*'.
- Click the *Power up* button to power up the Target System.
- → The programmer will then switch on the programmer controlled power supply and the Target System should power up to the specified voltage.
- The measured '*Target Voltage*'. will be continuously displayed. If it is not, then you can simply click the *<Measure V/I>* button.
- The voltage should be within 3.0 and 3.3V.

If the programmer is <u>NOT</u> powering the Target System...

Switch on the independent power supply which is connected to the Target System.



Click the <*Measure*> button to measure the '*Target Voltage*'.

• The voltage should be within 3.0 and 3.6V.

4.5 Testing SPI communication with the 500 series device

4.5.1 Overview

To make sure that the programmer can communicate with the target 500 series device, try reading back the *Device Signature (Device ID)* as follows:

- Select the <Target Device> tab
- Click the <Check ID> button
 - \rightarrow The programmer will now try to communicate with the Target Chip via the JTAG Interface
 - → If the Target Chip responds correctly, then EDS will report '*Signature Check Result: Pass*'.

Informatio	on	7	×
1	Operation: Result:	Signature Check PASS	
	Signature Read: Target device: Prog. Interface:	0x7F7F7F7F1F0401 SD3502 LV SPI3-wire + RESET	
			Diagnostic Info >>
		ОК	

• The Signature (Device ID) is displayed e.g. 0x7F7F7F7F7F1F0401

This message means that the programmer has established a connection via the SPI interface to the specified target device and that the device has the correct '*Signature / Device ID*' as specified in the device library.

4.5.2 Diagnostic Info

Every time the programmer enters programming mode, it will return detailed diagnostic information about the target device. This information includes the Target Voltage, oscillator frequency and FLASH timings.

To view the 'Diagnostic information':

- Click the <Diagnostic Info> button on any EDS screen
- Select the <Diagnostic Information> tab
- \rightarrow The diagnostic information is displayed as shown below.....


4.5.3 Possible failure messages

The action of performing a *Check ID* can produce any of the following error messages:

- i. Error 3039 / 3044 Failed to enter programming mode
- ii. Error 44 / 3041 Signature failure: Read back: 0x?????? Expected: 0x??????

These errors are discussed in the next two sections.



4.6 NVR Area - reading / writing in EDS mode

4.6.1 Overview

The '*NVR Area*' of a Z-Wave device contains all the '*factory parameters*' which are programmed into the device when the device is calibrated in the factory by Sigma. This data is fundamental to the operation of the Z-Wave module so it is very important that this data is never erased or corrupted. The '*NVR Area*' contains calibration coefficients and also some parameters to which allow the specification of external components to the Z-Wave IC including a Non-Volatile Memory (NVM) and 'Saw filter' components.

Using the *EQTools - EDS* tool, it is possible to read, backup, modify and write data in the Z-Wave '*NVR Area*'. It is also possible to export a list of '*NVR parameters*' with specified values which can then be imported both into a '*Standalone Programming Project*' or an '*ISP-PRO programming script*'.

4.6.2 NVR Area tab - explanation

To read or write the '*NVR Area*' of a Z-Wave device, select the '*NVR*' tab of the EDS session. A sample of a typical '*NVR*' screen with all data set to 0xFF is shown below....

Overview Programmer Target Device SPI Settings Target Power Supply Eras	e Flash NVR Security SRAM T×Pow	er Settings	
NVR File C:\test\Sigma\CalibrationSequence2\ZM5202_EU_NVR_101014.hex Ill for before file load Ill Automatically reload into buffer on change Automatically upload to target o	n change 🛛 Show NVR parameter view	Updated: 10/10/2014 09:45:03	Edit Buffer Ele Open
0x00: FF	NVR Parameters Report NVR Layout Revision VNR layout revision (REV): 0xFF	Non-Volatile Memory (NVM) Parameters Type (NVMT): 0xFF	Eil Eil Re Calc. CRC Target Device:
0x50: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF 0x60: FFFFFFFFFFFFFFFFFFFFFFFFFFF 0x70: FFFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF 0x80: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	☑ Pin Swapping (PINS): 0xFF	Size (NVMS): 0xFFFF Page Size (NVMP): 0xFFFF Chip Select (NVMCS): 0xFF	Reset Power up Check ID
0×0 : FF	USB Parameters USB Vendor ID (IDVEN): 0xFFFF USB Product ID (IDPROD): 0xFFFF	SAW Parameters ✓ SAW Center Frequency (SAWC): 0xFFFFFF ✓ SAW Bandwidth (SAWB): 0xFF	Blank Check
HEX Buffer view of the 'NVR Area'	Calibration Parameters Crystal Calibration (CCAL): 0xFF	Frequency Calibration 868.4MHz (TXCAL1): 0xFF Frequency Calibration 868.4MHz (TXCAL2): 0xFF	♥ <u>W</u> rite ♥ <u>V</u> erify
	Viniversally Unique Identifier (UUID): DXFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	Import Export	Re Calc CRC

- The 'HEX Buffer' view of the 'NVR Area' on the left shows a hex representation of the entire 256 bytes of the 'NVR Area'.
- The 'NVR Parameter' view of the 'NVR Area' on the right lists the main 'NVR Parameters' and their associated values.



4.6.3 Reading the 'NVR parameters' from a device

The 'NVR parameters' can be read from the 'NVR Area' of a Z-Wave device and displayed on screen.

Instructions:

- Select the EDS 'NVR' tab
- Click the 'Read' button on the right-hand side of the screen
- EDS will now prompt you for the address range. Click 'OK' to read the entire 256 bytes back from the target device 'NVR Area'.
- The read back 'NVR data' will now be displayed both in the 'HEX Buffer' view and the 'NVR Parameter' view - see screenshot below.....

Overview Programmer Target Device SPI Settings Target Power Supply	Erase Flash NVR Security SRAM Tx	Power Settings	
NVR File C:\test\Sigma\CalibrationSequence2\ZM5202_EU_NVR_101014.hex C:\test\Sigma\Calibratio	net on channe. 📝 Show NVR parameter view	Updated: 10/10/2014 09:45:03	ad Edit Buffer
Automatically reload into buffer on change Automatically upload to tar 0×00 : FF	get on change Show NVR parameter view NVR Parameters Report NVR Layout Revision 0.01 VINR layout revision (REV): 0.01 Pin Swapping (PINS): 0.01 USB Parameters 0.01 USB Parameters 0.01 USB Parameters 0.01 USB Parameters 0.01 USB Product ID (IDVEN): 0.07FFF Calibration Parameters 0.080 Universally Unique Identifier (UUID): 0.0.7FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	Non-Yolatile Memory (NYM) Parameters Type (NVMT): 0x02 Size (NVMS): 0x0080 Page Size (NVMP): 0x0100 Chip Select (NVMCS): 0x04 SAW Parameters 0x02 SAW Center Frequency (SAWC): 0x0D1B78 SAW Bandwidth (SAWB): 0x20 Frequency Calibration 868.4MHz (TXCAL1): 0xFF Frequency Calibration 868.4MHz (TXCAL2): 0xFF Import Export	Erase Ell Reset Calc. CRC Target Device: Reset Power up Check ID Check ID Erase Reset Bank Check Reset Check ID Check ID

The most important parameters to check are:

1. Crystal Calibration (CCAL)

If this field is 0x80, then the device is not XTAL (crystal) calibrated.

2. Frequency calibration (TXCAL1 / TXCAL2)

If either parameter is 0xFF, then the device has not been 'TX calibrated'.

3. Non-volatile (NVM) parameters

These parameters set up the type of external 'Non-volatile memory (NVM)' device attached to the Z-Wave device or module.



4.6.4 Exporting specified 'NVR parameters' to an NVR parameter file

If you need to fix certain '*NVR parameters*' to specified values during the programming process, then this can be achieved by creating an '*NVR parameter file (*.NEF)*'. This file contains a list of specified parameters with their required 'fixed' values.

A 'NVR parameter file (*.NEF)' can then be imported either into a 'Standalone programming project' or an 'ISP-PRO programming script' and will then 'fix' the values of the specified parameters.

Instructions:

- Select the EDS 'NVR' tab
- Read the 'NVR area' back from a golden sample device which has all the correct values already programmed into it.

or

• Manually enter the required values into the relevant 'NVR Parameter' fields.

In the 'NVR Parameter' view

- Untick all the parameters which you do NOT wish to export to the 'NVR parameter file (*.NEF)'.
- Tick all the parameters which you wish to export to the 'NVR parameter file (*.NEF)'.
- Check that the values of all the parameters you want to export are the correct value.
- In the example screenshot shown below, only the external 'Non-volatile memory (NVM)' parameters are selected for export.....

NVR Parameters Report		
NVR Layout Revision	Non-Volatile Memory (NVM) Paramete	rs
NVR layout revision (REV): 0×01	V Type (NVMT):	0x02
Pin Swapping (PINS): 0x01	Size (NVMS):	0×0080
	🔽 Page Size (NVMP):	0×0100
	Chip Select (NVMCS):	0x04
USB Parameters	SAW Parameters	
USB Vendor ID (IDVEN): 0×FFFF	SAW Center Frequency (SAWC): 0x0	D1B78
USB Product ID (IDPROD):	SAW Bandwidth (SAWB):	0x20
Calibration Parameters		
Crystal Calibration (CCAL): 0x80	Frequency Calibration 868.4MHz (TXCAL1):	0×FF
	Frequency Calibration 868.4MHz (TXCAL2):	0×FF
🔲 Universally Unique Identifier (UUID):		
0×FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF		
CRC (CRC16): 0x9CDF		
Calculated CRC 0x9CDF	Import	Export



To export the selected 'NVR Parameter' fields

- Click the <*Export*> button
- The 'Export NVR Parameters' dialog box will now be displayed.....

Even out NIVP Doctoremeters	
export work Parameters	
The following NVR Parameters will be export	ed to an NVR Parameter file
Parameter	Value
Non-Volatile Memory Type (NVMT)	0x02
Non-Volatile Memory Size (NVMS)	0x0100
Non-Volatile Memory Page Size (NVMP)	0x02
Non-Volatile Memory Chip Select (NVMCS)	0x04
	OK Cancel

- The parameters you selected should now be displayed with the 'fixed values' which you want to program into the target device.
- Check that all the required parameters have been exported and that the values are correct.
- Once you are happy the exported parameter list is correct, click the <OK> button
- It is now possible to save the 'NVR Parameter list' to a file.
- Enter a suitable file name e.g. *NVR_Exported_parameters.NEF* and then click 'Save' to save the parameter list to a *.*NEF* file
- The 'NVR Parameter list' is now available as a *.NEF file. This *.NEF file can be imported into a 'Standalone programming project' or an 'ISP-PRO programming script'.

4.6.5 Importing an 'NVR parameter file' back into EDS

If you need to check the parameters and parameter values contained in an 'NVR Parameter file' file, then the simplest way to achieve this is to import the file back into EDS.

To import an 'NVR Parameter file' file back into EDS....

- Select the EDS 'NVR' tab
- Click the <*Import*> button
- Select the relevant 'NVR Parameter file (*.NEF)' and click <Open> to import this file.
- The following screen will now be displayed....





- Click <OK> to load the parameters / values from the 'Parameter list' into the EDS 'NVR' page.
- The specified parameters / values from the file should now be merged into the existing '*NVR Area*' data displayed on the '*NVR*' tab.



Appendix 1 – Standalone Programming Mode setup

1.0 Overview

This section describes how to make a '*Standalone Programming Project*' for an Equinox programmer. A '*Standalone Programming Project*' is a single file (*.prj) which contains all programming actions and user data required to program a target Z-Wave device including the FLASH CODE File, '*NVR Parameter file*', '*Configuration Fuse*' settings and '*Security Fuse*' settings.

A '*Standalone Programming Project*' is generated using the '*EQTools – Project Builder*' utility. The illustration below shows the process of creating a project.



For most Equinox programmers, it is possible to have up to 64 x '*Standalone Programming Projects*' resident in the programmer memory at any point in time. A '*Standalone Programming Project*' can be executed by an Equinox programmer without requiring a PC to control the programmer. This makes it possible for the programming sequence to be triggered by an ASCII

AN145- In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices



command via the RS232 / USB port, via the 'START' button on the programmer, via the 4-wire '*Remote Control Port*' or via a '*Jig switch*' on the programming fixture lid. As there is no requirement for a PC to control the programmer, this 'Standalone mode' operation is ideal for interfacing the programmer to systems such as ICTs, PLCs etc.

1.1 Information required to create a Standalone Programming Project

The following information is required about your Target Board / application in order to create a 'Standalone Programming Project' for a 'Z-WAVE 500 series' device.....

#	Information / data required	Example
1	Sigma Z-WAVE device part number	SD3502
2	Programming interface	SPI or UART
4	Target System Vcc voltage	e.g. 3.3V
5	Target System maximum current consumption	e.g. 100mA
6	FLASH area 'Program File'	Binary (*.bin) or Intel Hex (*.hex)
7	NVR Parameter File	e.g. NVR_parameters.NEF (*.NEF)
8	Tx Power parameters	The ' Tx Power parameters' should be obtained by from RF testing of your final product and then merged into your ' Final FLASH hex file' .
9	Configuration Fuse settings	See Z-wave datasheet for details (if unsure, leave as default settings)
10	Security Fuse settings	See Z-wave datasheet for details (if unsure, leave as default settings)

1.2 Software versions for Standalone Programming mode

The functionality of being able to program a Z-Wave 500 series device in *'Standalone Programming Mode'* requires the following versions of both EQTools and programmer firmware to be installed....

#	Software / firmware	Version required
1	EQTools - programmer configuration software	Version 4 build 3660 or above
2	Programmer firmware version	6.28 or above

Important note:

If you use an earlier version of firmware, the **'NVR parameter list'** will not be programmed into the target device. Please ensure you are using the more up-to-date firmware version.



2.0 Explanation of 'standalone mode projects'

A **'Standalone Programming Project'** is used for high-speed programming of a Z-Wave device where no unique data per DUT such as a serial number needs to be programmed.

A 'Standalone Programming Project' supports the following functionality:

- Applies power to the DUT
- Enters programming mode
- Checks that the 'Device ID / Signature' of the target device is correct.
- Erases the 'FLASH area' and 'NVR Area' of a Z-Wave device
- Reads back and then validates the existing 'factory calibration data' stored in the 'NVR Area' of the Z-Wave device.
- Validates that the 'factory calibration data' stored in the 'NVR Area' is not corrupted (checks REV and CRC16 checksum)
- Validates that the '**TX Calibration**' parameters are within limits (i.e. the device is correctly TX calibrated)
- Validates that the '**XTAL Calibration**' parameters are within limits (i.e. the device is correctly XTAL calibrated)
- Validates that the 'Non-Volatile Memory (NVM)' parameters are not erroneous
- Backs up the existing 'factory calibration data' stored in the 'NVR Area' of the Z-Wave device before a 'Chip Erase' operation and automatically restoring this data after the 'Chip Erase'.
- Programs fixed value '*NVR parameters*' into the '*NVR Area*' of a Z-Wave device. e.g. '*Non-volatile memory (NVM)*' fixed values.
- Programs a fixed 'FLASH Firmware' hex or binary file into the 'FLASH area' of the Z-Wave device
- Programs the 'Configuration Fuses' of the Z-Wave device
- Checks that the data programmed into the 'FLASH Area' is correct by validating the CRC32 FLASH checksum
- Locks the device (setting the 'Lock bytes')
- Power down / discharge the DUT

A 'Standalone Programming Project' does <u>NOT</u> support the following functionality:

- Re-calibrating the 'TX Calibration' parameters if the Z-Wave device is not calibrated correctly.
- Re-calibrating the 'XTAL Calibration' parameters if the Z-Wave device is not calibrated correctly.
- Programming unique data e.g serial numbers, MAC addresses etc into the 'NVR Area'
- Recovering the data in the 'NVR Area' if it has been accidentally erased or corrupted.



3.0 Running a 'Standalone Programming Project'

Once a '**Standalone programming project**' has been created, it must be uploaded to the programmer using the '**Upload Wizard**' utility (part of EQTools) first before it can be executed.

It is then possible to execute a '*Standalone programming project'* using any of the following control methods:

Control method	Control method overview	Number of standalone projects supported
START button	Pressing the 'START' button on the programmer	1
Lid switch	Closing the ' <i>Fixture Lid Switch</i> ' contacts on the programmer	1
4-wire TTL port	Remote System e.g. ICT controls programmer execution via 4 x TTL control signals	1
ASCII protocol	Simple ASCII serial protocol	64



4.0 Creating a 'Standalone Programming Project'

4.1 Getting started

To make a 'Standalone programming project' please follow the instructions detailed below...

- Start EQTools
- From the top menu bar, select 'File New'
- Select the 'Project source' icon and click <OK>

v Items Iew			×
Development Project	Project Collection	Project Source Scrip	t Wizard
		QK	Cancel

- Select the 'Project source' icon and click <OK>
- The 'Project Builder Wizard' will now be displayed.
- Do NOT change any settings on this screen. Click the <Next> button --> the 'Task options' screen will be displayed
- Click the <Next> button --> the 'Header Information' screen will be displayed.

4.2 Header Information screen

The 'Header Information' screen is used to set up the following:

- **Project name (UniqueID)** This is used to reference projects once they have been uploaded to the programmer.
- Project File Name This is the actual file name of the project.
- **Project Version Control information** This information allows the project to be given a unique and traceable version control string.

🕞 Project Builder Wizard Untitled	
Header Information Enter the required information for th	e project header. The 'Unique ID' is also used as the filename.
Unique Project Id (filename) ZWFIRMWARE1	Append Build Date (mmddyy) Append Version Append Custom + + + + + + + + + + + + + + + + + + +
Project Name: ZWFIRMWARE1	Author: John Marriott
Project <u>V</u> ersion: <u>M</u> ajor version: <u>Min</u> or version: <u>R</u> elease 1 🗶 0 💭 0	e Build
Court from Darla	Auto-increment build number
Creation Date	Comments:
15/01/2015	
Embed actual build date and time in PF	RJ (If unchecked will embed Creation Date)

AN145- In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices



Instructions:

- Type in a suitable 'Project name / file name' in the 'Unique Project Id (filename)' field. e.g. 'ZWFIRMWARE1'
- This name will be used for both the project 'UniqueID' reference and also as the 'Project file name'.

4.3 Programmer and Project Type options

The 'Programmer and Project Type options' screen is used to set up the following:

- The programmer which the project is to be used with.
- The 'Project Type' which defines how the 'standalone project' will be controlled.

		pe or project	~
rogrammer and Project Type Programmer:	• Get Info	Minimum Firmware Version:	
Project <u>T</u> ype	det mon	6.00 😇	
Standalone - Keypad Control		•	
In this mode the programmer can	be controlled from the	programmer keypad.	
In this mode the programmer can	be controlled from the p Programmer Keypad L	programmer keypad. ock	
In this mode the programmer can Skip Image Check Enable ces:	be controlled from the p	programmer keypad.	

Instructions:

- 1. Select the 'Programmer' you want the project to work with
 - Select the correct programmer from the drop-down list e.g. *ISPnano Series III / IV*.

or

- Click the **<Get info>** to detect the attached programmer automatically (The programmer must be already connected to the PC COM / USB port).
- 2. Set the 'Project Type' to match your required control method for the project.
 - It is possible to leave the 'Project Type' set to 'Standalone keypad control' for now.
 - The 'Project Type' setting can be changed globally for all projects via the 'Global options' in EQTools overriding the setting in any individual project.
- 3. Leave all other settings as the default values
- 4. Click the *<Next>* button to go to the next screen.



4.4 Select the 'Target Device'

The 'Target Device' tab is used to select the device / IC to be programmed and also the 'programming interface' which is being used to connect the programmer to the target device.

elected Device				
Manufacturer:	Family:			
Device Code:	Device	Algorithm Version:	Silicon Revision:	elect Device
Flash Size:	Flash Page:	Eeprom Size:	Eeprom Page:	
Target Programming I	nterface	Polling	Method	
Signature <u>1</u>	Sign	ature <u>2</u>	Signature 3	
Signature <u>1</u>	Sign	ature <u>2</u>	Signature <u>3</u>	
Signature <u>1</u>	Sign	ature <u>2</u>	Signature <u>2</u>	•

Click the <Select Device> button

🔝 Select Target Device	
Search for Device by Name SD3502	Device Details Notes Timings Manufacturer:
Search by Signature	Sigma Designs Family:
<u>T</u> arget Programming Interface All ▼ Search <u>N</u> ow	Z-Wave 500 series - Serial Interface SoC Device Code: SD3502 (SPI)
Image: Second	Iarget Programming Interface SPI 3-wire + RESET_N (Z-Wave) Flash Size:
> - ⊌s Rohm > - 12 ST > - 12 Seiko Instruments ▲ - 12 Sigma Designs	131072 (0x20000) Flash Start Address: 0 (0x0)
Z-wave Single Chips Z-Wave 500 series - Serial Interface SoC Jost S03502 (SPI) S03502 (UART)	EEPROM Size: 256 (0x100) Signature:
> - 32 Winbond > - 32 Xicor > - 32 Zensys	0x7F1F0401 Algorithm Version: 0.20
Library: SD3502 (SPI).XML Version: 0	.20 1145 devices loaded

To select the device you wish to program:

• Type the 'Device code' e.g. SD3502 into the 'Search for Device by name' field and then click the <Search> button

or

• Find 'Sigma Designs' in the tree in the bottom window and then click the triangle symbol to open the tree up to display the 'Z-Wave 500 series' devices.

Select the required device depending on which 'programming interface' is being used:



- SD3502 (SPI) SPI Interface (MOSI, MISO, SCK, RESET_N)
- **SD3502 (UART)** UART Interface (TXD, RXD, RESET_N)

The 'Target Device' tab should now display the selected device and 'programming interface'

elected Device								
Manufacturer:	Family:							
Sigma Designs	Z-₩av	e 500 serie:	s - Serial	Interfa	ce SoC		Select	Device
Device Code:	De	evice Algorithr	n Version:	Silicon I	Revision:			
SD3502 (SPI)	0	.20		?				
Flash Size:	Flash Pa	ge: NVR Siz	e:		NVR Page:			
131072 (0×20000)	256	256 (0:	×100)		1	-		
Target Programming In	nterface		Polling	Method				
SPI 3-wire + RESET_M	N (Z-Wave)		Conve	ntional B	YTE polling			
Signature <u>1</u> 0x7F7F7F7F1F04	101							

Click the *<Next>* button to go to the next screen.

4.5 Select the 'SPI frequency' for programming

The 'SPI Settings' tab is used to set up the 'SPI frequency' which is used by the programmer to communicate with the target Z-Wave device.

SPI Settings Select the required SPI frequency and delay times	
External Oscillator O Internal RC Oscillator	
Target Oscillator Freq: 32.00 MHz	Target Max SPI Freq: 1.00 MHz
Fast SPI Frequency: 921.6 KHz	
Slow SPI Frequency 115.2 KHz	
	Set <u>F</u> astest <u>S</u> et Default
otes:	

- It is best to leave both the 'Slow' and 'Fast' SPI frequencies at the default settings at this stage.
- If you encounter reliability problems during programming, then try reducing the 'Fast SPI Frequency'.
- Click the **<Next>** button to go to the next screen.



4.6 Setting up the 'Chip Erase' / 'Amend NVR parameters' options

The 'ERASE' tab is used to configure the following options:

1. Erase Device

When enabled, the programmer will perform a 'Chip Erase' option which will erase both the FLASH and NVR areas

2. Validate / amend NVR parameters

When enabled, the programmer will validate / check the values of the main 'NVR parameters' and it will also merge any customer specific 'NVR parameters' from an 'NVR parameter file (*.NEF)'.

🖁 'Project Builder Wizard ZWFIRMWARE1	_ 0 ×
Erase Options Tick check box to Erase target device, and set post erase delay	**
Notes:	
	< Back Next > Close

Instructions for 'standalone programming projects'....

1. The 'Erase device' option must always be enabled.

2. The 'Validate / amend NVR parameters' option should always be enabled

The programmer will then automatically backup the 'NVR area' before the 'Chip Erase', validate all required factory calibration parameters, merge the 'Custom NVR parameters' from a 'NVR Parameter file' and finally re-program the merged 'NVR data' back into the 'NVR Area' after the 'Chip Erase'.



4.7 FLASH Area - programming options

The 'FLASH Area Programming options' tab allows you to specify the 'FLASH firmware' file which is to be programmed into the 'FLASH Area' of the target device.

🖁 'Project Builder Wizard ZWFIRMWARE1		
FLASH Area Programming Options Select the required programming options for the FLASH men	nory area	て
Blank Check Flash		
Operation: None Operation: None Program only		
Ele: C:\\serialapi_controller_static_ZM5202_U5_CR4	C32.hex Browse	
Status: Loaded OK Buffer	Type: Intel Hex (Generic)	
☑ Discard leading 0xFF ☑ Discard trailing 0xFF		
Min Add. 0x0000 Max Add. 0x1FFFF	Bytes: 0x20000 CRC32 0xC3044916	
Auto Range Custom: Write From 0x0000	Io 0x1FFFF Bytes: 0x1FFFF	
<u>V</u> otes:		
		^
		Ŧ
		_

Instructions:

- Check that the 'Flash file' which you plan to load already has the correct 'CRC32 Checksum' set up in the file.
- Check that the 'Flash file' which you plan to load already has the correct 'Tx Power parameters' set up in the file.
- Click the **<Browse>** button
- Browse to and select the file which you wish to program into the 'FLASH area' of the device.
- Click <OK> to select the file
- The 'File Preview' window will now be displayed.
- If a warning about *'Invalid CRC32 checksum'* is shown, please consult the relevant section in this document to fix this problem.
- If a warning about *'Invalid Tx Power parameters'* is shown, please consult the relevant section in this document to fix this problem.
- Click the *<Next>* button to go to the next screen.



4.8 Final 'Congratulations' screen - saving your project

The 'Congratulations' tab allows you to save your project and even change the 'File name / Unique ID' if you wish.



Instructions:

- Click the <Finish> button to save the project
- You will now be prompted for a file name. The file name you chose at the start of the Wizard will be automatically used unless you choose to change it here.
- Click <Save> to finally save the project.

The project is now displayed in the 'Project Builder' view as shown below



- A series of 'tabs' is displayed along the top of the screen.
- Select the 'Tasks' tab
- Make sure the 'Security Settings' and 'CRC32 Checksum' tasks are ticked (enabled)



4.9 Importing an 'NVR parameter file' into a standalone project

Once the 'standalone programming project' has been set up, it is then possible to import an 'NVR **Parameter file'** which will merge customer specified 'NVR parameter' values with the 'NVR factory **data'** already resident in the 'NVR Area' of the target Z-Wave device.

Please note:

You will need to create an '*NVR Parameter file (*.NEF)* using the EQTools - EDS utility to create this file - see section 4.6 of the main application note for further instructions.

Instructions:

- Select the 'NVR' tab of your project
- Click the **<Browse>** button
- Browse to and select the 'NVR Parameter file (*.NEF)' which contains the list of 'NVR parameters' which you want to merge into the 'NVR Area data' already resident in the Z-Wave device.
- Click <OK> to load this file
- A list of the 'NVR parameters' and associated 'Fixed values' should now be displayed.....

VB Area Programming Ontions	odeDemo\SD3502-SF	PI-SM.PPM							
Select the required programming options for the NVR memory area									
s Header Info Programmer and Project Type	e 🔚 Target Device	Target Oscillator	SPI Settings	Target Power Supply	Pre-Program State Machine	Erase	Elash	NVR	
Blank Check Eeprom									
peration: <u>N</u> one Program only									
NVR File Timings									
Eile: C:\test\Sigma\CalibrationSequence2	\NVR_Exported_param	neters.NEF	Browse	2					
Eile: C:\test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found	\NVR_Exported_param	neters.NEF /pe: Equinox NVR I	Browse Export	2] 661					
Elle: C:\test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay	\NVR_Exported_paran Ty	neters.NEF /pe: Equinox NVR I	Brows e Export	2] &					
Eile: C:\test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay Parameter	\NVR_Exported_param Ty Value	neters.NEF /pe: Equinox NVR	Browse Export	⊶] <i>6</i> 6°					
Eile: C:\test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay Parameter Non-Volatile Memory Type (NVMT):	\NVR_Exported_param Ty Value 0x02	neters.NEF Pquinox NVR I	Brows e Export	.					
Eile: C:\test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay Parameter Non-Volatile Memory Type (NVMT): Non-Volatile Memory Size (NVMS)	VWR_Exported_param Ty Value 0x02 0x0080	neters.NEF	Brows						
Elle: C:\test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay Parameter Non-Volatile Memory Type (NVMT): Non-Volatile Memory Size (NVMS) Non-Volatile Memory Page Size (NVMP)	VWR_Exported_param Ty Value 0x02 0x0080 0x0100	neters.NEF	Brows e						
Elle: C:{test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay Parameter Non-Volatile Memory Type (NVMT): Non-Volatile Memory Page Size (NVMP) Non-Volatile Memory Chip Select (NVMCS)	\NVR_Exported_param Ty Value 0x02 0x0080 0x0100 0x0100 0x04	neters.NEF	Browse						
Elle: C:{test\Sigma\CalibrationSequence2 Status: 4 overlay parameters found NVR Export File Preview of parmeters to overlay Parameter Non-Volatile Memory Type (NVMT): Non-Volatile Memory Page Size (NVMP) Non-Volatile Memory Chip Select (NVMCS)	\NVR_Exported_param Ty Value 0x02 0x0080 0x0100 0x0100 0x04	neters.NEF	Export						

Notes:

- Check that the imported list of 'NVR parameters' is correct.
- EQTools will automatically store all of these parameters in the 'Standalone programming project'.

Important note....

When executing the 'standalone programming project', the programmer will automatically merge the 'NVR parameters' specified in the 'NVR Parameter file (*.NEF)' with the 'Sigma factory programmed data' which is already resident in the 'NVR Area'. The programmer will NOT change any other parameter values in the 'NVR Area'.

AN145- In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices



4.10 Compiling your standalone project

Once you are happy that all the various settings in your 'Standalone programming project' are correct, it is necessary to compile all these settings into the final 'Standalone Programming Project' file.

Instructions:

- Make sure you have your *project file (*.ppm)* open in EQTools.
- Check that all settings in the project are correct.
- On the top EQTools menu bar, click the 'Compile' icon....

Equinox EQTools	
🔄 🖻 <u>N</u> ew 👌 <u>O</u> pen 🔒 Save 🔚 Save All 🗃 Setup 💿 Programmer Info	🔍 Detect Programmer(s) 📕 Download Wizard 🔶 Upload Wizard 🎡 Erint 🛛 🕼 Exit 🖕
🞬 Compile 뺆 Test in EDS 🖧 Inspect Project 🔈 View Log File 🔗 Test Script	🕞 Add Project 🕼 Edit Project 🗋 Delete Project 🕜 Move Up 🕐 Move Down 🛽 🗊 Startup Project
<u>Eile Edit P</u> rogrammer <u>W</u> indow <u>H</u> elp	

Your project will now be compiled to produce a project file *.prj e.g. SD3502-SPI-SM.ppm.

The following 'Information' screen is now displayed.....

Informat	ion							
1	Operation: Result:	Project Compilation PASS						
	Warning(s):	1						
	1. Invalid Sigma	Tx Power Settings value. Please change to a valid setting						
	Source file: Compiled file: Path:	SD3502-SPI-SM.PPM SD3502-SPI-SM.PRJ C:\test\Sigma\StandaloneModeDemo\						
	Options:							
	New Log	g file						
	🛱 🗛 🖂 🖂	ject file to a new Project Collection						
	<u> U</u> pdate t	his project in an existing Project Collection						
	mil <u>T</u> est this	project in Equinox Development Suite						
	ОК							

Important note:

The warning shown means that the '*Tx Power settings*' have not been set up in the '*FLASH File*'. To correct this problem, use the *EQTools – EDS* utility to re-save your 'FLASH File' with valid '*Tx Power settings*'.



4.11 Adding your standalone project to a Project Collection

The compiled '*Project File* *.*prj*' must now be added to a '*Project Collection File* (*.*ppc*)' so that it can be uploaded to a PC.

To add your project to a new 'Project Collection'.....

• Click the 'Add Project file to a new Project Collection' button

Informat	ion		X
	Operation:	Project has been added to a new Project Collection.	
	You now need The name can	to Save the Project Collection file with a suitable name e.g. myprojectcollection.ppc. contain up to 128 characters and can be a mix of upper, lower and alphanumeric charac	ters.
		ОК	

- Click <OK>
- Enter a suitable name for your new 'Project Collection'.....

File name:	SD3502-SM1_Project_Collection
Save as type:	Project Collection Files (*.PPC)
) Hide Folders	Save Cancel

- Click the <Save> button
- Your project will now be added to the new Project Collection and the 'Project Manager' window will now be displayed....

Project	Manager - C:\test	\Sigma\Sta	ndaloneModeDemo\SI	03502-SM1_Project_Colle	ction.PPC					- • ×
Details	iles Power Supp	ly Fuses	Security State Machin	e Programming Speed R	tetries Memory Map					
Number	Unique Id	Version	Build Date	Target Device	Target Programming Interface	Signature	JTAG ID	Project Type	SFM Module required	EQTools Build
0	SD3502-SPI-SM	1.0.0.0	18/01/2015 at 21:28	SD3502 (SPI)	SPI 3-wire + RESET_N (Z-Wave)	0x7F7F7F7F1F0401	N/A	Standalone - Keypad Con	No	4.0.0.3660
•				III						Þ
					Ģ	Edit Project	Test Project i	EDS Upload selected	l project	ad all projects
Programmer	Type: ISPnano Seri	es III/IV To	tal FLASH usage - Used:	130.3 KB (0.8%), Free: 15	.9 MB bytes (99.2%) Progr	ammer Flash Page Size =	= 256			

4.12 Uploading your standalone project to a programmer

To upload the 'Project Collection' to the attached programmer(s).....

- Click the 'Upload all projects....' button
- The 'Upload Wizard' utility will now start.
- Follow the on-screen instructions to upload the project(s).



Appendix 2 - ISP-PRO script - how to set up the PRODUCTION standalone project(s)

1.0 Overview

This section describes how to set up the 'standalone programming projects' for use with the Equinox ISP-PRO software. These instructions are only for updating the project collection used with ISP-PRO and an ISP-PRO programming / calibration sequence script for Z-Wave 500 series devices.

2.0 Explanation of 'standalone projects'

The programming / calibration sequence for a 'Sigma Z-Wave 500 series device' when executed within the Equinox ISP-PRO software, is made up of 4 x 'Standalone Programming Projects' as shown in the illustration below....





The projects within this 'Project Collection' are described below....

Project 1: SD3502-SPI

The first project called 'SD3502-SPI ' is the so-called 'Base Project'.

This project is used by the 'programming script' in ISP-PRO to define the target device, voltage, SPI speed etc.

*** Do NOT edit this project - this is for Equinox use only. ***

Project 1: CALIBRATION

This project programs the 'Calibration firmware' into the FLASH area of the target device. This 'Calibration firmware' firmware must have been compiled for the correct Sigma device and be the correct algorithm for the hardware being calibrated. This project has been pre-prepared by Equinox and you should not need to change this project.

*** Do NOT edit this project - this is for Equinox use only. ***

Project 2: RUNTARGET

This project programs simply powers up the Target System and then forces the **'Calibration** *firmware*' to run / execute the **'Calibration firmware'** by asserting the RESET pin of the DUT. This project has been pre-prepared by Equinox and you should not need to change this project. ***** Do NOT edit this project - this is for Equinox use only. *****

Project 3: PRODUCTION

This project programs the **'Production firmware'** into the FLASH area of the target device. The **'Production firmware'** is the final **'customer firmware'** which needs to be programmed into the device before the product leaves the factory.

Please note:

For most applications, the only project which needs to be modified is the **'PRODUCTION'** project. This project programs the **'Customer firmware file'** and so will probably be unique to each customer application. The next sections explain how to change the **'Customer firmware file'** in the **'PRODUCTION'** project.



2.1 Opening the Sigma Project Collection

To open the 'Sigma Project Collection'....

- Start EQTools
- From the top menu bar, select '*File open*' and browse to and select the '*Project Collection*' file '*Sigma_SD3502_project_collection.PPC*'
- → The Project Collection should open in 'Project Manager' view see screenshot below.....

🔐 Projec	t Manag	ger - C:\tes	t\Sigma\Ca	alibrationSe	equence\Sigm	a_SD3502_project_col	lection.P	PC 🗖 🗖 💌
Details	Files	Power Sup	ply Fuses	Security	State Machine	Programming Speed	Retries	Memory Map
Number	Uniq	jue Id	Version	Build Date		Target Device	Target	Programming Interface
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	SD3 CAL RUN PRC	502-SPI IBRATION ITARGET DUCTION	1.0.0.0 1.0.0.0 1.0.0.0	02/01/201 02/01/201 02/01/201 02/01/201	4 at 19:03 4 at 19:04 4 at 19:05 4 at 19:06	SD3502 (SPI) SD3502 (SPI) SD3502 (SPI) SD3502 (SPI)	SPI 3-1 SPI 3-1 SPI 3-1 SPI 3-1	wire + RESET_N (Z-Wave) wire + RESET_N (Z-Wave) wire + RESET_N (Z-Wave) wire + RESET_N (Z-Wave)
•		III						4
	<u>E</u> dit I	Project] <u>T</u> est Projec	t in EDS	Upload <u>s</u> elected pro	ject	Upload all projects
rogramme	r Type:	ISPnano Ser	ries III/IV T	otal FLASH	usage - Used: 2	260.8 KB (1.6%), Free:	15.7 MB b	oytes (98.4%) Program

As you can see, the 'Project Collection' comprises of 4 x 'Standalone Programming Projects'.

Please see section 2.0 for an explanation of the function of each project.



2.2 Changing the 'Product firmware' file

The test '*Project Collection*' is shipped with an example '*Production firmware*' from Sigma Designs. To program your own '*Product firmware*' into the target Z-Wave device, you will need to change the '*Firmware file*' specified in the '*PRODUCTION*' project to your own '*Product firmware*' file.

To change the 'Product firmware file'....

1. Open the 'Sigma Project Collection' - see section 2.1 for instructions

2. Highlight the 'PRODUCTION' project by clicking on the project name once

--> see screenshot below....

Project	Manag	ger - C:\test	\Sigma\Ca	librationSe	equence\Sigm	a_SD3502_project_col	lection.Pl	
Details	Files	Power Supp	ly Fuses	Security	State Machine	Programming Speed	Retries	Memory Map
Number	Uniq	ue Id	Version	Build Date	۱	Target Device	Target	Programming Interface
📴 0	SD3	502-SPI	1.0.0.0	02/01/201	4 at 19:03	SD3502 (SPI)	SPI 3-v	vire + RESET_N (Z-Wave)
	CALI	IBRATION	1.0.0.0	02/01/201	4 at 19:04 9	SD3502 (SPI)	SPI 3-v	vire +RESET_N (Z-Wave)
6 2	RUN	TARGET	1.0.0.0	02/01/201	4 at 19:05 9	SD3502 (SPI)	SPI 3-v	vire + RESET_N (Z-Wave)
GP 3	PRO	DUCTION	1.0.0.0	02/01/201	4 at 19:06 9	SD3502 (SPI)	SPI 3-v	vire + RESET_N (Z-Wave)
•								4
(🔰 <u>E</u> dit F	Project		<u>T</u> est Projec	t in EDS	1 Upload selected pro	ject	Upload all projects
grammer	Type: 1	ISPnano Serie	es III/IV To	otal FLASH	usage - Used: 2	260.8 KB (1.6%), Free:	15.7 MB b	ytes (98.4%) Progra

3. To open the 'PRODUCTION' project

- Click the 'Edit Project' button or double-click the 'PRODUCTION' project name in the list.

--> The 'PRODUCTION' project will now open in 'Project Builder' view.



4. Overview of the 'Task options' tab

The project will open on the 'Tasks' tab which shows the various 'Tasks' which can be performed by a 'standalone project'....

Task O Select	ptions which of the opti	onal tasks you wish to perform					₹	
Target Power Supply Pre-Program State Machine Erase Elash NVR CRC32 Checksum								
Tasks	<u>H</u> eader Info	Programmer and Project Type	Tar	rget Device	Target Osc	til <u>l</u> ator	SPI Setting	
 ✓ Heade ✓ Target ✓ Atmel ✓ Target 	r Information t Programmer ELF File							

5. Select the 'FLASH' tab.

er Info Ipply Pr ash Progr ne Progr	Programmer and Project Type e-Program State Machine Entire Area am only	e <u>mil T</u> ar E <u>r</u> ase	rget Device Flash	Target Osc NVR	illator CRC	<u>S</u> PI Settin <u>c</u> 32 Checksum			
ne Progr	e-Program State Machine	E <u>r</u> ase	<u>F</u> lash	NVR	CRC	C32 Checksum			
ash √ i ne ⊚ <u>P</u> rogr	ntire Area am only								
(Calibration	Sequence \serialapi_controller		ux_EU_crc.he	X Bro	wse	66			
		i ype	e. pricernex (ochency					
Burrer I Discard leading 0xFF I Discard trailing 0xFF									
Min Add. 0x0000 Max Add. 0x1FFFF Bytes: 0xDFB0 CRC32 0xFFA83B10									
	\Calibration ided OK eading 0xFF x0000	\CalibrationSequence \serialapi_controller ided OK eading 0xFF V Discard trailing 0xFF x0000 Max Add. 0x1FFFF	\CalibrationSequence \serialapi_controller_static_ZW05 ided OK Typ eading 0xFF IDiscard trailing 0xFF x0000 Max Add. 0x1FFFF Byt	\CalibrationSequence \serialapi_controller_static_ZW050x_EU_crc.he ided OK Type: Intel Hex (eading 0xFF Intel Hex (x0000 Max Add, 0x1FFFF Bytes: 0xDFB0	Intel Hex (Generic) aded OK Type: Intel Hex (Generic) eading 0xFF Intel Trailing 0xFF Max Add. 0x1FFFF Bytes: 0xDFB0 CRC32	Intel Hex (Generic) aded OK Type: Intel Hex (Generic) eading 0xFF Image: Discard trailing 0xFF x0000 Max Add. 0x1FFFF Bytes: 0xDFB0 CRC32 0xFFA8			



The project is currently setup with an example 'firmware file' from Sigma as follows.....

Operation:	None O Program only
Flash File	Timings
<u>F</u> ile:	C:\\CalibrationSequence\serialapi_controller_static_ZW050x_EU_crc.hex
Status:	Loaded OK Type: Intel Hex (Generic)
burrer	

6. To change the 'Firmware file'....

Click the **'Browse'** button and then browse to and select the file you want to load. This file can be a binary, Intel Hex or Motorola S-Record format file

--> The 'FLASH File Preview' window is now displayed.....

Flash File Prev	view							- • •
🔲 View entire a	area	Discard leading 0x	(FF 🔽 Discard tra	ailing 0xFF up to page	boundary			
Ox00000:	02 18 00	0 02 18 03 6C	88 25 E8 22 I	D2 18 0B 6D C3		. %è"mÃ		A H
0x00010:	41 62 22	2 02 18 13 03	47 03 58 FF (D2 18 1B 7F 01	Ab"	.G.[ÿ∎.		
0x00020:	22 D3 22	2 02 18 23 A2	42 92 30 22 (D2 18 2B C2 36	"Ó"#	;B′O"+Â6		1010
0x00030:	22 FF FI	7 02 18 33 A2	44 92 37 22 (D2 18 3B 02 21	"ÿÿ3	D17";.!		
0x00040: 1	B3 FF FI	7 02 18 43 02	2A D4 FF FF	D2 18 4B E4 A1	³ÿÿC	.*ÔÿÿKäi		
Ox00050:	70 FF FI	7 02 18 53 C2	21 D2 1E 22	D2 18 5B 02 61	pÿÿS	λ!Ò."[.a		
Ox00060: 1	E4 FF FI	7 02 18 63 <mark>E</mark> 4	FF 02 0F 5B	D2 18 6B E5 F7	äÿÿci	iÿ[kå÷		
0x00070:	13 22 FI	7 02 18 73 C2	D5 E8 30 E7 I	OF B2 D5 E4 C3	."ÿs	ÅÕèOç.²ÕäÃ		-
Sigma 500 Checksum: Input file CRC = 0xFFA83B10, Calculated CRC = 0xFFA83B10							<u>o</u> ĸ	
Min Addr. = 0x00	00	Max Addr. = 0x1F	FFF Bytes	Loaded = 57264 (0xD	DFBO) (RC = 0xFFA83B10	File Format: In	ntel Hex (Generic)

This window displays the following information about the selected input file....

- A preview of the data in the file in both Hex and ASCII format
- The 'Sigma CRC32 FLASH Checksum' value stored in the file (if present)
- The 'CRC32 FLASH Checksum' calculated by EQTools when the file was loaded

!!! Important !!!

i. The input file must have a 'Sigma CRC32 FLASH Checksum' value stored in the last 4 bytes of the file.

ii. The 'Sigma CRC32 FLASH Checksum' value stored in the file must be the same as the 'CRC32 FLASH Checksum' calculated by EQTools when the file was loaded



Once you are happy that the selected input file is OK, click the 'OK' button to load it into the project.

7. The selected 'Firmware file' should now be displayed.

The 'CRC32 FLASH Checksum' calculated by EQTools when the file was loaded is also displayed. See screenshot below....

Target Power Supply	Pre-Program State Machine	E <u>r</u> ase <u>F</u> lash	NVR CRC32 Checksum
Blank Check Flash	✓ Entire Area		
Operation: <u>N</u> one Flash File <u>Timings</u>	<u>P</u> rogram only		
Eile: C:\\Calib Status: Loaded OK	rationSequence\serialapi_controller_s	tatic_ZW050x_EU_crc.hex Type: Intel Hex ((Generic)
Buffer	xFF 🛛 Discard trailing 0xFF		
Min Add. 0x0000	Max Add. 0x1FFFF	Bytes: 0xDFB0	CRC32 0xFFA83B10
🔘 Auto Range (Custom: Write From 0x0000	To 0x1FFFF	Bytes: 0x1FFFF



8. Compile the project

The revised **'PRODUCTION'** project must be compiled and then updated in the Project Collection before it can be uploaded to the programmer.

On the top EQTools icon bar, click the 'Compile' icon.

This will compile the project and then display the following Info screen....

Informat	ion	—
1	Operation: Result:	Project Compilation PASS
	Source file: Compiled file: Path:	PRODUCTION.PPM PRODUCTION.PRJ C:\test\Sigma\CalibrationSequence
	Options:	
	No View Log) file
	🛱 <u>A</u> dd Proj	iect file to a new Project Collection
	<u> U</u> pdate t	his project in an existing Project Collection
	mini <u>T</u> est this	project in Equinox Development Suite
		ОК

- Now click the 'Update this project in an existing Project Collection' button.
- Select the 'Sigma_SD3502_project_collection.PPC' file and click the 'Open' button see screenshot...

Select an Equinox file to Open				— ×			
COO V 🎍 « Local Disk (C:)	▶ test ▶ Sigm	a 🕨 CalibrationSequence 🗸 👻	Search Calibrations	equence 🔎			
Organize 🔻 New folder							
📌 Favorites	N	Name	Date modified	Туре			
🧾 Desktop		Sigma_SD3502_project_collection.PPC	02/01/2014 19:06	PPC File			
🕌 Downloads 📃 Recent Places	E						
🥽 Libraries							
Documents							
Music							
Pictures							
Videos							
Computer	▼ ₹	m		Þ			
File <u>n</u> ame:	File name: Sigma_SD3502_project_collection.PPC						
			<u>O</u> pen ▼	Cancel			



The 'Update Project Collection' process should now report 'PASS'......

Informat	ion									
1	Operation: Result: Project Name Path:	Update Project Collection PASS Sigma_SD3502_project_collection.PPC C:\test\Sigma\CalibrationSequence								
	Options:									
	Select project	from collection								
	Edit chosen Project									
	Test chosen Project in EDS									
		ОК								

Click the 'OK' button to exit this screen.

The '**PRODUCTION'** project should now have been updated with the new 'firmware file' and the 'Build date' should show the new date and time of the file.

Details	Files	Power Sup	oply	Fuses	Security	State Machin	e Programming Spee	d Retries	Memory Map	
Number	Uniq	jue Id	Ver	sion	Build Date		Target Device	Target	Programming	Interface
0	SD3	502-SPI	1.0	.0.0	02/01/201	l4 at 19:03	SD3502 (SPI)	SPI 3-	wire + RESET_	N (Z-Wave)
1	CAL	IBRATION	1.0	.0.0	02/01/201	l4 at 19:04	SD3502 (SPI)	SPI 3-	wire + RESET	N (Z-Wave)
	DUIN	TADCET	1.0	0.0	02/01/201	4 st 10:05	CD2502 (CDT)	CDT 2	uiro I DECET	N (7 Waya)
<mark>⊔</mark> #3	PRC	DUCTION	1.0	.0.0	08/01/201	l4 at 23:11	SD3502 (SPI)	SPI 3-	wire + RESET_	N (Z-Wave)
<										,
<										



2.3 Uploading the new Project Collection to the programmer

Once you have updated the '**PRODUCTION'** project with your '**firmware file**', it is then necessary to upload the entire '**Project Collection'** to the programmer.

To upload the 'Project Collection' to the programmer....

- 1. Make sure the '*Project Collection*' is already open in EQTools
- 2. Make sure the programmer is attached to the PC and is powered on
- 3. Click the 'Upload all projects' button (bottom right of the Project Manager window)

4. If everything is OK, then the 'Upload Wizard' utility will start and the following screen will be displayed.....

🏦 Equinox Upload Wizard								
Upload Project(s) to programmer Select <upload> to upload your selected project(s) to the attached programmer(s)</upload>								
Click 'Upload and Verify' to upload selected project(s) to seleted programmer(s).								
Page:								
<u>T</u> ransfer Baud Rate 230400 ▼								
Dpload and Verify								
Verify only								
< Back Next > Cancel								

5. Click the 'Upload and Verify' button to start the upload process

6. Follow the on-screen instructions to upload the Project Collection

7. Once complete, the projects will then be permanently resident in the '*Programmer FLASH* memory Store'.

Important note:

It is also possible to upload the *Project Collection* using ISP-PRO. In ISP-PRO, select the *'Programming Script File'* and then click the *'Upload Project'* button.



Appendix 3 - Setting up the 'Tx Power' parameters

1.0 Overview

This section describes how to set up the '*Tx Power*' parameters for a Z-Wave 500 series module or SOC device.

The values for the '*Tx Power*' parameters must be derived by experimentation during the development and final R&D testing stages of the customer product. The values are then usually fixed for this product and must be programmed into certain specific locations in the '*FLASH area*' of the Z-Wave device. As these parameter values are fixed values, then these values should be placed in the '*FLASH hex file*' which is used to program the device in production.

Important note:

It is not possible to over-program the '*Tx Power*' parameters after the main FLASH firmware has been programmed. This is because the '*CRC32 checksum*' used to validate that the FLASH has been programmed correctly would need to be changed when the '*Tx Power*' parameters were programmed.

1.1 Why do I get a 'TX Power parameter' error?

If you try to load a HEX or BINARY file into the FLASH buffer in the EQTools software, then the file loader automatically checks the input file to make sure that the '*Tx Power Parameters*' have been configured in the file.

If EQTools detects that the '*Tx Power Parameters*' are all set to 0xFF in the input file, then the following warning message will be displayed.....

Sigma Tx Power S	Sigma Tx Power Settings								
Warning! The input file contains invalid 'Tx Power' parameters. Please set the 'Tx Power' parameters to valid values and then resave the input file.									
	Normal Tx Power:	Low Tx Power:							
Channel 0:	0xFF	0xFF]						
Channel 1:	0xFF	0xFF]						
Channel 2:	0xFF	0xFF							

This error message means that the '*Tx Power Parameters*' are invalid. It is therefore necessary to enter valid values for these parameters which match the required power output for your target product. Please refer to the instructions in the next section for further instructions on how to configure the '*Tx Power Parameters*'.



1.2 Where do I find the 'Tx Power' settings in Sigma's SDK

The **'Tx Power'** parameters for a Z-Wave 500 series module or SOC device will usually have already been setup / tested by an RF engineer during the RF testing of the customer product. If the development engineer has used Sigma's own SDK software, then you should be able to obtain the required **'Tx Power'** parameters from the following screen in Sigma's GUI interface – see screenshot below.....

Programming Inte Current interfac	rface e: SPI			
Flash Code Memo	ry SRAM Extern	al Non-Volatile Memory	NVR	
HEX File: v10	0.5_OTA_ZM5202_I	US_BOOTLOADER 2014-	0	
			1	
Read	Calibrate, Program	n and Verify Compa	are	
Read Erase	Calibrate, Program	n and Verify Compa	are	
Read Erase Options	Calibrate, Program	n and Verify Compa	are	
Read Erase Options	Calibrate, Program	r: Low Tx Power:	are	
Read Erase Options Channel 0:	Calibrate, Program	r: Low Tx Power: 1b	Get Options	
Read Erase Options Channel 0: Channel 1:	Calibrate, Program	r: Low Tx Power: 1b 1b	Get Options Set Options	

If you have a 'golden sample' of your product which already have the correct 'Tx Power' parameters programmed into it, then it is possible to read out the values by clicking the 'Get options' button on the above screen.

Unfortunately, there is no way to export these settings to the Equinox EQTools software, so please make a note of the values for each parameter. You will need these values to enter into hex file.

For many customer products, it may be possible to simply use the default values (supplied by Sigma) for these settings. However, the values used should always be double-checked either with your 'RF engineer' or with Sigma Designs technical support service.



1.3 'Tx Power' parameters – overview of merging process

The '*Tx Power*' parameters for your product must be merged into the '*Production FLASH hex file*' so that they are automatically programmed into the target device at the same time as the '*production firmware*'.

An overview of the steps to integrate your '*Tx Power*' parameters into your final '*Production FLASH Area hex file*' is shown below....

1. Obtain the correct values for '*Tx Power*' parameters either by experimentation or by reading out the values from a '*golden sample*' device which has the correct parameters in it.

2. Overlay these '*Tx Power*' parameters into your '*Production FLASH Area hex file*'
This can now be done using a utility within EQTools.

3. Recalculate the '*FLASH CRC32 checksum*' to take account of the values for the '*Tx Power*' parameters.

- This task is performed by EDS.

4. Save the amended hex file which now has the updated '*Tx Power*' parameters + updated '*FLASH CRC32 checksum*'.

5. This updated hex file should now be used as your 'Production FLASH Area hex file'.

The 'Production FLASH Area hex file' now contains:

- Your 'Z-Wave Firmware data'.
- The correct '*Tx Power*' parameters for your end product.
- The correct 'CRC32 FLASH checksum' for the entire file including the 'Tx Power' parameters.



1.4 'Tx Power' parameters – merging into the FLASH hex file

The '*Tx Power*' parameters for your product must be merged into the '*Production FLASH hex file*' It is possible to perform this task using the EQTools – EDS (Development Mode) utility.

Instructions:

1. Start the Equinox - EQTools software

2. Select the option to 'Create an EDS - Development Project' and follow the wizard to create the EDS project

Or

Open an existing Sigma 500 series EDS project (*.eds)

3. Once the EDS session has started, select the 'FLASH' tab. See screenshot below

Quantian	Description	Tayaah Dawisa	CDI Calificat	Taxaat Daway Supelu	Even	Flach	NUD	Convillation	CD AM	Ty Dewey California	
Overview	Programmer	Target Device	SPI Settings	Target Power Supply	Erase	1 IGSI1	INVR	Securicy	SRAM	TX Power Sectings	
Flash File Update						Updated:				Edit Buffer	
serialapi_	serialapi_controller_static_ZM5202_US_CRC32.hex					25/04/2014 21:41:28					
Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: Constraint of the load Image: C						Save as					
0x0000 0x0001)0: 02 10 (.0: 4E E4 3	00 02 10 03 0 22 02 10 13 0	68 62 68 7I 03 47 03 5E	0 22 02 10 0B 32 3 FF 02 10 1B C2	2C 4E Nä'	hl	oh}" G.[ÿ	.2, /		Erase	
0x0002 0x0003	20: 02 0E 4 30: C8 FF 1	47 02 10 23 (FF 02 10 33)	02 10 70 D3 7F 01 22 FE	3 22 02 10 2B 02 7 FF 02 10 3B A2	1A(4B Èÿy	G#. 73∎	.pÓ" "ÿÿ	+ ;≎K		№ <u>C</u> alc. CRC	

4. Select your 'Production FLASH hex file' as follows...

- Tick the 'Edit buffer' check box on the right-hand side of the screen see screenshot above
- Click the 'File open' button on the right-hand side of the screen
- Browse to and select the 'FLASH Firmware file' which you want to program into the target device.

5. Enter the required 'Tx Power Options' for your product as follows...

• Select the 'Tx Power Options' tab → the following screen should be displayed...

Overview	Programmer	Target Device	<u>S</u> PI Settings	Target Power	Supply	Erase	Flash	NVR	Security	SRAM	Tx Power Settings
Tx Powe	r Options										
			No	rmal Tx Power:	Low 1	Tx Power	:	e			
>>	 Transfer from 	n Buffer Cha	nnel 0: 0>	(3F	0×04	ł		De	fault		
<	< Transfer <u>t</u> o I	Buffer Cha	nnel 1: 0>	(3F	0×04	ł		B	ead		
		Cha	nnel 2: 0>	(3F	0×04	ł		<u>w</u>	rite		
								📝 Re	-calculate a	and Write	e CRC

• Enter the correct values for the 'Tx Power' parameters for your product in the relevant fields

AN145- In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices



• The correct values may be custom to your Z-Wave module or final product so please check the values with your RF engineer or with Sigma's technical support department.

6. Transfer your 'Tx Power' parameter settings to the 'FLASH Buffer'

- Click the 'Transfer to buffer' button to transfer your settings to the 'FLASH Buffer'.
- You should then see the following Information screen

Informat	ion
i	Operation: Update FLASH Buffer with 'Tx Power' parameter settings Result: PASS
	The 'FLASH Buffer Area' has been updated with the 'Tx Power' settings. The data has been written to the address range: 0x07BB2 to 0x07FB7 The FLASH Buffer CRC32 checksum has also been updated to: 0xB7905BF7
	Please check these settings are correct and then save the entire buffer area to a hex file. This new hex file should contain your original 'FLASH firmware' + 'Tx Power' settings + updated CRC32 checksum.
	ОК

- 7. Check the correct values have been transferred to the 'FLASH Buffer'....
 - Select the 'FLASH' tab again
 - Go to address 0x7FB2 in the '*FLASH Buffer*' (Select CTRL + G + then enter the address: 0x7FB2).
 - You should see the 6 bytes values you entered for the '*Tx Power*' parameters now stored at the address range: 0x07FB2 0x07FB7 in the FLASH buffer.

- The 'FLASH CRC32 checksum' (found in the last 4 bytes of the 'FLASH buffer' will also have been automatically updated by EQTools so it is now correct for the new data you have entered.
- 8. Save the updated 'FLASH Buffer' back to your 'FLASH hex file'.....
 - Click the 'Save as' button and then save the entire FLASH area to a new hex file
 - The saved hex file now contains your original FLASH data + '*Tx Power*' parameters + updated '*FLASH CRC32 checksum*'.
 - This hex file can be used to program the final *'production firmware'* into the FLASH area of the target Z-Wave device.

The 'Production FLASH Area hex file' now contains:

- Your 'Z-Wave Firmware data'.
- The correct 'Tx Power' parameters for your end product.
- The correct 'FLASH CRC32 checksum' for the entire FLASH file.




Appendix 4 - Configuring the Z-Wave 'External Non-volatile memory (NVM)' parameters

1.0 Overview

It is possible to connect an **'External Non-volatile memory (NVM)**' device to a Z-Wave 500 series device. This **'NVM device**' is NOT fitted on a Z-Wave module as the module does NOT require the **'NVM device**' to operate. Instead, the **'NVM device**' can be fitted on the customer's target board and is then connected to the Z-Wave device on the Z-Wave module via the 'SPI1' port of the Z-Wave device. The **'NVM device**' is not required for most Z-Wave applications and hence is usually either not catered for on the target board (no footprint provided) or the **'NVM device**' is simply not fitted during the assembly process.

However, for some Z-Wave applications, it is necessary to fit the '*External Non-volatile memory* (*NVM*)' device on the customer target board. In this case, it is essential that the relevant '*External Non-volatile memory* (*NVM*)' parameters are configured so that the Z-Wave device knows the relevant settings of the external '*NVM*' device.

This section describes how to configure / program the Z-Wave '*External Non-volatile memory* (*NVM*)' parameters to match the configuration of your target Z-Wave board.

1.1 NVM memory – configuration parameters overview

The parameters which are used to configure a Z-Wave 500 series device to interface to an '*External Non-volatile memory (NVM)*' device are detailed in the table below.

NVM Parameter	Parameter description	Function of NVM parameter
NVMT	Non-volatile memory	This parameter defines the type of external memory device fitted to the target board.
NVMS	Non-volatile memory SIZE	This parameter defines the physical size in 'kbytes' of the external memory device fitted to the target board.
NVMP	Non-volatile memory PAGE SIZE	This parameter defines the physical 'page size' of the external memory device fitted to the target board.
NVMCS	Non-volatile memory CHIP SELECT	This parameter configures which pin on the Z-Wave device is used to control the 'Chip Select (CS) signal line of the external memory device fitted to the target board.

2.0 No external NVM (memory) device fitted on target board

If you do not have or plan to fit an **'External Non-volatile memory (NVM)**' device on your Z-Wave target board, then there is usually no need to change any of the parameters in the programming script from their default values. If you are programming a Z-Wave **'module'** then the memory parameters should already have been factory programmed by Sigma to declare **'No external NVM fitted'**.

The default factory values of the 'NVM parameters' are shown in the table below

NVM Parameter	Parameter description	Default value
NVMT	Non-volatile memory	0x00
	TYPE	This indicates that an NVM device is NOT fitted.
NVMS	Non-volatile memory	0xFFFF
	SIZE	
NVMP	Non-volatile memory	0xFFFF
	PAGE SIZE	
NVMCS	Non-volatile memory	0x04 ???
	CHIP SELECT	

Please note:

The default version of the Sigma programming script will simply use the default **'NVM settings'** read from the target device and will not change these settings in any way. This plan should work OK for all Z-Wave modules which should have had the **'NVM settings'** pre-programmed by Sigma at the factory.

3.0 Custom NVM (memory) device fitted on target board

If you plan to fit an 'External Non-volatile memory (NVM)' device connected to the Z-Wave device on your Z-Wave target board, then it is necessary to configure a set of specific 'External Non-volatile memory (NVM)' parameters in the Z-Wave 'NVR memory area' of the target Z-Wave device. This configuration tells the Z-Wave device what type of 'External Non-volatile memory (NVM)' device is fitted to the device, what the 'NVM memory size' is, what the 'Page Size' is and also which pin on the Z-Wave device should be used to control the 'Chip Select' pin.

Warning!

Failure to declare the settings for the *'External Non-volatile memory (NVM)'* device correctly could cause the Z-Wave device to malfunction with newer versions of Z-Wave firmware. It is therefore essential that the *NVM* device is correctly configured.



3.1 How to work out the NVM (memory) device parameter values

If you are inheriting a Z-Wave design from an R&D department, then it is likely that the **'External Non-volatile memory (NVM)'** device parameters have already been defined for your Z-Wave target board. In this case, you should be able to simply transfer the values you are given by your R&D department directly into the Equinox programming script. If you have been given a working 'golden sample' of your target board, then it may also be possible to read out the relevant **'External Non-volatile memory (NVM)'** device parameters from the Z-Wave device on this target board.

However, if you do not know the relevant values for the '*External Non-volatile memory (NVM*)' device parameters, please make a note of the part number of the memory device fitted on your target board and then contact Sigma technical support quoting the memory device part number. They should hopefully be able to tell you what values to use for the configuration.

3.2 Configuring the script file to program custom NVM (memory) device parameter values

If you need to program custom values for the '*External Non-volatile memory (NVM)*' device parameters, you will need to amend the '*Sigma programming script*'.

Instructions:

- Start EQTools
- Select *File Open* and then browse to and open the latest version of the Sigma script source file e.g. *Sigma_ZW500_ProgCal_V17-14.ESW*
- The *Script source file (*.esw)* should now open in the Script Builder utility and the following script related tabs will be displayed.....

C	😽 Script Build	er C:\Equi	nox\Sigma'	Calibratio	onSequence2\Sig	ma_ZW500_Prog	gCal_V17-14.ESW	/					- • •
	Select So Select w	c ript Option hich scripting ta	I S asks to exec	ute from t	he list below								22
	Run Target	Write Data to	File Chip	Erase 2	NVR Write Tar	get AutoProg3	Final NVR Read	Final R	Read From File	Final Che	ck Database Values	Flash Checksum	NVR Checksum
	Script Tasks	Base Project	Database	Target Co	onnect/Disconnect	Read Signature	Pre-Erase NVR	Read	Pre-Erase Read	From File	Check Database Valu	es Chip Erase	Target AutoProg1

- Select the 'Write data to File' tab
- On this tab you should see a list of the available 'NVR parameters' which can be configured see screenshot below....

Write Da Add the	ata to File Tasks required file write tas	ks								
Script Tasks	Base Project Datab	ase Target Co	onnect/Disconne	ect Read Signature	Pre-Erase NVR	Read	Pre-Erase Read	From File	Check Database Valu	ues Chip E
Run Target	Write Data to File	Chip Erase 2	NVR Write	Target AutoProg3	Final NVR Read	Final	Read From File	Final Che	ck Database Values	Flash Che



Write data to File

Enabled	Name	Start	DataType	Size (Bytes)	Source	Value	File to update
0	CCAL (XTAL Calibration Byte)	17	BYTE	1	Database	Sigma.CALCULATED_CCAL	%NVR_TEMP%
V 1	TXCAL1	49	BYTE	1	Database	Sigma.CALCULATED_TXCAL1	%NVR_TEMP%
V 2	TXCAL2	50	BYTE	1	Database	Sigma.CALCULATED_TXCAL2	%NVR_TEMP%
3	PINS	18	BYTE	1	Fixed	255	%NVR TEMP%
4	NVMCS	19	BYTE	1	Fixed	0x04	%NVR_TEMP%
5	SAWC	20	BLOCK	3	Database	Sigma.NVR2_SAWC	%NVR_TEMP%
6	SAWB	23	BYTE	1	Database	Sigma.NVR2_SAWB	%NVR_TEMP%
7	NVMT	24	BYTE	1	Fixed	0x02	%NVR_TEMP%
8	NVMS	25	WORD	2	Fixed	0x0100	%NVR_TEMP%
9	NVMP	27	WORD	2	Fixed	0x0100	%NVR_TEMP%
10		32	BLOCK	16	Database	Sigma NVR2 LILITD	%NVR TEMP%
•							

- The 'NVR parameters' which are used to set up the Z-Wave external 'Non-volatile memory (NVM)' are highlighted in the list in red: NVMCS, NVMT, NVMS, NVMP
- By default, the programming script will simply use the value of each parameter (*NVMCS, NVMT, NVMS, NVMP*) which it read from the target device at the start of the script.
- If you wish to configure custom values for the external '*Non-volatile memory (NVM)*' parameters to match the hardware configuration of your target board, then please follow the instructions below....



3.3 Configuring individual NVM (memory) device parameters

The instructions in this section describe how to configure a custom fixed value for each '*Non-volatile memory (NVM)*' parameter. This will allow you to set up the programming script to program custom values into the following NVR parameters: *NVMCS, NVMT, NVMS, NVMP.*

Instructions:

- Double-click the 'NVMCS' parameter in the 'NVR parameter' list
- The 'File Write Parameter' screen for the 'NVMCS' parameter should now be displayed

File Write Para	meter			×
Number:	4			
<u>N</u> ame:	NVMCS			
Error Message				
Data Type:	BYTE		•	
Start <u>A</u> ddres	s rt Address 00013	🔿 Auto Size	O Get from Database	
Data Size(by Fixed Size	tes): : 00001	Auto Size	Get from Database	
S <u>o</u> urce:	Fixed	•		
Value:	0x04		•	
File to write o	lata to 'EMP%			

- Set the 'Value' field to the value you wish to program into the 'NVMCS' parameter. e.g. 0x04
- Check the settings match the screenshot above and leave all other settings unchanged!
- Click **<OK>** to save your amended settings.
- You will then be returned to the '**NVR parameters**' list and the '**NVMCS**' parameter should now have been automatically enabled and should show your amended settings. e.g. Value = 0x04.

	3	PINS	1	8	BYTE	1	Fixed	255		%NVR_TEMP%
	V 4	NVMCS	1	9	BYTE	1	Fixed	0x04		%NVR_TEMP%
٦	5	SAWC	2	0	BLOCK	3	Database	Sigma.N	VR2_SAWC	%NVR_TEMP%

 Repeat the above procedure for each of the 'Non-volatile memory (NVM)' parameters (NVMCS, NVMT, NVMS, NVMP)



• Once you have configured all the relevant parameters, the '*NVR parameters*' list should then show all these parameters as '*Enabled*' and with the correctly configured values....

Enabled	Name	Start	DataType	Size (Bytes)	Source	Value	File to update
V 0	CCAL (XTAL Calibration Byte)	17	BYTE	1	Database	Sigma.CALCULATED_CCAL	%NVR_TEMP%
V 1	TXCAL1	49	BYTE	1	Database	Sigma.CALCULATED_TXCAL1	%NVR_TEMP%
V 2	TXCAL2	50	BYTE	1	Database	Sigma.CALCULATED_TXCAL2	%NVR_TEMP%
3	DING	18	RVTE	1	Fixed	255	%_NIVD TEMD%_
V 4	NVMCS	19	BYTE	1	Fixed	0x04	%NVR_TEMP%
5	SAWC	20	BLOCK	3	Database	Sigma.NVR2_SAWC	%NVR_TEMP%
6	SAWB	23	BYTE	1	Database	Sigma, NVR2_SAWB	%NVR_TEMP%
V 7	NVMT	24	BYTE	1	Fixed	0x02	%NVR_TEMP%
V 8	NVMS	25	WORD	2	Fixed	0x0100	%NVR_TEMP%
V 9	NVMP	27	WORD	2	Fixed	0x0100	%NVR_TEMP%
10		32	BLOCK	16	Datahase	Sigma NVR2 LILITD	%NVR TEMP%

- The script is now configured to automatically program your custom values for the 'Non-volatile memory (NVM)' parameters into the 'NVR Area' of the target Z-Wave device.
- Click the 'Compile' icon on the top EQTools icon bar to generate the amended script file (*.esf).

🔚 Equinox EQTools - [Script Builder C:\Equinox\Sigma\CalibrationSequence2\Sigma_ZW500_ProgCal_V17-14-customNVM.ESW]
Eile Edit Programmer Window Help
<u>New</u> 👌 Open 🔒 Save 🔒 Save All 🖾 Setup 💿 Programmer Info 🔍 Detect Programmer(s) 🐺 Download Wizard 🔝 Upload Wizard 📚 Brint 📳 Exit
🛗 Compile 📓 Jest in EDS 🖧 Inspect Project 🔊 View Log File 🖋 Test Script 🛱 Add Project 🕼 Edit Project 🛱 Delete Project 🕥 Move Up 🕐 Move Down 🚺 Start

• This script file (*.esf) can now be executed within the ISP-PRO production utility.

3.4 Testing the custom NVM (memory) device parameters

To test whether your custom '**NVM parameter values**' have been programmed correctly, you will need to follow the instructions below....

- Execute your customised 'programming script (*.esf)' in the ISP-PRO utility
- Program a virgin Z-Wave target board using your customised 'programming script (*.esf)'
- Exit the ISP-PRO utility
- Start the EQTools software utility
- Open or create a new 'EDS Development project'
- Select the 'NVR' tab
- Click the '*Read*' button → the current values of the '*NVR Area*' will be transferred to the '*Buffer window*'
- Check that the read back values for the '*NVM parameter values*' match the values you declared in your programming script.
- The final test that you have programmed the correct values is to try running Sigma's own firmware on the Z-Wave device and check that it executes correctly. This is not a comprehensive test as some Sigma firmware does not actually require or use the external NVM device!



Appendix 5 – Restore 'NVR Area' script

1.0 Overview

This section describes how to restore the '*NVR Area*' of a 500 series Z-Wave device which has a blank (erased to 0xFF) or corrupt '*NVR Area*'.

The '*Restore NVR Script*' is a special version of the Sigma '*Programming / calibration*' script which will restore / re-program the '*NVR parameters*' back into a Z-Wave device.

1.1 Modifying the default 'Restore NVR' script

The '*Restore NVR Script*' has been set up by Equinox with default values for all the '*NVR parameters*'. Each customer Z-Wave target board / device will require customised values for the '*NVR parameters*' as these parameters will differ from device to module to target board.

If the Z-Wave device has been erased and the entire '**NVR Area**' has been blanked (set to 0xFF), then it is necessary to modify the '**Restore Script**' to restore / reprogram all the NVR parameters back into the target device. To set up the NVR parameter values, you will need to work out what the values should be.

Instructions:

- Start EQTools
- Select *File Open* and then browse to and select the '*Restore NVR*' script e.g. Sigma_ZW500_RESTORE_NVR_V17-15.ESW
- Select the 'Write to file' tab.....

Write Dat Add the r	a to File Task	s tasks									₹
Fi	nal Read From Fil	e		Final Check	Database Valu	ies 🚽	Flash Che	cksum		NVR Ch	ecksum
Script Task	s Base I	Project	Databas	e Targ	et Connect/Di	connect R	ead Signature	Pre-Erase NVR R	ead	Pre-Eras	e Read From File
heck Databa	ase Values	Chip Erase	Target A	utoProg1	Run Target	Write Data to File	Chip Erase 2	NVR Write	Target	AutoProg3	Final NVR Rea
💌											
Enabled	ta to File Name	Start	DataType	Size (Bytes)	Source	Value	File to upd	Error Messa	ige	•	Add
Enabled	Name CCAL (XT	Start 17	DataType BYTE	Size (Bytes)	Source Database	Value Sigma.CALCUL	File to upd	Error Messa	ige	- Î	Add
Enabled	Name CCAL (XT TXCAL1	Start 17 49	DataType BYTE BYTE	Size (Bytes)	Source Database Database	Value Sigma.CALCUL Sigma.CALCUL	File to upd %NVR_TE %NVR_TE	Error Messa	age	- G	Add
Enabled	Name CCAL (XT TXCAL1 TXCAL2	Start 17 49 50	DataType BYTE BYTE BYTE	Size (Bytes)	Source Database Database Database	Value Sigma.CALCUL Sigma.CALCUL Sigma.CALCUL	File to upd %NVR_TE %NVR_TE %NVR_TE	Error Messa	age		Add

- At the top of this tab, you will see a list of all the available '**NVR parameters**' which can be programmed into the target Z-Wave device.
- It is possible to pre-set each '*NVR parameter*' to a custom value by simply selecting the relevant parameter and then entering the value you want to set the parameter to.



The following parameters must not be touched.....

- The 'CCAL' and 'TXCAL1 / TXCAL2' parameters are automatically calibrated / calculated when the script executes, so these parameters must not be changed by the user.
- The 'Calculate NVR CRC16' parameter is the CRC16 checksum. This parameter is automatically calculated based on the values of all the other parameters in the list. This parameter must not be changed by the user.

Setting the remaining 'NVR parameters' to your customised values....

- All the remaining parameters must be manually set to either the 'Sigma factory default values' or your custom values which are required for your Z-Wave target board.
- To set the *'External Non-Volatile Memory (NVR)'* parameters (*NVMCS, NVMT, NVMS, NVMP*), please refer to the instructions in appendix 3.
- To set the '**PINS**' parameter, you will need to find out what the value should be by reading out this parameter from a working Z-Wave module.
- To set the **'SAWB / SAWC'** parameters, please refer to the original parameter settings or consult the designer of the **'Saw filter'** on your product.
- The '**UUID**' parameter is used to store a unique serial number of MAC address. If you are not using this parameter, then the value can simply be set to 0xFF (x16).
- The '*VID*' and '*PID*' parameters are used to set the USB '*Vendor ID*' and '*Product ID*' respectively of the device. If you plan to use the Sigma default USB values, then you can simply set both fields to 0xFFFF and 0xFFFF respectively. Otherwise, you will need to enter your custom '*VID*' and '*PID*' values.

Once you have set up the values for all the '**NVR parameters**', the table should look something like the example below....

Enabled	Name	Start	DataType	Size (Bytes)	Source	Value	File to update
V 0	CCAL (XTAL Calibration Byte)	17	BYTE	1	Database	Sigma.CALCULATED_CCAL	%NVR_TEMP%
V 1	TXCAL1	49	BYTE	1	Database	Sigma.CALCULATED_TXCAL1	%NVR_TEMP%
V 2	TXCAL2	50	BYTE	1	Database	Sigma.CALCULATED_TXCAL2	%NVR_TEMP%
🔽 3	PINS	18	BYTE	1	Fixed	0x00	%NVR_TEMP%
V 4	NVMCS	19	BYTE	1	Fixed	0x04	%NVR_TEMP%
V 5	SAWC	20	BLOCK	3	Fixed	0×FFFF	%NVR_TEMP%
V 6	SAWB	23	BYTE	1	Fixed	FF	%NVR_TEMP%
7	NVMT	24	BYTE	1	Fixed	0x02	%NVR_TEMP%
V 8	NVMS	25	WORD	2	Fixed	0×0100	%NVR_TEMP%
V 9	NVMP	27	WORD	2	Fixed	0×0100	%NVR_TEMP%
V 10	UUID	32	BLOCK	16	Fixed	0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	%NVR_TEMP%
V 11	VID	45	WORD	2	Fixed	0xFF	%NVR_TEMP%
V 12	PID	47	WORD	2	Fixed	0xFF	%NVR_TEMP%
V 13	Calculate NVR CRC16	126	WORD	2	CRC16	CalculateCRC16	%NVR_TEMP%

- All the parameters must have the '*Enabled*' option ticked.
- All parameters except 'CCAL', 'TXCAL1 / TXCAL2' and 'Calculate NVR CRC16' parameters must have the 'Source' set to 'Fixed' and the correct value defined in the 'Value' field.

To save your modified 'Restore NVR' script file.....

- Select *File Save* to save your changes back to the Script File (*.esw).
- Click the '*Compile*' icon on the EQTools top icon bar → This will compile your changes into the ISP-PRO script file (*.esf).



1.2 Testing the 'Restore NVR' script in ISP-PRO

The '*Restore NVR Script*' can be executed within the ISP-PRO application just like any other script. This script should only be used if the '*NVR Area*' of the target board / device being programmed is known to be blank (contains all 0xFF). If this script is run on a target device which has valid NVR settings, then it will fail immediately.

Instructions:

- Start the ISP-PRO application
- Login and then click the 'Setup' icon
- Select the 'Restore NVR Script' e.g. Sigma_ZW500_RESTORE_NVR_V17-15.ESW
- Start the programming network
- ISP-PRO will prompt you for the 'Project Collection' and 'Standalone Programming Project' to be used
- Start execution of the script by clicking the 'Connect' button or using the selection connection method.
- When the script runs, it first checks that the '*NVR Area*' of the target Z-Wave device is blank (all parameters are 0xFF).
- If the '*NVR Area*' is not blank, then the script will fail immediately. This failure indicates that the target device already has valid 'NVR data'.
- If the *'NVR Area'* is blank (all parameters are 0xFF), then the script will program the 'NVR parameters' with the custom values defined in the *'Write data to* file' tab in the script
- The 'Calibration firmware' will always be programmed and then executed and the CCAL and TXCAL1 / TXCAL2 will be calculated and then stored into the 'NVR Area'.
- The '*Production firmware*' will then be programmed

1.3 Checking the final 'NVR settings' are correct

It is very important to check that the values which are re-programmed into the '**NVR Area**' of a target device are actually correct. Once you have executed the '**Restore NVR Script**', it is recommended that you then read back the '**NVR Area**' using the EQTools – EDS utility and compare the values with a known working sample of the target board.





Appendix 6 - ISP-PRO - Quick Start Guide

1.0 Overview

This section offers an overview / quick-start guide to running a 'Sigma calibration / programming' script with the ISP-PRO application.

ISP-PRO executes '*Programming Scripts*' in order to control a target Equinox Programmer. These scripts are created using the *EQTools – Script Builder* utility and can be tested / debugged using the *EQTools – Script Debugger* utility. Once the scripts have been fully tested using EQTools, they are then ready for executing within the ISP-PRO application.

This section details how to take the files from EQTools and install / execute them within ISP-PRO.

2.0 Installing the Sigma scripts and projects

2.1 Overview

The 'Sigma calibration / programming' scripts and projects are supplied in a single zip file by Equinox.

Instructions:

- Copy the zip file to your PC hard disk
- Unzip the files to a suitable folder on your PC hard disk e.g. *c:\Equinox*

Important note:

The zip file contains all the 'development' source files which were used to make the projects and scripts. These 'development' files are not required for ISP-PRO and so do NOT need to be copied to your production PC.

The only files which ISP-PRO actually requires to run the script in 'production mode' are as follows:

- *.esf Script File(s)
- *.PPC Project Collection File(s)
- *.prj Compiled Project File(s) only the 'Base Project' is required
- *.NEF NVR Parameter file (if NVR parameters are being merged into the NVR)

All other files are only required for maintaining the scripts / projects by the developer or production supervisor and therefore do not have to be copied to the 'production PC'.



2.2 Overview of files required for ISP-PRO software

The illustration below shows which files are required by ISP-PRO and what files ISP-PRO creates when executing a programming script...





2.3 Setting up ISP-PRO to run the Programming Script(s)

In order to execute the 'programming scripts', it is necessary to install them into your ISP-PRO 'Scripts folder' and also to set up ISP-PRO so that it knows which script(s) to execute.

Please follow the steps below before attempting to execute a 'programming script':

2.4 Start the ISP-PRO application and log in

- Select <**Start><Programs><Equinox><ISPPRO> →** ISP-PRO application should start up..
- Click the <Login> button

Configure	Actions Security	Help										
8 Abort	@ Clear Errors	Start Auto	O S <u>h</u> utdown	Project Check	5 etup	Upload Project	Repository	Programmer Info	ADO Explorer	8读 Login	E <u>x</u> it	
									_		_	
									_	22		

• You will now be asked to enter your password.

Supervisor Lo	gin			X
?	Please Ente to Logon to	r the required Sup the System	perviso	r password
<u>P</u> as	sword	•••••		
		ОК		Cancel

- Type in your password (default password is: equinox)
- The **<Setup>** icon should now be selectable.

ISP Pro - C:\t	est\Sigma\Calibration	Sequence\Sigma_	Z-Wave_500.PM	Р	100.001	Magati Serve	e Sheet in	an i programma i	Altern Westmann	the start	(arrise)	
<u>File</u> <u>Configure</u>	e <u>Actions</u> <u>Security</u>	Help										
Abort	☐ Clear Errors	Start Auto	O S <u>h</u> utdown	Project Check	<mark>⊠</mark> <u>S</u> etup	😭 Upload Project	🚑 Repository	Programmer Info	ADO Explorer	8ð Logout	E <u>x</u> it	
												^
					<u>S</u> etup.							1.441

• Click the **<Setup>** icon



--> The 'PPM Setup' screen will now be displayed with the 'Programmer settings' tab selected....

🔜 PPM Setup - C:\test\Sig	gma\CalibrationSequen	ce\Sigma_Z-Wave	e_500.PMP				—
<u>F</u> ile <u>V</u> iew <u>H</u> elp							
👌 🔒 🚔 🕼							
Custom Bitmaps	Database Options	Incremental F	epository	Global Optic	ons	Global Strings	Zebra Printer
Programmer Settings	MUX Options	Zip File	Administr	ator Options	Co	mmunications	Barcode Scanning

2.5 Setting up the correct COM port

If this is the first time that you have used ISP-PRO, then it is likely that the COM port needs to be configured.

Instructions:

- Click the **<Setup>** icon
- Select the 'Communications' tab

Programmer Settings	MUX Options	Zip File	Administrator Options	Communications	Barcode Scanning
Programmer Settings Communication Options Use Port COM1 Timeout (ms) COM2 Baud Rate 38400 Attempts 3 3	MUX Options Refresh Port Li Stop Bits	Zip File	Administrator Options	Lommunications	Barcode Scanning
Scan Interval 333 🐑					

- Click the <Refresh Port List> button
- A list of the available 'COM ports' on your PC should now be displayed.
- Select the COM port which the programmer is attached to
- Click the *<Test>* button
- ISP-PRO will attempt to communicate with the programmer and should display a message to tell you that it has found the attached programmer.





2.6 Detecting the attached programmer(s)

The very first time you run ISP-PRO, it is necessary to detect the attached programmer(s) and also assign **'programmer names'** to each programmer.

Please follow the instructions below...

- Click the <**Setup>** icon
- Select the <Programmer Settings> tab

Programmer Settings	Communications	MUX Options	Zip File	Administrator Options	Barcode Scanning
Attached programmer details Detect Programmer(s). Number of programmers: 1	: 				
Assign a programmer nar Channel Name: ISPnano Series III/N <u>A</u> ddress: 0 💭	me and script to each availa / @address:0	ble programming channe	4		

• If you have not already detected the attached programmer(s), then click the 'Detect Programmer(s)' button.

---> The detection process will provide a list of all attached programmers....in this case it has detected a single programmer at node address 0.

• Each attached programmer is automatically given a 'programmer name' e.g. 'ISPnano Series IIIIV @address:0'.

2.7 Selecting a Script File to run

To select the required 'Script File', follow the instructions below ...

- Click the <Setup> icon
- Select the <**Programmer Settings**> tab

E Unbedded Solutions Company		A	ppli	cation	Note
Programmer Settings	Communications	MUX Options	Zip File	Administrator Options	Barcode Scanning
Attached programmer details Detect Programmer(s) Number of programmers: 1 Assign a programmer nar Channel Name: 	: Use Broadcast Mode me and script to each availa	ble programming channel			
ISPnano Series III/IV <u>A</u> ddress:	/ @address:0				
0 Script File Name: Sigma_2	2W500_Res	tore_NVR	_only_:	121014.ESF	Browse

Script File Directory:

C:\test\Sigma\CalibrationSequence2

- If you wish to select a *Script File (*.esf)* for a single channel, click the *<Browse>* button.
- Browse to and select the required 'Script File' which will have the file extension *.esf. The 'Script File' should be located in your 'Scripts' directory / folder.
- If you selected 'Script File' is not in the 'Script File directory' then you will receive the following warning:

Warnin	ng 🔀
⚠	Warning this file is not in the current script directory.Would you like to update your script path now?.
	Yes No

- If you click <**Yes>** then ISP-PRO will automatically set the '**Script File directory**' to the directory where your selected script is located.
- Once you've selected your 'Script File', then you just need to save your settings by selecting 'File - Save As...' and then specifying a file name. It's a good idea to put this 'ISP-PRO Settings' file in your 'Script File Directory'.

ISP IRC P	PPM Setup - C:\test\Sigma\CalibrationSequence\Sigma_Z-Wave_500.PMP									
<u>F</u> ile) <u>V</u> iew <u>H</u> elp									
2	Open	4								
	Save		Database Options	Incremental F	Pepository	Global Optic	ons	Global Strings	Zebra Printer	
	Save As	ngs	MUX Options	Zip File	Administra	tor Options	Cor	mmunications	Barcode Scanning	
	Close	ier detai	ls:							

• You should then see a 'Confirm' screen similar to this one.....





 You will then be returned to the main ISP-PRO screen where you should see that your selected 'Script File' is now displayed.

ISP Pro - C:\test\Sigma\Sigma_Z-V	Vave_500.PMP	
<u>File</u> Configure <u>Actions</u> Security	Help	-
🛛 🙆 Abort Clear Errors 🙆 Start	Auto 💿 Shutdown 🔂 Project Check 🥂 Setup 🙎 Upload Project 😂 Repository 🐠 Programmer Info 🎭 Logout 🔯 Run Database Explorer 🤑 Exit 🚬	
Parameter Drogrammor Namo	Value ISDaano Sorior III /IV @addrocc:0	
Script Name	Sigma ZW500 Restore NVR only 121014 ESE	111
		WAIT
		Database ID
		Script Time: 00:00
Channel Status	Messages	ID
1 IN PROGRESS	Programmer is Waiting	
	विषे -	
		eport
	A A DR-C000005	
senai numper=000000000000000000000000000000000000	AMBBCC VUVVVJ Supervisor Mode Auto Programming Stopped!	

• ISP-PRO is still in 'Supervisor Mode' allowing you to change any other settings before going into 'Production mode'.

2.8 Uploading your Project Collection to the programmer(s)

Before running any 'Script File' which uses 'Standalone programming projects', it is necessary to upload these projects to the programmer 'FLASH Memory Store'.

To upload a 'Project Collection' to the programmer, follow the instructions below ...

• Click the <Upload Project> button on the ISP-PRO icon bar.



ISP Pro - C:\t	est\Sigma\Calibratior	Sequence\Sigma_	Z-Wave_500.PM	P	100.002	Margari Sarra	ny liferana in	na i poperna -	other Western	iter; there	(arrise)	
<u>File</u> <u>C</u> onfigure	<u>Actions</u> <u>Security</u>	Help										
8 Abort	☐ Clear Errors	Start Auto	O Shutdown	Project Check	<mark>⊠</mark> <u>S</u> etup	Lipload Project	nepository	Programmer Info	ADO Explorer	8ठू Logout	E git	
												<u>^</u>
						余						
1.1					Lie	load Pro	iect					1 - 1

• If you have already selected the script file to execute, then ISP-PRO should automatically notify you of which *Project Collection (*.ppc)* file to upload.

Informat	ion	
	Operation: Upload Prog	ramming Projects which are referenced in the Script File
	The Script File which yo	nu have selected requires that new Programming Project(s) contained in the specified Project Collection are uploaded.
	Programmer: Script File Name:	ISPnano Series III/IV @address:0 Siama ZW500 Restore NVR only 121014.ESF
	Project Collection:	C:\test\Sigma\CalibrationSequence2\Sigma_Z-Wave500_project_collection.PPC
	Press <ok> to launch t</ok>	he Upload Wizard and then follow the on-screen instructions to upload the specified Project Collection.
		ОК

• If ISP-PRO detects the attached programmer(s), then the 'Upload Wizard' utility will be automatically started and the following screen will be displayed....

🏦 Equinox Upload Wizard	×
Upload Project(s) to programmer Select <upload> to upload your selected project(s) to the attached programmer(s)</upload>	}
Click 'Upload and Verify' to upload selected project(s) to seleted programmer(s). Page:	
<u>I</u> ransfer Baud Rate 230400	fy
< Back Next > Ca	incel

- Click the <Upload and Verify> button to upload the Project Collection (*.ppc) to the attached programmers.
- Once the upload of the projects is complete, the 'Upload Wizard' will display a list of the uploaded projects....



^	🟦 Equinox Upload Wizard 💽 💌											
	010											
P	Programmer Preview (4 projects found)											
	No.	Name	Version	Build Date	Author							
	0 1 2	SD3502-SPI CALIBRATION RUNTARGET	1.0.0.0 1.0.0.0 1.0.0.0	02/01/2014 19:03 02/01/2014 19:04 02/01/2014 19:05	John							
	5	PRODUCTION	1.0.0.0	00/01/2014 23:11	John							
	•				•							
			< <u>B</u> ack	<u>N</u> ext >	Cancel							

- Click the *<Next>* button on this screen and then the *<Finish>* button on the next screen to complete the project upload process.
- You will then be taken back to the main ISP-PRO screen.
- ISP-PRO is still in 'Supervisor mode' with the programming network stopped.



2.9 Running the programming Script(s)

To execute your selected *Programming Script(s)*, please follow the steps details below:

1. Click the **<Start Auto>** icon on the ISP-PRO Icon Bar

	0		and a							
Tions II Start A	uto Shutdov	n Project Check	<u>S</u> etup	Upload Project	nepository	• Programmer Info	ADO Explorer	8b Logout	📮 Egit	
6										
	6									

 \rightarrow All enabled programming channels should now go to the *Connect* state

👷 ISP Pro - C:\test\Sigma\Sigma_Z-V	/ave_500.PMP			_ 0 🔀
<u>File Configure Actions</u> Security	Help			•
🙆 Abort 🥔 Clear Errors 🕜 Start	Auto 🕥 Shutdown 🗟 Project Check 🖾 Setup 👔 Upload Project 😂 Repo	sitory 🕥 Program	mer Info 🎭 Logout 혽 Run Database Explorer	🕼 Exit 🖕
SIGMA SD3502 - PR	OGRAMMING / CALIBRATION SEQUENCE			
Press the 'Connect'	outton to start the sequence			
	•			
Parameter	Value			
Programmer Name	ISPnano Series III/IV @address:0			CONNECT
Script Name	Sigma_ZW500_Restore_NVR_only_121014.ESF			
Selected Projects	Calibration Firmware=CALIBRATION_1.0.2.0			
	Run Calibration Firmware=RUNTARGET_1.0.2.0			
				Database ID
				Script Time: 00:00
Channel Status	Messages			ID
1 IN PROGRESS	Waiting to Connect			
				Report OCOnnect
Serial number=000000000000000000000000000000000000	AARBCC000805			
0:SYSTEM_STATUS PASS - CommsErrS	tatus = NONE, SystemErrStatus = NONE, SystemStatus = Programmer is waiting		Supervisor Mode Now Auto-programming	

The message '*Now Autoprogramming*' should be displayed at the bottom right-hand corner of the ISP-PRO window.

Supervisor Mode Now Auto-programming...



2.10 Executing the programming / calibration sequence

To execute the programming / calibration sequence on a Target System (DUT)....

- 1. Connect a Target System (DUT) to the relevant programming channel
- 2. Click the *Connect>* button on the bottom-right of the channel icon to commence the programming operation on the selected channel.

SIGMA SD3502 - TX (SIGMA SD3502 - TX CALIBRATION SEQUENCE									
Connect the Targe	et Board (DUT) to the programmer									
Press the Connect button	n to start the sequence									
Parameter	Value	CONNECT								
Programmer Name	ISPnano Series III/IV @address:0	COMMECT								
Database ID	Sigma_ZwSUU_TX_Cal_Z.ESF									
	00:00									
Status: Waiting to Conne	ect	Connect								

2. The script will then start to execute....

Now programming	g Calibration Firmware into DUT	
Please wait		
Parameter	Value	
Programmer Name	ISPnano Series III/IV @address:0	AUTU
Script Name	Sigma_ZW500_TX_Cal_2.ESF	PROGRAM
Script Time	00:02	
AutoProgram 1	CALIBRATION	
Status: Applying power	to Line Drivers / Target System	

--> The icon will display 'Auto Program'

--> The 'Script Timer' will now start timing the execution of the script.

Important note:

- You can abort the execution of the script at any time by pressing the 'Abort' button.
- However, this will leave the 'NVR area' blank so the Target IC is now scrap !!!

AN145- In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices



2.11 Programming sequence - PASS

If the programming sequence is successful (executes without any errors), then the following screen will be displayed.....

SIGMA SD3502 - TX	CALIBRATION SEQUENCE	
Disconnect the Ta	rget Board (DUT) from the programmer	
Press the Disconnect but	ton to reset the sequence	
Parameter	Value	PASS
Programmer Name Script Name	ISPnano Series III/IV @address:0 Sigma_ZW500_TX_Cal_2.ESF	
Database ID Script Time	12 00:14	
AutoProgram 3	PRODUCTION	
Status: Programmer is V	Vaiting	Disconnect

- ISP-PRO will display 'PASS Disconnect now'
- The Target Board (DUT) can now be disconnected from the programmer.
- Press the 'Disconnect' button to reset the sequence ready for the next Target Board (DUT)



2.12 Programming sequence - FAIL

If the programming or calibration fails for any reason, then the following screen will be displayed.....

SIGMA SD3502 - TX CALIBRATION SEQUENCE										
The programming	The programming / calibration sequence has failed.									
Press the Disconnect button to reset the sequence										
Parameter	Value									
Programmer Name	ISPnano Series III/IV @address:0									
Script Name	Sigma_ZW500_TX_Cal_2.ESF									
Database ID	11									
Script Time	00:01	DISCONNECT								
		NOW								
Status: Programmer is W	aiting									
Verify Error - read NVR V	ER = 0xFF expected = 0x01									
Verify Error - read NVR C	RC16 = 0xFFFF expected <> 0xFFFF									

- ISP-PRO will display 'PASS Disconnect now'
- The Target Board (DUT) can now be disconnected from the programmer.
- Press the 'Disconnect' button to reset the sequence ready for the next Target Board (DUT)

Important note:

If the programming fails for any reason, the 'NVR Area' will be left blank (all 0xFF).

This means that the Target IC / Module can never be re-programmed again because the **'Factory NVR calibration data'** has now been permanently erased.



3.0 Setting up an Incremental Serial Number

3.1 Overview

It is possible to setup an *'Incremental Serial Number'* within ISP-PRO to generate any format of serial number, RFID, MAC address etc. There is a special utility within ISP-PRO called the *'Incremental Repository'* which is used to generate and store any form of 'serial number'. The generated 'Serial Number' can then be programmed into the FLASH or 'NVR Area' of a target device.

The 'Sigma calibration / programming' scripts and projects are supplied in a single zip file by Equinox.

3.2 Importing a Serial Number from file

In many instances, a 'Serial number' or 'MAC address' may have already been setup for a particular application. It is possible to import this 'Serial number' from a file directly into the 'Incremental Repository' - see instructions below.

Instructions:

- Start ISP-PRO
- Login
- On the top icon bar, click the 'Repository' button

ISP Pro - C:\test\Sigma\Sigma_Z-Wave_500.PMP	
<u>File</u> <u>C</u> onfigure <u>A</u> ctions <u>S</u> ecurity <u>H</u> elp	
🔇 Abort 🥔 Clear Errors 🕼 Start Auto 🔘 Shutdown 🗟 Project Check 🔯 Setup 🤮 Upload Project 😂 Repository 🕥 Programmer Info 🕸 Logout 🕵 Run Database Explorer 👔	E <u>x</u> it 🖕

The 'Incremental Repository' screen will then be displayed

In	crementa	al Repository								X
	🔘 Get 🛛	ata from Registry (Get Data from Databas	E						
	No.	Description	Current Value	Increment	Min	Max	Format	Increment at End	Preview	Add
										Remove
										Edit
										Clear
										Import
										Export

- Click the 'Import' button on the right-hand side
- Browse to and select the required 'Serial number file' eg. UUID.ERE
- This should import the 'Serial number' value, increment, format etc into the 'Incremental repository'.
- In this example, the 'Serial number' is called 'INC1' and is a 16-byte number which is written into the 'UUID' NVR field.



• The 'Current value' is the value which will be programmed next which is represented in the actual display format in the 'Preview' column.....

I	ncremental Repository									
	Get Data from Registry (@) Get Data from Database									
		ata nomregioa]	
	No.	Description	Current Value	Increment	Min	Max	Format	Increment at End	Preview	
	INC1	UUID	2009	1	0	16777215	00000000	NO	0000000000000000000000ABBCC0007D9	
11										

- Once you have setup the *'Incremental repository'* then this saves *'INC1'* to the database so the value of this *'Serial number'* will always be remembered.
- Click 'Close' to save the 'Incremental repository' settings.





Appendix 7 - FLASH File - CRC32 Checksum

1.0 Overview

The Z-Wave 500 series devices feature a 32-bit (4 byte) FLASH checksum *(CRC32)* which is used by both the Equinox programmer and the Z-Wave device itself at run-time to validate (verify) that the FLASH contents have been programmed correctly are not corrupt. The *'CRC32 Checksum'* is calculated over the entire FLASH address range except for the last 4 bytes. The calculated *'CRC32 Checksum'* is stored in the top 4 bytes of the FLASH.

The table below shows the various components which make up the *'Final FLASH File'* which is used to program the FLASH area of the Z-Wave device....



Important notes:

1. The Equinox EQTools configuration software expects that a valid 'CRC32 Checksum' is pre-stored in any input BINARY or HEX file for a Z-Wave 500 series device. If a valid 'CRC32 Checksum' is not detected, then EQTools will display an error message 'Invalid CRC32 Checksum'.

2. It is recommended that you update the '*Tx Power Parameters*' before you calculate the '*CRC32 Checksum*' because the checksum will need to be changes if you update the '*Tx Power Parameters*'.



1.1 Using EQTools - EDS to update the CRC32 checksum

The **EQTools - EDS** utility can be used to update your existing 'Firmware FLASH file' with the correct 'CRC32 Checksum' and 'Tx Power Parameters'. EDS takes the 'Original FLASH file' and the 'Tx Power parameters' as input, calculates the 'CRC32 checksum' of the entire FLASH image and then outputs the 'Final FLASH File' - see illustration below.



1.2 Why do I get an invalid FLASH CRC32 checksum error?

When a BINARY or HEX file is loaded into the '*FLASH buffer*' of the Equinox EQTools software, the file loader automatically checks to make sure that a valid '*CRC32 Checksum*' for the data in the file is stored in the last 4 bytes of the file data image.

The screenshot below shows a valid 'CRC32 Checksum' stored in last 4 bytes of the 'FLASH file' (address range: 0x1FFFC - 0x1FFFF).....





If the input file does not contain a valid 'CRC32 Checksum' then the following error message will be displayed.....

This error message simply means that the input BINARY or HEX file either has no **'CRC32 Checksum'** setup in it as the CRC32 checksum of the file is 0xFFFFFFFF.

1.3 Loading standard Z-Wave Firmware releases from Sigma

If you are trying to load a standard Z-Wave '*firmware file*' released by Sigma Designs for a Z-Wave 500 series device, then none of these 'standard' hex files contain a valid '*CRC32 Checksum*'. It is therefore necessary to use the Equinox *EQTools - EDS* development software to resave the hex file with the correct '*CRC32 Checksum*'.

1.4 Loading a custom Z-Wave Firmware release

If you have created your own BINARY or HEX file to program into the FLASH area of a Z-Wave 500 series device using a third party compiler, then your files will definitely NOT contain the 'CRC32 checksum' as part of the file image. It is therefore necessary to use the Equinox EQTools - EDS development software to resave the hex file with the correct 'CRC32 Checksum'.

1.5 How to correct / add a valid CRC32 checksum to an input file

If your input **'FLASH file'** does not have a valid **'CRC32 Checksum'**, then the file cannot be used for programming within EQTools. It is necessary to generate a **'CRC32 Checksum'** for the file and then re-save the file with this checksum before the file can be used with EQTools to program a 500 series Z-Wave device.

Please follow the instructions starting in section 1.6 to create a hex file with a valid embedded *'CRC32 Checksum'*.



1.6 Opening the project in EDS (Development mode)

A simple way to open the project in EDS (Development mode) is as follows:

- Open the Project Collection
- Click the project you want to test in EDS once so that it is highlighted
- Now click the 'Check project in EDS mode' button

Project	t Manag	ger - C:\test\	.Sigma\Ca	librationSe	equence\Sigm	a_SD3502_project_col	lection.Pl	РС		
Details	Files	Power Supp	ly Fuses	Security	State Machine	Programming Speed	Retries	Memory Map		
Number	Uniq	ue Id	Version	Build Date		Target Device	Target	Programming Interfa	ce Sigr	nature
0	SD3	502-SPI	1.0.0.0	02/01/201	4 at 19:03	SD3502 (SPI)	SPI 3-v	vire + RESET_N (Z-W	ave) 0x7	F7F7F7F1F0401
1	CALI	IBRATION	1.0.0.0	02/01/201	4 at 19:04	SD3502 (SPI)	SPI 3-v	vire + RESET_N (Z-W	ave) 0x7	F7F7F7F1F0401
6	RUN	TARGET	1.0.0.0	02/01/201	4 at 19:05	SD3502 (SPT)	SPI 3-v	vire + RESET N (7-W	ave) 0x7	F7F7F7F1F0401
0	PRO	DUCTION	1.0.0.0	13/01/201	4 at 21:15	SD3502 (SPI)	SPI 3-v	vire + RESET_N (Z-W	ave) 0x7	F7F7F7F1F0401
•										Þ
	🕼 Upload selected project									
ogramme	grammer Type: ISPnano Series III/IV Total FLASH usage - Used: 141.0 KB (0.9%), Free: 15.9 MB bytes (99.1%) Programmer Flash Page Size = 2									r Flash Page Size = 2

The selected project should then launch in 'EDS - Development mode'....

Image: Equinox Development Suite (EDS) C:\test\Sigma\CalibrationSec	quence\Test_PRODUCTION.EDS	- • •
Overview Programmer Target Device SPI Settings Target Power Su	pply E <u>r</u> ase Flash NVR Security	SRAM
Flash File	Updated:	Edit Buffer
C:\\ZM5202_calibration_en_P37_ref_P23.hex	22/12/2013 21:11:32 📄 Reload	File Open
✓ Erase buffer before file load ✓ Skip all leading and tr	Save as	
Automatically reload into buffer on change Automatically upload	to target on change	📎 Erase

1.7 Calculating the CRC32 checksum of the input file

The 'CRC32 checksum' of the input file can be calculated as follows...

- Select the 'FLASH' tab
- Your selected input file should be displayed in the buffer area
- Use the scroll bar on the right-hand side of the EDS window to scroll down to the end of the target device FLASH area.



🛲 Equinox Development Suite (EDS) C:\test\Sigma\CalibrationSequence\Test_PRODUCTION.EDS			
Overview Programmer Target Device SPI Settings Target Power Supply Erase Flash NVR Security SRAM			
C:\\ZM5202_calibration_en_P37_ref_P23.hex 22/12/2013 21:11:32 2 Reload Reload			
Grand buffer before file land Image of the file and the file of the f			
Save as			
Automatically reload into buffer on change Automatically upload to target on change			
⇒ <u>F</u> rase			
0x1FE70: FF			
Ux1FEBU: FF			
Owifeso, FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF			
Ovi TECO: FE			
0x1FED0: FF			
0x1FEE0: FF			
0x1FEF0: FF			
0x1FF00: FF			
0x1FF10: FF			
0x1FF20: FF			
OxIFF30: FF			
Ux1FF4U: FF			
0x1FF50; FF			
$0 \neq 1 \neq 2 = 1 \neq 2 \neq$			
0x1FF80: FF yyyyyyyy			
0x1FF90: FF			
0x1FFA0: FF			
0x1FFB0: FF			
0x1FFC0: FF			
0x1FFD0: FF			
UXIFFEU: FF			
Size = 131072 (0x20000) CRC : 0x85BC696B 0 (0x0000) Last Non FF (0x21F1)			

- As you can see, the last 4 bytes of FLASH are set to 0xFFFFFFF. This means that the 'CRC32 checksum' is invalid.
- Now click the 'Calc CRC' button
- EDS will now calculate the 'CRC32 checksum' for the file loaded into the buffer area.



- Click the 'Yes' button to update the 'Calculated CRC' value into the last 4 bytes of the 'FLASH buffer'
- If you look at the last 4 bytes of the 'FLASH buffer', they have now been updated with the 'Calculated CRC checksum value'....

0x1FFE0: FF1 0x1FFF0: FF1	FF FF FE FF FF FE	FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	F FF FF FF FF F FF FF FF FF	6B 69 BC 85	yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyy <mark>ki%</mark> ∎
Size = 131072 (0x2	0000)	CRC: 0x85BC6	596B 0 (0x0000)	L	ast Non FF (0x21F1)

AN145- In-System Programming (ISP) of Sigma Z-Wave 500 series modules and SOC devices



1.8 Saving the revised file with the CRC32 checksum

You now just need to save the modified 'FLASH buffer' back to your original hex file.

- Click the 'Save as' button
- The 'Save FLASH buffer' window is now displayed.....

Save Flash Buffer	X	
Use file start and end a	ddresses	
Start Address		
0x000000	Entire Device	
<u>N</u> o of bytes	Remove preceeding '0xFF'	
131072	(
End Address	Remove trailing '0xFF'	
0x01FFFF		
<u>O</u> ffset		
0x000000		
	<u>O</u> K <u>C</u> ancel	

- Click the '*Entire device*' button --> This selects the entire address range of the device which includes the '*CRC32 checksum*' value stored in the last 4 bytes of the FLASH.
- Click 'OK' and then browse to the folder where you want to save the file.

	File name:	Test_CRC32_File.hex		•
Save as type: Flash Files (*.HEX, *.A90, *.BIN, *.ROM, *.EEP)		Flash Files (*.HEX, *.A90, *.BIN	N, *.ROM, *.EEP)	•
ら Hid	e Folders		Save	Cancel

- Click the 'Save' button to save the file.
- The contents of the 'FLASH buffer' including the now valid 'CRC32 checksum' is saved to the specified file name. You may want to change the file name to append 'CRC32' on the end.
- This file can now be loaded back into the 'PRODUCTION' project as it has a valid 'CRC32 checksum'.



Appendix 8 - Sigma SD3502 Evaluation Module

1.0 Overview

This section describes how to interface an Equinox ISPnano programmer to a 'Sigma SD3502 *Evaluation Module'* via the 'UART' programming interface.

1.1 Equipment required

The following equipment is required for the programming evaluation.....

- ISPnano Series IV programmer
- IOMOD6 I/O Connector Module
- Sigma SD3502 Evaluation Module (available to order from Sigma Designs)
- 10-way IDC ribbon cable



1.2 Connecting the programmer to the Sigma eval module

This section describes how to interface an Equinox ISPnano programmer to a 'Sigma SD3502

Instructions:

- Insert the 'IOMOD6' I/O Connector Module into the 'ISPnano Series IV' programmer
- Plug one end of the 10-way IDC cable into the 'Equinox header' 10-way IDC connector on the 'IOMOD6' module.
- Plug the other end of the 10-way IDC cable into the 10-way IDC connector on the bottom (base) PCB of the 'Sigma SD3502 Evaluation Module' - see illustration below





Appendix 9 – Sigma connector definitions

1.0 Sigma - ISP Header Selection

#	ISP Header	Description / Function	ISP Header Pin-out
1	J6	Equinox 10-way Header(a) Device support: Sigma Z-Wave 500 series devices via the 'SPI' interface.	PROG_VCC 1 2 PROG_SPARE PROG_SPARE 3 4 PROG_MOSI N/C 5 6 PROG_MISO PROG_GND 7 8 PROG_SCK1 PROG_GND 9 10 PROG_RESET_N


Application Note



Appendix 10 – Connecting the Z-Wave target board to an ISPnano programmer

1.1 IOMOD10 module - connections to Z-Wave target board (SPI interface)

The diagram below shows the typical connections between the Equinox IOMOD10 connector module and a Z-Wave target board using the SPI programming interface.



Number	Description
1	IOMOD10 calibration / connector module
2	10-way ISP cable
3	Customer Target Board featuring Z-Wave module or IC
4	Z-wave module or IC
5	External power supply (used to power the Target Board) if the programmer is not supplying power)
6	Fixture main GROUND connection. The Target Board, programmer and External Power Supply should all be star connected to this common GROUND point.

Important notes...

1. It is recommended that separate thick wires are used for the "Power 0V" and "Power VCC" between the programmer and the target board.





1.2 IOMOD7 module - connections to Z-Wave target board (SPI interface)



Number	Description
1	IOMOD7 connector module
2	10-way ISP cable
3	Customer Target Board featuring Z-Wave module or IC
4	Z-wave module or IC
5	External power supply (used to power the Target Board) if the programmer is not supplying power)
6	Fixture main GROUND connection. The Target Board, programmer and External Power Supply should all be star connected to this common GROUND point.

Important notes...

1. It is recommended that separate thick wires are used for the "Power 0V" and "Power VCC" between the programmer and the target board.



1.3 IOMOD4 connector module - connections to Z-Wave target board (SPI interface)

The illustration below shows how to connect a 'Z-Wave module / target board' to an ISPnano programmer using an IOMOD4 connector module.



Number	Description
1	IOMOD4 connector module
2	Customer Target Board featuring Z-Wave module or IC
3	Z-wave module or IC
4	External power supply (used to power the Target Board) if the programmer is not supplying power)
5	Fixture main GROUND connection. The Target Board, programmer and External Power Supply should all be star connected to this common GROUND point.

Important notes...

1. It is recommended that separate thick wires are used for the "Power 0V" and "Power VCC" between the programmer and the target board.



1.4 IOMOD4 connector module + Clock Buffer - connections to Z-Wave target board (SPI interface)

The illustration below shows how to connect a 'Z-Wave module / target board' to an ISPnano programmer using an IOMOD4 connector module and also a remote 'Clock Buffer'. The 'Clock Buffer' is used to buffer the SPI SCK signal at the target end of the wire to ensure good signal integrity.



Number	Description
1	IOMOD4 connector module
2	10-way ISP cable
3	Clock Buffer
4	Customer Target Board featuring Z-Wave module or IC
5	Z-wave module or IC
6	External power supply (used to power the Target Board) if the programmer is not supplying power)
7	Fixture main GROUND connection. The Target Board, programmer and External Power Supply should all be star connected to this common GROUND point.

Important notes...

1. It is recommended that separate thick wires are used for the "Power 0V" and "Power VCC" between the programmer and the target board.