GreenPAK / RH850

GreenPAK configuration and programing with RH850

R00ZZ0017EE0100
Rev.1.00
October 11, 2023

Introduction

GreenPAK™ is a broad family of cost-effective non-volatile memory (NVM) programmable devices. The RH850 is a family of 32-bit automotive microcontrollers. Together they enable innovators to integrate many system functions into a single custom circuit and change it on PCB board and allows to reconfigure it in the application by the MCU. This document describes the mechanism of configuring, programing and reprograming GreenPAK™ device on PCB board via I2C communication, implemented in most of the RH850 device family.

Target Device

All RH850 devices which include I2C Bus Interface (RIIC).

All GreenPAK™ programmable devices.

Used equipment:

- GreenPAK™ Advanced Development Board 1, rev.1.3.1;
- GreenPAK™ TSSOP-20 #2;
- GreenPAK™ device SLG46827-A;
- GO Configure Software Hub;
- Main board RH850-X2X-MB-T1-V1;
- Piggyback board RH850-U2A-373PIN-PB-T1-V1;
- Renesas E2 emulator;
- GHS Multi

Contents

1. GreenPAK™ SLG46827-A

For demonstration purposes this application note uses GreenPAK™ SLG46827-A auto AEC-Q100 Qualified which provides a small, low power component for commonly used Mixed-Signal functions. The user creates the circuit design by programming the multiple time Non-Volatile Memory (NVM) to configure the interconnect logic, the IOs, and the macrocells. Dual power supply allows to flexibly interface two independent voltage domains.

1.1 Overview

1.1.1 Block diagram of SLG46827-A

Other block diagram of GreenPAK™ devices may look different but also provide possibilities to reprogram it via the I²C interface.

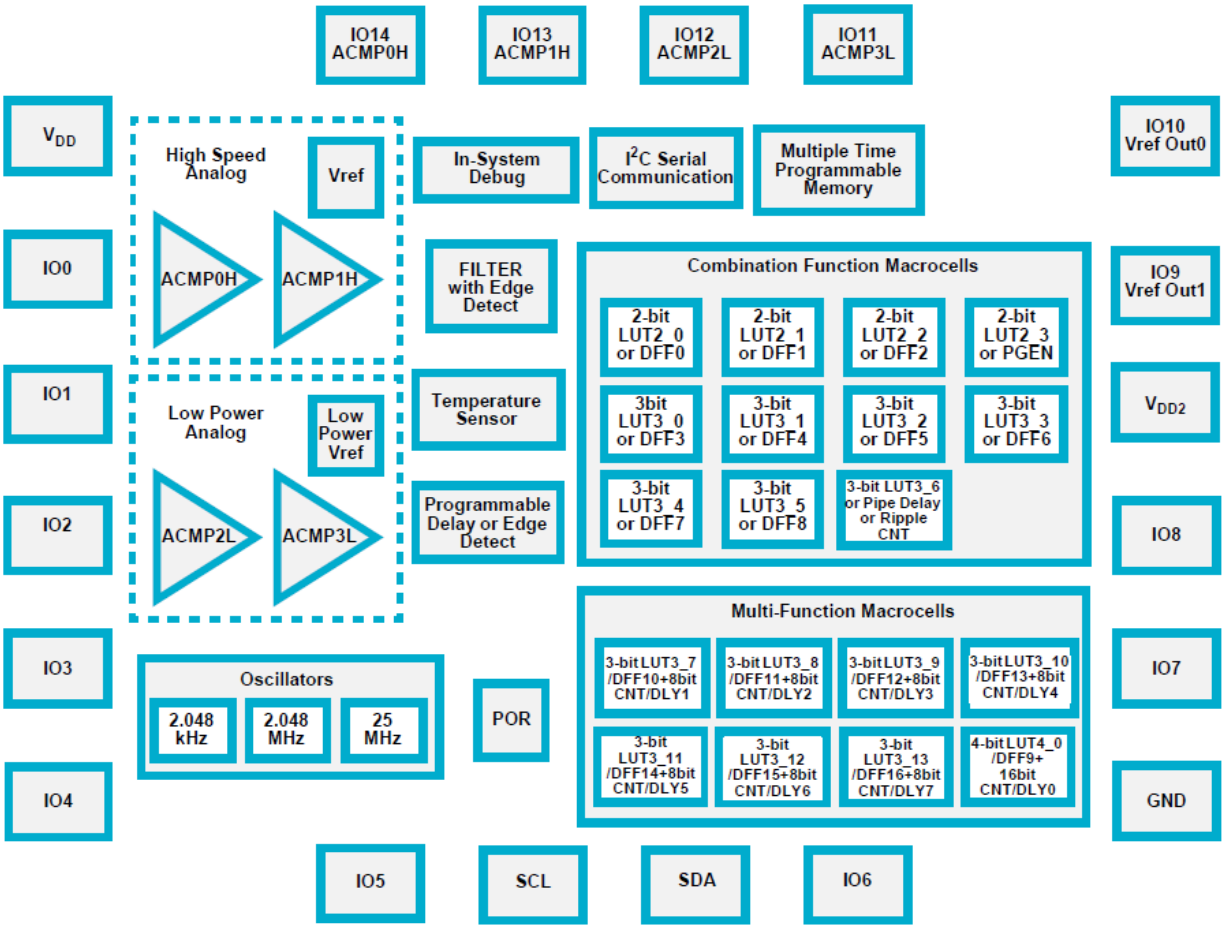

Figure 1: Block diagram of SLG46827-A

1.1.2 Memory

The SLG The SLG46827-A provides 2,048 bits of Non-volatile Serial Electrically Erasable Configuration Register memory that is used for device configuration and 2,048 bits of volatile RAM register memory. In the standard use case for the GreenPAK devices, the configuration choices made by the user are stored as bit settings in the Non-Volatile Memory (NVM), and this information is transferred at startup time to volatile RAM registers that enable the configuration of the macrocells.

The SLG46827-A utilizes a scheme that allows a portion or the entire Register and NVM to be inhibited from being read or written/erased. There are two bytes that define the register and NVM access or change. The first byte RPR defines the 2k RAM register read and write protection. The second byte NPR

defines the 2k NVM data configuration read and write protection. If desired, the protection lock bit (PRL) can be set so that protection may no longer be modified, thereby making the current protection scheme permanent.

1.2 I²C serial communication microcell overview.

The I²C Serial Communications Macrocell in this device allows an I2C bus Master to read and write information via a serial channel directly to the RAM registers and NVM memory, allowing the remote re-configuration of macrocells, and remote changes to signal chains within the device. An I2C bus Master is also able to read and write other register bits that are not associated with NVM memory. As an example, the input lines to the Connection Matrix can be read as digital register bits. These are the signal outputs of each of the macrocells in the device, giving an I2C bus Master the capability to remotely read the current value of any microcell and change it if needed. This gives developers the opportunity to change parameters of macrocells or read pin levels during execution of the program.

1.3 I²C serial communication device addressing.

Each command to the I2C Serial Communications macrocell begins with a Control Byte. The bits inside this Control Byte are shown in Figure 2. After the Start bit, the first four bits are a control code. Each bit in a control code can be sourced independently from the register or by value defined externally by IO5, IO4, IO3, and IO2. The LSB of the control code is defined by the value of IO2, while the MSB is defined by the value of IO5. The default control code is '0001'b. The Block Address is the next three bits (A10, A9, A8), which will define the most significant bits in the addressing of the data to be read or written by the command. The last bit in the Control Byte is the R/W bit, which selects whether a read command or write command is requested, with a "1" selecting for a Read command, and a "0" selecting for a Write command. This Control Byte will be followed by an Acknowledge bit (ACK), which is sent by this device to indicate successful communication of the Control Byte data.

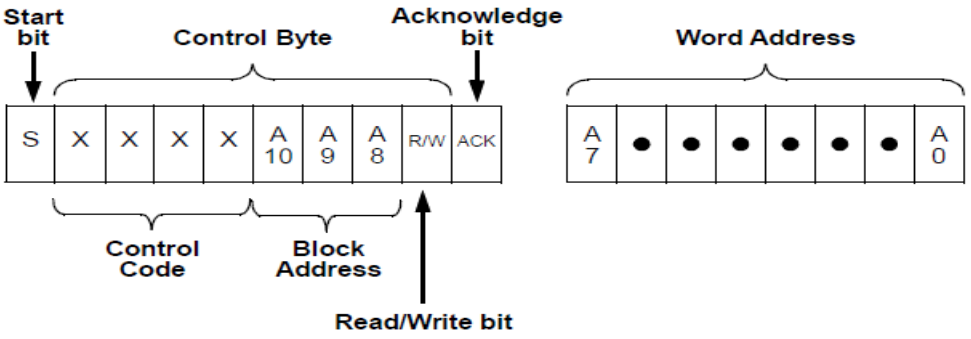

Figure 2: GreenPAK™ device addressing

Control code by default is '0001'b and can be changed from "Properties" of the I²C Serial Communications Macrocell in "Go configure software hub" to be fixed or by value defined externally by IO5, IO4, IO3, and IO2.

Control code could be considered like GreenPAK device "address", together with Block address user can address different space either NVM memory or RAM registers. Block addressing codes are shown in Table 1.

Table 1: Block address

I²C Block address			Memory space
A10 = 0	A9 = 0	A8 = x	2 Kbits Register Data Configuration (RAM)
A10 = 0	A9 = 1	A8 = 0	2 Kbits NVM Data Configuration
A10 = 0	A9 = 1	A8 = 1	Not used
A10 = 1	A9 = x	A8 = x	Not used

1.4 I²C serial communication commands.

1.4.1 Byte Write command.

Following the Start condition from the Master, the Control Code [4 bits], the Block Address [3 bits], and the R/W bit (set to “0”) are placed onto the I2C bus by the Master. After the SLG46827-A sends an Acknowledge bit (ACK), the next byte transmitted by the Master is the Word Address. The Block Address (A10, A9, A8), combined with the Word Address (A7 through A0), together set the internal address pointer in the SLG46827-A, where the data byte is to be written. After the SLG46827-A sends another Acknowledge bit, the Master will transmit the data byte to be written into the addressed memory location. The SLG46827-A again provides an Acknowledge bit and then the Master generates a Stop condition. The internal write cycle for the data will take place at the time that the SLG46827-A generates the Acknowledge bit.

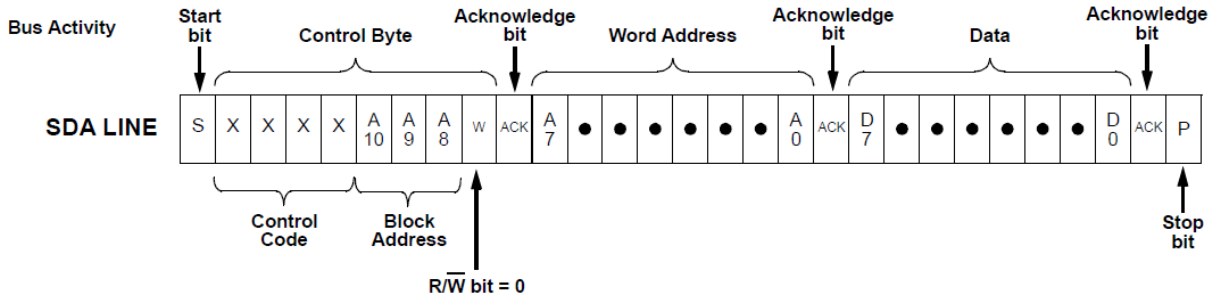

Figure 3: Byte write command

1.4.2 Sequential Write command

The write Control Byte, Word Address, and the first data byte are transmitted to the SLG46827-A in the same way as in a Byte Write command. However, instead of generating a Stop condition, the Bus Master continues to transmit data bytes to the SLG46827-A. Each subsequent data byte will increment the internal address counter and will be written into the next higher byte in the command addressing. As in the case of the Byte Write command, the internal write cycle will take place at the time that the SLG46827-A generates the Acknowledge bit.

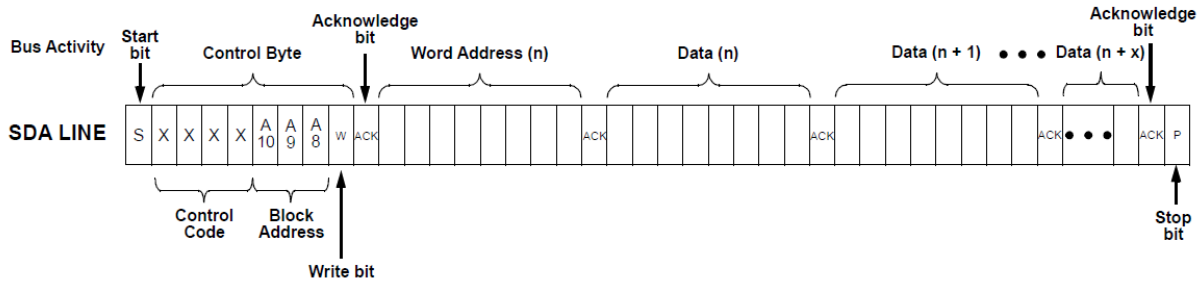

Figure 4: Sequential write command

1.4.3 Random Read command.

The Random Read command starts with a Control Byte (with R/W bit set to “0”, indicating a write command) and Word Address to set the internal byte address, ***followed by a Start bit***, and then the Control Byte for the read (the same as the Byte Write command). The Start bit in the middle of the command will halt the decoding of a Write command but will set the internal address counter in preparation for the second half of the command. After the Start bit, the Bus Master issues a second

control byte with the R/W bit set to “1”, after which the SLG46827-A issues an Acknowledge bit, followed by the requested eight data bits.

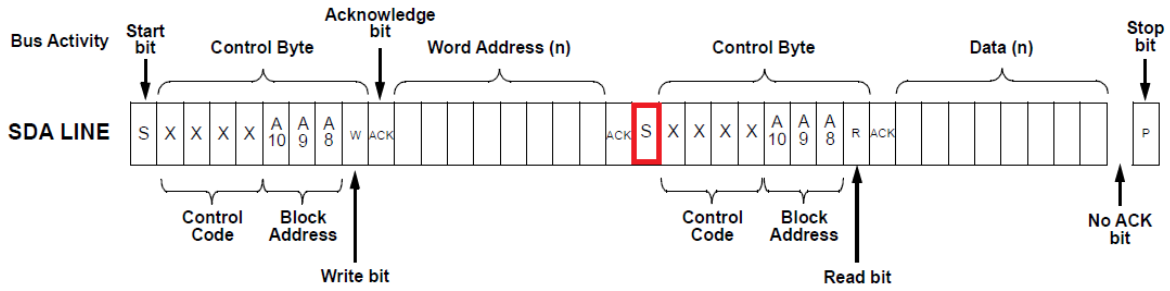

Figure 5: Random Read command

1.4.4 Sequential Read command.

The Sequential Read command is initiated in the same way as a Random Read command, except that no Word address is sent, SLG46827-A will start read where internal address counter pointing, point one address above last I²C command. Once the SLG46827-A transmits the first data byte, the Bus Master issues an Acknowledge bit as opposed to a Stop condition in a random read. The Bus Master can continue reading sequential bytes of data and will terminate the command with a Stop condition.

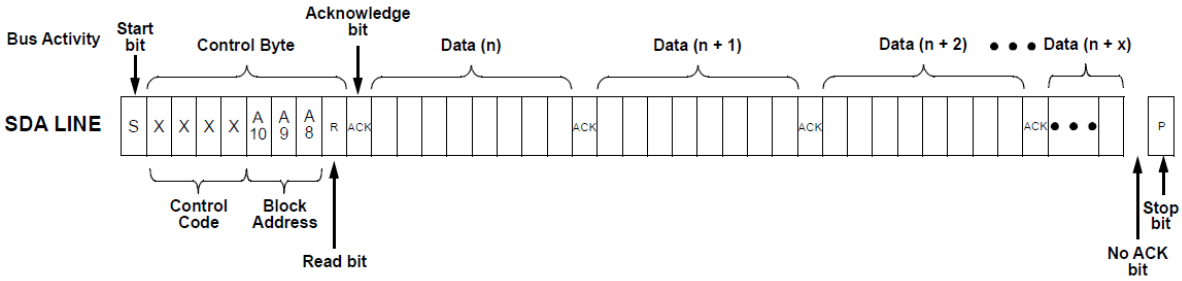

Figure 6: Sequential read command

1.4.5 Current Address Read command

The Current Address Read Command reads from the current pointer address location. The address pointer is incremented at the first STOP bit following any write control byte. For example, if a Sequential Read command (which contains a write control byte) reads data up to address n, the address pointer would get incremented to n + 1 upon the STOP of that command. Subsequently, a Current Address Read that follows would start reading data at n + 1. Control Byte sent by the Master, with the R/W bit = “1” will issue an Acknowledge bit, and then transmit eight data bits for the requested byte. Then Master shall not issue an Acknowledge bit but issue a Stop condition.

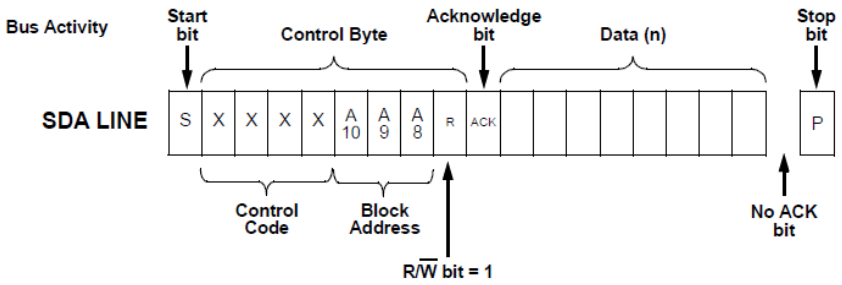

Figure 7: Current address read command

1.4.6 Reset command.

It is possible to reset the device to initial power up conditions, including configuration of all macrocells, and all connections provided by the Connection Matrix. This is implemented by setting bit [1601] on byte address 0xC8, to "1", which causes the device to re-enable the Power-On Reset (POR) sequence, including the reload of all register data from NVM. During the POR sequence, the outputs of the device will be in tri-state. After the reset has taken place, the contents of bit [1601] on byte address 0xC8 will be set to "0" automatically.S

Note: *Byte 0xC8 of RAM consists of two bits, bit [1601] responsible for device reset and bit [1602] responsible for IO latching enable/disable during I²C write operations, other bits are reserved. Initiate Reset command with caution of bit [1602].*

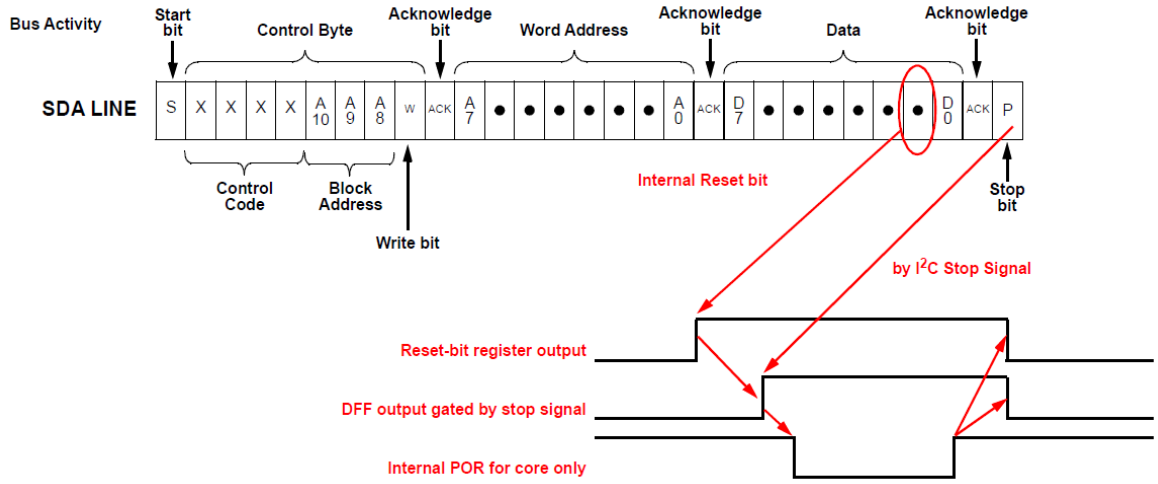

Figure 8: Reset command

1.4.7 NVM Write command.

Write access to the NVM is possible by setting A3, A2, A1, A0 to "0000", which allows serial write data for a single page *only*. Upon receipt of the proper Control Byte and Word Address bytes, the SLG46827-A will send an ACK. The device will then be ready to receive page data, which is 16 sequential writes of 8-bit data words. The SLG46827-A will respond with an ACK after each data word is received. The addressing device, such as a bus Master, must then terminate the write operation with a Stop condition after all page data is written. At that time the device will enter an internally self-timed write cycle, which will be completed within t_{wr} = 20ms. While the data is being written into the NVM Memory Array, all inputs, outputs, internal logic, and I2C access to the Register data (RAM) will be operational/valid.

Note: *The 16 programmed bytes should be on the same page. Any I2C command that does not meet specific requirements will be ignored and NVM will remain unprogrammed. Data "1" cannot be re-programmed as data "0" without erasure. Each byte can only be programmed one time without erasure.*

Caution: *Erase page before writing.*

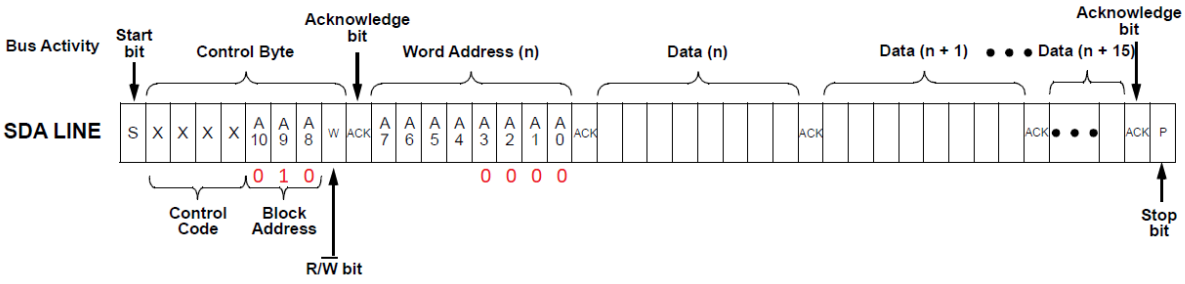

Figure 9: NVM Write command

1.4.8 NVM Read command.

Reading NVM memory is different from writing NVM memory by page of 16 bytes, NVM Read command is the same as serial Read commands.

- Random read command;
- Sequential Read command;
- Current address Read command;

1.4.9 NVM Erase command.

The erase scheme allows a 16 bytes page in the NVM chip configuration space to be erased by modifying the contents of the Erase Register (ERSR). When the ERSE bit is set in the ERSR register, the device will start a self-timed erase cycle which will complete in a maximum of t_{er} = 20 ms.
The VDD pin requires a voltage ranging from 2.5 V to 5.5 V for Programming and Erase operations.
Changing the state of the ERSR is accomplished with a Byte Write command with the requirements outlined in this section.

The ERSR register is located on I^2C Block Address = 000b, I2C Word Address = 0xE3.

After the erase has taken place, the contents of ERSE bits will be set to “0” automatically. The internal erase cycle will be triggered at the time the Stop Bit in the I2C command is received.

Table 2: ERSR register bit functions

bit	Name	Function	Description
7	ERSE		Setting b7 bit to "1" will start an internal erase cycle on the page defined by ERSEB4-0
6	-	-	reserved
5	-	-	reserved
4	ERSEB4	Page selection for erase	Define the page address, which will be erased. ERSB4 = 0 corresponds to the Upper 2K NVM used for chip configuration
3	ERSEB3		
2	ERSEB2		
1	ERSEB1		
0	ERSEB0		

2. RH850/U2A I2C Bus Interface (RIIC)

For demonstration purposes this application note uses RH850/U2A device, but can be use like reference for all RH850 devices line with implemented I2C Bus Interface (RIIC).

2.1 Overview

2.1.1 Functional Overview

Master mode or slave mode selectable, automatic securing of the various set-up times, hold times, and bus-free times according to the specified transfer rate. Transfer rate Up to 400 kbps. For Master operation, the duty cycle of the SCL clock is selectable in the following range: 0% < Duty < 100%. Issuing and detecting conditions: start, restart, and stop conditions are automatically generated.

2.1.2 External Input/Output signals

I2C interfaces need two signals SCL and SDA. For each of the available units alternative function can be selected to use the SCL and SDA signals at a port pin.

Table 3: Alternative port signal name

Unit signal name	Description	Alternative port Pin Signal
RIIC0		
RIIC0SCL	Serial clock I/O pin	RIIC0SCL
RIIC0SDA	Serial data I/O pin	RIIC0SDA
RIIC1		
RIIC1SCL	Serial clock I/O pin	RIIC1SCL
RIIC1SDA	Serial data I/O pin	RIIC1SDA

2.1.3 Clock supply

The RIIC must be supplied with a clock. Depending on the used RH850 MCU, further settings like clock configuration and releasing possible module stand-by settings might be needed.

Table 4: Supply clock name

Unit name	Unit clock Name	Supply Clock Name	Description
RIICn	PCLK	CLK_LSB	Bus clock

Note: Set PCLK of RH850/U2A to a value that is less than 1/2 the SCL clock (high level width).

2.1.4 Reset sources.

RH850 MCU's offer different reset sources, which can be used to reset the RIIC. Table 5 show the reset sources of RH850/U2A

Table 5: Reset sources of RH850/U2A

Unit name	Register name	Reset condition						
		Power On Reset	System Reset1	System Reset2	Application Reset	DeepSTOP Reset	Module Reset	JTAG Reset
RIICn	All registers	Yes	Yes	Yes	Yes	Yes	Yes	No

2.1.5 Interrupts requests

The assignment of interrupts and DMA/DTS channels depend on the used RH850 device. Below table show the interrupt and DMA/DTS channel assignment for RH850/U2A

Table 6: Interrupts request of U2A

Unit interrupt signal	Description	Interrupt number	DMA trigger number	DTS trigger number
RIIC0				
ITRIIC0EE	RIIC communication error/event generation interrupt	685	-	-
ITRIIC0RI	RIIC receive end interrupt	686	Group0-150	Group0-122
ITRIIC0TI	RIIC transmit data empty interrupt	687	Group0-151	Group0-123
ITRIIC0TEI	RIIC transmit end interrupt	688	-	-
RIIC1				
ITRIIC1EE	RIIC communication error/event generation interrupt	689	-	-
ITRIIC1RI	RIIC receive end interrupt	690	Group0-152	Group0-124
ITRIIC1TI	RIIC transmit data empty interrupt	691	Group0-153	Group0-12
ITRIIC1TEI	RIIC transmit end interrupt	692	-	-

2.1.6 Interrupt sources

Interrupts of RIIC can be triggered because of different sources. Table 7 shows the RIIC internal sources of interrupts.

Table 7: RH850/U2A RIIC module Interrupts sources

Symbol	Interrupt Source	Interrupt flag	DMA Launching	Interrupt condition
INTRIICnTI	Transmit data empty	TDRE	Possible	TDRE=1 & TIE=1
INTRIICnTEI	Transmit end	TEND	Not possible	TEND=1 & TEIE=1
INTRIICnRI	Receive end	RDRF	Possible	RDRF=1 & RIE=1
INTRIICnEE	Transfer Error/event generation	AI	Not possible	AL=1 & ALIE=1
		NACK		NACKF=1 & NAKIE=1
		TMOF		TMOF=1 & TMOIE=1
		START		START=1 & STIE=1
		STOP		STOP=1 & SPIE=1

2.1.7 Block diagram

Bellow figure shows one of the RIIC module in RH850/U2A

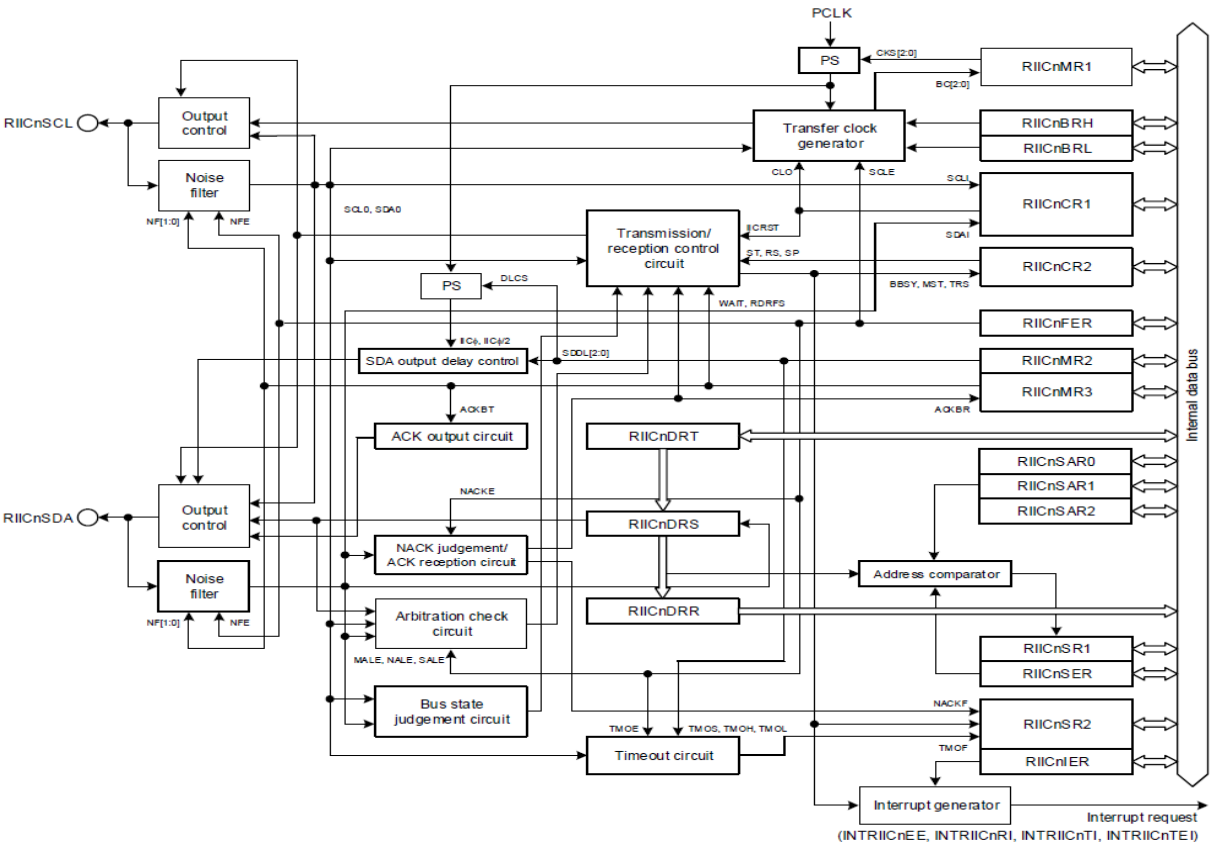

Figure 10: RH850/U2A RIIC module

2.1.8 Communication Data Format

The I2C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a start condition or restart condition is an address frame used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a stop condition is issued.

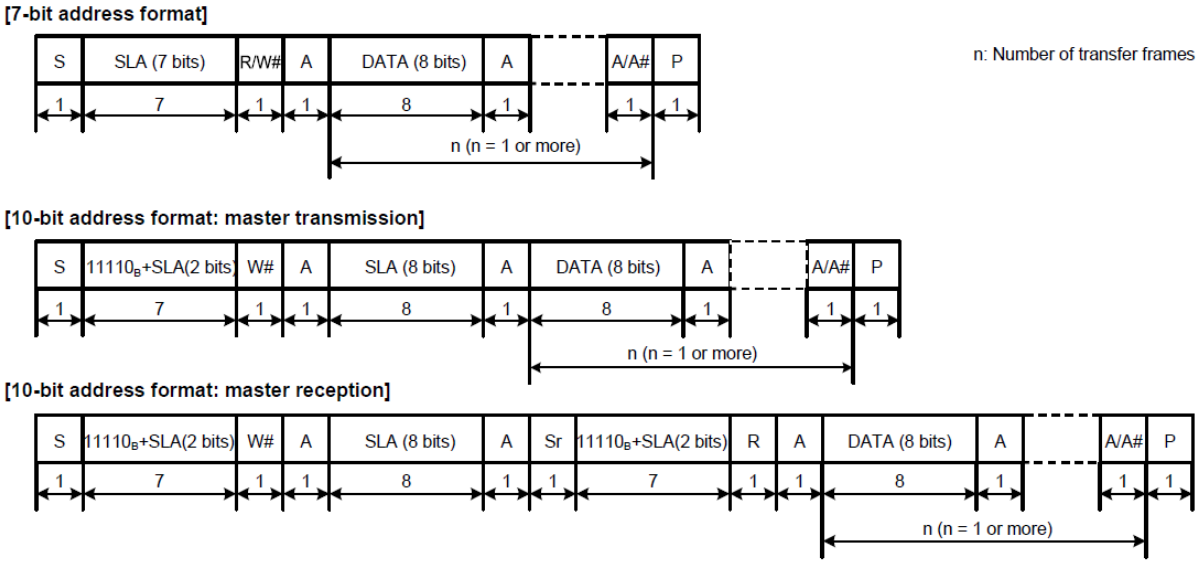

Figure 11: RIIC data format

3. Software and hardware tools

This section contains information about what tools and hardware development platform are used to implement GreenPAK configuration and programing with RH850 devices.

3.1 Software development tools

- Compiler used for sample software is GHS v2020.1.5 for RH850
- GO Configure Software Hub v.6.38 for sample software for SLG46827-A

3.2 Hardware development tools

- Main board RH850-X2X-MB-T1-V1, the latest datasheet can be obtained from the following web location: RH850/X2X User's Manual Main Board
- Piggyback board RH850-U2A-373PIN-PB-T1-V1 with mounted device RH850/U2A R7F702300EBBB-C, the latest datasheet can be obtained from the following web location: Piggyback board RH850-U2A-373PIN-PB-T1-V1
- Advanced Development Board 1, rev.1.3.1 with mounted device SLG46827-A, C, the latest datasheet can be obtained from the following web location: GreenPAK Advanced Development Platform User Guide
- Debug probe – Renesas E2 emulator, the latest datasheet can be obtained from the following web location: E2 Emulator User's Manual

3.3 Setup hardware tools

3.3.1 Connections

Sample software uses RIIC1 periphery to establish I^2C connection to GreenPAK device. In table below is described signals from U2A to GreenPAK advance development board:

Signal name	Piggyback board RH850-U2A	GreenPAK advance development board
SCL	P22_4/ CN16 22pin	SV4 TP8
SDA	P22_3/CN16 18pin	SV4 TP9
VDD	JP15 3.3Vpin	SV4 VDD
GND	CN15 27pin	SV4 GND

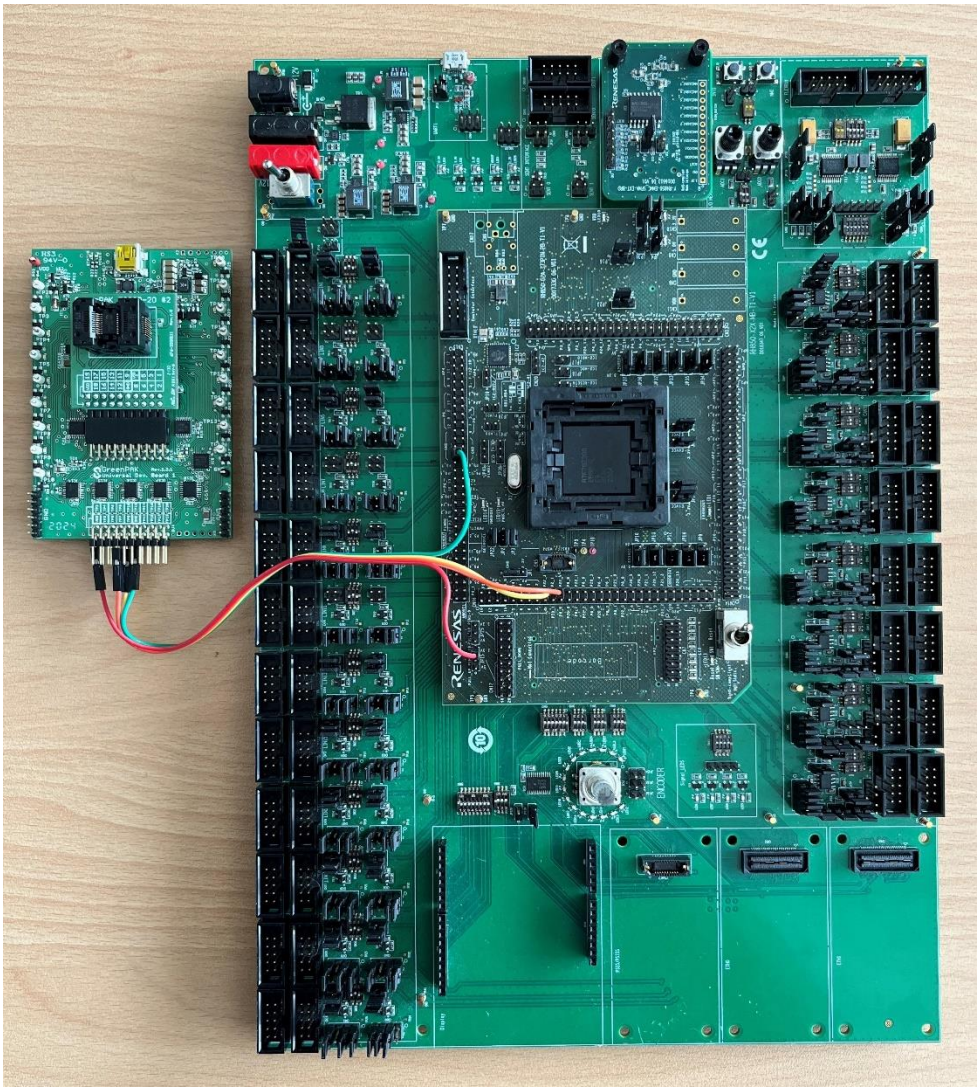

Figure 12: Connections

3.3.2 GreenPAK™ Advance development board setup

To prepare GreenPAK™ Advanced Development Board to can interact with external world it is needed to follow following steps:

1. Open GO configure Software Hub
2. Open sample software for GreenPAK™ SLG46827-A -> SLG46827AG.gp6
3. Press debug button or F9 to open debug session
4. *"Debugging Controls"* will appear on the left.
5. Unselect *"Int VDD"* and select *"Ext.VDD"*
6. Select pin 9 and pin 8 of external connector (SV4) representation.

In picture below is shown how *"Debugging Controls"* should look like.

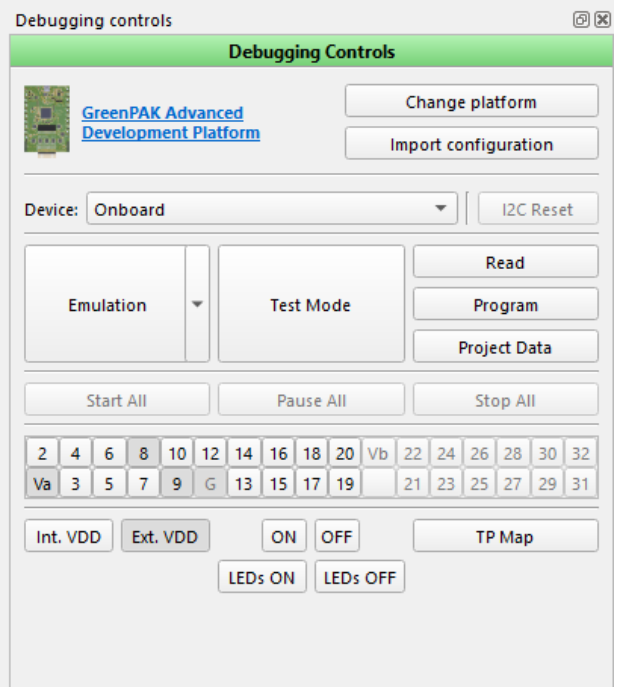

Figure 13: Debugging controls – set for external power and I²C communication

7. Now GreenPAK Advanced Development Board is ready to accept external I²C communication.
8. Press "Emulation" then the sample software will start execution, the LED of TP3 to TP7 will start some "running light".

4. Sample software

Sample software implements basic commands to interact with GreenPAK™ devices, byte write, sequential write, random read, sequential read. These commands are essential to send Reset command, read and write NVM memory, check and change parameters of macrocell in RAM or set lock bits.

Sample software for GreenPAK™ is built in “GO Configure Software Hub”, you can find it archive with demo code under name – “SLG46827AG.gp6”. This software implements basic delay scheme for OSC0 output and drives 5 LEDs to present “running light”, also provide I²C functionality.

Sample software for RH850/U2A device is built in Green Hills Multi, use “RIIC_GP.gpj” to open the project which is in archive with demo code in sub archive “R7F702300_RIIC_GP.zip”. In main function is shown reading and writing to RAM and change OSC0 predivider to change speed to “running light” immediately after write is finished. Also is shown reading and writing to entire NVM memory to change OSC0 predivider to change speed to “running light” – here the changed value takes effect after GreenPAK™ device is reset.

In sample software the setup and interrupt handling of RIIC1 are designed with respect to GreenPAK devices, it is not universal RIIC routine.

Device address is different depending on what address space you need to access please refer to I2C serial communication device addressing.

To test sample software, you need first to connect both demo boards how is shown in chapter 3.1.1 Conections, then start debug session first on RH8650/U2A device and after that on GreenPAK™ device. Runing demo code for RH850/U2A step by step you can observe results how is shown in chapters below.

4.1 Functions

4.1.1 Read byte

```
uint8_t R_RIIC1_Master_ReadByte(uint8_t slave_addr, uint8_t word_addr);
```

The read byte functions implement GreenPAK mechanism to read from random address, where first need to address the device and address to read make restart condition on I²C line and send again device address.

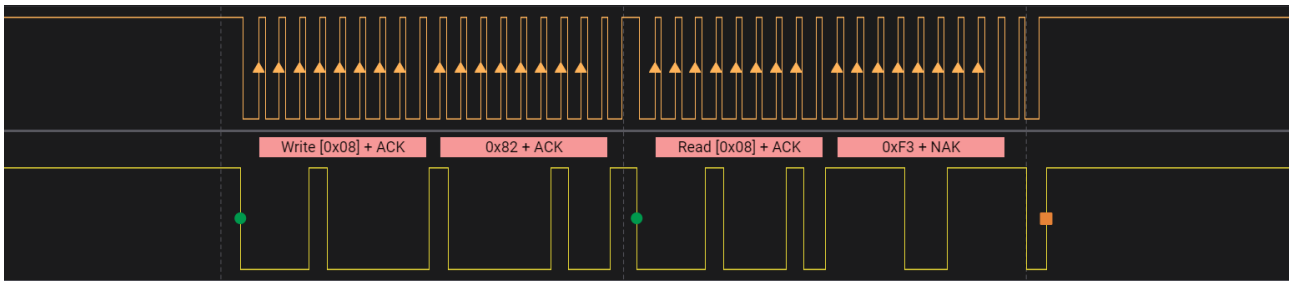

Figure 14: Master read byte from GreenPAK™ device

4.1.2 Write byte

```
void R_RIIC1_Master_WriteByte(uint8_t slave_addr, uint8_t word_addr, uint8_t data);
```

The write byte function is basic for sending commands to GreenPAK devices, usually is used to change the content of RAM and send command like “Reset”.

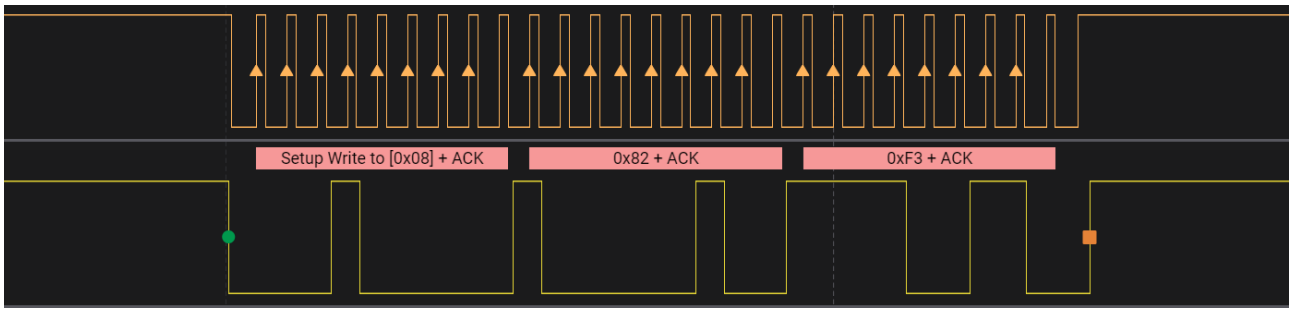

Figure 15: Master write byte to GreenPAK™ device

4.1.3 Read buffer

```
void R_RIIC1_Master_ReadBuffer(uint8_t slave_addr, uint32_t len , uint8_t *r_data);
```

The read buffer function implements sequential read functionality of GreenPAK, user must take care for current pointer of address space is addressed. The pointer to address space is incremented with 1 on the end of each read/write command. If you want to read entire NVM space you need to reset GreenPAK device to set internal pointer to 0 or to make random read from address 0 and after that to read entire NVM without first byte, because internal pointer is set to 1 after first random read command.

4.1.4 Write buffer

```
void R_RIIC1_Master_WriteByte(uint8_t slave_addr, uint8_t word_addr, uint8_t data);
```

The write buffer function implements sequential write routine of GreenPAK devices in NVM memory space. Writing in NVM memory can be done only by page of 16 bytes. Before writing it is needed this page to be erased. Both commands erase/write, one page, took 20ms to finish internal erase/write cycle after the command is send.

4.2 Commands

4.2.1 Reset command

```
void GP_SLG46827A_Reset(void);
```

To reset GreenPAK device you need to change bit [1601] to 1, after reset is finished bit [1601] will became 0 again. Bit [1601] is in byte address 0xC8 of RAM, it is recommended first to read this byte, change only reset bit and write it again. This sequence is implemented in function GP_SLG46827A_Reset.

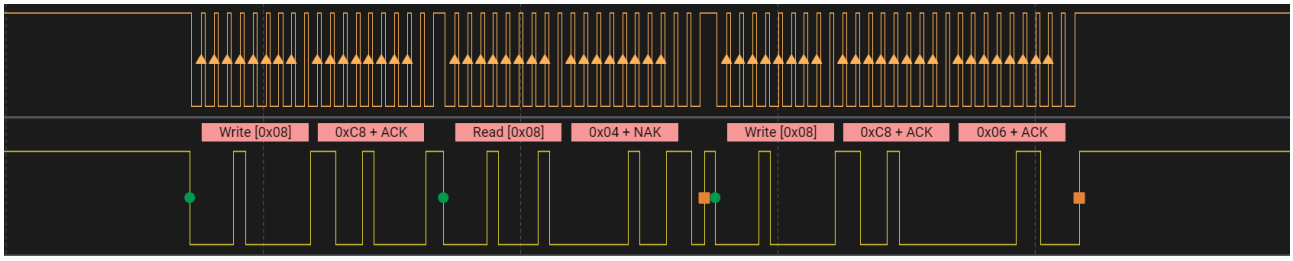

Figure 16: Master reset GreenPAK™ device

4.2.2 Erase command

```
void GP_SLG46827A_PageErase(uint8_t nvmpage);
```

The erase command implements page erase in NVM memory space in GreenPAK including wait time to internal erase cycle to finish.

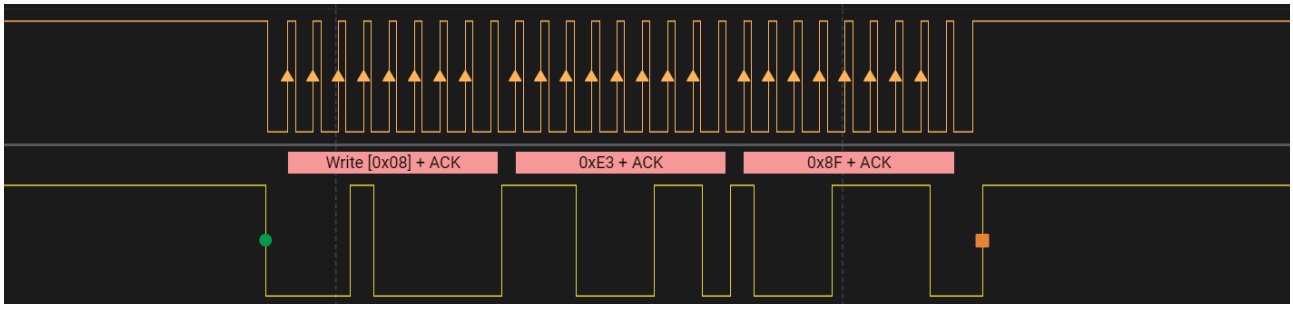

Figure 17: Master erase page in NVM memory

4.3 Test results

In sample software the I²C speed is set to 200kbps

4.3.1 Reading NVM memory time

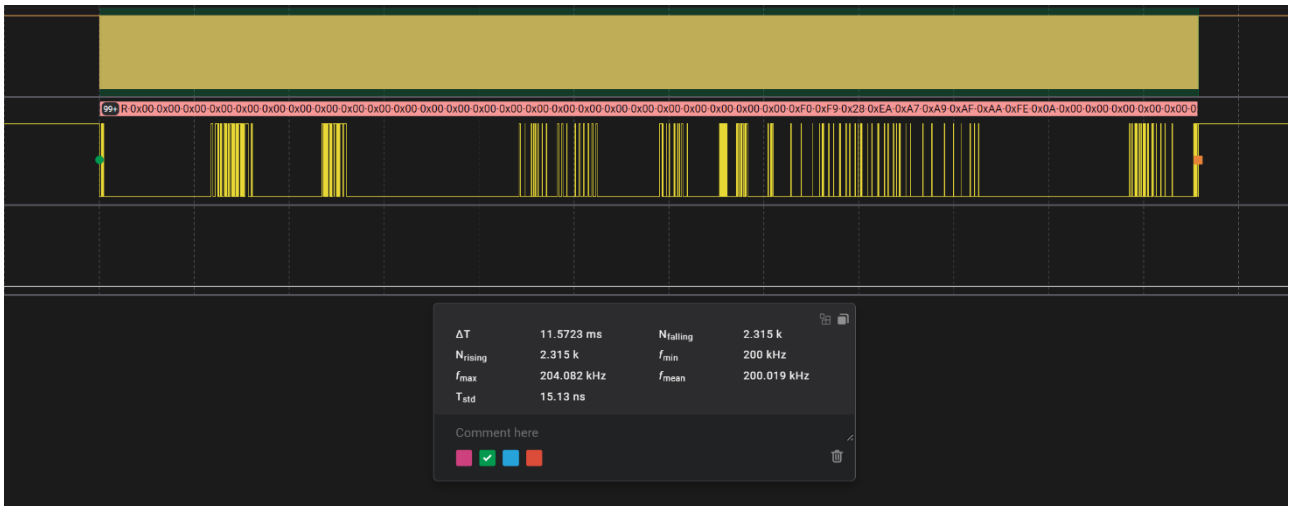

Figure 18: Master read entire NVM memory

4.3.2 Writing NVM memory time

Figure 19: Master write entire NVM memory

4.3.3 Writing to RAM

Writing to RAM to change some parameters takes immediate effect of GreenPAK program execution. In this test is changing predivider of OSC0 make “running lights” to speed up immediately after write operation is done.

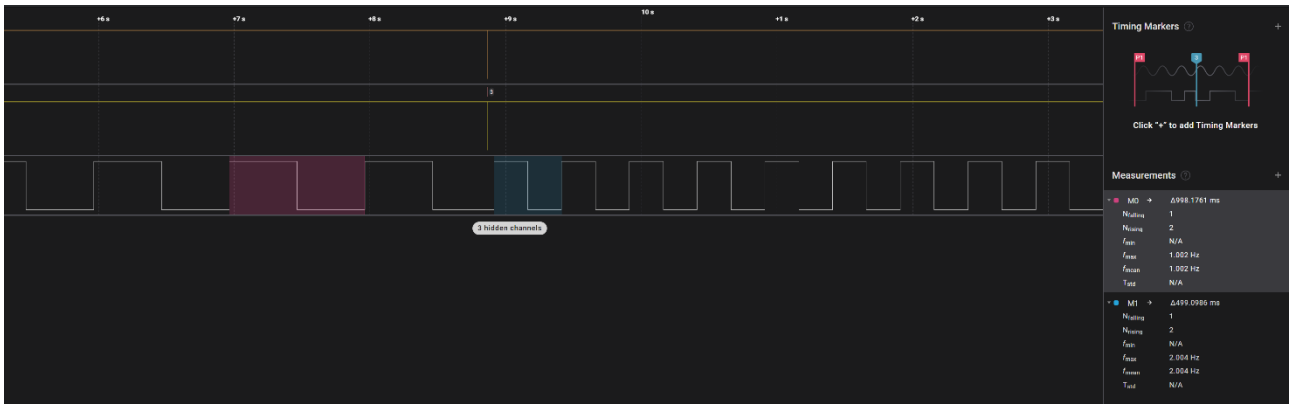

Figure 20: Master change OSC0 predivider

4.3.4 Writing to NVM and reset

Writing to NVM memory does not take effect on running program on GreenPAk until power on reset is executed and NVM memory is loaded to RAM. In this test changing of predivider of OSC0 is done in NVM memory and after execution of reset command, which trigger power on reset, changes are valid in running program of GreenPAK.

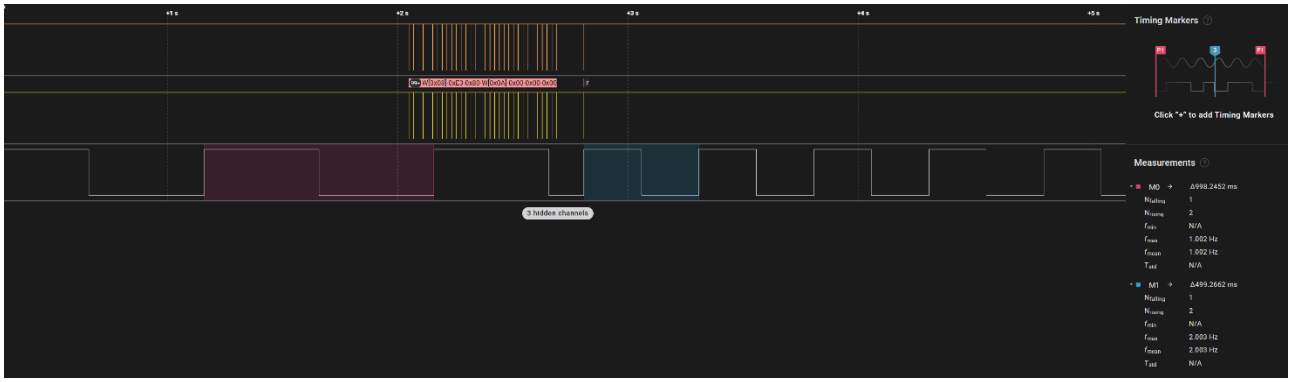

Figure 21: Master writing entire NVM memory and reset device

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	11 October 2023		Initial release

Used Documents

		Description	
Rev.	**Doc. Name**	**Type**	**Device / development tool**
1.30	r01uh0864ej0130_rh850u2a	User's manual: Hardware	RH850/U2A-EVA Group
3.12	REN_SLG46827-A_ds_3v12_DST_20230226	Datasheet	SLG46827-A
2.4	REN_UM-GP-002_User_Manual_GreenPAK_Advanced_Development_Platform_2v3_MAT_20220321_1	User manual	GreenPAK - Advanced Development Board 1

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HALII Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141