User Manual

DA9168 Performance Board

UM-PM-051

Abstract

This document describes the hardware and software used to evaluate the DA9168. It is applicable to the DA9168-03-A1 performance board.

Contents

Figures

Tables

1 Terms and Definitions

GPIO	General purpose input / output
GUI	Graphical user interface
IOs	Inputs / outputs
OTP	One time programmable
MUX	Multiplexer
PC	Personal computer
PCB	Printed circuit board
SAM3U	I^2C-USB interface (Microchip® Arm®-based ATSAM3U4E® microcontroller)
SCH	Schematic
GND	Ground
USB	Universal serial bus

2 References

[1] DA9168, Datasheet, Dialog Semiconductor.

[2] DA9168-03-A1_sch.pdf, Dialog Semiconductor.

[3] DA9168-03-A1_pcb.pdf, Dialog Semiconductor.

3 Introduction

The DA9168 performance board enables the measurement, evaluation, and programming of the DA9168 device.

Dialog's control software SmartCanvas™, uses a graphical user interface (GUI) to control DA9168 via the USB port of a PC. The I²C-USB connector is on the bottom side of the performance board.

The board has jumper links to provide access to alternative configurations and measurement test points.

4 DA9168 Performance Board Hardware

The DA9168 performance board functionality is organized in seven discrete sections, see Figure 1.

1. VBUS input section: Power supply and USB power supply connectors.
2. VSYS output section.
3. VBAT input/output section.
4. VOUT1 and VOUT2 outputs section.
5. REFLDO output section.
6. GPIOs network section.
7. GPIOs signal monitors section.

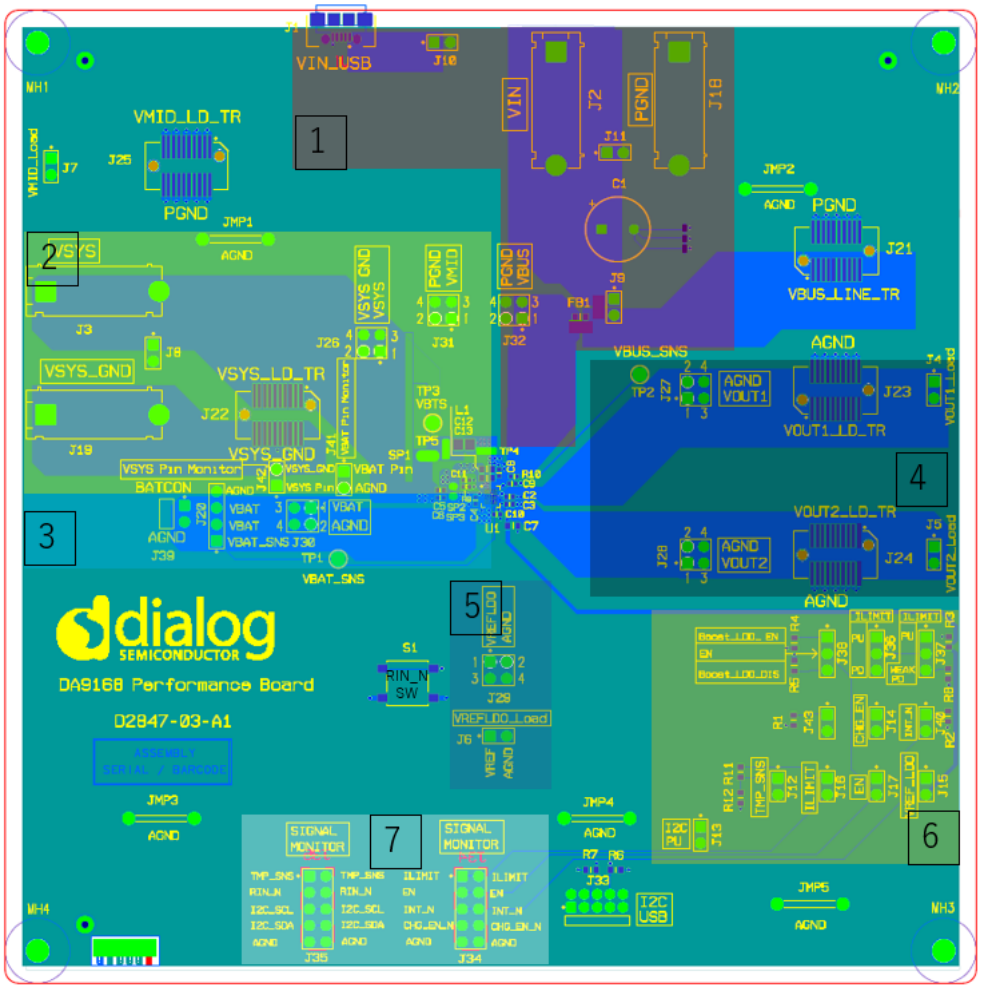

Figure 1: DA9168 Performance Board Overview

4.1 Default Jumper Positions and Connector Definitions

DA9168 performance board default jumper connections are shown as Figure 2.

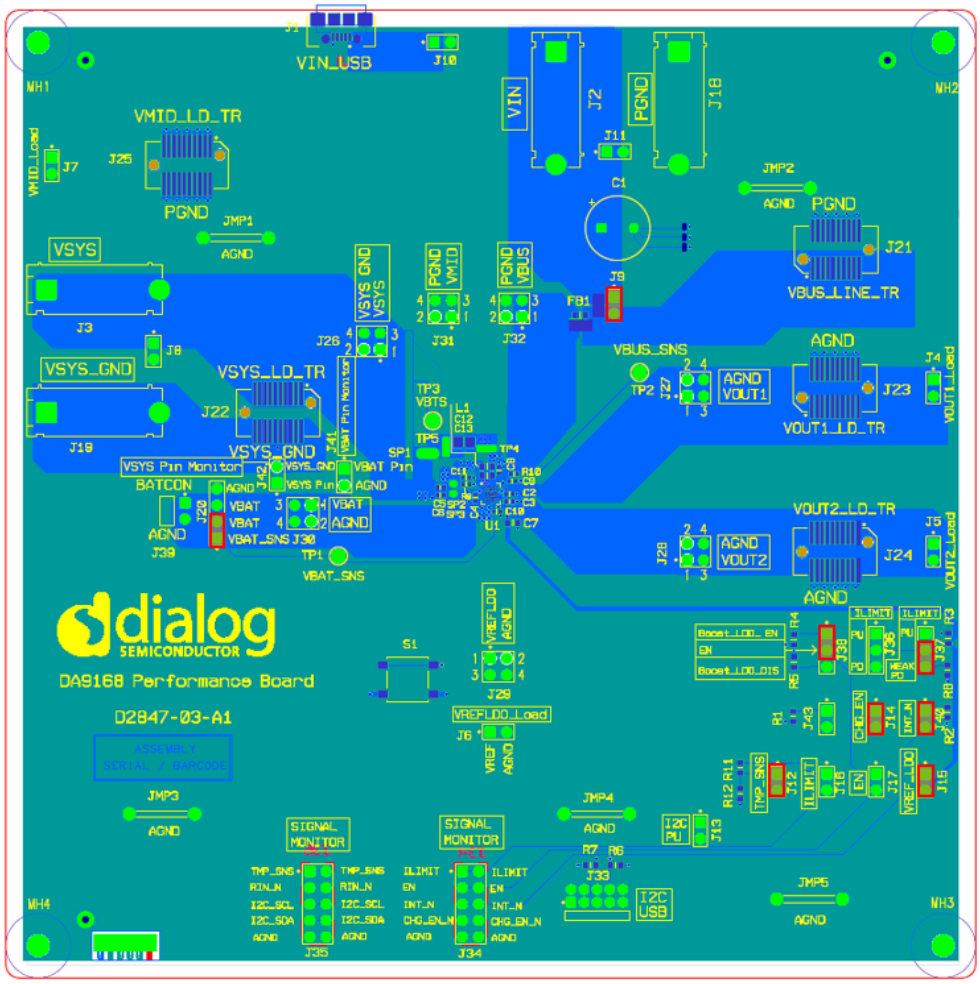

Figure 2: DA9168 Performance Board Default Jumper Position

Table 1: Header and Switch Definitions

Note: Default jumper positions are indicated in **bold**.

Reference Designator	Position	Function
J1	n/a	Micro USB-2.0 connector
J2	n/a	VIN power
J3	n/a	VSYS load
J4	n/a	VOUT1 load
J5	n/a	VOUT2 load
J6	n/a	VREFLDO load
J7	n/a	VMID load
J8	1	VSYS load voltage sense
	2	VSYS load GND sense
J9	**1-2**	Connects VIN to VBUS
J10	1-2	VIN_USB to VBUS

Reference Designator	Position	Function
J11	1	VIN power sense
	2	VIN GND sense
J12	**1-2**	Connects TMP_SNS network to TMP_SNS pin
J13	1-2	Reserved
J14	**1-2**	CHG_EN_N pull-down connector
J15	**1-2**	TMP_SNS pull-up connector
J16	1-2	Reserved
J17	1-2	Reserved
J18	n/a	VIN GND
J19	n/a	VSYS GND
J20	**1-2/3**	Connects VBAT_SNS and VBAT
	4	AGND
J21	n/a	VBUS line transient connector
J22	n/a	VSYS load transient connector
J23	n/a	VOUT1 load transient connector
J24	n/a	VOUT2 load transient connector
J25	n/a	VMID load transient connector
J26 (VSYS Kelvin sensing)	1	VSYS output voltage sense
	2	VSYS GND voltage sense
	3	VSYS load
	4	VSYS GND
J27 (VOUT1 Kelvin sensing)	1	VOUT1 output voltage sense
	2	VOUT1 GND voltage sense
	3	VOUT1 load
	4	VOUT1 GND
J28 (VOUT2 Kelvin sensing)	1	VOUT2 output voltage sense
	2	VOUT2 GND voltage sense
	3	VOUT2 load
	4	VOUT2 GND
J29 (REFLDO Kelvin sensing)	1	REFLDO output voltage sense
	2	REFLDO GND voltage sense
	3	REFLDO load
	4	REFLDO GND
J30 (VBAT Kelvin sensing)	1	VBAT voltage sense
	2	VBAT GND voltage sense
	3	VBAT load
	4	VBAT GND
J31 (VMID Kelvin sensing)	1	VMID output voltage sense
	2	VMID GND voltage sense

Reference Designator	Position	Function
	3	VMID load
	4	VMID GND
J32 (VBUS Kelvin sensing)	1	VBUS voltage sense
	2	VBUS GND voltage sense
	3	VBUS load
	4	VBUS GND
J33	n/a	I^2C-USB module connector
J34	1-10	GPIOs signal monitor
J35	1-10	GPIOs signal monitor
J36	1-2	ILIMIT pull-up connector
	2-3	ILIMIT pull-down connector
J37	1-2	ILIMIT pull-up connector
	3-4	ILIMIT weak pull-down connector
J38	**1-2**	EN pull-up connector
	2-3	EN pull-down connector
J39	1	Connects to real battery positive
	2	Connects to real battery GND
J40	**1-2**	INT_N pull-up connector
J41	1	Device C1 pin (VBAT) voltage sense
	2	VBAT GND voltage sense
J42	1	Device C2 pin (VSYS) voltage sense
	2	VSYS GND voltage sense
J43	1-2	CHG_EN_N pull-up connector
S1	n/a	RIN_N reset push button

4.2 I²C-USB Module

The DA9168 performance board uses an I²C-USB module (Figure 3) for the I²C communication. This I²C-USB module integrates a Microchip® Arm®-based ATSAM3U4E® (SAM3U) microcontroller.

Note: The SAM3U microcontroller VDDIO voltage is fixed at 3.3 V; therefore, the I²C-USB module only supports 3.3 V level I²C communication.

Figure 3: I²C-USB Module

5 Basic Operation

5.1 Starting Fast Charge Mode

Before starting Fast Charge mode, confirm the jumper connections are correct; VBUS and VBAT voltage level:

- VBUS voltage is higher than VINDPM, typical setting is 5.0 V
- VBAT voltage is higher than fast charge voltage threshold, typical setting is 3.7 V

Once power is applied, OTP registers are loaded and I²C communication is available.

Set register CHG_EN (0x16[0]) to high. DA9168 enters Fast Charge mode.

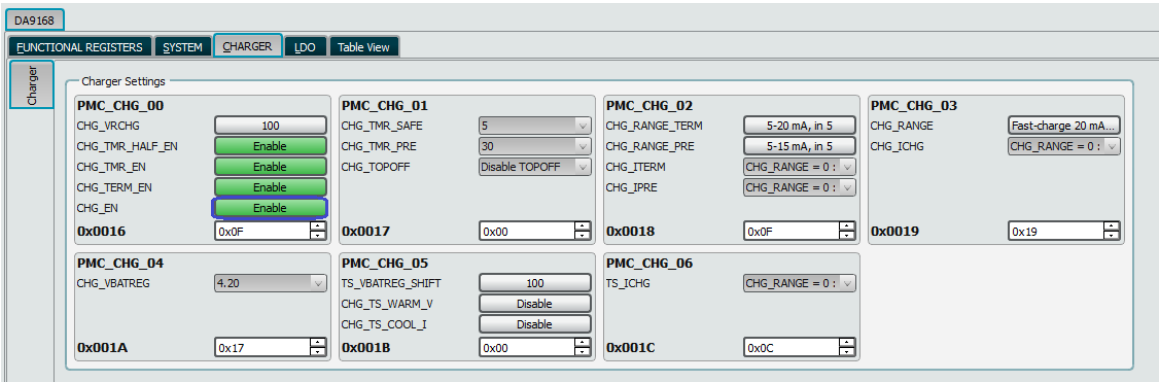

Figure 4: Enable Fast Charge Mode

5.2 Starting Reverse Boost Mode

To enable DA9168 Reverse Boost mode, confirm the following conditions:

- VBUS power supply is disconnected
- VBAT voltage is higher than reverse boost voltage threshold, typical setting is 3.7 V
- Bits BOOST_VOUT(0x14[3:0]) setting is 0.5 V higher than VBAT

Set register BOOST_EN (0x13[0]) to high. DA9168 enters Reverse Boost mode.

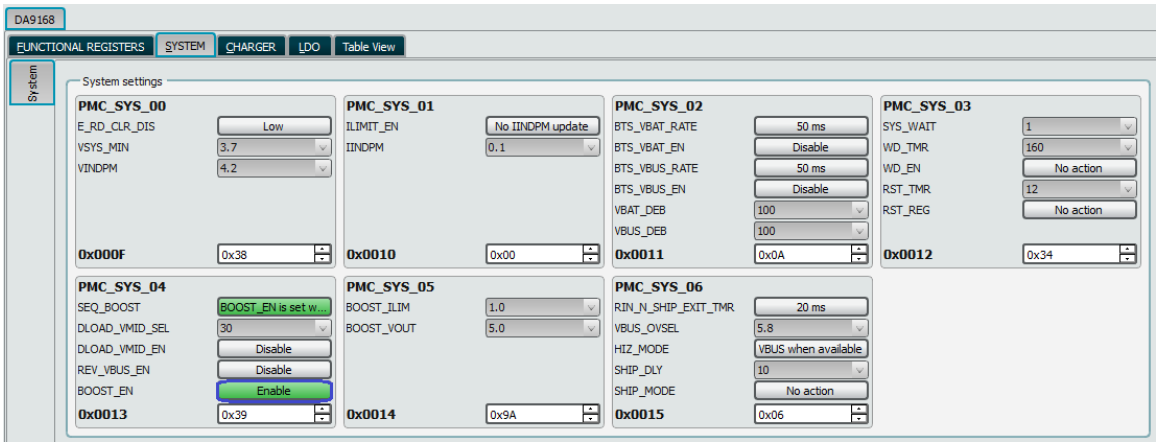

Figure 5: Enable Reverse Boost Mode

5.3 Signal Monitoring

The Kelvin sensing method is used on the DA9168 performance board to improve the measurement accuracy. On J26, J27, J28, J29, J30, J31, and J32, header pins 1 and 2 are used as voltage sensing pins, while pins 3 and 4 can be used for measuring the current flows.

- J41 monitors the DA9168 C1(VBAT) pin voltage directly
- J42 monitors the DA9168 C2(VSYS) pin voltage directly

⚠	**CAUTION**
Apply high current to J26, J27, J28, J29, J30, J31, J32, J41 and J42 headers pin 1 and 2 may cause the voltage sensing traces to burn out.	

Other signals:

- switching node is monitored at TP4
- VBTS signal is monitored at TP3
- GPIO signals are monitored at J34 and J35 headers

5.4 Fault Indicators

Faults are indicated in the **System Events** tab in the SmartCanvas GUI.

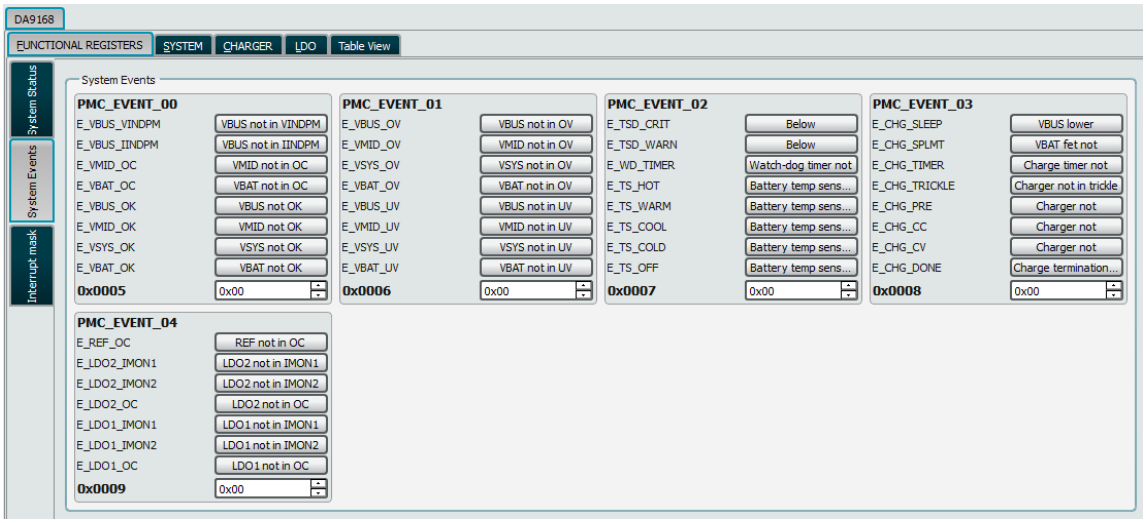

Figure 6: System Events Tab

5.5 Interrupt Signals

DA9168 interrupt signals are masked in the **Interrupt Mask** tab.

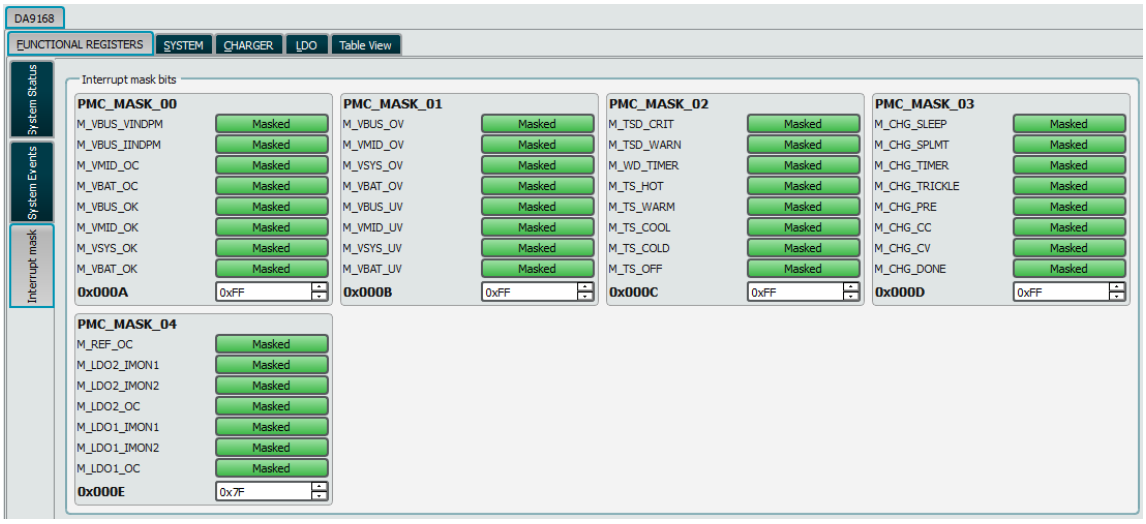

Figure 7: Interrupt Mask Tab

Note: A masked fault is still indicated as high in the read-only registers 0x0000 to 0x0004 in the **System Status** tab.

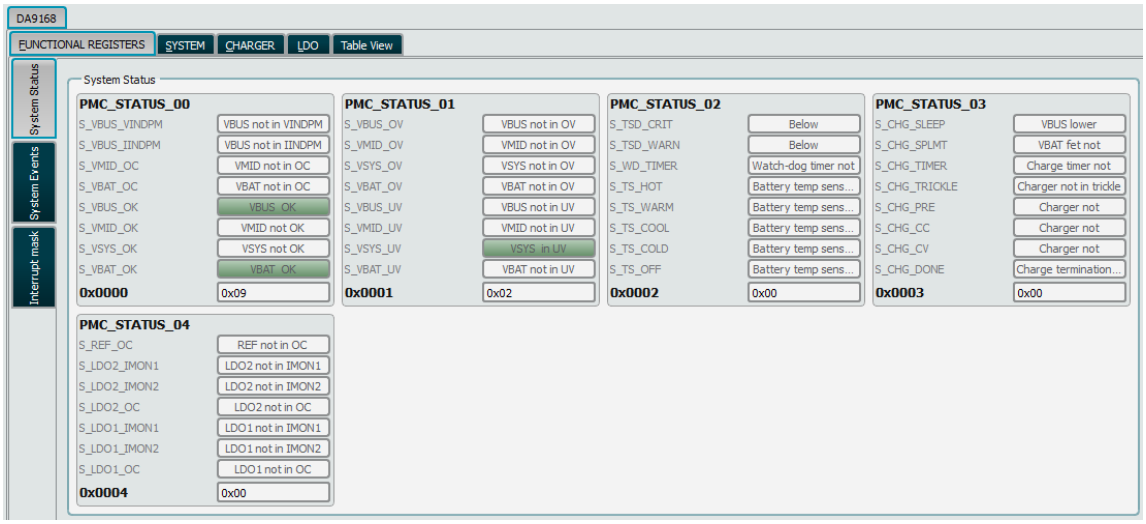

Figure 8: System Status Tab

6 DA9168 Performance Board Software

The board is controlled using the SmartCanvas software graphical user interface (GUI), which requires a PC operating Windows® 2000/XP/Vista/7/10 with a USB1.1 or USB2 interface. The GUI allows the user to:

- configure the DA9168 device
- perform write and read operations to all control registers
- monitor the device status, including faults

6.1 SmartCanvas GUI Installation

The files required to install the software are available on the supplied USB drive. To install the DA9168 SmartCanvas software:

1. Run **setup_DA9168_GUI_0.0.0.3.exe**. The program default install location is: **C:\Dialog Semiconductor\Power Management\DA9168 GUI**.
2. On completion, plug in the performance board I²C-USB and apply power to VBUS or VBAT. The software must be started after the I²C-USB is plugged in, otherwise communication with the board may fail.
   a. For first time users, Windows should detect the attached USB device. If this is not the case, it may be necessary to install the driver by navigating to the required driver file in the USB driver directory, see Section 6.2. For guaranteed operation, it is recommended that a PC reboot be carried out after installing the driver.
3. Once installation is complete, run the DA9168 SmartCanvas software: **DA9168 GUI.exe**.

6.1.1 SmartCanvas GUI Installation Step-by-Step Guide

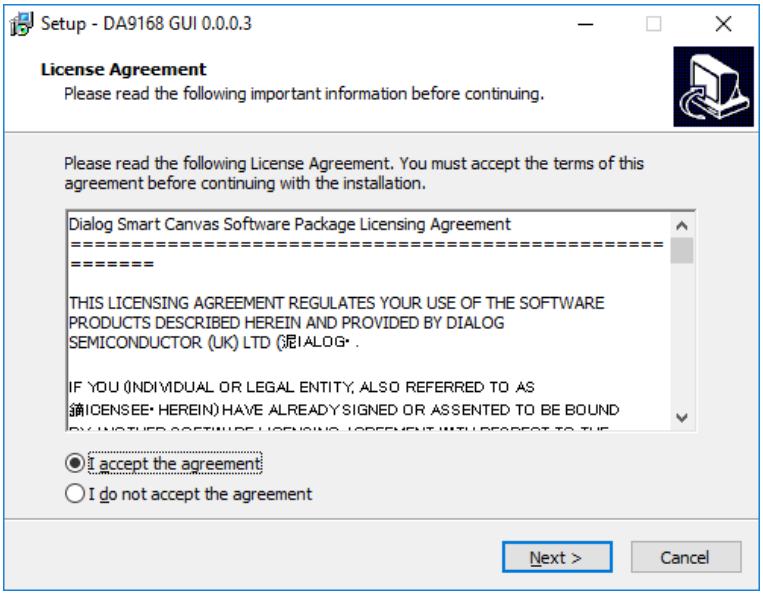

Figure 9: GUI Setup License Agreement

4. Select **I accept the agreement** then click **Next**.

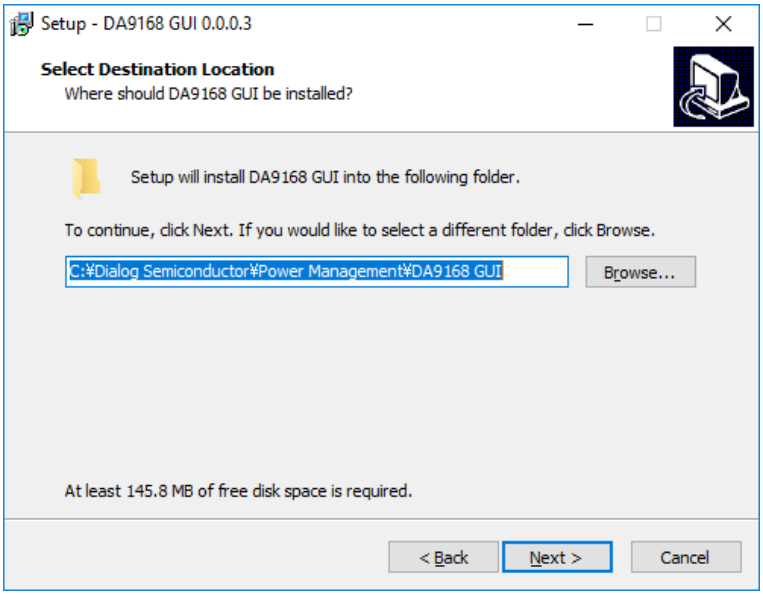

Figure 10: GUI Setup Destination Location

5. Click **Next**.

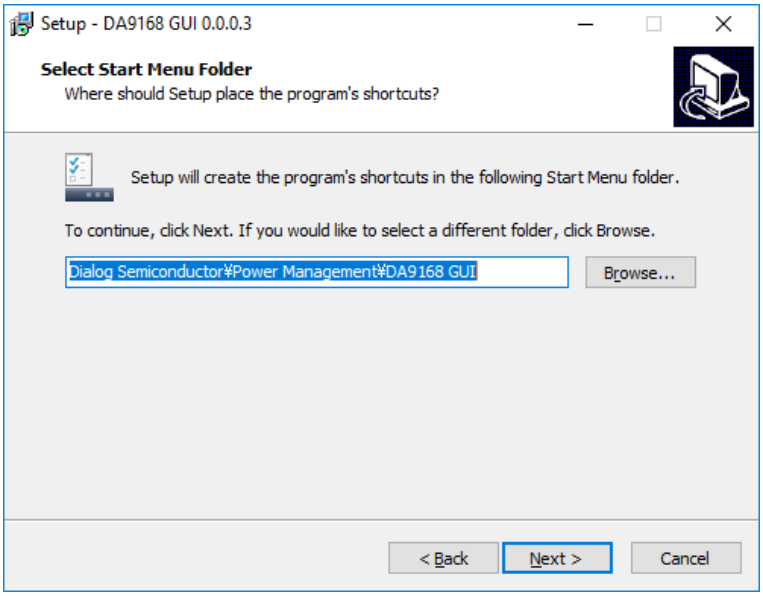

Figure 11: GUI Setup Shortcuts Location

6. Click **Next**.

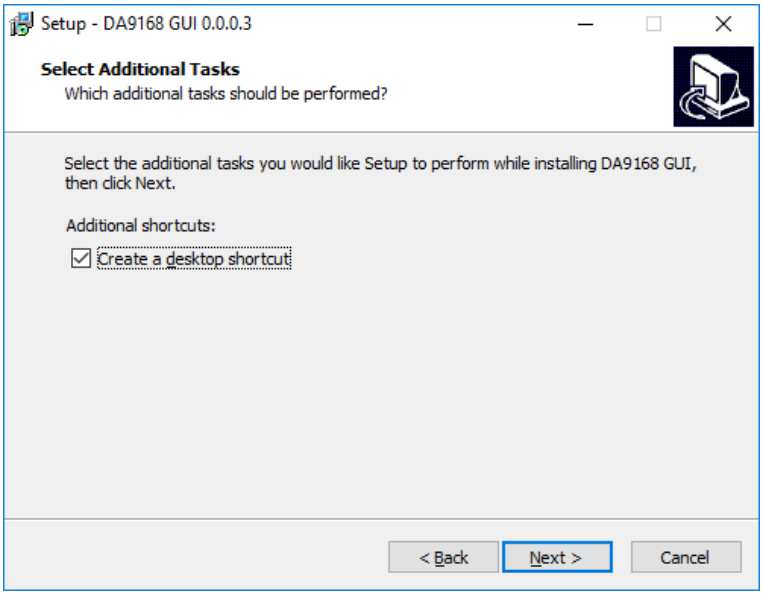

Figure 12: GUI Setup Create Shortcut

7. Select the **Create a desktop shortcut** check box and click **Next**.

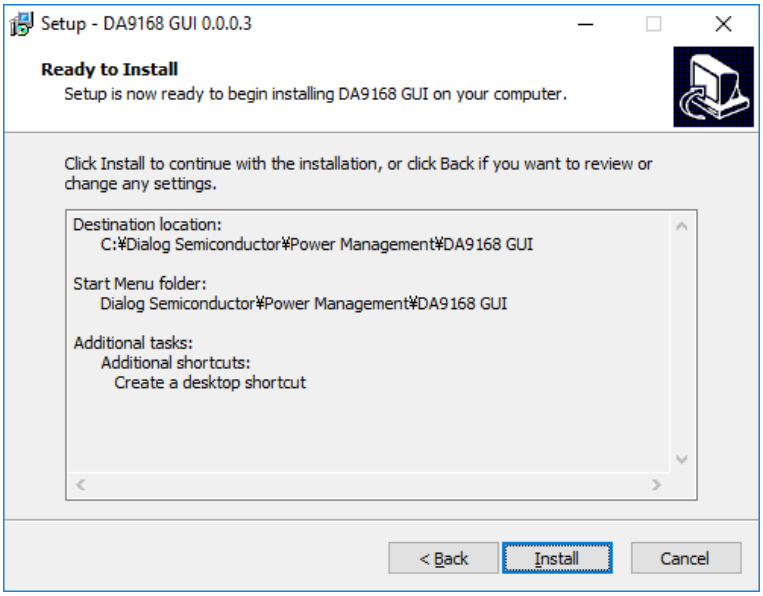

Figure 13: GUI Setup Installation

8. Click **Install**.

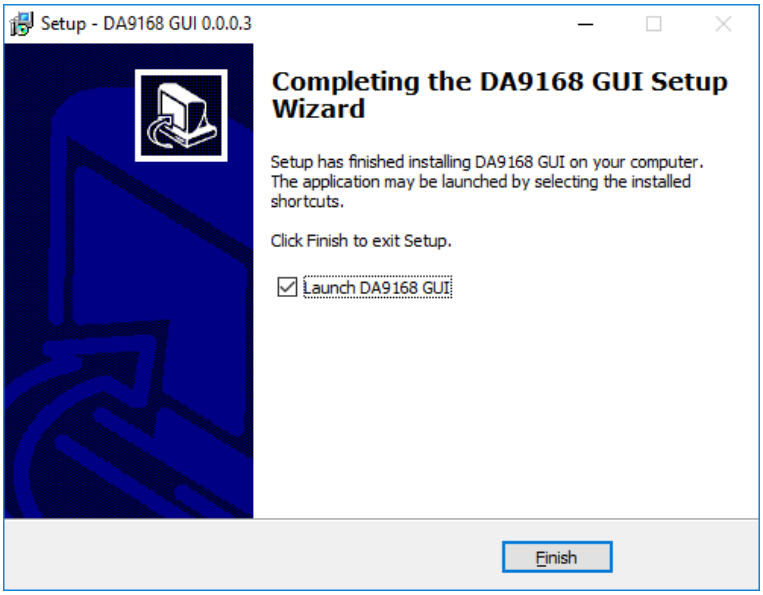

Figure 14: GUI Setup Finish and Launch

9. Click **Finish**.

Once the installation is complete, a PC restart may be required.

When the software is installed, insert the I²C-USB cable before applying power. The DA9168 SmartCanvas software can be started after power up.

6.2 Initial I²C-USB Connection

On connecting the I²C-USB to the PC for the first time, the SAM3U USB driver requests driver updating/installation from the Windows operating system. On Windows 7/10 32-bit operating systems the driver usually installs automatically. On Windows 7/10 64-bit machines it is common for the complete driver installation to fail. If this happens you must install the driver manually by following these steps:

1. **Control Panel** → **Devices and Printers** (double-click device with yellow exclamation sign).
2. **Update Driver**.
3. **Browse my computer for driver software**.
4. Select the **Driver** folder location: **C:\Dialog Semiconductor\Power Management\DA9168 GUI\driver**.
5. If Windows warns about the driver, select **Install anyway**.
6. Remove the I²C-USB cable then reinsert it into the performance board.

7 SmartCanvas Software

Run the DA9168 SmartCanvas software by clicking the shortcut on the appropriate item in the **Start menu** (**All Programs** → **Dialog Semiconductor** → **Power Management** → **DA9168 GUI**). The main GUI interface is displayed, see Figure 15.

The minimum recommended setting for the PC display size is 1024x768 pixels. Font size on the PC display should be Normal (95 dpi).

NOTE
It is important to note that a display size other than the recommended setting affects the way in which the panels appear.

7.1 Register Controls

Device registers are displayed as a group of controls. Selected bit ranges within a register make up a control. Register data is always a standard bit width dictated by the device register map, but a control can be anything from 1-bit to the full register data width.

Registers are grouped together on tabs to assist with identification of device function or registers of the same type.

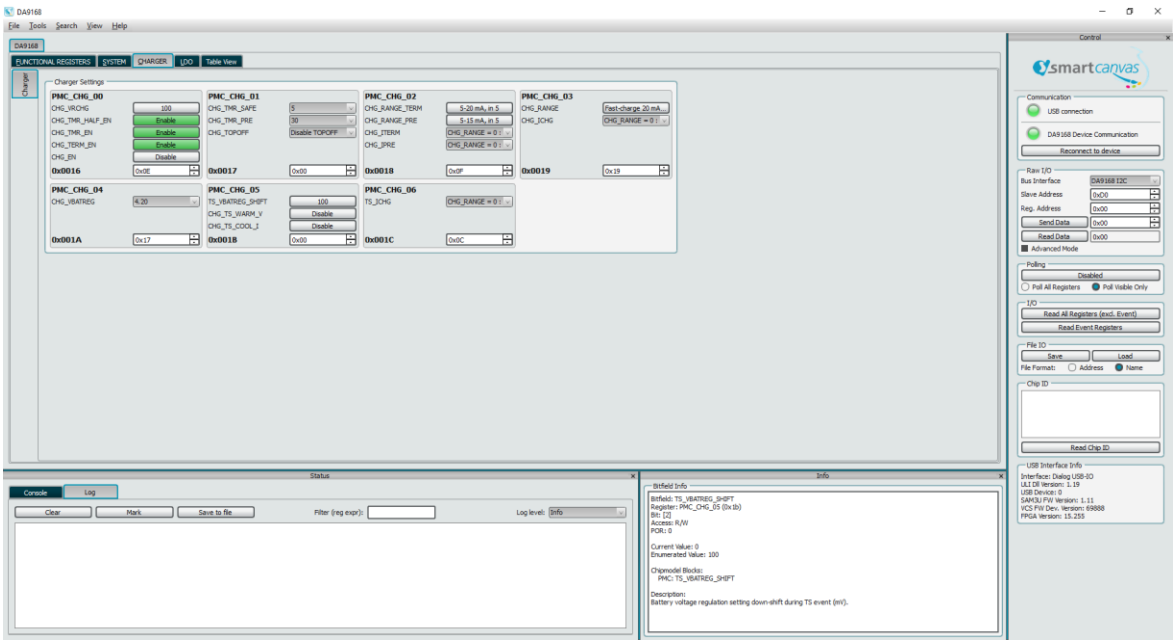

Figure 15: Main Interface

7.2 Table View Tab

The **Table View** tab shows the complete register map, see Figure 16. From this tab all the registers can be set or read back. Clicking on any of the bit groups on the map allows access to the full control settings. These controls read and write the same value elsewhere on the interface.

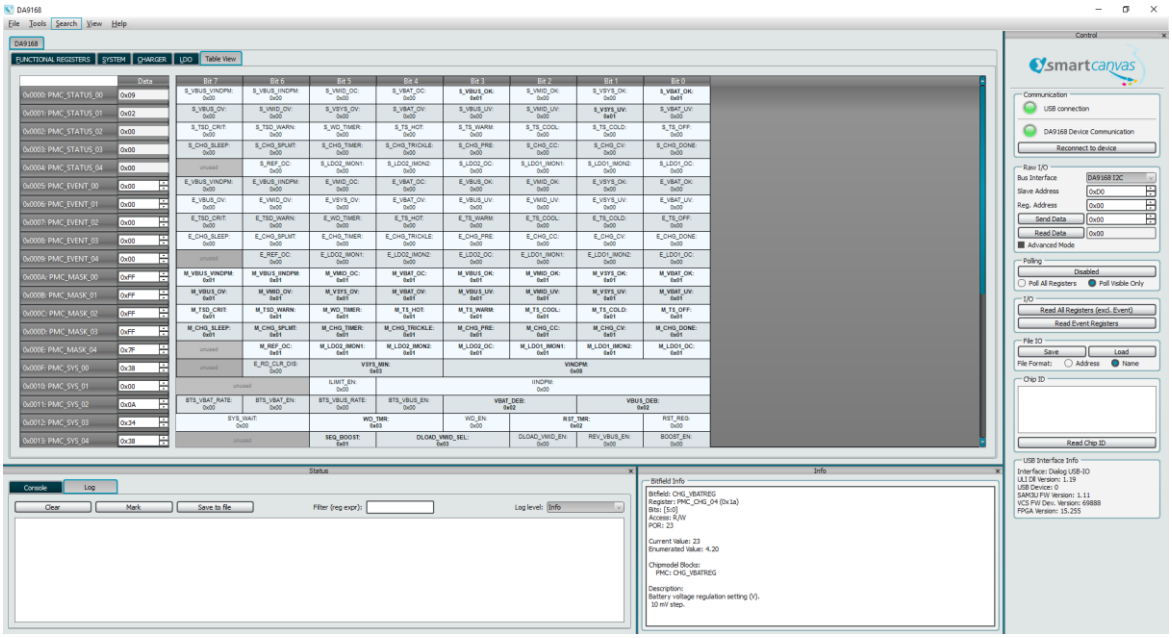

Figure 16: Table View

7.3 Control Windows

All dockable control windows, either on the right or bottom of the main window add additional functionality or monitoring to the GUI. The windows can be undocked by clicking on and dragging the title bar of that window and can be placed anywhere on the screen.

7.3.1 Communication

The **Communication** window has indicators to show when the SAM3U I²C-USB module is plugged in and when I²C communication is possible; the currently active I²C-USB device number is also shown, see Figure 17. Most of the time the communication link automatically connects if the I²C-USB is active and the device is powered up. On rare occasions, sequence of events may prevent recognition of the active communication link, pressing the **Reconnect to device** button recovers the link.

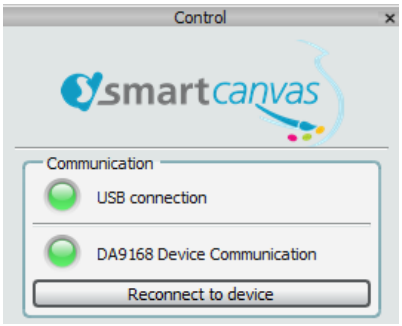

Figure 17: Communication Control

7.3.2 Enable/Disable Polling

The **Enable/Disable Polling** button enables or disables polling of the device registers and the refreshing of the registers controls on the GUI interface, see Figure 18.

Figure 18: Polling

The **Enable/Disable Polling** button also allows the polling rate to be changed. There is an option to poll all or just the visible registers; however, selecting **Poll Visible Only** may have adverse effects on the functionality of the automation controls and is therefore not recommended.

7.3.3 I/O

The I/O update actions are as follows:

- **Read All Register (excl. Event)** – even if the hardware device is not being actively polled then all the registers can be polled once by pressing this button.
- **Read Event Registers** – all event registers can be polled once by pressing this button.

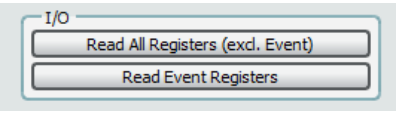

Figure 19: I/O

7.3.4 Raw I/O

The **Raw I/O** control sends the entered device address, register address and data on the I^2C communications interface, see Figure 20. If the information sent is not valid then the I^2C message returns NACK and an error message is displayed in the **Status** window. Data from an individual address can also be READ from this window.

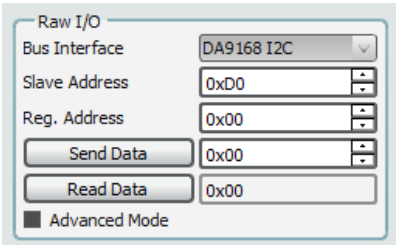

Figure 20: Raw I/O

The optional **Advanced Mode** allows control of the I^2C frequency.

7.3.5 File I/O

The **File I/O** control **Load** button allows formatted text files to be loaded into the device registers, which is then be reflected on the **Registers** display of the GUI interface, see Figure 21.

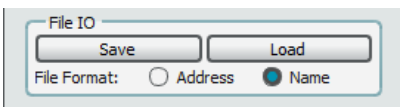

Figure 21: File I/O

The **Save** button saves the register values displayed on the GUI interface. If saving with the extension .txt or .csv, one of the **File Format** radio buttons can be selected to save by either register name or register address.

A predefined automated sequence is present in this window. This single-shot button allows a separate script to be loaded as required.

7.3.6 Info

The **Info** window displays a description of an interface control including: name, parent register and the bits to which this control corresponds, current value, whether it is read only or R/W, and finally a description of each possible setting, see Figure 22.

For long descriptions the window may either be undocked and made larger; or docked at the left-hand side of the main window.

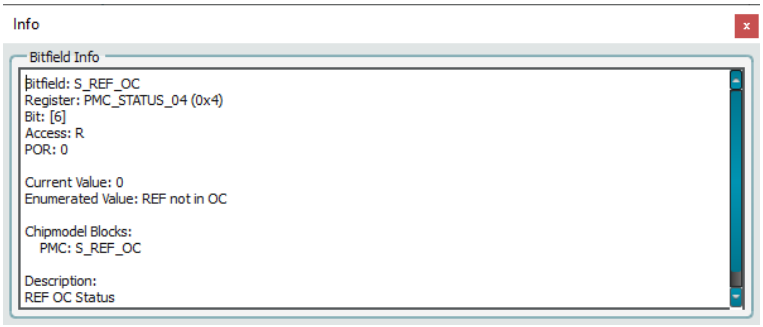

Figure 22: Info

7.4 Menu Items

There are several menu items found on the top line of the interface, which allows the selection of enhanced functionality. These are not required in most cases of interface use.

7.4.1 File -> Open Python Script

The **Open Python script** option in the **File** menu opens a Python® script selected in the pop-up window.

7.4.2 File -> Run Python Script

The **Run Python script** option in the **File** menu runs the Python script selected in the pop-up window.

7.4.3 Tools -> Model IO

The **Model IO** option in the **Tools** menu allows access to all registers at a glance, see Figure 23.

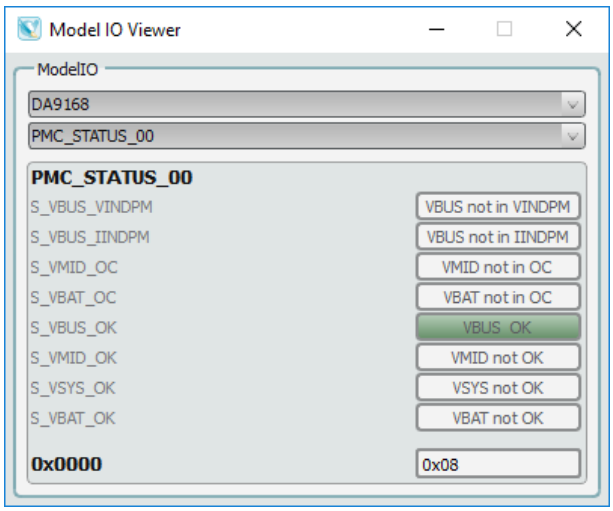

Figure 23: Model IO

7.4.4 Tools -> Scan I2C

The **Scan I2C** option in the **Tools** menu allows the scan of all slave devices on the I²C bus, see Figure 24.

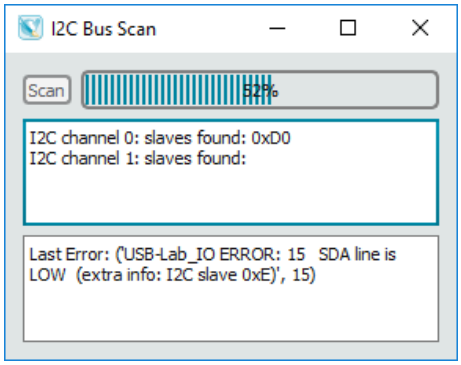

Figure 24: Scan I2C

7.4.5 Tools -> Custom Tabs

The **Custom Tabs** option in the **Tools** menu allows customized tabs to be created by dragging register widgets to the **Custom Tabs** control window, see Figure 25.

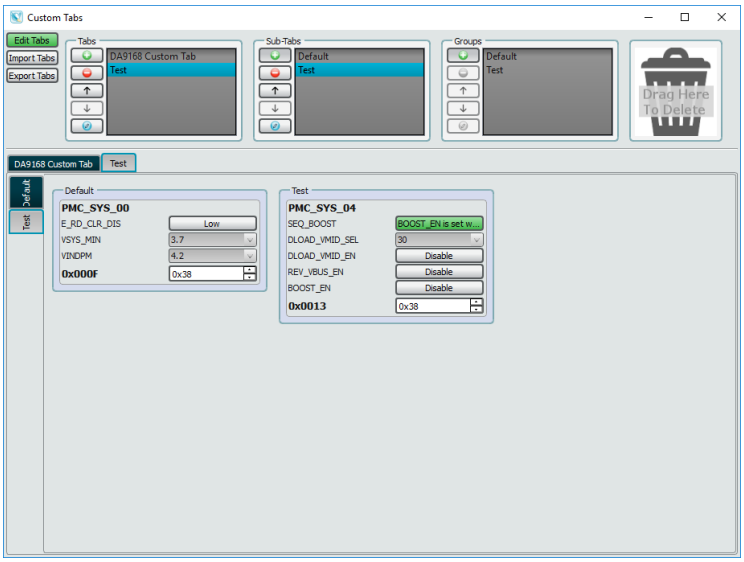

Figure 25: Custom Tabs

7.4.6 Search -> Find Register

The **Find Register** option in the **Search** menu searches by name, or number, or by a particular text term contained within the GUI, see Figure 26. An entry made in the **Search Name** box searches register names and display the instances found in the right-hand side list box. If the **All Text** radio button is selected all entries containing the written text are identified and listed.

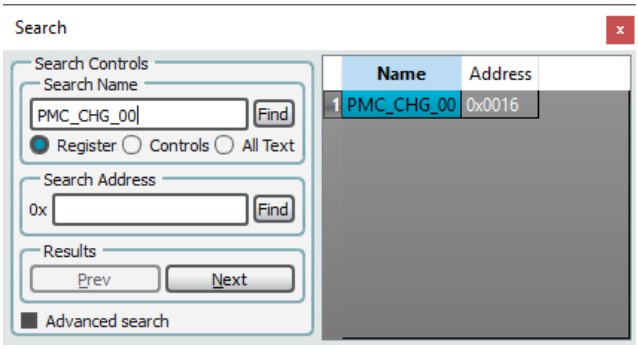

Figure 26: Find Register

If the **Advanced Search** checkbox has been selected, selecting **Reg expr** allows a search based upon Python regular expression functions.

7.4.7 View

The **View** menu reopens the docking windows if they have been previously closed.

8 Register Text File

The DA9168 SmartCanvas software can save and load a text file containing command codes representing the register addresses and data. This file is principally used to save and load setup data but may also be used to perform a small degree of automation.

The use of the **Save** button in the **File I/O** control transfers register contents to the user's software. This is a register dump of the entire device. If the **Name** radial button is selected instead of **Address**, then register names rather than register addresses are saved in the text file.

8.1 I²C Register Text File Format

The following formats are used for both read and write in the text file:

- WRITE: writes to the device: WRITE [device name] [register address] [register data value].
- READ: reads from the device: READ [device name] [register address]. The result of the read is passed to a **File Readback Values** pop-up window.
- DELAY: implements a time delay specified in milliseconds: DELAY [time in milliseconds] – no time suffix required.
- USBIO: controls the SAM3U USB device IOs: USBIO [USB IO pin index] [output value] – USB device output value = 0 or 1.
- ACTION: pauses the file being loaded until the pop-up message dialogue has been acknowledged: ACTION [text message].

Numbers are always expressed in Hex, separated by tabs. The use of 0x in front of the hex value is mandatory. Inline comments (lines beginning with //) are permitted in the file. The data is processed in the order written and written directly to the specified device.

For example:

```
WRITE DA9168_I2C 0x0016 0x0E
WRITE DA9168_I2C PMC_CHG_00 0x0E
READ DA9168_I2C 0x0016
READ DA9168_I2C PMC_CHG_00
DELAY 1000 // delay is 1000 ms or 1 s
USBIO 0 1 // "0" refers to the index number for the USB IO pin on the "USB_Ports" tab.
ACTION Please press the OK button to continue.
```

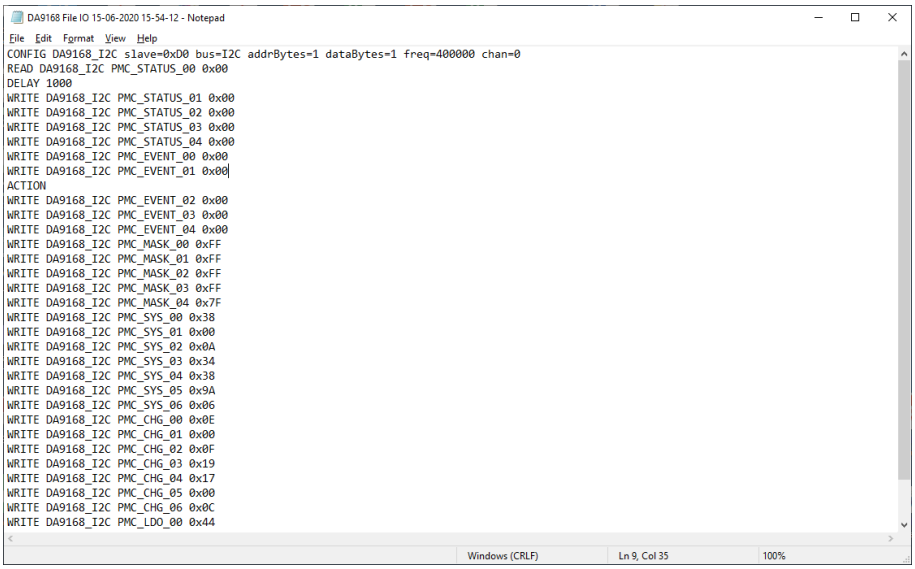

Figure 27: Register Dump (.txt File) Example

Revision History

Revision	Date	Description
3	03-Aug-2022	File was rebranded with new logo, copyright and disclaimer
2	17-Aug-2020	Minor spelling corrections and clarification of Caution in Section 5.3.
1	03-July-2020	Initial version

Status Definitions

Status	Definition
DRAFT	The content of this document is under review and subject to formal approval, which may result in modifications or additions.
APPROVED or unmarked	The content of this document has been approved for publication.

RoHS Compliance

Dialog Semiconductor's suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.

Important Notice and Disclaimer

(Rev.1.0 Mar 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu

Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

https://www.renesas.com/contact/

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.