

# User Manual

## Smartcanvas DA9061/2

### UM-PM-008

#### **Abstract**

*This document explains the DA9061/DA9062 Smartcanvas software and its functionality.*

## Contents

<b>Abstract .....</b>	<b>1</b>
<b>Contents .....</b>	<b>2</b>
<b>Figures.....</b>	<b>3</b>
<b>Tables .....</b>	<b>3</b>
<b>1 Terms and Definitions.....</b>	<b>4</b>
<b>2 References .....</b>	<b>4</b>
<b>3 Introduction.....</b>	<b>5</b>
<b>4 Installation.....</b>	<b>5</b>
<b>5 Power Commander Mode .....</b>	<b>9</b>
<b>6 GUI Overview .....</b>	<b>9</b>
6.1 Front Panel.....	10
6.1.1 GPIOs .....	10
6.1.2 Bucks and LDOs .....	11
6.1.3 RTC Time .....	11
6.1.4 INI File Control.....	11
6.1.5 Power Sequencer .....	12
6.2 Register View .....	13
6.3 Control Widgets.....	14
6.3.1 Raw I/O .....	14
6.3.2 Bus Interface.....	15
6.3.3 Communication .....	15
6.3.4 Power Commander Mode.....	15
6.3.5 ini file Control .....	15
6.3.6 Polling .....	16
6.3.7 I/O .....	16
6.4 USB Ports Table .....	17
6.5 ADC Plot .....	18
6.6 Table View .....	19
<b>7 OTP Programmer Tool .....</b>	<b>20</b>
<b>8 Project File Comparison Tool .....</b>	<b>22</b>
<b>9 Menu Bar .....</b>	<b>23</b>
9.1 Scan I <sup>2</sup> C .....	23
9.2 Search .....	23
9.3 Settings .....	24
9.3.1 Number Base.....	24
9.3.2 Polling .....	24
9.3.3 Watchdog Automation .....	24
9.3.4 Device Connect Error Settings .....	24
9.3.5 Comms Server.....	24
<b>Appendix A : Quick-Start Guides .....</b>	<b>25</b>
A.1 Modifying an .ini File .....	25
A.2 OTP Programming Procedure .....	25
A.3 Using Power Commander Mode.....	25

## Smartcanvas DA9061/2

<b>Appendix B : Legacy SmartCanvas ini Files .....</b>	<b>26</b>
<b>Revision History .....</b>	<b>27</b>

## Figures

Figure 1: Front Panel .....	10
Figure 2: GPIO Controls .....	10
Figure 3: Bucks and LDOs .....	11
Figure 4: RTC Time .....	11
Figure 5: INI File Control .....	11
Figure 6: Power Sequencer Controls .....	12
Figure 7: Register View .....	13
Figure 8: Control Widgets .....	14
Figure 9: Raw I/O 'Batch Mode' .....	15
Figure 10: USB_Ports Table View .....	17
Figure 11: ADC Plot Window .....	18
Figure 12: Table View .....	19
Figure 13: OTP Programmer Tool .....	20
Figure 14: Project File Comparison Tool .....	22
Figure 15: Bus Scan Tool .....	23
Figure 16: Search Tool .....	23
Figure 17: General Settings .....	24
Figure 18: Information Message Box .....	26

## Tables

Table 1: Communication Modes .....	15
------------------------------------	----

---

**Smartcanvas DA9061/2**

## 1 Terms and Definitions

DUT	Device under test (i.e. DA9061 or DA9062)
PMIC	Power management IC
GUI	Graphical user interface
ULI	USB Lab IO interface IC (Atmel)
CRC	Cyclic redundancy check

## 2 References

- [1] DA9061 Datasheet, Dialog Semiconductor, 2015
- [2] DA9062 Datasheet, Dialog Semiconductor, 2015
- [3] User Manual: DA9061-63 Motherboard, Dialog Semiconductor, 2015

## Smartcanvas DA9061/2

### 3 Introduction

The GUI (Graphical User Interface) provides easy access to the DUT and the supporting functionality located on the motherboard. The GUI is used to exercise the DUT using the I<sup>2</sup>C interface. Control or measurement of analogue and digital pins is also supported.

The GUI is non-invasive, meaning:

- It will not modify register contents without user interaction
- All ULI activities are stopped when the “Polling Enabled” button is set to “Polling Disabled”.

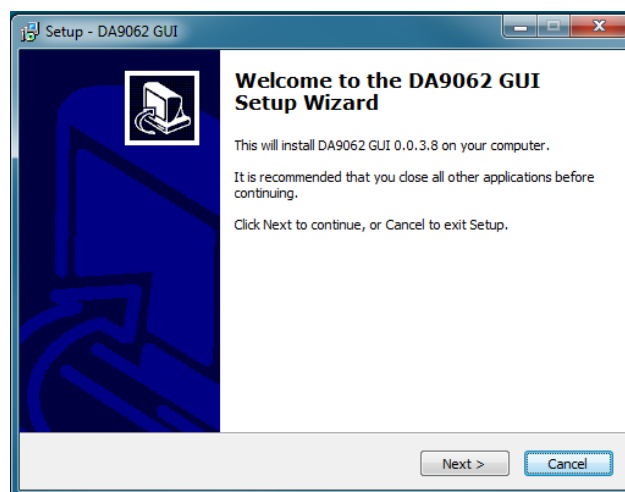
The document applies equally to DA9061 and DA9062. For brevity, some text refers to DA9062, but this is also applicable to DA9061.

The version of this document available from the GUI ‘Help Menu’ may not include *minor* document revisions. The latest manual is available from the Dialog customer portal.

### 4 Installation

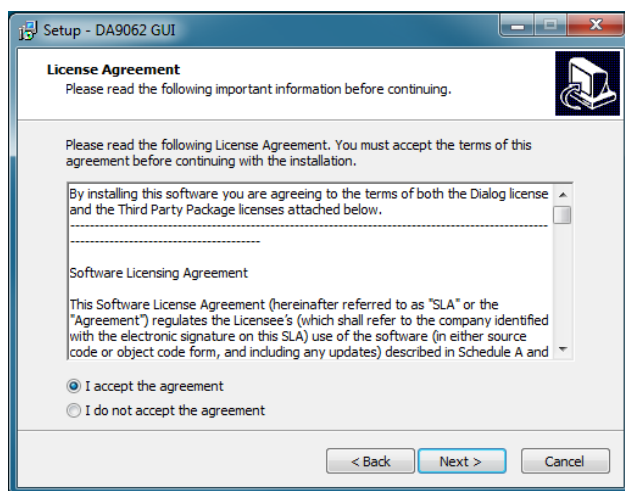
To install the software,

1. Ensure the evaluation board is connected to the PC using a USB cable.
2. Run setup\_DA906x\_GUI\_x.x.exe and click ‘Next’:

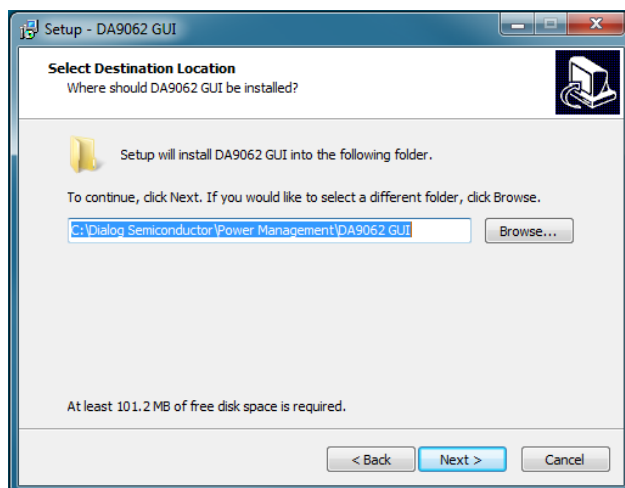


## Smartcanvas DA9061/2

3. Accept the license agreement and click 'Next':

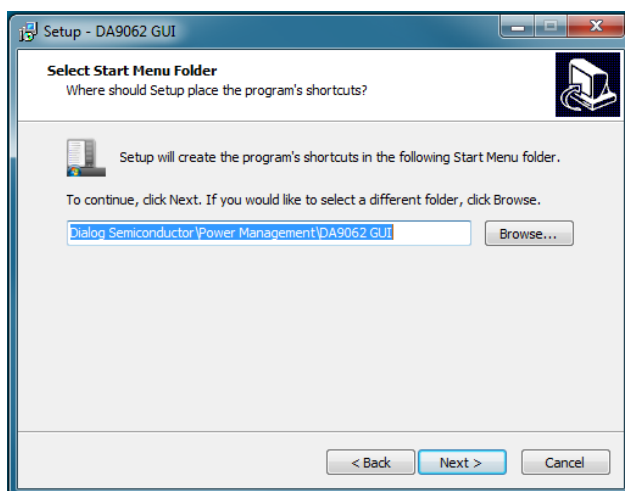


4. Choose the installation path using 'Browse...' or leave the default path and click 'Next':

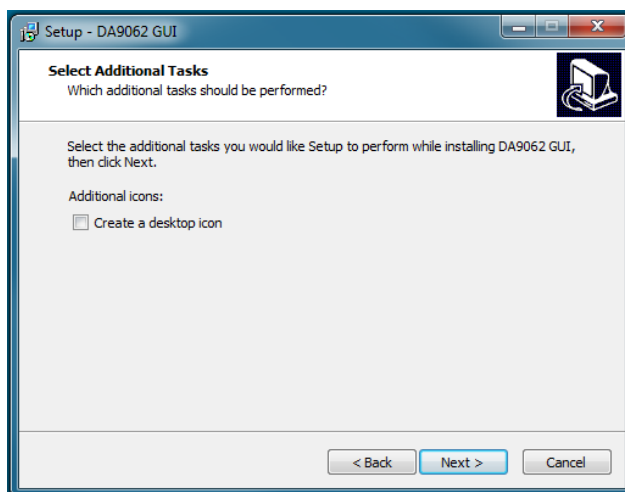


## Smartcanvas DA9061/2

5. Choose the 'Start Menu Folder' using 'Browse...' or leave the default folder and click 'Next':

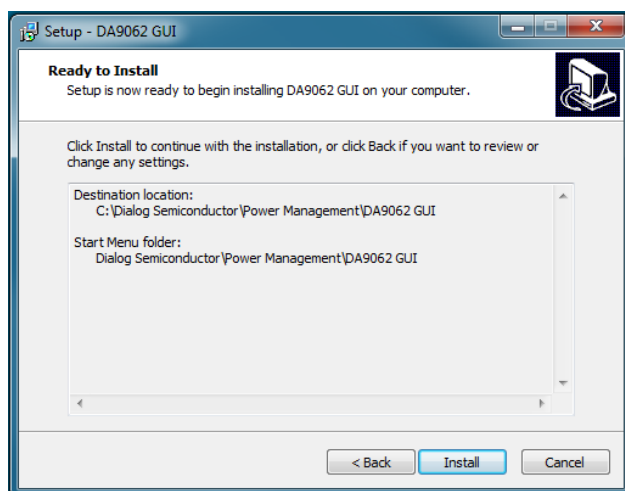


6. Check 'Create a desktop icon' to access the software directly from the desktop and click 'Next':

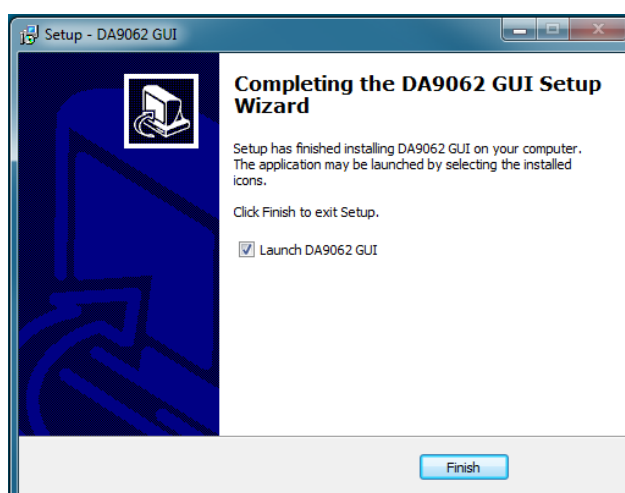


## Smartcanvas DA9061/2

7. Review installation information and click 'Next':



8. Check 'Launch DA906x GUI' if you want to launch the GUI immediately after installation and click 'Finish':



Once connected, the hardware will be detected and the drivers will install automatically. In some operating systems the driver installer window may pop up: accept the recommended settings.



## 5 Power Commander Mode

Power Commander mode ('PC Mode') is a powerful feature of Dialog PMICs that allows prototyping of OTP configurations without having to burn a device's OTP. PC Mode starts a device using the configuration taken from an external .ini file instead of the device's own programmed (or blank) configuration.

To start a device using PC Mode:

1. Select a project file. This can be done from the 'DA9062 INI File Control' section on the 'Front Panel' tab or the right-hand-side control
2. Ensure the motherboard toggle switch is to the left ('TP\_ON')
3. Turn off the device (VSYS = 0 V)
4. Click the 'Power Commander mode' button to turn on the feature.
5. Turn on the PMIC
6. The device should start up.
  - a. Depending on the configuration in the .ini file, the device may go into the 'Power Down' state or 'Active' state.
  - b. To wake-up a device that is in 'Power Down', press the 'Start Device' button (which performs an nONKEY pulse).
7. The device will now have started up using the selected project .ini file content.
8. Leaving PC Mode switched on allows the device to continue to use the external ini file contents when moving between certain states (For example, after performing a SHUTDOWN command, or a further wake-up after entering the Power Down state).

Note: Devices with an OTP programmed to a non-default slave address cannot be used with Power Commander mode.

## 6 GUI Overview

GUI elements (also known as 'widgets') are grouped by functions and separated by tabs as described below.

Some elements, such as the 'Info' and 'Status' widgets, can be undocked from their default locations and resized. Dragging them back to their default locations and holding them there momentarily will automatically create a space for them at which instant they can be dropped and re-docked.

## Smartcanvas DA9061/2

## 6.1 Front Panel

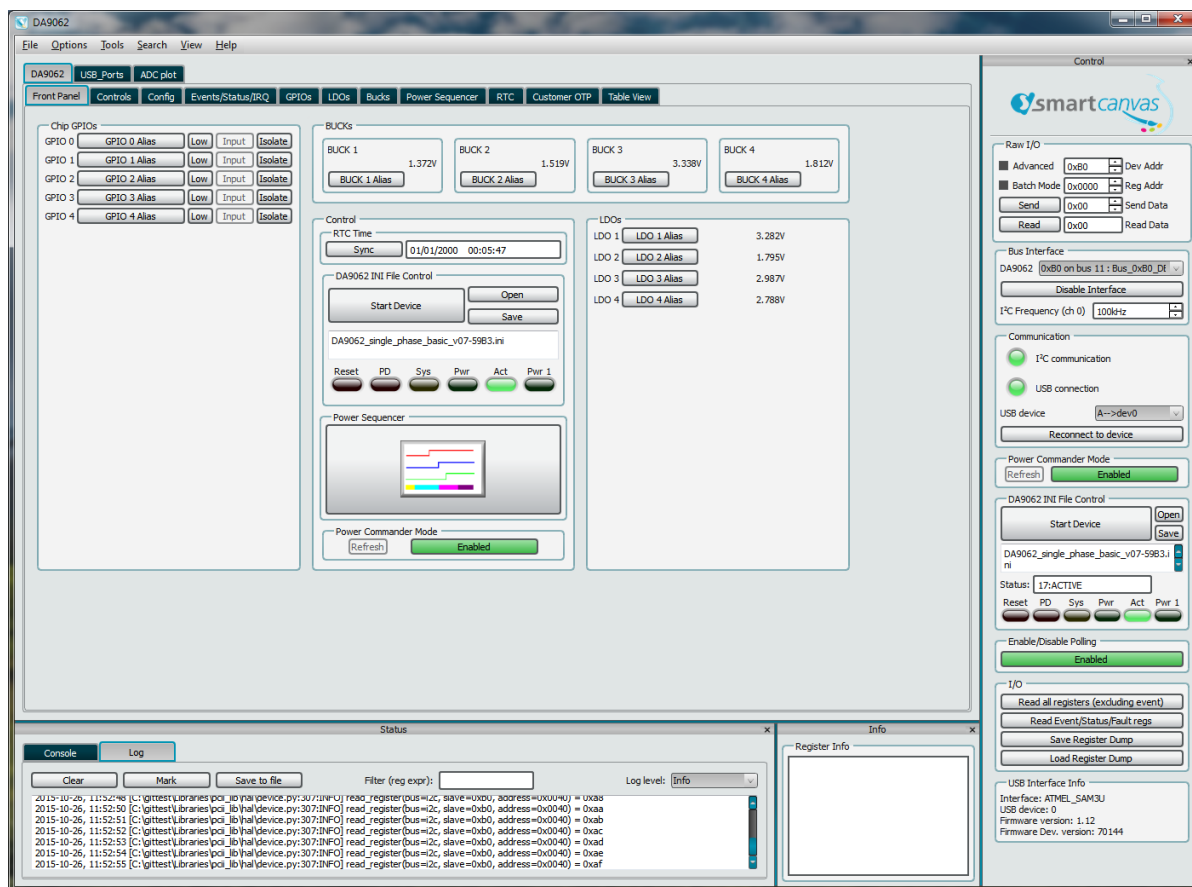


Figure 1: Front Panel

The 'Front Panel' (Figure 1) shows an overview of the DUT's GPIO, LDO and Buck settings; the aliases associated with this project file (.ini format); an RTC that can be synced to the computer's system time and the Power Sequencer window.

## 6.1.1 GPIOs

The 'Chip GPIOs' (Figure 2) can be used to report the level on DUT GPIO pins configured as outputs, or to drive a level on those DUT GPIOs that are configured as inputs:

- When set to 'Input', the 'High/Low' button will be available and drives either a logic '1' or '0' onto the DUT input.
- When set to 'Output', the 'High/Low' is an indicator that reports the level being output from the DUT.
- The 'Isolate' button puts the ULI pin into high-Z state so that it does not interfere with the DUT behaviour. Note that this button will override the other settings in this control.



Figure 2: GPIO Controls

## Smartcanvas DA9061/2

Smartcanvas automatically sets the directions for these controls. The configuration is based on the powered DUT register settings. If the DUT is not powered, then the settings are obtained from the project ini file. These 'Chip GPIOs' settings are updated on-the-fly when the GUI is used to change the DUT GPIO register settings.

### 6.1.2 Bucks and LDOs

The Bucks and LDOs sections on the Front Panel show the aliases associated with these regulators and, with polling enabled, display a live ADC measurement of their output voltages.

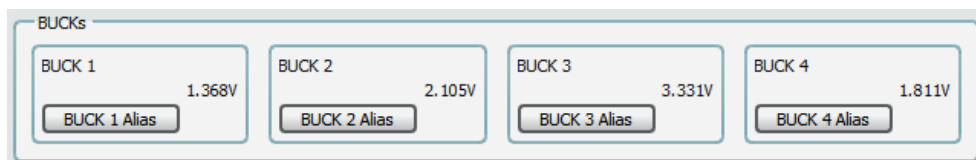


Figure 3: Bucks and LDOs

### 6.1.3 RTC Time

The RTC on the DUT can be synchronised to the System Clock of the PC controlling the Smartcanvas software. If polling is enabled, the time updates periodically and continues to reflect the System Clock (Figure 4).

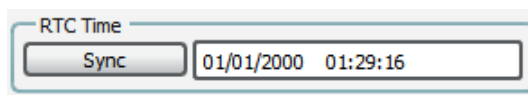


Figure 4: RTC Time

### 6.1.4 INI File Control

The 'Start Device' button triggers a wake-up event by toggling the nONKEY signal low for a short duration. This wake-up event can be used while using Power Commander mode to progress the DUT state from Power Down to Active, as illustrated by 'PD' and 'Act', respectively, in the series of LED-type indicators along the bottom of this control (Figure 5).

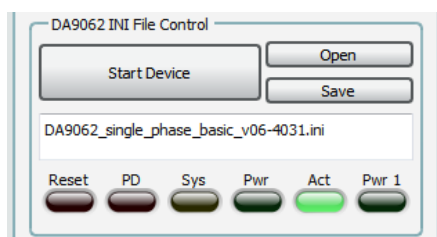


Figure 5: INI File Control

'Open' chooses the project file to be used by Power Commander mode.

#### NOTE

ini files created by the old 'LabPy' GUI prior to version 1.2 will not load into Smartcanvas. Such files need to be opened in a later version of the LabPy GUI such as 1v42 and resaved in the newer format.

#### NOTE

When opening an ini file, if a message box is shown stating, '*The file being loaded appears to have been created using a different version of the GUI*', then simply clicking 'OK' to continue will ensure the correct behaviour. See [Appendix B](#).

## Smartcanvas DA9061/2

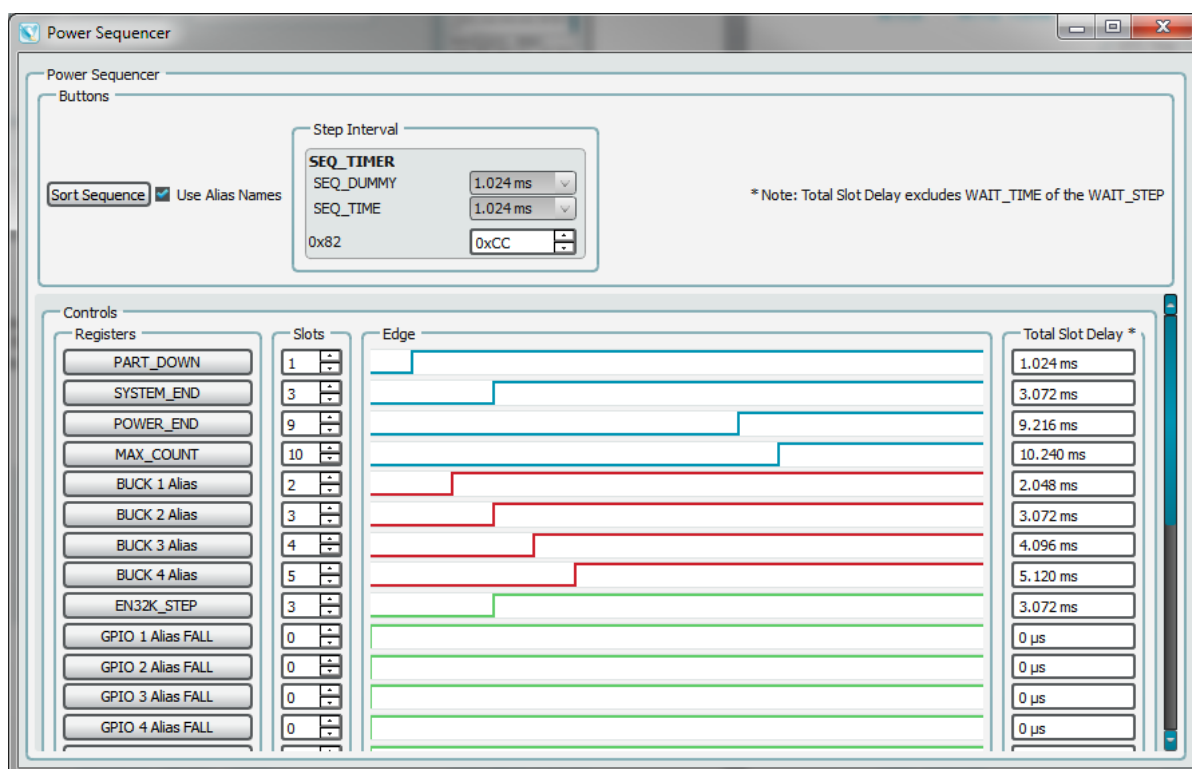
'Save' is used to create a new project file. The current state of the applicable registers in the device denotes the contents of this newly created .ini file.

The INI file controls are duplicated in the widget on the right side so that they are visible when displaying tabs other than the Front Panel.

### 6.1.5 Power Sequencer

The Power Sequencer window (Figure 6) can be used to define the boot-up sequence. Edges can be placed by clicking where an edge is desired, or by entering the slot number in the spin-box.

Within the 'Buttons' section at the top, 'Sort Sequence' re-orders the regulators into ascending slot order.



**Figure 6: Power Sequencer Controls**

The check box 'Use Alias Names' toggles the display between DUT default regulator names and those aliases entered in the Front Panel.

## Smartcanvas DA9061/2

## 6.2 Register View

The register view presents groups of registers, each group selected using tabs along the top. For example, [Figure 7](#) shows the 'Controls' tab. (The tabs down the left side are not used for DA9062, but allow for future GUI expansion).

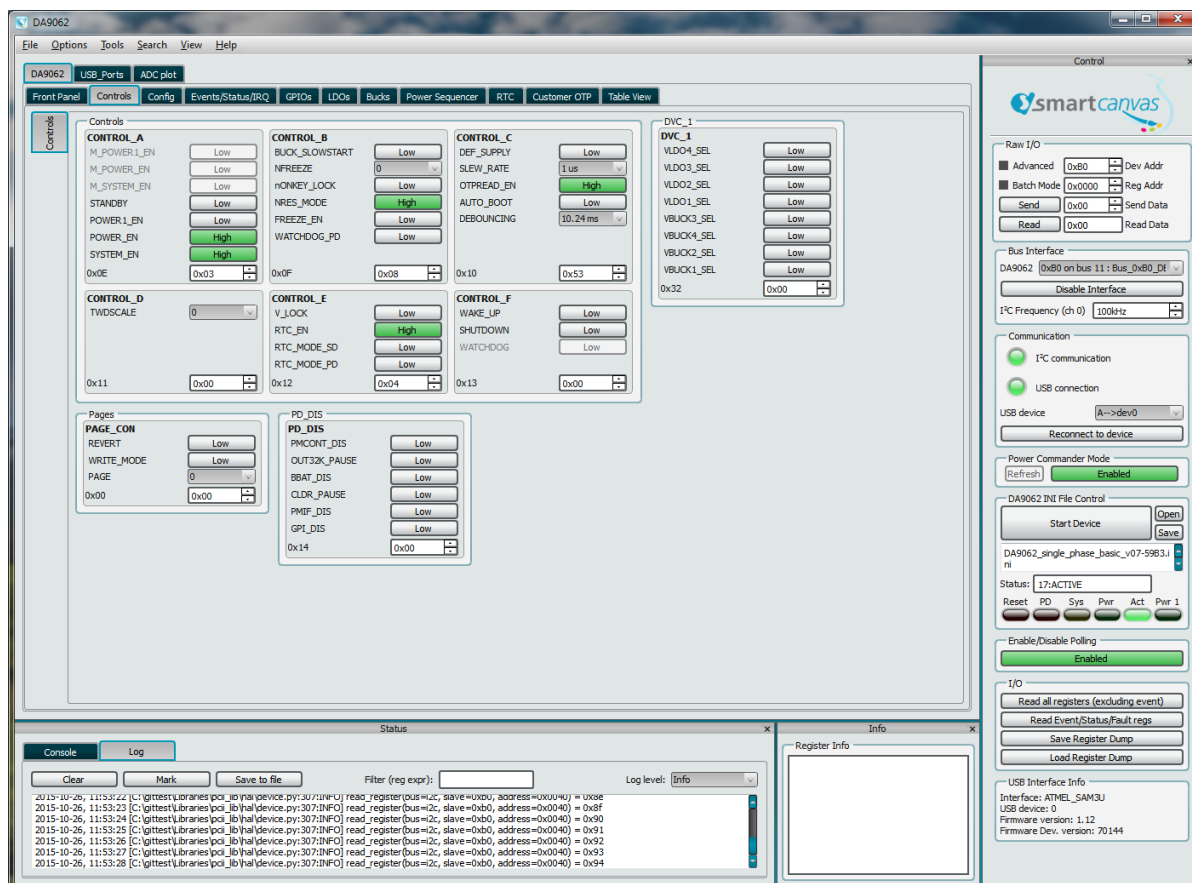


Figure 7: Register View

Greyed-out buttons represent a read-only bit field; drop-down selections represent multiple bit fields; Boolean buttons represent single bit fields. With polling set to 'enabled' on the right-hand side control, these registers are continuously updated from the DUT registers (as long as I<sup>2</sup>C communication is available). If working offline (for example, running the GUI with no hardware connected), the register changes are stored locally within the GUI. These values can then be saved to an .ini file for use later using the OTP Programmer tool (see [Section 7](#)).

Hovering the mouse over a control displays the associated help text in the Info window.

## Smartcanvas DA9061/2

### 6.3 Control Widgets

The right side panel contains 'Control' widgets (Figure 8). This contains features related to IC communications, I<sup>2</sup>C communications, file manipulation and miscellaneous I/O control.

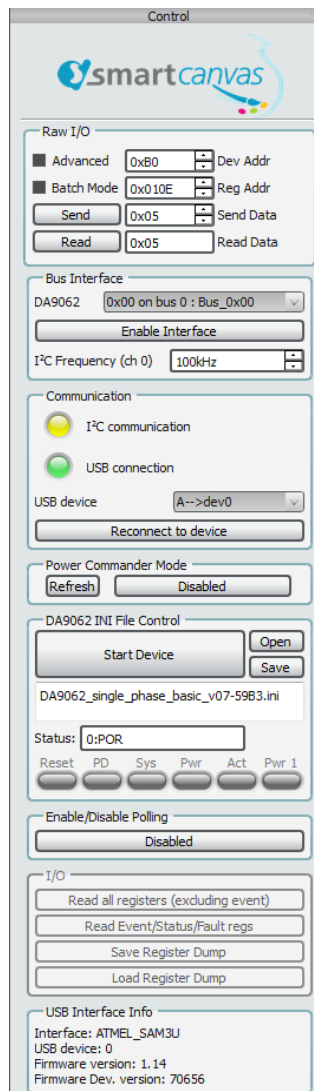


Figure 8: Control Widgets

#### 6.3.1 Raw I/O

This widget allows low-level access to the I<sup>2</sup>C bus (Figure 9). However, for convenience of writing to all pages of the device, the 'Reg Addr' field manages the page address automatically. For example, to read from address 0x107, set 'Reg Addr' = 0x0107: it is not necessary to manually write a 0x2 to the PAGE\_CON register at address 0x00 followed by a read from address 0x07.

The 'Advanced' option allows control of the base address. 'Batch Mode' allows multiple reads and writes (Figure 9).

The syntax for a write is: WRITE [base\_address] [register\_address] [data]

The syntax for a read is: READ [base\_address] [register\_address]

The results can be traced in the status window on the 'Log' tab (Log level = 'Info').

Smartcanvas DA9061/2

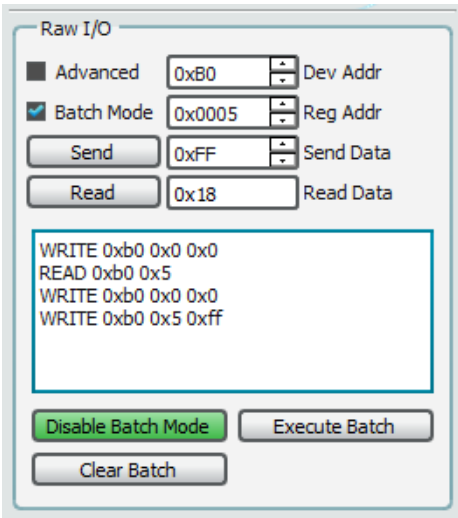


Figure 9: Raw I/O ‘Batch Mode’

6.3.2 Bus Interface

This control allows selection of the DUT base address. The default for DA9061/2 is 0xB0 and is marked as ‘DEFAULT’ in the drop-down list.

The interface can be completely disabled using the ‘Disable Interface’ button. However, the GUI observes the bus and highlights that a device is available for communications by flashing the ‘I<sup>2</sup>C communication’ LED yellow (See Section 6.3.3).

The I<sup>2</sup>C frequency can be selected. 100 kHz is sufficient for normal GUI use.

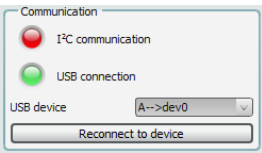
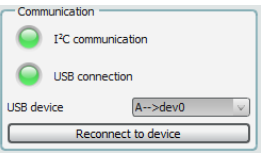
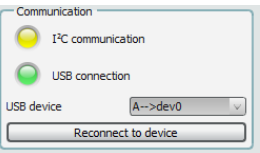
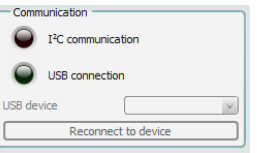
6.3.3 Communication

Two LEDs show the status of the bus and the GUI. There are several modes of the GUI-to-DUT interface, as illustrated in Table 1.

It is recommended to work in ‘Offline Mode’ when making changes to an ini file. See Appendix A.1.

The ‘USB device’ selector is not used for the DA9062 GUI. It is only used when there are multiple devices available on the I<sup>2</sup>C bus.

Table 1: Communication Modes

 <p>No DUT comms on I<sup>2</sup>C</p> <p>Eval board connected</p>	 <p>DUT comms on I<sup>2</sup>C</p> <p>Eval board connected</p>	 <p>‘Offline Mode’: DUT comms available but disabled</p> <p>Eval board connected</p>	 <p>‘Offline Mode’: DUT comms unavailable</p> <p>Eval board disconnected</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------

6.3.4 Power Commander Mode

Please refer to Section 5.

6.3.5 ini file Control

See Section 6.1.4.

---

**Smartcanvas DA9061/2****6.3.6 Polling**

The 'Enable/Disable Polling' control is the same as the one accessible from the GUI Options Settings menu. Disabling polling can be useful to silence the bus. This allows direct user control. For example, the 'Raw I/O' controls can be used to issue specific I<sup>2</sup>C commands. See Section 9.3.2 for further details.

**6.3.7 I/O**

The I/O buttons provide miscellaneous tools. Reading all device registers is useful when polling is disabled.

The 'Register Dump' buttons allow a simple transfer of register values to or from a text file.



## Smartcanvas DA9061/2

## 6.4 USB Ports Table

The USB ports table presents a low-level view of the motherboard SAM3U controller configuration. The SAM3U port name is shown in the left column. The columns on the right show the evaluation motherboard line that each SAM3U port is connected to. Double-clicking the names provides additional information, as shown in Figure 10 for PB25\_ONKEY. This configuration is set by the GUI and is read-only for the user.

	Data	Bit 3	Bit 2	Bit 1	Bit 0
0x00: PA2_ULI_ALIVE_LED	0x3	PA2_ULI_ALIVE_LED_MODE: 0x00	PA2_ULI_ALIVE_L...	PA2_ULI_ALIVE_L...	
0x01: PA23_PC_TM_EN	0x1	PA23_PC_TM_EN_MODE: 0x00	PA23_PC_TM_EN...	PA23_PC_TM_EN...	
0x02: PB15_GPIO_7	0x6	PB15_GPIO_7_MODE: 0x01	PB15_GPIO_7_STA...	PB15_GPIO_7_DIR...	
0x03: PB18_GPIO_10	0x6	PB18_GPIO_10_MODE: 0x01	PB18_GPIO_10_ST...	PB18_GPIO_10_DI...	
0x04: PB20_GPIO_12	0x6	PB20_GPIO_12_MODE: 0x01	PB20_GPIO_12_ST...	PB20_GPIO_12_DI...	
0x05: PA31_ADC_4	0x3	PA31_ADC_4_MODE: 0x00	PA31_ADC_4_STATE: 0x01	PA31_ADC_4_DIRE...	
0x06: PB11_GPIO_3	0x3	PB11_GPIO_3_MODE: 0x00	PB11_GPIO_3_STA...	PB11_GPIO_3_DIR...	
0x07: PB17_GPIO_9	0x6	PB17_GPIO_9_MODE: 0x01	PB17_GPIO_9_STA...	PB17_GPIO_9_DIR...	
0x08: PB16_GPIO_8	0x6	PB16_GPIO_8_MODE: 0x01	PB16_GPIO_8_STA...	PB16_GPIO_8_DIR...	
0x09: PB25_nONKEY	0x3	PB25_nONKEY_MO...	MODE Floating	PB25_nONKEY_ST...	PB25_nONKEY_DI...
0x0A: PA22_OTP_PROG_EN	0x1	PA22_OTP_PROG_EN_MODE: 0x00	PA22_OTP_PROG...	PA22_OTP_PROG...	
0x0B: PB9_GPIO_1	0x3	PB9_GPIO_1_MODE: 0x00	PB9_GPIO_1_STATE: 0x01	PB9_GPIO_1_DIR...	
0x0C: PB2_TP	0x0	PB2_TP_MODE: 0x00	PB2_TP_STATE: 0x00	PB2_TP_DIRECTION: 0x00	
0x0D: PB26_NOFF_USB	0xB	PB26_NOFF_USB_MODE: 0x02	PB26_NOFF_USB...	PB26_NOFF_USB...	
0x0E: PB23_GPIO_FB_3	0x6	PB23_GPIO_FB_3_MODE: 0x01	PB23_GPIO_FB_3...	PB23_GPIO_FB_3...	
0x0F: PB22_GPIO_FB_2	0x6	PB22_GPIO_FB_2_MODE: 0x01	PB22_GPIO_FB_2...	PB22_GPIO_FB_2...	
0x10: PB27_LV_PROG_ENA	0x3	PB27_LV_PROG_ENA_MODE: 0x00	PB27_LV_PROG_E...	PB27_LV_PROG_E...	
0x11: PB21_GPIO_13	0x6	PB21_GPIO_13_MODE: 0x01	PB21_GPIO_13_ST...	PB21_GPIO_13_DI...	
0x12: PA27_MUX_EN	0x1	PA27_MUX_EN_MODE: 0x00	PA27_MUX_EN_STA...	PA27_MUX_EN_DI...	
0x13: PB14_GPIO_6	0x6	PB14_GPIO_6_MODE: 0x01	PB14_GPIO_6_STA...	PB14_GPIO_6_DIR...	
0x14: PB13_GPIO_5	0x6	PB13_GPIO_5_MODE: 0x01	PB13_GPIO_5_STA...	PB13_GPIO_5_DIR...	
0x15: PA28_ADC_1	0x1	PA28_ADC_1_MODE: 0x00	PA28_ADC_1_STATE: 0x00	PA28_ADC_1_DIRE...	
0x16: PB10_GPIO_2	0x3	PB10_GPIO_2_MODE: 0x00	PB10_GPIO_2_STA...	PB10_GPIO_2_DIR...	
0x17: PB24_nSHUTDOWN	0x3	PB24_nSHUTDOWN_MODE: 0x00	PB24_nSHUTDOWN...	PB24_nSHUTDOWN...	
0x18: PA30_ADC_3	0x1	PA30_ADC_3_MODE: 0x00	PA30_ADC_3_STATE: 0x00	PA30_ADC_3_DIRE...	

Figure 10: USB\_Ports Table View

## Smartcanvas DA9061/2

## 6.5 ADC Plot

The ADC plot tab provides a utility to plot the PMIC regulator voltages (see Figure 11).

The regulator voltages to be plotted are selected using the check boxes next to the regulator names.

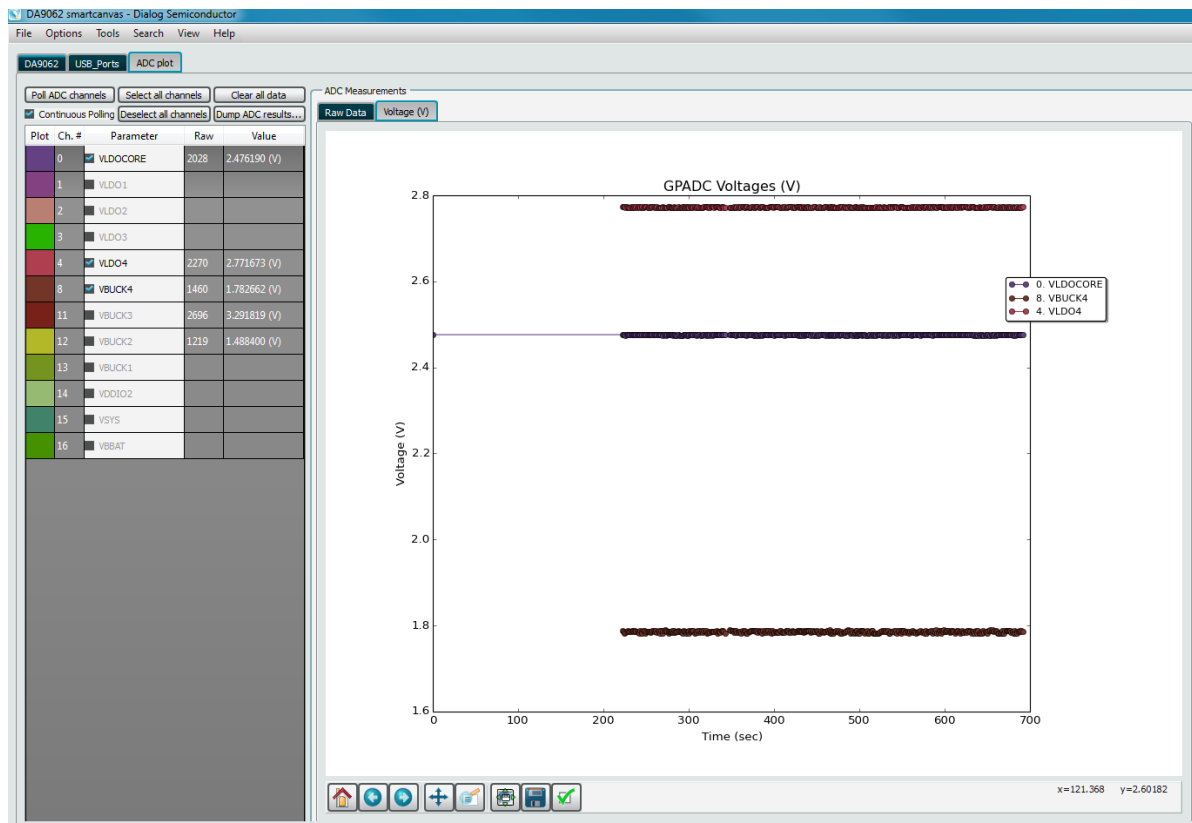


Figure 11: ADC Plot Window

The utility is controlled using the following buttons:

- 'Poll ADC channels': performs a single snap-shot measurement of the selected voltages. (Greyed-out if no voltages are selected)
- 'Continuous Polling' check-box: enables plotting of each voltage versus time. (This polling uses a separate I<sup>2</sup>C bus on the motherboard and therefore can be operated independently from the PMIC I<sup>2</sup>C bus.)
- 'Dump ADC results...': saves data to a csv file.

The 'ADC measurements' window has two tabs. The 'Raw Data' tab plots the raw ADC register value on the graph y-axis. The 'Voltage (V)' tab displays the readouts on the y-axis as voltages.

A set of buttons below the graph provides control of the display and the data. Mouse-over the buttons to show context help for each button.

## Smartcanvas DA9061/2

## 6.6 Table View

The 'Table View' tab (Figure 12) displays all the registers on the DUT. The colours only denote difference in bit fields, with grey being the usual colour for 'unused' sections. Double-clicking on a bit field brings up its control, allowing the user to change the value for writable registers.

DA9062

FileOptionsToolsSearchViewHelp

DA9062USB\_Ports

BucksConfigControlsCustomer OTPEvents/Status/IRQGPIOsLDOsPower SequencerRTCTable ViewFront PanelGPADC plot

	Data	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00: PAGE_CON	0x02	REVERT_I7: 0x00	WRITE_MODE_I8: 0x00	PAGE_I0: 0x02					
0x01: STATUS_A	0x00	unused					DVC_BUSY_I2: 0x00	unused	NONKEY_I0: 0x00
0x02: STATUS_B	0x00	unused				GPI4_I4: 0x00	GPI3_I3: 0x00	GPI2_I2: 0x00	GPI1_I1: 0x00
0x04: STATUS_D	0x00	unused				LDO4_ILIM_I3: 0x00	LDO3_ILIM_I2: 0x00	LDO2_ILIM_I1: 0x00	LDO1_ILIM_I0: 0x00
0x05: FAULT_LOG	0x02	WAIT_SHUT_I7: 0x00	NRESETREQ_I8: 0x00	KEY_RESET_I5: 0x00	TEMP_CRIT_I4: 0x00	VDD_START_I3: 0x00	VDD_FAULT_I2: 0x00	POR_I1: 0x01	TWD_ERROR_I0: 0x00
0x06: EVENT_A	0x00	unused	EVENTS_C_I8: 0x00	EVENTS_B_I5: 0x00	E_SEQ_RDY_I4: 0x00	E_WDG_WARN_I3: 0x00	E_TICK_I2: 0x00	E_ALARM_I1: 0x00	E_NONKEY_I0: 0x00
0x07: EVENT_B	0x00	E_VDD_WARN_I7: 0x00	unused	E_DVC_RDY_I5: 0x00	unused	E_LDO_LIM_I3: 0x00	unused	E_TEMP_I1: 0x00	unused
0x08: EVENT_C	0x00	unused				E_GPI4_I4: 0x00	E_GPI3_I3: 0x00	E_GPI2_I2: 0x00	E_GPI1_I1: 0x00
0x0A: IRQ_MASK_A	0x00	unused				M_SEQ_RDY_I4: 0x00	M_WDG_WARN_I3: 0x00	M_TICK_I2: 0x00	M_ALARM_I1: 0x00
0x0B: IRQ_MASK_B	0x00	M_VDD_WARN_I7: 0x00	unused	M_DVC_RDY_I5: 0x00	unused	M_LDO_LIM_I3: 0x00	unused	M_TEMP_I1: 0x00	unused
0x0C: IRQ_MASK_C	0x00	unused				M_GPI4_I4: 0x00	M_GPI3_I3: 0x00	M_GPI2_I2: 0x00	M_GPI1_I1: 0x00
0x0E: CONTROL_A	0x00	unused	M_POWER1_EN_I8: 0x00	M_POWER_EN_I5: 0x00	M_SYSTEM_EN_I4: 0x00	STANDBY_I3: 0x00	POWER1_EN_I2: 0x00	POWER_EN_I1: 0x00	SYSTEM_EN_I0: 0x00
0x0F: CONTROL_B	0x18	BUCK_SLOWSTAR...	NFREEZE_I5: 0x00	nONKEY_LOCK_I4: 0x01	NRES_MODE_I3: 0x01	AUTO_BOOT_I3: 0x00	FREEZE_EN_I2: 0x00	WATCHDOG_PD_I1: 0x00	unused
0x10: CONTROL_C	0x53	DEF_SUPPLY_I7: 0x00	SLEW_RATE_I5: 0x02	OTPREAD_EN_I4: 0x01	DEBOUNCING_I0: 0x03				
0x11: CONTROL_D	0x00	unused					TWDSCALE_I0: 0x00		
0x12: CONTROL_E	0x02	V_LOCK_I7: 0x00	unused				RTC_EN_I2: 0x00	RTC_MODE_SD_I1: 0x01	RTC_MODE_PD_I0: 0x00
0x13: CONTROL_F	0x00	unused					WAKE_UP_I2: 0x00	SHUTDOWN_I1: 0x00	WATCHDOG_I0: 0x00
0x14: PD_DIS	0x00	PMCONT_DIS_I7: 0x00	OUT32K_PAUSE_I8: 0x00	BBAT_DIS_I5: 0x00	CLDR_PAUSE_I4: 0x00	unused	PMIF_DIS_I2: 0x00	unused	GPI_DIS_I0: 0x00
0x15: GPIO_0_1	0x55	GPIO1_WEN_I7: 0x00	GPIO1_TYPE_I8: 0x01	GPIO1_PIN_I4: 0x01		GPIO0_WEN_I3: 0x00	GPIO0_TYPE_I2: 0x01	GPIO0_PIN_I0: 0x01	
0x16: GPIO_2_3	0x55	GPIO3_WEN_I7: 0x00	GPIO3_TYPE_I8: 0x01	GPIO3_PIN_I4: 0x01		GPIO2_WEN_I3: 0x00	GPIO2_TYPE_I2: 0x01	GPIO2_PIN_I0: 0x01	
0x17: GPIO_4	0x05	unused				GPIO4_WEN_I3: 0x00	GPIO4_TYPE_I2: 0x01	GPIO4_PIN_I0: 0x01	
0x1C: GPIO_WKUP_MODE	0x00	unused				GPIO4_WKUP_MO... GPIO4_MODE_I4: 0x00	GPIO3_WKUP_MO... GPIO3_MODE_I3: 0x00	GPIO2_WKUP_MO... GPIO2_MODE_I2: 0x00	GPIO1_WKUP_MO... GPIO1_MODE_I1: 0x00
0x1D: GPIO_MODE0_4	0x00	unused				GPIO4_WKUP_MO... GPIO4_MODE_I4: 0x00	GPIO3_WKUP_MO... GPIO3_MODE_I3: 0x00	GPIO2_WKUP_MO... GPIO2_MODE_I2: 0x00	GPIO1_WKUP_MO... GPIO1_MODE_I1: 0x00
0x1E: GPIO_OUT0_2	0x00	GPIO2_OUT_I8: 0x00		GPIO1_OUT_I3: 0x00			GPIO0_OUT_I0: 0x00		
0x1F: GPIO_OUT3_4	0x00	unused				GPIO4_OUT_I3: 0x00		GPIO3_OUT_I0: 0x00	
0x20: BUCK2_CONT	0x00	unused	VBUCK2_GPI_I5: 0x00	unused	BUCK2_CONF_I3: 0x00	BUCK2_GPI_I1: 0x00		BUCK2_EN_I0: 0x00	
0x21: BUCK1_CONT	0x00	unused	VBUCK1_GPI_I5: 0x00	unused	BUCK1_CONF_I3: 0x00	BUCK1_GPI_I1: 0x00		BUCK1_EN_I0: 0x00	
0x22: BUCK4_CONT	0x00	unused	VBUCK4_GPI_I5: 0x00	unused	BUCK4_CONF_I3: 0x00	BUCK4_GPI_I1: 0x00		BUCK4_EN_I0: 0x00	

Figure 12: Table View

## Smartcanvas DA9061/2

## 7 OTP Programmer Tool

The OTP Programmer tool (see [Figure 13](#)) is used to program a configuration into a device. It is also used to prepare the configuration, and allows manipulation of register settings.

The three basic steps for programming are outlined in the 'Info' area at the top left corner of the window.

Register values can be loaded from an ini file, a device's OTP or from current register settings.

These can then be manipulated, if required. Programming a device is started using the button at the top right of the window.

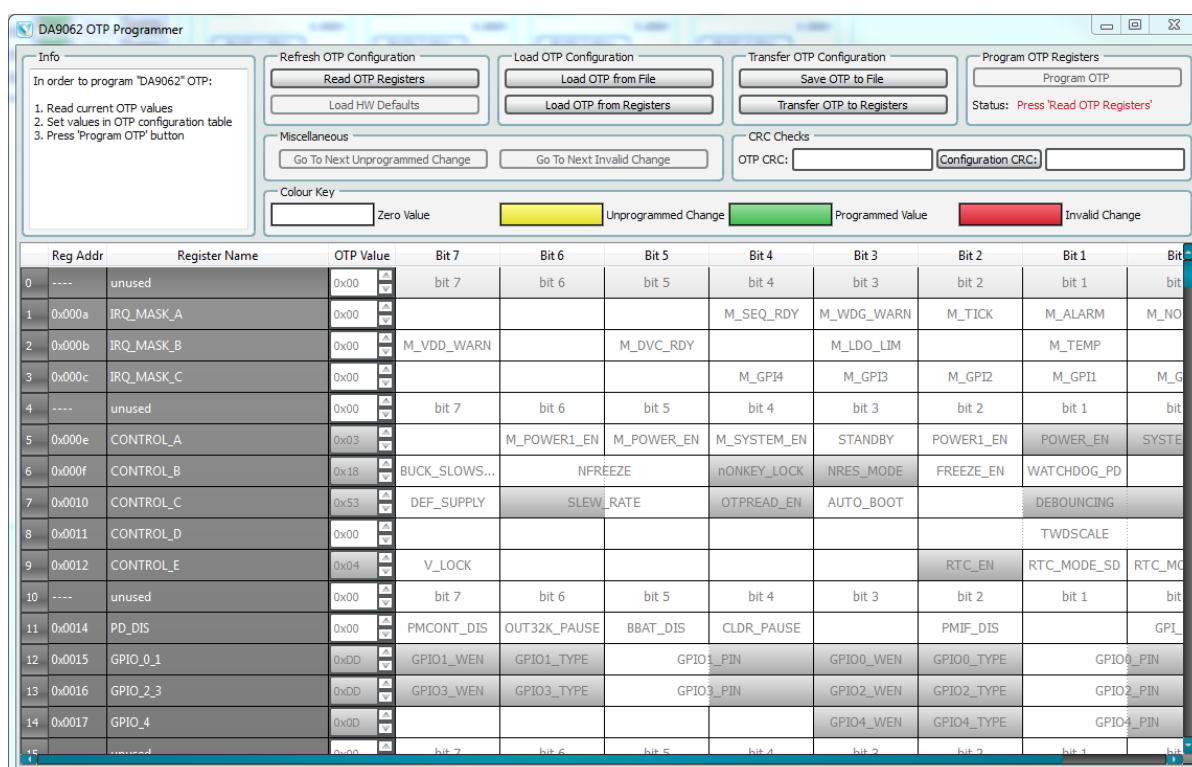


Figure 13: OTP Programmer Tool

## Smartcanvas DA9061/2

The OTP Programmer tool provides a visual comparison of the present OTP content of a DUT and the target configuration. The register table shows the results of this comparison and is colour-coded using the key, which is further explained below:

Unused	The bit has no function or is reserved
Zero Value	Bit is to be set to '0'.
Unprogrammed Change	The bit is to be set to '1' but it is presently unprogrammed in the DUT.
Programmed Value	The bit is to be set to '1' and is already programmed in the DUT
Invalid Change	The bit is to be set to '0' but the bit in the DUT has previously been programmed to a '1'. Therefore, this bit cannot be correctly configured.

### NOTE

It is not possible to program the device if there are one or more bits flagged as 'Invalid Change'.

The CRCs provide an important tool for checking the configuration. Dialog uses these CRCs for traceability of configuration files. A correctly-programmed device always returns the same CRC as that contained in the corresponding configuration filename. If a different CRC is returned, then a problem has occurred.

## 8 Project File Comparison Tool

The project file comparison tool (see [Figure 14](#)) is accessed under 'Tools' on the menu bar. It allows two sets of register values to be compared. This is particularly useful when checking a set of changes. Values can be loaded from a file, from a device OTP, or from the device registers. Differences are highlighted in red. Matching values are shown as white ('0' bits) or green ('1' bits).

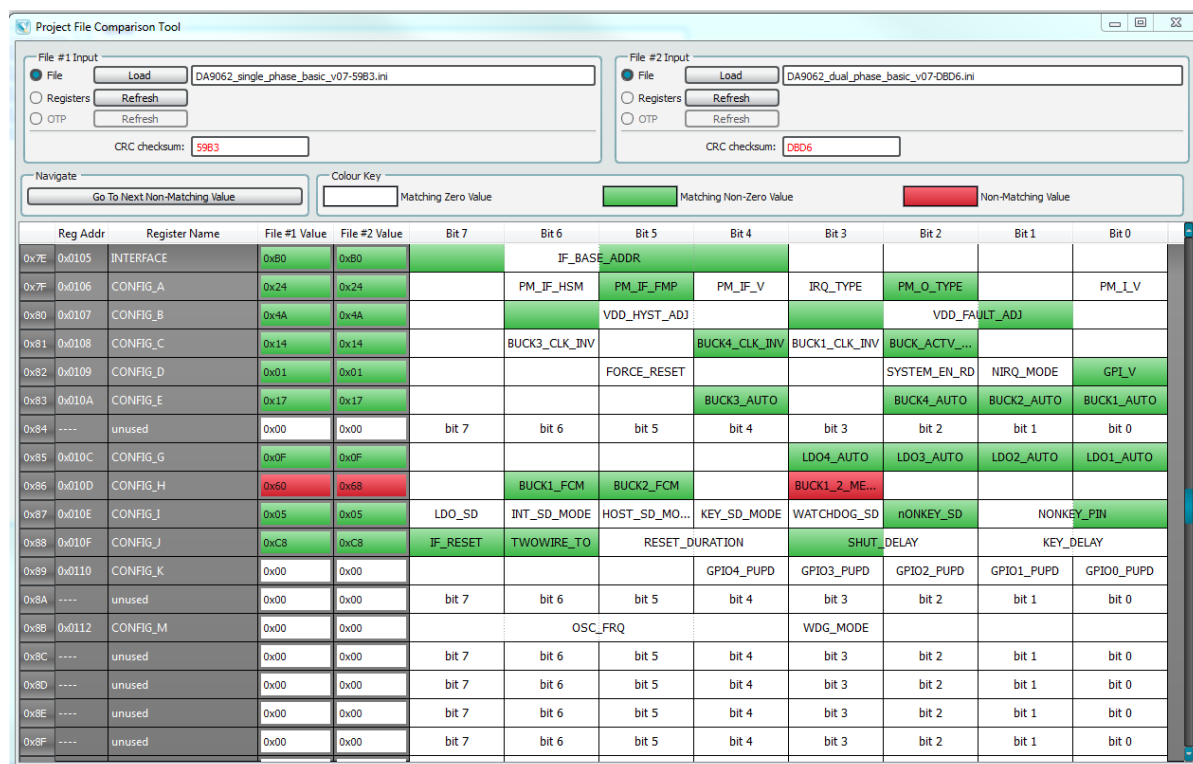


Figure 14: Project File Comparison Tool

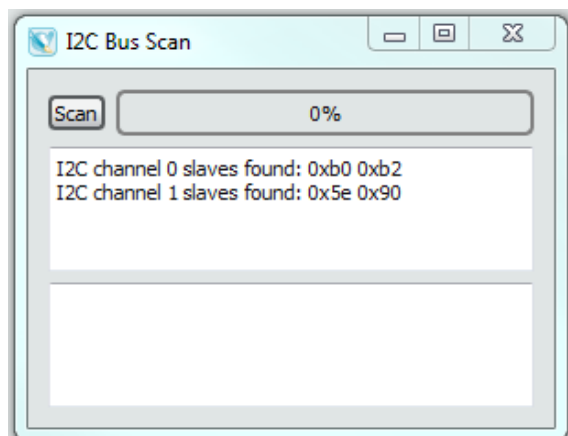
## Smartcanvas DA9061/2

## 9 Menu Bar

The menu bar provides access to the OTP programmer tool ([section 7](#)), project file comparison tool ([section 8](#)) and other miscellaneous features.

### 9.1 Scan I<sup>2</sup>C

This tool scans all 128 I<sup>2</sup>C addresses (0x00 to 0xFE; read and write). Responding slaves are reported by the tool ([Figure 15](#)).



**Figure 15: Bus Scan Tool**

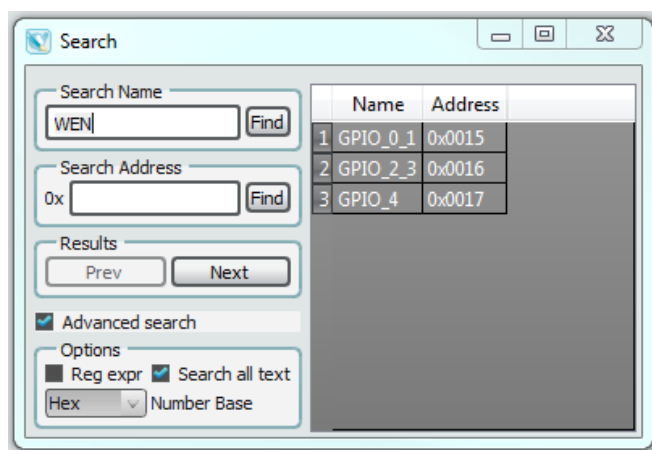
The DA9062 is usually accessible via its default address of 0xB0. Address 0xB2 provides direct access to Page 2 (register address range 0x100 to 0x1FF) without having to write to the PAGE\_CON register.

The DA9061/2 motherboard includes ADCs for regulator voltage readouts. These ADCs are at fixed I<sup>2</sup>C addresses of 0x5E and 0x90.

### 9.2 Search

The search tool provides a fast method of finding a register. It performs dynamic string-matching to show all register names that contain the typed string. The advanced search option enables the search of control names as well as register names (see [Figure 16](#)). Double-clicking on a search result displays the register in the main window.

The tool can also be opened using CTRL+F.

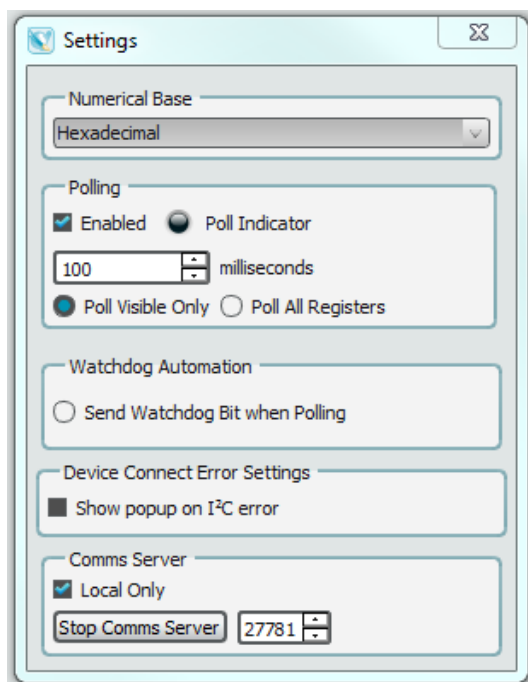


**Figure 16: Search Tool**

## Smartcanvas DA9061/2

### 9.3 Settings

General settings are accessible via the menu bar under Options > Settings (see [Figure 17](#)).



**Figure 17: General Settings**

#### 9.3.1 Number Base

The default number base is hexadecimal. Binary is occasionally useful for interpreting bit fields.

#### 9.3.2 Polling

The polling 'enabled' control is the same as on the GUI main menu on the right-hand side. Disabling polling can be useful to silence the bus. This allows direct user control. For example, the 'Raw I/O' controls can be used to issue specific I<sup>2</sup>C commands.

It is usually necessary only to poll the registers that are currently visible in the tab being displayed. This is the default setting of 'Poll Visible Only'. The alternate option is to read all registers. The frequency of polling can also be set. Polling every 100ms is usually adequate.

#### 9.3.3 Watchdog Automation

When a device has the watchdog feature enabled (TWDS<sub>SCALE</sub> ≠ 0), it is necessary for the system to periodically write to the WATCHDOG bit. The GUI can perform these periodic writes by using the Watchdog Automation feature (enabling 'Send Watchdog Bit when Polling').

This feature relies on polling being enabled and in order for the device not to enter POWERDOWN, the polling must be set more than tWDMIN as per the device specification.

#### 9.3.4 Device Connect Error Settings

Setting 'show popup' displays message boxes when there is an I<sup>2</sup>C error.

#### 9.3.5 Comms Server

The Comms Server allows connection to the GUI over TCP/IP connection and is mainly for internal usage.



## Appendix A : Quick-Start Guides

### A.1 Modifying an .ini File

1. **Turn off** the device or disable the interface by clicking the button in the 'Bus Interface' widget. *It is important to work without a device powered up or with the GUI in 'Offline Mode'. This ensures the behaviour of the DUT does not automatically change any of the register values.*
2. Always start from a known-good Dialog .ini file.
3. From the 'INI File Control' widget, select 'Open' and choose the file to load.
4. The configuration can now be changed via the register values in the GUI tabs.
5. Write the new configuration to a file using the 'Save' button in the 'INI File Control' widget. Note that the new CRC is automatically appended. Dialog recommends leaving this CRC at the end of the filename.

### A.2 OTP Programming Procedure

The following is a basic guide to programming a device:

1. Ensure the device is turned off and the motherboard toggle switch is to the left ('TP\_ON').
2. Insert an unprogrammed (-00 blank) device into the socket.
3. On the GUI Front Panel, enable Power Commander mode.
4. Turn on the PMIC.
5. Open the 'OTP Programmer Tool' from the menu bar (under 'Tools').
6. Update the GUI's table from the OTP by clicking the 'Load from device OTP' button
7. Open the target ini file using the 'Load from ini file' button.
8. Check there are no 'Invalid Changes' highlighted (This is unnecessary with a blank device).
9. Click the 'Program OTP' button.
10. After a short delay, check that the GUI reports 'OTP matches table configuration' (below the 'Program OTP' button).
11. Verify that the 'Read OTP CRC' matches the 'Table CRC'.

### A.3 Using Power Commander Mode

Power Commander mode allows a device to be started using the configuration taken from an external ini file instead of the device's own programmed (or blank) configuration. To start a device in Power Command mode ('PC Mode'):

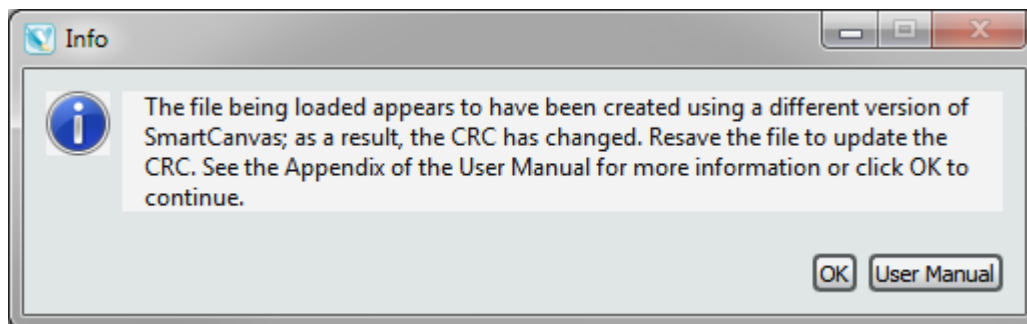
1. Select a project ini file using the 'Open' button in the 'DA9062 INI File Control' widget, located on the 'Front Panel' tab or the right side of the main window.
2. Ensure the motherboard toggle switch is to the left ('TP\_ON').
3. Turn off the device (VSYS = 0 V).
4. Click the 'Power Commander mode' button to enable PC mode (located on the GUI 'Front Panel' tab or the right side of the main window).
5. Turn on the device.
6. The device should now have started up using the selected project ini file contents. (The device will typically power-up into POWERDOWN when the configuration has AUTOBOOT=0, or may power-up into the ACTIVE state. The behaviour is therefore dependent on the OTP configuration).

#### NOTE

After starting up a device, Power Commander mode can be left enabled. This is useful if power-cycling the device, using commands such as SHUTDOWN, POWERDOWN, using the nRESETREQ button, and so on, as Power Commander mode will continue to service requests from the DUT for configuration transfers.

## Appendix B : Legacy SmartCanvas ini Files

SmartCanvas releases up to v0.0.3.15 use different internal reference information compared to later versions of SmartCanvas (The reference information is also known as the 'OTP mask file'). This may result in the message box of [Figure 18](#) being displayed.



**Figure 18: Information Message Box**

This is only for information, and clicking 'OK' allows work to continue. Resaving the ini file will prevent this message box showing again.

The same applies when opening ini files in the OTP Programmer Tool (see [Section 7](#)) and the File Comparison Tool (see [Section 8](#)).

## Revision History

Revision	Date	Description
1.0	03-Nov-2015	Initial release.
1.1	22-Dec-2015	Added description of 'project file comparison tool'. Typographical edits.
1.2	05-Jan-2016	Revised installation instructions (Hardware to be plugged in before installation started).
1.3	19-Sep-2016	Added Appendix B to describe the message box relating to OTP masks. Updated 'INI file control' section.
1.4	19-Oct-2016	Revised Section A.1, 'Modifying an .ini file'.
1.5	21-Feb-2022	Rebranded file with new logo, copyright and disclaimer

---

**Smartcanvas DA9061/2****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

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](http://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/](http://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.