Simple I2C I/O Controllers with SLG46537
SLG46537

The application note gives guidelines for creating simple I2C I/O controllers with SLG46537 device. A unique set of components of the SLG46537 allows the creation of such a system.

The application note comes complete with design files which can be found in the Reference section.

Contents

Figures

Tables

1. Terms and Definitions

MCU Microcontroller Unit

GPIO General-Purpose Input/Output

2. References

For related documents and software, please visit:

GreenPAK Programmable Mixed-Signal Products | Renesas

Download our free Go Configure Software Hub [1] to open the .gp5 files [2] and view the proposed circuit design. Use the GreenPAK development tools [3] to freeze the design into your own customized IC in a matter of minutes. Renesas provides a complete library of application notes [4] featuring design examples as well as explanations of features and blocks within the Renesas IC.

[1] Go Configure Software Hub, Software Download and User Guide, Renesas Electronics

[2] AN-1090 Simple I2C I/O Controllers with SLG46537.gp, GreenPAK Design File, Renesas Electronics

[3] GreenPAK Development Tools, GreenPAK Development Tools Webpage, Renesas Electronics

[4] Application Notes, GreenPAK Application Notes Webpage, Renesas Electronics

[5] AN-1107, Application note, Renesas Electronics

3. Introduction

This app note explains how to use the SLG46537's I2C macrocell as an input/output controller. This is a convenient way to add additional inputs and outputs in an MCU system such as shown in Figure 1.

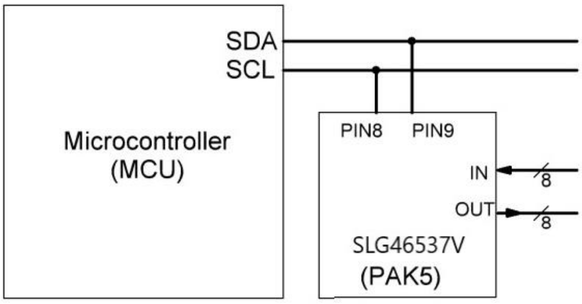

Figure 1. I/O Controller System View

4. Building an Output Controller – Write

The I2C interface in SLG46537 is very powerful because it can read and write all its configuration bits, including output states. However, directly writing configuration bits could be cumbersome, as they can be scattered among the 2K addressable bits. To facilitate simple, single-byte control, the SLG46537 has an I2C macrocell which can easily be connected to other cells graphically. Figure 2 shows the I2C macrocell outputs simply connected to output pins. The I2C macrocell states are referred to as Virtual Inputs, and are stored contiguously at Register Bit Addresses 1952 through 1959 as documented in SLG46537's Datasheet Appendix A.

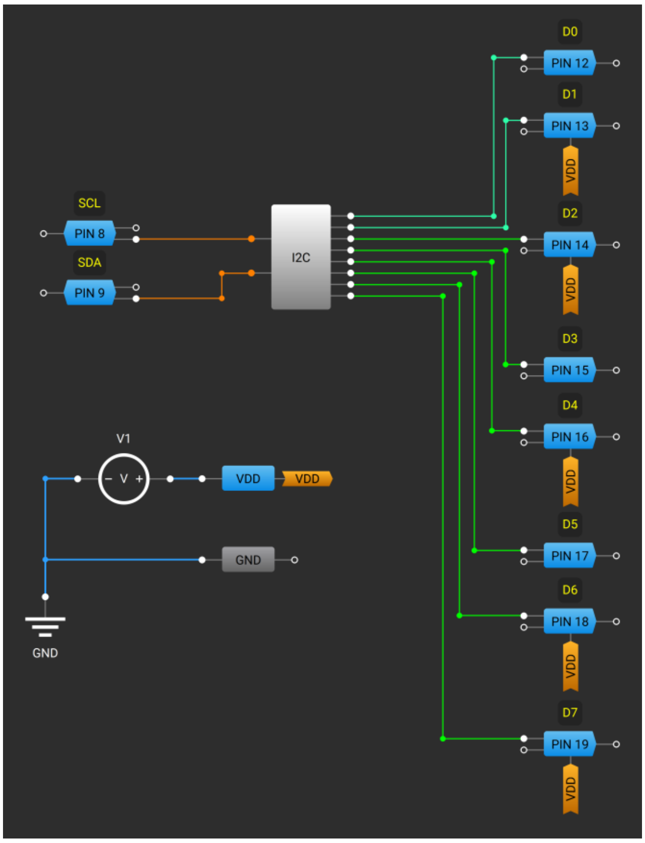

Figure 2. GUI connection of I2C macrocell to pins

To map a Register Bit to an I2C write, simply divide by the Register Bit Address by 8 to get the Word Address, and take the modulo 8 remainder to get the bit position in the Data Byte. For example, Register Bit 1959 corresponds to Word Address 0xF4, bit position 7 for the I2C command. See Table 1 for the mapping of all the Virtual Inputs.

An I2C write can be structured as follows:

1. Start
2. Control Byte (includes device address, R/W bit=0)
3. Word Address
4. Data
5. Stop

(Please see datasheet section **19.4.1 "I2C Byte Write"** for detailed description).

Table 1. Virtual Inputs Register Bit Addresses

Word Address		Data Bit Position	Register Bit Definition	SLG4657 Register Bit
244	0xF4	0	Virtual Input 0	1952
"	"	1	Virtual Input 1	1953
"	"	2	Virtual Input 2	1954
"	"	3	Virtual Input 3	1955
"	"	4	Virtual Input 4	1956
"	"	5	Virtual Input 5	1957
"	"	6	Virtual Input 6	1958
"	"	7	Virtual Input 7	1959

Examples of I2C writes to the Virtual Inputs are shown in Table 2. Please note the following:

1. Open bracket ' [' denotes START, and close bracket ']' denotes STOP

2. The upper 4 bits of the first byte (the Control Byte) are used for device addressing, and are programmable in each SLG46537.

This allows up to 16 SLG46537 devices to be individually addressed on the same bus. The next 3 bits are hard-wired to 000 in SLG46537. The final bit of the Control Byte is R/W which is 0 for write. For example, a device programmed with "0011" is written to with 0x30 as the Control Byte.

Table 2. Example Commands

I2C Command	Result
[0x30, 0xF4, 0xFF]	Write to device with programmed address bits "0011". Register at word address 0xF4 will be written with 0xFF, resulting in all virtual bits set HIGH.
[0x10, 0xF4, 0x00]	Write to device with programmed address bits "0001". Register at word address 0xF4 will be written with 0x00, resulting in all virtual bits set LOW.

5. Waveforms – Writing Virtual

I2C commands to write virtual bits are captured in the waveforms below. Channel 1 and 2 are the I2C clock (SCL) and data (SDA), respectively. Digital signals D0 through D7 correspond to the virtual bits connected as shown Figure 2. The I2C interface is running at 400 kHz with 10kohm resistors pulling up to 3.3 V on the SDA and SCL lines.

D0- PIN#12 (OUT0)

D1- PIN#13 (OUT1)

D2- PIN#14 (OUT2)

D3- PIN#15 (OUT3)

D4- PIN#16 (OUT4)

D5- PIN#17 (OUT5)

D6- PIN#18 (OUT6)

D7- PIN#19 (OUT7)

Channel 1 (yellow/second bottom) - PIN#8 (SCL)

Channel 2 (light blue/bottom) - PIN#9 (SDA)

The captures show the complete I2C sequence: START, Control Byte, Word Byte, Data Byte, STOP As can be seen, the outputs (pins 12 through 19) are updated at the last (8th) SCL falling edge at the end of the Data Byte.

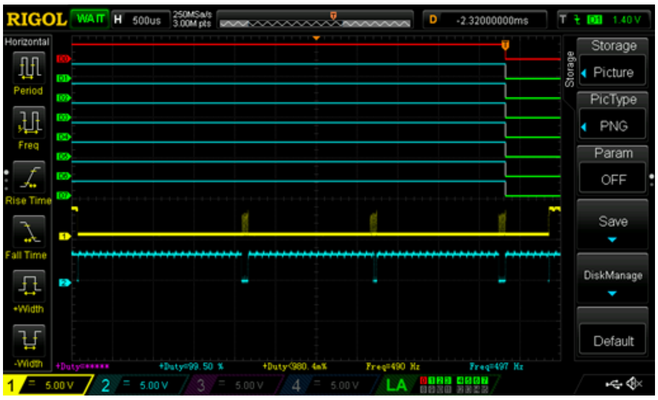

Figure 3. I2C Command [0x00, 0xF4,0x00]

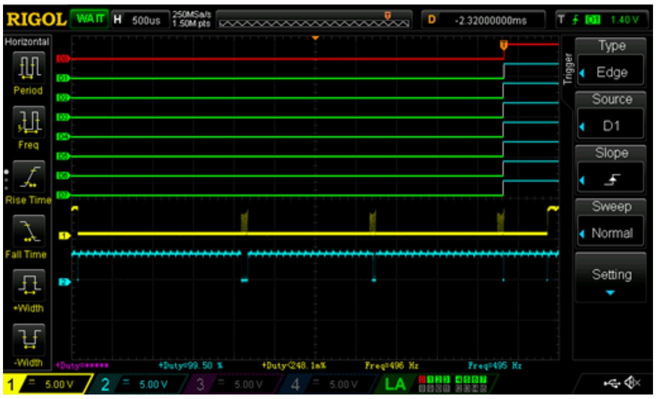

Figure 4. I2C Command [0x00, 0xF4, 0xFF]

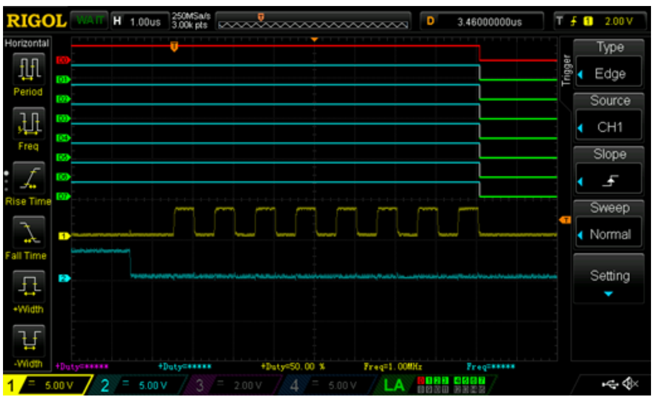

Figure 5. 1us scale of Data Byte 0x00

Figure 6. 1us scale of Data Byte 0xFF

6. Building an Input Controller – Read

An I2C read can be structured as follows:

1. Start
2. Control Byte (R/W=0)
3. Word Address (sets the internal address counter on device)
4. Start
5. Control Byte (R/W=1)
6. Data (from device)
7. Stop

(Please see datasheet section **19.4.4 “I2C Random Read”** for detailed description).

For an example, we will read GPIO 2 through 20 (no pin11 as it is GND). Those Digital Input bits are stored at Register bit addresses <1921:1927>, and <1968:1976> as documented in SLG46537’s Datasheet Appendix A. The relevant information from that appendix is shown in Table 3 below.

Table 3. GPIO Input Register Bit Address

Word Address		Data Bit Position	Register Bit Definition	SLG46537 Register Bit
240	0xF0	1	Pin2 Digital Input	1921
“	“	2	Pin3 Digital Input	1922
“	“	3	Pin4 Digital Input	1923
“	“	4	Pin5 Digital Input	1924
“	“	5	Pin6 Digital Input	1925
“’	“’	6	Pin7 Digital Input	1926
“	“	7	Pin10 Digital Input	1927
246	0xF6	0	Pin12 Digital Input	1968
“	“	1	Pin13 Digital Input	1969
“	“	2	Pin14 Digital Input	1970
“	“	3	Pin15 Digital Input	1971
“	“	4	Pin16 Digital Input	1972
“’	“’	5	Pin17 Digital Input	1973
“	“	6	Pin18 Digital Input	1974
“	“	7	Pin19 Digital Input	1975
247	0xF7	0	Pin20 Digital Input	1976

Examples of I2C reads to the Virtual Inputs are shown in Table 4. Please note the following:

1. Open bracket ‘ [‘ denotes START, and close bracket ‘]’ denotes STOP

2. The upper 4 bits of the first byte (the Control Byte) are used for device addressing, and are programmable in each SLG46537. This allows up to 16 SLG46537 devices to be individually addressed on the same bus. The next 3 bits are hard-wired to 000 in SLG46537. The final bit of the Control Byte is R/W which is 0 for write. For example, a device programmed with “0011” is written to with 0x30 as the Control Byte.

3. The ‘r’ byte is the read data sent from the slave device. Acknowledges from the master will cause the slave device to send another byte of data until there is a STOP.

The second command in Table 2 follows the same format as the previous example except two bytes are read, corresponding to Pins 12 through 20. Note that pin20 data is the 1st bit in the second data byte.

Table 4. Example Commands

Command	Result
[0x00, 0xF0, [0x01, r]	Read from device with programmed address bits “0000”. The contents at Word Address 0xF0 will be read.
[0x30, 0xF6, [0x31, r, r]	Read from device with programmed address bits “0011”. The contents at Word Address 0xF6 and the following byte 0xF7 will be read.

7. Bench testing – Reading GPIOs

For this bench test, the GPIOs 2 through 10 are read through I2C using a blank chip. First place a blank device into the development kit, start the emulator and power the SLG46537 in emulation mode. Be sure that all External Connectors for SCL and SDA, Pins 8 and 9, are enabled. Connect the master device SCL and SDA to the External Connectors and begin sending I2C commands.

The default I2C macrocell address on a blank chip is “0000”. Below is a demonstration of the first command from Table 4 if Pins 2 and 3 are HIGH. Figure 7 shows the results in a terminal program. The result is as expected with a read data byte of 0x06, this indicates that bit positions 3 and 2 of 0xF0 were logic HIGH. Referring back to Table 3, it is clear that these positions belong to Pins 2 and 3.

```
I2C>[0x00 0xF0 [ 0x01 r]
I2C START BIT
WRITE: 0x00 ACK
WRITE: 0xF0 ACK
I2C START BIT
WRITE: 0x01 ACK
READ: 0x06
NACK
I2C STOP BIT
I2C>
```

Figure 7. Terminal of Read Command

8. Conclusion

This App Note explained how to use the SLG46537’s I2C macrocell as an input/output controller. This technique can be useful if the user needs a way to transmit data to or from a microcontroller without using multiple GPIOs. The ability to write to Virtual Inputs 7:0 and read GPIOs via I2C makes the SLG46537 a more flexible device with more possible configurations.

9. Revision History

Revision	Date	Description
1.00	Aug 4, 2015	Initial release.
2.00	Apr 7, 2026	The part number has been changed from SLG46531V to SLG46537V.

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/.