Laboratory Short Course

S12UB Embedded Systems Lab

Lab Experiment #1

- ➢ S12UB Hardware
- ➢ Background Debug Module (BDM)
- ➢ D-Bug12 and Port I/O

www.freescale.com/universityprograms

Document Number: S12UBL1/2.0

freescale™ University Programs

Reading this Document

Answers provided to the Instructor assume that the reader is using Freescale S12UB, and CodeWarrior development software.

Overview

Understand S12UB Debug (POD) and Target hardware design. Read D-Bug12 and BDM user guides. Review S12 DG128 PORTA and PORTB operation.

On the DG128 all of the I/O registers are memory mapped. That is, they can be found in the same memory map, or organization, as the microcontroller RAM and FLASH program memory. Some non-memory mapped I/O microcontrollers have the I/O registers separate from the program memory. Conceptually memory mapped I/O is easier to understand because all processor resources are referenced from a single combined map. The tradeoff is memory taken by I/O cannot be used for program storage.

Open S12MEBIV3.pdf data sheet and reference register description to understand PORTA, PORTB, DDRA and DDRB register function and location in the device memory map. The data direction register (DDR) is used to configure the port. After reset PORTA and PORTB are configured as digital inputs. The DDR is used to reconfigure port pins to outputs. On the DG128 the majority of the ports exit reset as high impedance inputs.

Learning Objectives

Establish serial communication between the PC and POD (running D-Bug12). Use D-Bug12 to halt the DG128 and modify port registers.

- **PART – A Background Debug Mode**

Communicate from the PC to D-Bug12 using PC Hyperterm application over an RS-232 interface. Communicate BDM from Pod BDM OUT header to the Target BDM IN header over a 6 wire BDM cable.

- **PART – B Simple Port I/O Using D-Bug12**

Use D-Bug12 to read and write Target port registers.

Success Criteria

- Familiar with S12UB hardware design, D-Bug12 and BDM operation
- Understand PORTA and PORTB operation and register location in the DG128 memory map
- Know which DG128 pin is used to communicate BDM
- Understand How to enter active BDM mode
- Understand the purpose of the ASM command in D-Bug12
- Know the starting address of the I/O registers after DG128 is reset

Prerequisites - Equipment

- Freescale S12UB MC9S12DG128 (DG128) University Board
- S12UB (Distribution Disk) CD
- Personal Computer with a serial port interface
- Adobe Acrobat Reader
- Terminal emulation program like Hyperterm or TeraTerm
- Power supply; 120VAC to 12V DC mini-jack (2 to 4 Amp)
- DB9 RS-232 Serial Interface cable (Straight through no null modem)

Prerequisites - Study

Complete Prelab #1

- Familiar with S12UB hardware design, D-Bug12 and BDM operation
- Understand PORTA and PORTB operation and register location in the DG128 memory map
- Know which DG128 pin is used to communicate BDM
- Understand How to enter active BDM mode
- Understand the purpose of the ASM command in D-Bug12
- Know the starting address of the I/O registers after DG128 is reset

More Resources and Further Information

Read all background material before the lab. Hand in Prelab report on the day of the lab at the beginning of the lab session. The professor will review and grade. Prelab report will constitute ½ of the actual lab points.

Hand in Lab report before next lab session. The professor will review and grade.

Lab Studies and Activities

S12UB Hardware and DG128 Ports

- **PART – A Background Debug Mode**

Power S12UB and connect Debug (Pod) DB9 to PC COM port. Be careful, there are two DB9 connectors on the S12UB, make sure to connect the Debug (Pod) DB9 to the PC. Check that there is a BDM ribbon cable connection between the Pod BDM OUT and Target BDM IN headers, and verify proper orientation. Pin1 on Pod BDM OUT should connect to Pin1 on Target BDM IN. The Pod communicates to the Target over the BDM cable. If it is not in place D-Bug12 will not be able to find the Target.

1) Create Hyperterm session for 9600 Baud, 8 Data Bits, No Parity, 1 Stop bit, using XON / XOFF flow control.
2) Reset Pod micro by pressing SW2.
3) D-Bug12 Version will be displayed followed by the S> or R> prompt.
4) Enter RESET and return to halt and reset the target. When the Target is stopped the S> prompt will be displayed.
5) Enter HELP and return for command list. Press any key to scroll through the complete list. Take time to look at the different D-Bug12 commands available.
6) Enter DEVICE and return to determine target microcontroller type and memory map.
7) Should see MC9S12DG128 (DG128) among the listed part numbers, with a target speed of 8000 KHz (8 MHz). If not, check that the ribbon cable between the Pod and Target is present and indexed correctly to pin one (Red Strip on ribbon cable).

- **PART – B Simple Port I/O Using D-Bug12**

1) Enter RESET and return to reset the Target.
2) Jumper J12 pins 1 and 2, located under the 7-Seg display. Pin 1 will have the enlarged silk screen marking.
3) PORTA defaults to input port after reset. Configure PORTA as output by writing to Port A Data Direction Register (DDRA). Use the Memory Modify command MM. Write 0xFF to DDRA; enter MM 0002 FF and return. All data in HEX.
4) Write PORTA; enter MM 0000 FF and return, all LEDs should be ON.
5) Writing PORT A with 0F, F0, 55, and AA. Should see a corresponding change in the LED pattern.
6) Configure PORTB as input by writing to Port B Data Direction Register (DDRB). Use Memory Modify command MM. Write 0x00 to DDRB; enter MM 0003 00 and return. All data in HEX.
7) Read PORTB with Memory Display command; enter MD 0001 and return. Confirm PORTB data matches dip switch setting SW8 and SW9.
8) Change the dip switch setting and read PORTB again. Changes made to the DIP switch setting should be reflected in the PORTB read value.

Reflection on Learning

Part A showed the basics of establishing a D-Bug12 session between the Pod and Target Microcontrollers and looking at the command set and device type.

Part B illustrated how D-Bug12 can be used to directly access I/O registers in the Target DG128 microcontroller memory map. Even though this was done with the Target stopped, it can also be done while the microcontroller is running. The BDM subunit runs transparent to the core in real-time (BDM Hardware Commands) while the Target is stopped or running. When the microcontroller is stopped (Active BDM mode), more powerful BDM firmware commands can be executed.text

Lab Student Report

Write a one page summary of this lab that includes:

- Institution name
- Lab # and title
- Student name and date
- Objective
- Materials
- Brief discussion of work performed
- Conclusion

Hand in Lab Report before next lab session. The Professor will review and grade.

Revision History

Revision	Comments	Author
1.2	Initial Release	Boyd Beckington
2.0	Format Revision	Mark Robbins

How to Reach Us:

Home Page:
www.freescale.com

e-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
1-800-521-6274
480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014
+81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate,
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor
Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447
303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Number: S12UBL1/2.0

freescale™ semiconductor