To all our customers

Regarding the change of names mentioned in the document, such as Mitsubishi Electric and Mitsubishi XX, to Renesas Technology Corp.

The semiconductor operations of Hitachi and Mitsubishi Electric were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Mitsubishi Electric, Mitsubishi Electric Corporation, Mitsubishi Semiconductors, and other Mitsubishi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Note : Mitsubishi Electric will continue the business operations of high frequency & optical devices and power devices.

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

M16C/62

Virtual Flash EEPROM Driver for the M16C/62

1. Declaration

Mitsubishi Electric & Electronics USA does not guarantee the performance or use of this source-code. The intended use of provided source-code is the sole responsible of the user. The files have been successfully compiled using Mitsubishi's NC30 compiler. Before using this software review the source and make any necessary changes to support your hardware and application.

2. Introduction

Mitsubishi's M16C/62 is a 16-bit MCU based on the M16C CPU core with 256k bytes of user flash. The device can erase and program the on-chip flash memory under control of a user's program with no external programming devices required. This feature is called "CPU Rewrite Mode".

This application note describes the usage of the accompanying program that allows the user to write to the flash as if it were EEPROM.

3. Compatibility

The included virtual EEPROM driver is compatible with M16C/6x microcontrollers with page write (256 bytes) flash memory. It is NOT compatible with word write MCUs such as the M30220. The driver is compatible with the above noted MCUs on any Starter Kit/evaluation system running under KD30 and the ROM monitor. It CANNOT be evaluated or demonstrated on any emulator using RAM to emulate flash (i.e. Mitsubishi's PC4701, Nohau, or Ashling emulator systems).

4. Theory of operation

The C0000h and D0000h flash blocks (blocks 5 and 6) are used for the virtual EEPROM, with provisions to switch to other blocks (see section 5 below). The D0000h is used as backup in case of a power failure or a write error on the C0000h block. The size of the EEPROM is written to the top of both blocks. The first call to writeEE() writes the word at the beginning of both blocks (at offset "tag") with an "end of file" written to the end of the EEPROM "record". If writeEE() is called again with the same "tag", then another EEPROM "file" is created after the previous one and so on until the block (and the backup block) is filled. At that point, the C0000h block is erased, the last EEPROM file in the D0000H block is copied to the start of the C0000h block, then the D0000h block is erased and the EEPROM file is copied over to the start of it as well. Then the process repeats. The most up-to-date data is determined as the last "end of file" in the block.

For more detailed information on flash operations, read the application note references given at the end of this document.

5. Implementing the VirtEE drivers in a User Programs

Include “flash.h” in any project file that requires driver functions. Add “RAM62.c” , “virtEE.c”, and “EE_fl_util.c” to your project. These files require the “fdriver.h” and “sfr62.h” header files. By default, the driver uses the two 64k blocks 5 and 6 for the virtual EEPROM. This can be changed to the two 8k blocks (1 and 2) at FA000h and FA000h by un-commenting the definition “#define BLOCKS8K “ near the top of file “flash.h”. The maximum size of EEPROM changes to FF0h. Note that using blocks 1 and 2 is not compatible with the ROM monitor and KD30.

The following is a list of the driver function calls and how to use them.

int setEE_size(size)

- This function MUST be called before calling any other of the following functions.
- Size = 2 to 7ff0h (or 2 to ff0h if using 8k blocks). Sets the maximum size of the EEPROM in words (16 bits). Once set, it can only be changed with the resetEE() function (size stored in flash).
- Returns 1 if no errors, 0 on invalid size, -1 on error, or the previously set size value (2 to 7ff0h). Errors (-1) are caused by flash write errors or the EEPROM “file” in flash cannot be found. To determine the flash write error call readsrd().
- When called for the first time (no size stored in flash), flash blocks C0000h and D0000h are completely erased.
- The user must call this function before calling checkEE().

int void checkEE(void)

- This function verifies that the flash blocks used for EEPROM contain the same data (see theory of operation). This function should always be called after reset but after calling setEE_size().
- Returns 1 if no errors, -1 on error. These errors can result from an interruption of power or non-maskable interrupt occurring during a write to EEPROM If a nonrecoverable error is detected, call the resetEE() function.
- Obviously, this function cannot detect a loss of data due to a power loss before or shortly after a call to writeEE().

int void resetEE()

- Completely erases flash blocks C0000h and D0000h. Call this function only after checkEE() detects an error.
- After calling this function, call EE_size().

int writeEE(int value, int tag)

- Writes the word “value” to EEPROM at EEPROM location “tag” (0 to size -1). Returns 1 if no errors, -1 on SRD error, or 0 on invalid “tag”. Before writing, this function checks if the value to be written is the same as that stored in the EEPROM. If so, function returns with no error.

char readsrd()

- Returns the flash SRD register.

int clearsrd()

- Clears the flash SRD register.

int readEE(int tag, int *value)

- Upon return, *value contains the EEPROM value at the address “tag”. Returns 1 on no errors, 0 on invalid tag.

6. Notes

Interrupts assigned to the fixed vector table CANNOT occur during any write to flash operation. This includes all the function calls above except for readsrd() and readEE(). As a ”safety” feature, interrupts are suspended during flash operations. The worst case suspend time is over 600 ms (maximum erase block time). If this delay is unacceptable, remove the “interruptoff” line at the beginning of the function ramcodestrt() in the file RAM62.c, move the variable vector table to RAM, and then set the “intb” register accordingly.

Note as the size of the EEPROM increases, so can the time to execute the writeEE() function. For example, an EEPROM size of 3FF0h or greater requires an erase of both flash blocks and rewrite every time a value is updated. For sizes near the max. of 7FF0h a rewrite to an EEPROM location could take up to 4 seconds!

7. Demonstration Program

The driver file virtEE.c contains a demonstration “main.c” (remove or comment out when implementing as a driver). The program was developed to run on the MSV1632, but it also works with any M16C/62 flash system running the KD30 debugger in single chip mode.

The following steps are required to run the Demonstration (it is assumed the user has a general understanding of the KD30 debugger):

1. Connect the MDECE0620 board to power and the host PC.
2. Start KD30. In the “INIT” screen, click on the tab “run mode” and select “free run mode”.
3. Load the “virtEE.x30” file.
4. Insert a breakpoint in the main() function at the first “y= *read_val;”.
5. Run the code: debug-> go free.
6. Hit the red “stop” button on KD30 (in “free run” mode, KD30 does not respond automatically to breakpoints).
7. View the global “y” from the C watch window.
8. Remove the breakpoint and insert one at the next “y= *read_val;”.
9. Repeat steps 5–8 until you reach the end of main().
10. You can view the flash blocks and/or change the variables passed (e.g. size, tag, values), recompile, and run again to see how the changes affect operation.

8. References

Data sheets:

- 62AEDS.pdf – M16C/62A Specifications

Application Notes:

- cpurw62.pdf appnote - Programming the M16C/62 Flash in CPU Rewrite Mode
- Wflash.pdf - Applications of CPU Rewrite Mode

Manuals:

- 6020esm.pdf (Software Manual)
- 6020ec.pdf (C Manual)
- 6020easm.pdf (Assembler Manual)
- NC30ue.pdf (Compiler Manual)

RENESAS
Renesas Technology Corp.

Keep safety first in your circuit designs!

- Mitsubishi Electric Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always a possibility of trouble occurring. Trouble with semiconductors may lead to personal injury, fire, or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of backup, auxiliary circuits, (ii) use of non-flammable material, or (iii) prevention against any malfunction or mishap.

Notes regarding this information

- This information is intended as a reference to assist our consumers in the selection of the Mitsubishi semiconductor product best suited to the customer's application; it does not convey any license under any intellectual property rights, or any other rights, belonging to Mitsubishi Electric Corporation or a third party.
- Mitsubishi Electric Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
- All information contained in these materials, including product data, diagrams, charts, programs and algorithms, represents information on products at the time of publication of these materials, and is subject to change by Mitsubishi Electric Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for the latest product information before purchasing a product listed herein.
- The information described herein may contain technical inaccuracies or typographical errors. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability, or other loss arising from these inaccuracies or errors.
- Please also pay attention to information published by Mitsubishi Electric Corporation by various means, including the Mitsubishi Semiconductor home page (http://www.mitsubishichips.com).
- When using any or all of the information contained in these materials, including product, data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability, or other loss resulting from the information contained herein.
- Mitsubishi Electric Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
- The prior written approval of Mitsubishi Electric Corporation is necessary to reprint or reproduce these materials in whole or in part.
- If these products or technologies are subject to Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
- Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
- Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for further details on these materials or the products contained therein.

RENESAS
Renesas Technology Corp.

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

USA Headquarters

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

1050 East Arques Avenue
Sunnyvale, CA 94085-4601
Phone: 408-730-5900
FAX: 408-732-9382

Direct Sales Offices – USA

Northwest

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

1050 East Arques Avenue
Sunnyvale, CA 94085-4601
Phone: 408-730-5900
FAX: 408-737-1129

Southwest

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

20 Fairbanks, Suite 181
Irvine, CA 92618
Phone: 949-859-9453
FAX: 949-859-9450

Rocky Mountain

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

PO Box 273377
Fort Collins, CO 80527
Phone: 970-669-4068
FAX: 970-669-4129

North Central

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

3300 Edinborough Way, Suite 511
Edina, MN 55435
Phone: 952-837-9053
FAX: 952-837-9059

Mitsubishi Electric & Electronics USA, Inc.
(Automotive Market Only)

50 West Big Beaver Rd., Suite 136
Troy, MI 48084
Phone: 248-526-9580
FAX: 248-526-9583

South Central

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

8310 Capital of Texas Hwy. N.,
Suite 260
Austin, TX 78731
Phone: 512-346-4200
FAX: 512-346-4434

Northeast

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

92 Montvale Avenue, Suite 2500
Stoneham, MA 02180
Phone: 781-245-6956
FAX: 781-245-4233

Mid-Atlantic

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

92 Montvale Avenue, Suite 2500
Stoneham, MA 02180
Phone: 781-245-6956
FAX: 781-245-4233

South Atlantic

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

2625 Meridian Parkway, Suite 100
Durham, NC 27713
Phone: 919-767-7900
FAX: 919-767-7902

Southeast

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

2810 Premiere Parkway, Suite 400
Duluth, GA 30097
Phone: 678-258-4518
FAX: 678-258-4519

Mexico Inquiries

Western Mexico

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

20 Fairbanks, Suite 181
Irvine, CA 92618
Phone: 949-859-9453
FAX: 949-859-9450

Eastern Mexico

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

8310 Capital of Texas Hwy. N.,
Suite 260
Austin, TX 78731
Phone: 512-346-4200
FAX: 512-346-4434

Puerto Rico Inquiries

Puerto Rico

Mitsubishi Electric & Electronics USA, Inc.
Electronic Device Group

2810 Premiere Parkway, Suite 400
Duluth, GA 30097
Phone: 678-258-4518
FAX: 678-258-4519

Mitsubishi Electric Sales Canada, Inc.

Direct Sales Offices – Canada

Central & Western Canada

Mitsubishi Electric Sales Canada, Inc.

4299 14th Avenue
Markham, Ontario,
Canada L3R 0J2
Phone: 905-475-7728
FAX: 905-475-1918

Eastern Canada

Mitsubishi Electric Sales Canada, Inc.

340 March Road, Suite 300
Kanata, Ontario,
Canada K2K 2E4
Phone: 613-591-3348
FAX: 613-591-3948

JQA-QS0161

Mitsubishi Electric MCU products sold into the Automotive marketplace are QS-9000 certified.

Printed in U.S.A.

www.mitsubishichips.com