

## M16C/62P, M16C/26

### Simple Flash API for the M16C/62P and M16C/26

---

#### 1.0 Abstract

A simple Application Program Interface (API) has been created to allow programmers of the M16C/62P (M30626FHP) and M16C/26 (M30262F8) to easily integrate reprogramming abilities into their applications using CPU Rewrite. CPU Rewrite is the term used to describe a Renesas MCU's ability to reprogram its own internal Flash memory while running in its normal operational mode.

#### 2.0 Introduction

The M16C/62P and M16C/26 both use a flash technology referred to as HND. This Flash technology allows a user to program the internal flash memory of the MCU a WORD (16 bits) at a time. Since the flash technologies are the same, the source code for this API is virtually the same.

This API uses Erase-Write Mode 1 (or EW1 Mode) of CPU Rewrite. This mode allows the code responsible for erasing and programming to execute from flash memory as opposed to earlier MCUs that required the code to run from RAM when performing CPU Rewrite operations.

The API source files comply with Renesas' NC30 compilers only.

#### 3.0 API Functions

##### 3.1 FlashErase Function

This function allows the user to erase any Flash block within the MCU including the near area memory (below address 0xFFFF).

```
int FlashErase( int block);
```

##### Return Value

Returns the outcome of the erase operation: 1 if successful and 0 if not successful.

##### Parameters

*block*

Specifies the block to erase. The blocks are numbered in the same fashion as they are in the device specifications; the highest address block is labeled "0". Furthermore, the near area Flash blocks are labeled in succession after the User Area blocks (i.e. "4"&"5" for the M16C/26 and "11" for the M16C/62P).

### 3.2 FlashWrite Function

This function allows data to be written into Flash. The number of bytes to write **MUST** be an even number because the flash controller has to write a WORD (16 bits) at a time. The flash address **MUST** also be an even number as well because the flash controller needs to write WORDS to even addresses only.

**int FlashWrite(unsigned long *flash\_addr*, far unsigned char \* *buffer\_addr*, unsigned int *bytes*);**

#### Return Value

Returns the outcome of the write operation: 1 if successful and 0 if not successful.

#### Parameters

*flash\_addr*

This is the address of the Flash area to write to. This must be an even address in Flash.

*buffer\_addr*

This is the address of the data buffer containing the data to write to Flash. This address can be either a location in RAM or Flash.

*Bytes*

The number of bytes contained in the data buffer. This must always be an even number because the MCU can only program a WORD of data at a time. This parameter represents the number of “bytes” instead of “words” in order to avoid confusion of how much data is being passed.

## 4.0 Reference

Renesas Technology Corporation Semiconductor Home Page

<http://www.renesas.com>

E-mail Support

[support\\_apl@renesas.com](mailto:support_apl@renesas.com)

## 5.0 Software Code

### 5.1 Main Program

The following is an example of using the API in order to erase and program a Flash block in a M16C/26 device.

```
#include "flash-262-ewl.h"

unsigned char data_buffer[256];

/* We have predefined a section in the sect30 file to hold some */
/* const data at a fixed location (0xF000). When the program is loaded */
/* into KD30, we will see this data has programmed into Flash at address */
/* 0xF000. We will use this to verify that the Erase function works. */
#pragma SECTION rom fixed
/* This will pre-fill our 256 byte area */
const far long watch_here[8][8] =
{0xAABBCCDD,0xAABBCCDD,0xAABBCCDD,0xAABBCCDD,
 0xAABBCCDD,0xAABBCCDD,0xAABBCCDD,0xAABBCCDD};

void main() {

    int i;
    int ok;

    /* Fill our data buffer */
    for(i=0;i<256;i++)
        data_buffer[i] = i;

    /* Place a breakpoint on the following line in order to view the flash */
    /* contents at address 0xF000 before it is erased. You can use the Memory */
    /* Window in KD30 to do this. */

    i = 1;                                /* PLACE BREAKPOINT ON THIS LINE */
                                          /* View address 0xF000 */

    ok = FlashErase( 5 );                  // Erase Block 5 (0xF000 - 0xF7FF)
        if( !ok )                          // Note that interrupts are disabled during this function
            while(1);                       /* ERROR ERASING */
```

```
i = 2;                                /* PLACE BREAKPOINT ON THIS LINE */
                                        /* View address 0xF000, should be all FF */

ok = FlashWrite( 0xF000, // Flash address write to (must be on a 256-byte page boundary)
                data_buffer, // Address of buffer containing data to write
                256); // Number of byte in buffer
                                // Note that interrupts are disabled during this

// function
if( !ok )
    while(1); // ERROR PROGRAMMING */

i = 3;                                /* PLACE BREAKPOINT ON THIS LINE */
                                        /* View address 0xF000, should look like our data_buffer */

while(1); // The End */
}
```

## Keep safety first in your circuit designs!

- Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. 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 substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

## Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
- Renesas Technology 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 are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.  
The information described here may contain technical inaccuracies or typographical errors.  
Renesas Technology 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 Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (<http://www.renesas.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. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Renesas Technology 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 Renesas Technology Corporation or an authorized Renesas Technology Corporation 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 Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
- If these products or technologies are subject to the 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 Renesas Technology Corporation for further details on these materials or the products contained therein.