

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

Renesas MCU

Utility To Convert S-Record Data To A C-Structure

Introduction

In embedded applications it is often desirable to store program code in one area or type of memory but execute it from another. An example would be routines to program internal Flash memory. Such programming routines cannot execute from the same Flash memory that is being programmed and so must be run from elsewhere. In single chip systems the only other memory available is internal RAM. In such cases the program code must be stored in non-volatile Flash memory and then copied into RAM at runtime for execution. Here lies a problem. For correct execution the program code must be linked for the memory addresses where it will run (e.g. RAM) but moved into non-volatile memory (e.g. Flash) at build time. The utility, 'Motice_cl', described in this document provides one method of achieving this. It should be noted that there are many other approaches to tackling this problem.

Contents

| | |
|--|----|
| UTILITY TO CONVERT S-RECORD DATA TO A C-STRUCTURE | 1 |
| INTRODUCTION | 1 |
| CONTENTS | 1 |
| 1. MOTICE_CL | 2 |
| 2. CALLING MECHANISM | 2 |
| 3. OUTPUT FORMAT | 2 |
| 4. HEW INTEGRATION | 3 |
| SUMMARY | 9 |
| APPENDIX A : C SOURCE FILES FOR THE 'MOTICE_CL_DEMO' HEW WORKSPACE | 10 |
| WEBSITE AND SUPPORT | 16 |

1. Motice_cl

'Motice_cl' is a command line (DOS) application that converts the data presented in a Motorola s-record file into a C structure. The idea being that the function to be relocated is processed as follows:

1. Function is compiled to an object file.
2. Compiled object file is linked to RAM address.
3. S-record file representing linked absolute data is produced.
4. S-record file is processed using 'motice_cl' to produce a C structure containing constant data.
5. C structure is added to final application, compiled and linked.
6. At runtime the data in the structure is copied to RAM.
7. Relocated function is called using a function pointer.

This results in code that has been linked for RAM addresses being stored in non-volatile memory and then copied to and run from the original RAM addresses.

2. Calling Mechanism

The command line interface for the utility is:

MOTICE_CL source destination structname

Source: Specifies the name and path of the s-record file to be processed.

Destination: Specifies the name and path of the C file to be produced.

Structname: Specifies the name of the C structure to be generated.

3. Output Format

The C file output by 'motice_cl' has contains the following structure definition.

```
struct rom_data {  
    unsigned long start_address;  
    unsigned long data_length;  
    unsigned char data[4708];  
};
```

The size of the 'data' array is dependent on the amount of data in the processed s-record.

The format of the structure should be self explanatory. The 'start_address' is the base address where the byte 'data' should be loaded to prior to execution and 'data_length' is the number of data bytes in the 'data' array in the structure.

An example of the structure, in part, is shown below.

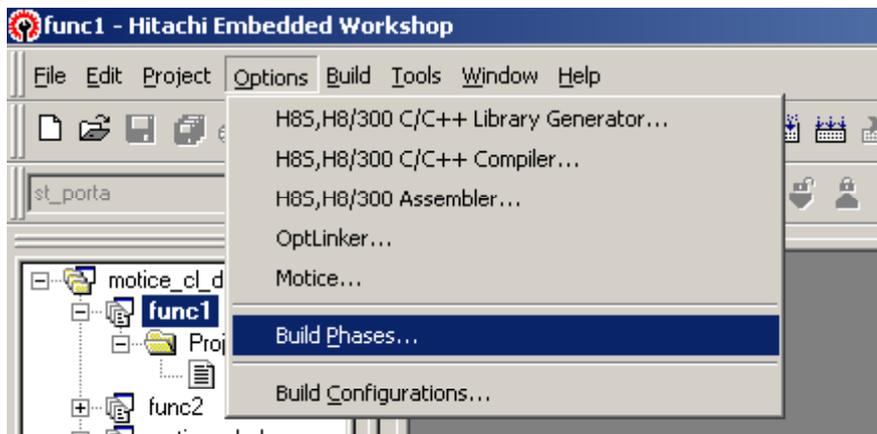
```
const struct rom_data datastruct = {
    0x00000000,
    0x00001264,
    0x00, 0x00, 0x08, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00,
    0x08, 0x00, 0xff, 0xff, 0xff, 0xf0, 0x00, 0x00, 0x04, 0x00,
    0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
    0x04, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x04, 0x00,
    0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
    .....
};
```

This example shows the data starting at address zero.

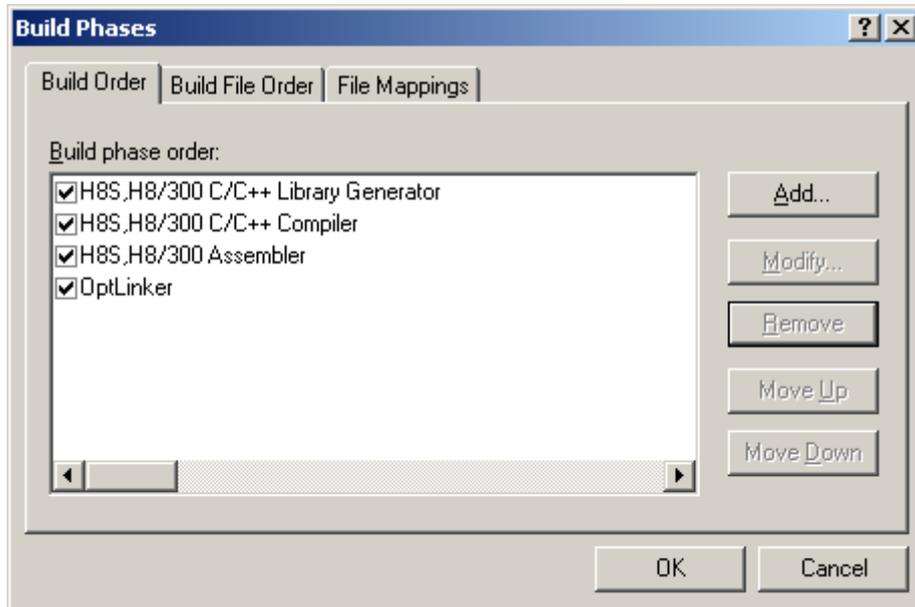
4. HEW Integration

One of the features of HEW (High-Performance Embedded Workshop) is the ability to produce custom build phases. This lends itself well to 'Motice_cl' as a separate build phase can run to call the utility each time an application is built. The following screenshots show the process of setting up the build phase.

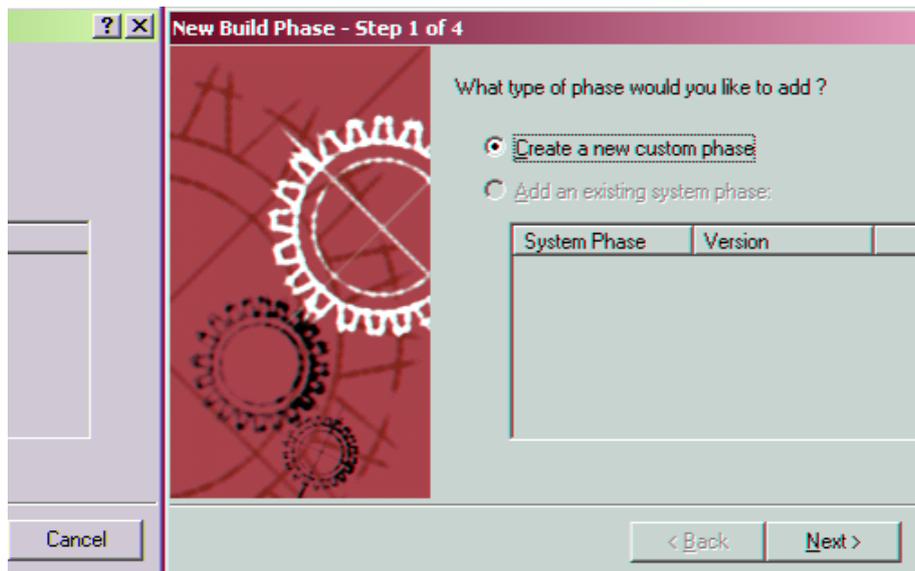
1. Select 'Build Phases' from the 'Options' menu.



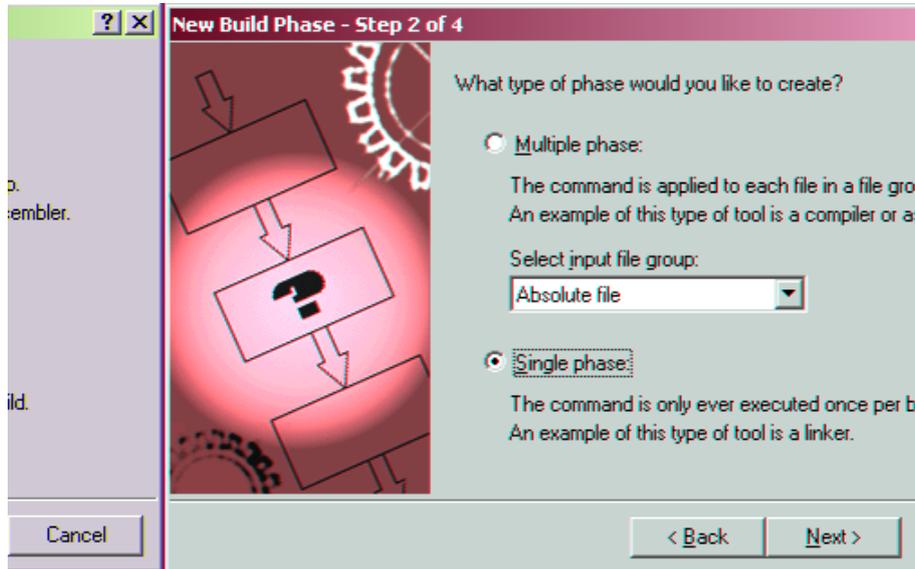
2. Click on the 'Add' button.



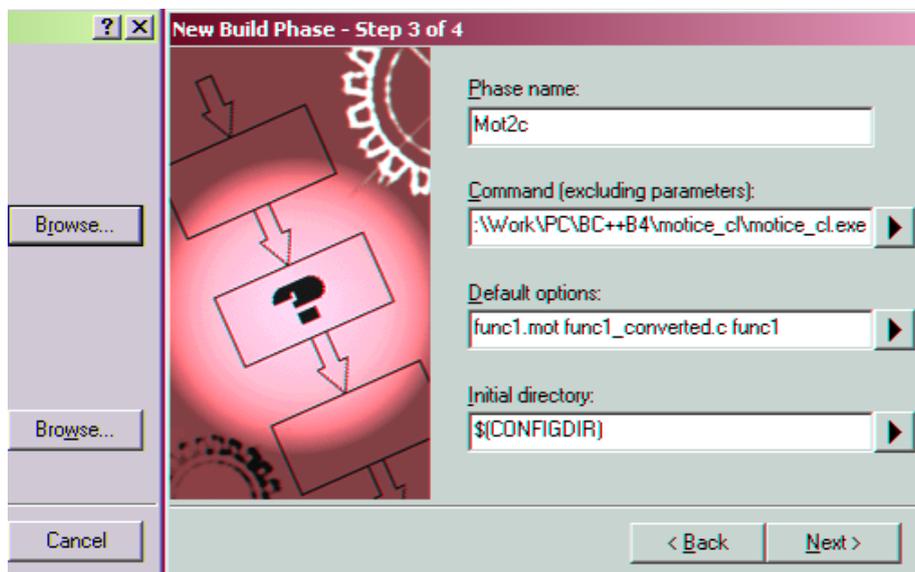
3. Create a custom build phase by clicking 'Next'.



4. Create a single build phase. This means that the phase will only execute once per build. The 'Motice_cl' utility is only required to run once when it processes the s-record file output from the linker.

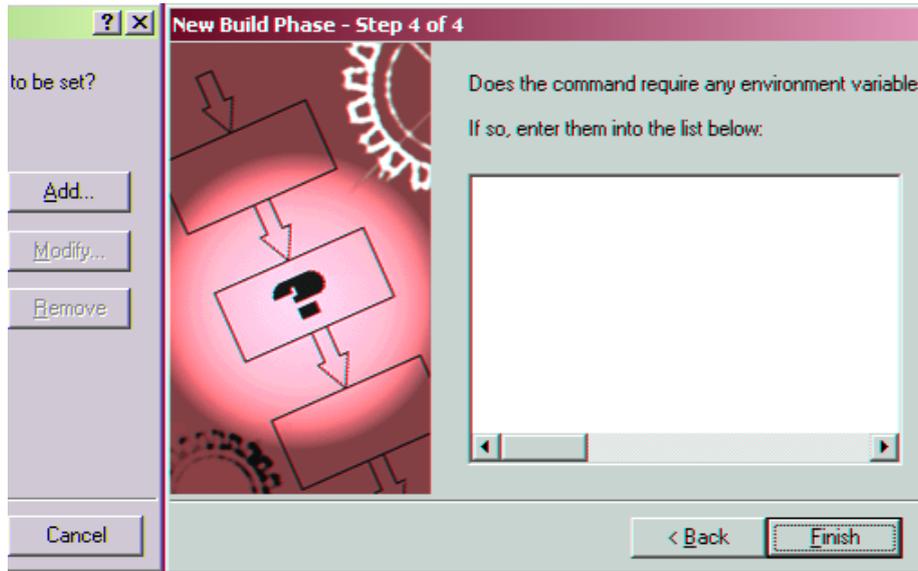


5. Configure the build phase to call 'Motice_cl' with the correct command line parameters.

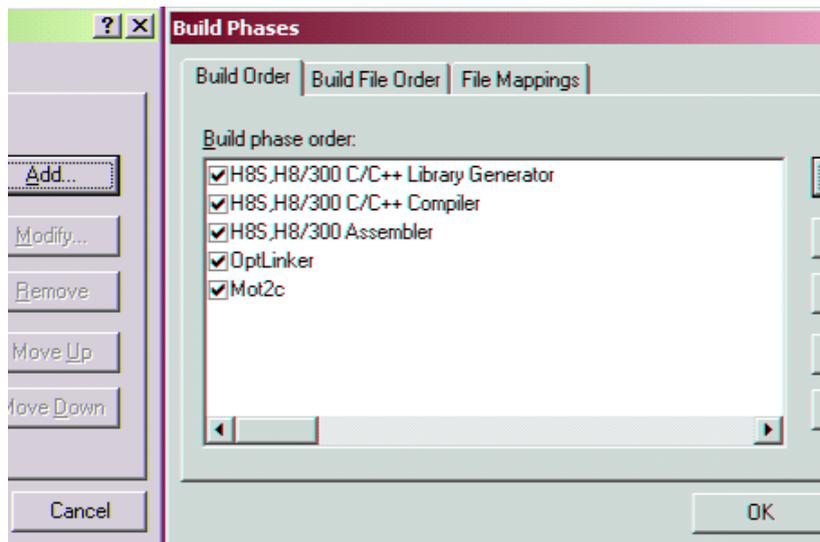


Here the phase has been called 'Mot2c', the full path for the 'Motice_cl' executable has been entered as have the default parameters for the command. In this case the default parameters are the command line parameters for the utility.

6. No environment variables are required for the utility so leave this blank and hit 'Finish'.



7. The 'Build Phases' form should now show the newly created 'Motc2c' build phase. Ensure that it is ticked so that the phase is executed. The order the build phases are executed can be changed. This is not necessary here as the 'Motc' phase must run after the linker otherwise it will have no s-record file to process!



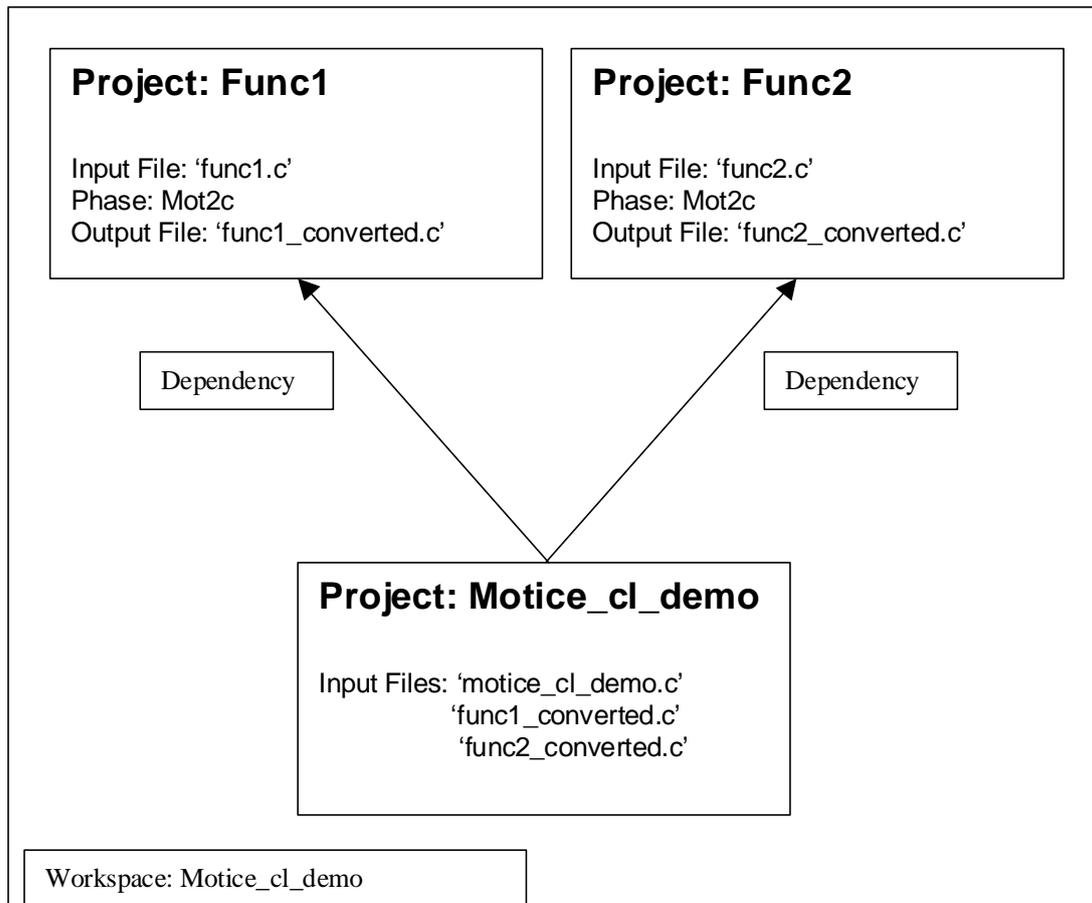
- 8 Click 'OK'. Now when the project is built the 'Mot2c' phase will execute and convert the s-record file to a C structure. The screen shot below shows the output from the phase in the HEW build window.

```

Phase Mot2c starting
Motice v0.1a (Command Line Version) (c) GAntelope 2001
Source file   = func1.mot
Dest file    = func1_converted.c
Struct name  = func1
Start Address = 0xffb000
End Address  = 0xffb042
Byte count   = 0x42 (66 bytes)
5 S records processed
Phase Mot2c finished
  
```

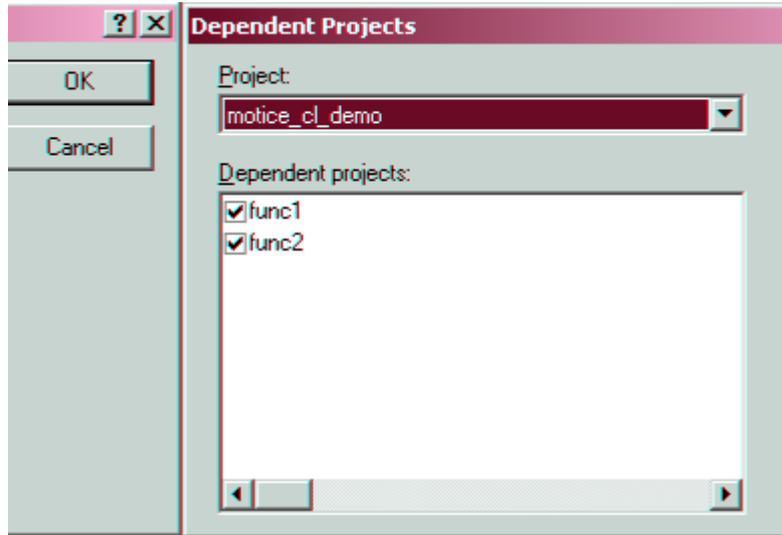
For simplicity it is recommended that each function to be processed be placed in its own project within a HEW workspace. Each function's project should be created as an 'Empty Project' using the HEW project generator with the linker sections located at the required runtime addresses. Each project should have its own custom 'Mot2c' build phase. A master project can be created that compiles and links the C files output from the individual projects.

To aid clarification a HEW workspace's hierarchy is shown by the following diagram.



Here there is a HEW workspace called 'Motice_cl_demo' which contains the projects 'motice_cl_demo', 'func1' and 'func2'. Both the 'func1' and 'func2' projects consist of a single file which is compiled and linked to an s-record file and then processed by 'Mot2c' build phases to produce the files 'func1_converted.c' and 'func2_converted.c' respectively. To ensure that any changes to the code in the 'func1' and 'func2' projects is picked up by the main 'motice_cl_demo' project then the 'motice_cl_demo' project is made dependent on both the 'func1' and 'func2' projects.

Project dependencies are set up in HEW via the ‘Projects/Dependent Projects’ menu. The screenshot below shows the project dependencies for ‘motice_cl_demo’.



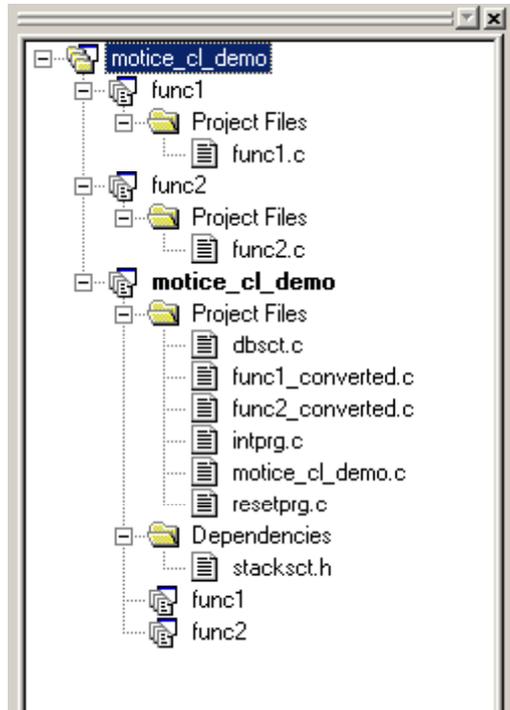
Now, whenever the ‘motice_cl_demo’ project is built both the ‘func1’ and ‘func2’ projects will be built first and so ensuring that very latest converted files are made available to ‘motice_cl_demo’.

Appendix 1 of this apps note shows the C source code for the files in the projects. In particular the file ‘motice_cl_demo.c’ may be of interest as it shows how to copy the program code from Flash to RAM using ‘memcpy’ and then call the copied functions via function pointers.

Summary

The utility ‘motice_cl’ can easily be integrated into HEW projects to provide the ability to relocate RAM based functions into Flash at build time for relocation and execution at runtime.

**APPENDIX A : C SOURCE FILES FOR THE 'MOTICE_CL_DEMO' HEW
WORKSPACE**



```
Funcl.c  
  
// funcl.c  
  
unsigned long funcl (unsigned long ul)  
{  
    return ++ul;  
}
```

```
Func2.c  
  
// func2.c  
  
unsigned long func2 (unsigned long ul)  
{  
    return ++ul;  
}
```

```
Func1_converted.c

//
// This file has been generated by 'motice_cl' (v0.1a)
//
// FILE: func1_converted.c
// DATE: 24/7/2001
// TIME: 13:50:53

struct rom_data {
    unsigned long start_address;
    unsigned long data_length;
    unsigned char data[4];
};

const struct rom_data func1 = {
    0x00ffb000,
    0x00000004,
    0x0b, 0x70, 0x54, 0x70
};
```

```
Func2_converted.c

//
// This file has been generated by 'motice_cl' (v0.1a)
//
// FILE: func2_converted.c
// DATE: 24/7/2001
// TIME: 13:50:53

struct rom_data {
    unsigned long start_address;
    unsigned long data_length;
    unsigned char data[4];
};

const struct rom_data func2 = {
    0x00ffb100,
    0x00000004,
    0x0b, 0x70, 0x54, 0x70
};
```

```

Motice_cl_demo.c

/*****
/*
/* FILE      :motice_cl_demo.c
/* DATE      :Tue, Jul 24, 2001
/* DESCRIPTION :Main Program
/* CPU TYPE  :H8S/2633
/*
/*
/*
/*
/*
/*****

#include <string.h>

struct rom_data {
    unsigned long start_address;
    unsigned long data_length;
    unsigned char data[0xffff]; // this value (0xffff) does not have to
        // reflect the true size of the data
        //but is required to satisfy the compiler
};

// externals
extern const struct rom_data func1;
extern const struct rom_data func2;

// function pointers
unsigned long (*ptr1) (unsigned long);
unsigned long (*ptr2) (unsigned long);

volatile unsigned long ul;

void main(void)
{
    // initialise the variable to be incremented
    ul = 0;

    // initialise function pointers
    ptr1 = (void *) func1.start_address;
    ptr2 = (void *) func2.start_address;

    // copy relocated functions to RAM
    memcpy ((void *) func1.start_address, func1.data, func1.data_length);
    memcpy ((void *) func2.start_address, func2.data, func2.data_length);

    while (1)
    {
        ul = ptr1 (ul);
        ul = ptr2 (ul);
    }
}

```

Website and Support

Renesas Technology Website

<http://www.renesas.com/>

Inquiries

<http://www.renesas.com/inquiry>

csc@renesas.com

All trademarks and registered trademarks are the property of their respective owners.

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (<http://www.renesas.com>)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
 - (1) artificial life support devices or systems
 - (2) surgical implantations
 - (3) healthcare intervention (e.g., excision, administration of medication, etc.)
 - (4) any other purposes that pose a direct threat to human life
 Renesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

© 2008. Renesas Technology Corp., All rights reserved.