R32C/111 Group

R01AN0001EK0100
Rev.1.00
Mar 14, 2014

IEC 60730 Self Test Sample Code

Introduction

All home appliances intended for the European Union market must today comply with the International Electrotechnical Committee's standard IEC 60730-1, "**Automatic electrical controls for household and similar use**". Both Europe and the U.S. have introduced IEC regulations defining safety requirements in the design of home appliances, but in the U.S. the standard UL1998 is still in use.

White goods microcontroller design teams therefore need to consider what is required in terms of hardware and software in order to comply with the IEC 607301 requirements. Appendix H of IEC 60730-1 specifically defines requirements for 'controls using software'.

This document explains example software test routines that have been developed as a help in fulfilling the standard's Appendix H software requirements for Class B controllers, 'control functions intended to prevent unsafe operation'. See table in 7(7).

Demonstration target
The Sample code was created for the R32C/111. Most routines can be used for devices in the R32C/111 Group. For M16C and H8, there exist corresponding CPU, RAM and ROM test source code.

Tools used

- Renesas R32C/100 Standard Toolchain version 1.02.01. (compiler, assembler, linker):
- E8a in-circuit debugger.

Time measurement conditions
Execution time measurements were done at 50 MHz CPU clock. No compiler optimizations. Measurement was taken with a scope. The pin (p0_0) accessible on jumper JA3 on the RSK2R32C111-3 was toggled before and after the function call.

Contents

1. Application software safety considerations

Apart from the test routines of the API, here are some safety considerations to take into account when writing a software application for equipment of Class B.

- Monitor for application state machine misbehavior, this could be done by checking for functions called in the wrong sequence using a safety checking state machine.
- Check for uncalled functions.
- Check for functions not finished.
- Ensure the circumstances are right before executing a function, e.g. 'Door locked before motor on'
- Dynamically enable / disable a function's ability to run.

2. Guide to the API

The following pages show each API test call's syntax and provides explanations.

2.1 CPU Test

This section describes the CPU tests routines. Reference IEC 60730: 1999+A1:2003 Annex H - Table H.11.12.1 CPU

The CPU test covers testing of CPU Registers by writing test values (like 0x55, 0xAA) into them and then reading them back. This can't be done using 'C' language so inline assembly code has been used.

These tests are testing such fundamental aspects of the CPU operation; the API functions do not have return values to indicate the result of a test. Instead the user of these tests must provide an error handling function with the following declaration:-

```
extern void ErrorHandler(void);
```

This will be called by the CPU test if an error is detected. This function must not return back to the test code.

The test functions all follow the rules of register preservation following a C function call as specified in the Renesas tool chain manual. Therefore the user can call these functions like any normal C function without any additional responsibilities for saving register values beforehand.

Specifically CPU registers R0-R7, A0-A3, FB, INTB, USP, ISP, SB and FLG are tested.
The source file 'CPU_Test.c' provides implementation of CPU test using "C" language functions that contain the inline assembly. The source file 'CPU_Test.h' provides the interface to the function CPU test. The file 'MisraTypes.h' includes definitions of MISRA compliant standard data types.

IMPORTANT NOTE: Please keep the "Optimisation" option "OFF" for the 'CPU_Test.c' file, to prevent modification of the test code.

The CPU test is categorised as follows:

- General purpose registers (R0, R1, R2, R3, R4, R5, R6, R7, A0, A1, A2, A3, FB for each bank).
- Interrupt Table registers (INTB).
- User Stack Pointer (USP), Interrupt Stack Pointer (ISP), Static Base (SB) register.
- Program counter (PC) register.
- Flag (FLG) or Status register.
- Test all above

They are tested by writes and subsequent readbacks of two different test patterns.

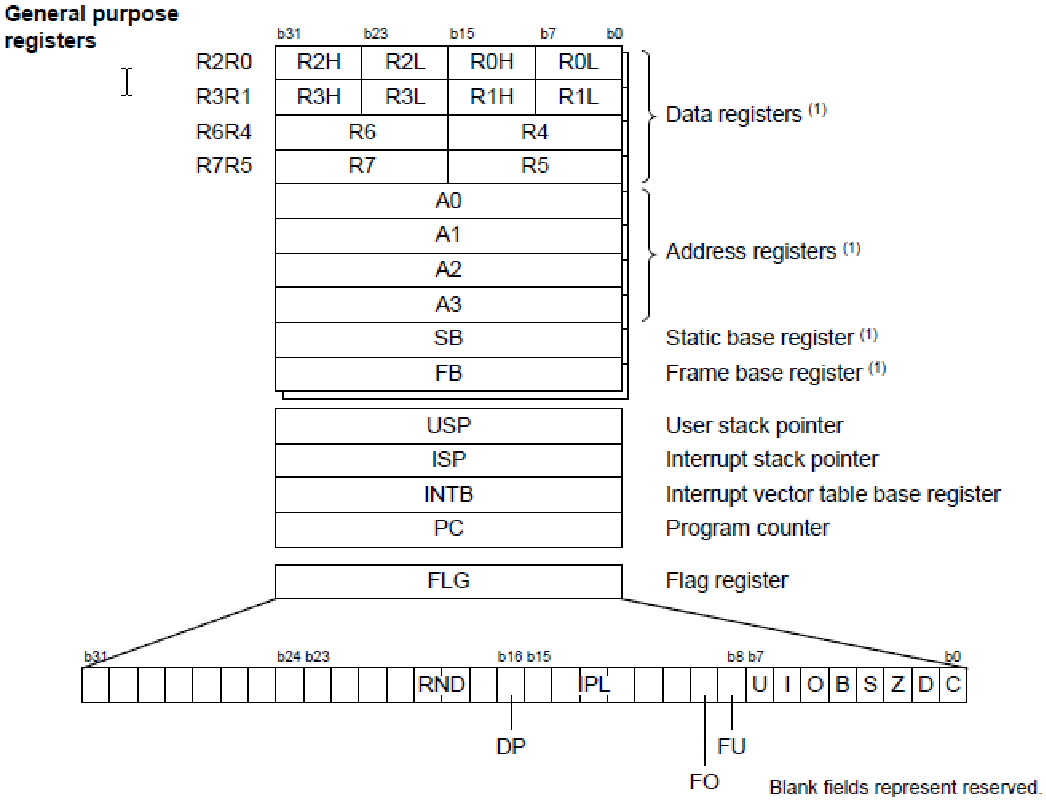

Figure 1. CPU registers of the R32C/111. All are tested by CPU_TestAll. Two different test patterns are written and read back to and from the registers.

2.1.1 Software API

Syntax	
void CPU_TestAll(void)	
Description	
Runs through all the tests detailed below in the following order:- 1. TestGPRs 2. TestIntRegs 3. TestStackRegs 4. TestPCReg 5. TestFlagReg It is the calling function’s responsibility to ensure no interrupts occur during this test. If an error is detected then external function ‘ErrorHandler’ will be called. See the individual tests for a full description.	
Input Pameters	
None	N/A
Output Parameters	
None	N/A
Return VALUES	
None	N/A

Syntax	
void TestGPRs(void)	
Description	
This function provides CPU General Purpose Registers Testing to test R0, R1, R2, R3, R4, R5, R6, R7, A0, A1, A2, A3 & FB in Register Bank0 and Bank1. If an error is detected then external function ‘ErrorHandler’ will be called.	
Input Pameters	
None	N/A
Output Parameters	
None	N/A
Return VALUES	
None	N/A

Syntax	
void TestIntRegs(void)	
Description	
This function provides Interrupt Table (INTB) register testing. If an error is detected then external function 'ErrorHandler' will be called. **Assumption**: R2R0 is used as a utility register in this test. This test will fail if R2R0 is faulty.	
Input Pameters	
None	N/A
Output Parameters	
None	N/A
Return VALUES	
None	N/A

Syntax	
void TestStackRegs(void)	
Description	
This function provides Stack (USP, ISP) and Static Base (SB) Registers testing. If an error is detected then external function 'ErrorHandler' will be called. **Assumption**: R2R0 and R3R1 are used as utility registers in this test. This test will fail if R2R0 or R3R1 are faulty.	
Input Pameters	
None	N/A
Output Parameters	
None	N/A
Return VALUES	
None	N/A

Syntax	
void TestFlagReg(void)	
Description	
This function provides the flag (FLG) register testing. If an error is detected then external function 'ErrorHandler' will be called. **Assumption**: R2R0 is used as a utility register in this test. This test will fail, if R2R0 is faulty.	
Input Pameters	
None	N/A
Output Parameters	
None	N/A
Return VALUES	
None	N/A

Syntax	
void TestPCReg(void)	
Description	
This function provides the Program Counter (PC) register test. This provides a confidence check that the PC is working. It tests that the PC is working by calling a function that is located in its own section so that it can be located away from this function, so that when it is called more of the PC Register bits are required for it to work. So that this function can be sure that the function has actually been executed it returns the inverse of the supplied parameter. This return value is checked for correctness. If an error is detected then external function 'ErrorHandler' will be called.	
Input Pameters	
None	N/A
Output Parameters	
None	N/A
Return VALUES	
None	N/A

3. TEST ENVIRONMENT

Development board: RSK-R32C11, 16MHz external clock, fcpu=50MHz.
MCU: R5F64112DFB
Tool chain: Renesas R32C/100 standard toolchain version 1.02.01
In-circuit debugger: E8a.

3.1 TOOL CHAIN SETTINGS

3.1.1 Linker

-L "nc100lib" -G -MS -O "$(CONFIGDIR)\$(PROJECTNAME).x30" –ORDER

4. BENCHMARKING RESULTS

The function execution time was measured using the pin(p0_0) and digital oscilloscope. A port pin was set high at function entry and low at function exit.

Code Size is the size of all functions in the specific file.

4.1 CPU Test Result

Note : Optimisation cannot be used for these tests.

Table 1 : R32C/111 CPU test result

Measurement	Result	
	Code size	Execution Time
CPU_TestAll	14 bytes	15.98 us
TestGPRs	436 bytes	10.01 us
TestIntRegs	56 bytes	1.15 us
TestStackRegs	148 bytes	2.40 us
TestPCReg	64 bytes	1.33 us
TestFlagReg	40 bytes	1.09 us

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/contact/

Revision History

Rev.	Date	Description Page	Summary
0.01	Mar 14 ‘14		First edition for review

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

Renesas Electronics Corporation

SALES OFFICES

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 4.0