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.

RENESAS

Notice

R8C/27

Frequency Measurement of the Low-Speed On-Chip Oscillator

Introduction

Many of the Renesas R8C/Tiny Series of microcontrollers include a low-speed on-chip oscillator with a nominal frequency of 125 kHz. This oscillator, however, is subject to wide variations in operating frequency and is not calibrated during factory test like the high-speed on-chip oscillator.

Using the flexibility of the Clock Generation Circuit on the R8C/Tiny Series it is possible to measure the frequency of the low-speed on-chip oscillator within an application. This allows the variations in frequency across temperature, voltage and different devices to be compensated for within the application code, and the oscillator to be used as an alternative clock source for the CPU or peripherals with much higher accuracy.

Use of the low-speed on-chip oscillator allows CPU and peripheral operation at much lower supply current than using the high-speed on-chip oscillator. This can save the cost of an external 32 kHz crystal, which has similar supply current characteristics. It also has the advantage of allowing higher CPU performance and better timer resolution than when using the 32 kHz crystal oscillator, which is important, for example, in networked detector applications.

This application note describes several methods of measurement of the R8C/Tiny Series low-speed on-chip oscillator using different reference clock sources and with and without the use of general purpose input/output pins. Example application software is included to demonstrate each method using the Renesas Starter Kit for the R8C/27 Group of microcontrollers.

Contents

1. Overview of the R8C On-Chip Oscillators

1.1 The High-Speed On-Chip Oscillator

The R8C/2x range of microcontrollers include a High Speed On-Chip Oscillator (HSOCO) with an approximate frequency of 40 MHz. This can be used as the clock source for the CPU and the peripherals, where a divider of at least 2 generates a system clock of maximum 20 MHz.

To obtain an accurate frequency on the HSOCO, on reset a value obtained during factory testing of the device and stored in a register (FRA1) trims the frequency to as near 40 MHz as possible. Trimming of the HSOCO is needed to compensate for the variations inherent in the CMOS manufacturing process used to implement the circuitry for the HSOCO.

The contents of the FRA1 register can be modified by the user to adjust the frequency of the HSOCO. Incrementing the value in FRA1 results in a lower frequency on the HSOCO, while decrementing it increases the frequency of the HSOCO.

The frequency of the HSOCO also varies with operating voltage and temperature. Figure 1 shows graphs of measured HSOCO frequency versus voltage and temperature for a sample of devices from the R8C/2x Group of devices.

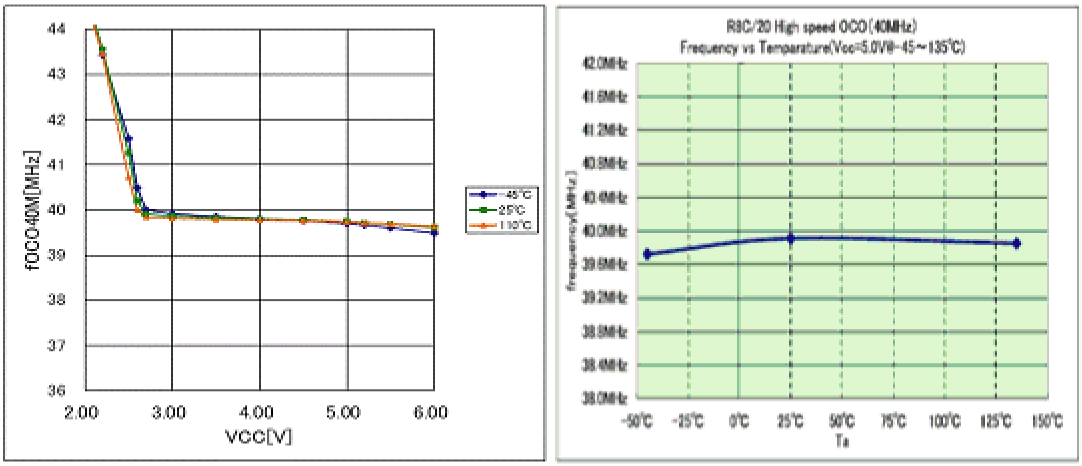


Figure 1: Effects of Operating Voltage and Temperature on HSOCO Frequency

The factory trimming value for the HSOCO is chosen to give the smallest possible frequency deviation from the nominal 40 MHz over the operating voltage and temperature of the device. At any set temperature and voltage the frequency of the HSOCO for a particular device can be expected to be stable over time.

1.2 The Low-Speed On-Chip Oscillator

The Renesas R8C/2x range of microcontrollers also contain a Low-Speed On-Chip Oscillator (LSOCO) with a nominal frequency of 125 kHz. This can also be used as the clock source for the CPU and peripherals, as well as providing a clock to system protection resources such as the Watchdog Timer and Voltage Detection circuits.

However, as there is no factory trimming of the LSOCO done at test, and indeed no register available to adjust the frequency, the variations inherent in the CMOS manufacturing process, as well as the variations due to operating voltage and temperature, mean that the frequency of the LSOCO can vary between 30 kHz and 250 kHz (for example, see R8C/26-27 Group Hardware Manual, Chapter 20: Electrical Specifications).

With such wide variation in operating frequency, it could be concluded that the LSOCO is not a useful resource for system designers beyond providing the initial system clock upon microcontroller reset and for providing a clock signal to system protection circuitry. In fact once the operating frequency of the LSOCO for a specific device is determined, the frequency stability over time for a given temperature and voltage is similar to that for the HSOCO.

Figure 2 shows the variations in frequency due to temperature and operating voltage for the Low-Speed On-Chip Oscillator for a small sample of devices. Note that this typical behaviour is not guaranteed on every device, and any design should allow for variation within the limits set out in the Datasheet specifications.

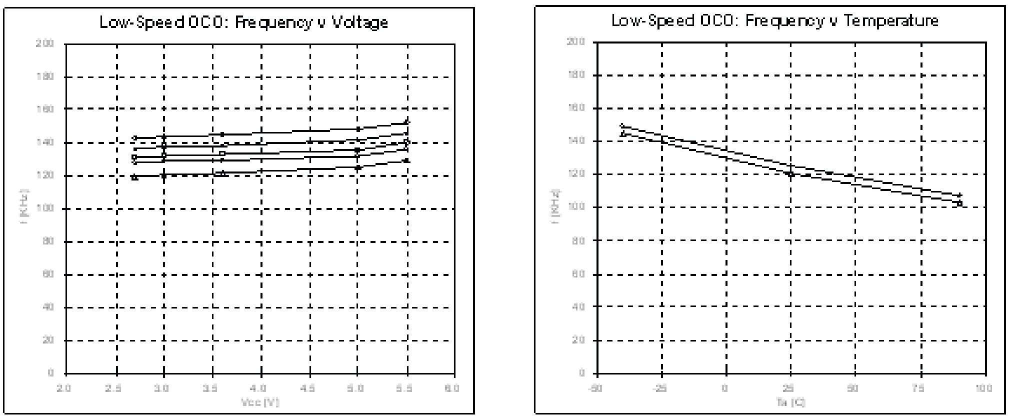


Figure 2: Effects of Operating Voltage and Temperature on HSOCO Frequency

If the actual frequency of the LSOCO for a specific device can be measured, and the variation in operating temperature and voltage is limited or compensated for, then the LSOCO can effectively be used as a relatively accurate system clock source.

2. Measuring the Frequency of the Low-Speed On-Chip Oscillator

Frequency measurement of the LSOCO using a high-speed external crystal or resonator

Timing components such as ceramic resonators and quartz crystals used for the input to microcontroller oscillator circuits have typically very high accuracy of 0.1% or better, and are stable over temperature and voltage. Using such a device adds cost to a system, but is often necessary to provide the required accuracy for communications timing, signal measurement or accurate time keeping over long periods.

In a system where such a component is used, the derived system clock can be used as a reference to determine the frequency of relatively inaccurate clock sources such as the on-chip oscillators found on the Renesas R8C/Tiny Series of microcontrollers.

The flexibility built into the R8C/2x Clock Generation Circuit allows, for example, the use of the clock provided by an external crystal or oscillator to be used for the CPU and main peripheral clock, while the LSOCO is used to provide a clock source for Timer RA.

Figure 3 shows a section of the Clock Generation Circuit of the R8C/27 Group of microcontrollers and the configuration of the CM01, OCD2 and FRA01 bits allowing Timer RA to be clocked from the LSOCO while Timer RC is clocked from XIN.

Timer RA is an 8-bit timer with 8-bit pre-scalar that can be configured in Pulse Output Mode to invert the output on pin TRA0 each time the timer underflows.

At the same time, 16-bit free running Timer C can be configured in Input Capture Mode to measure the time between edges of a signal on any of pins TRCIOA, TRCIOB, TRCIOC or TRCIOD. The clock source for Timer RC is provided by the main system clock derived from the external crystal or resonator.

[Note: Timer RD on the R8C/24 and R8C/25 Group can be configured in Input Capture Mode in the same way as Timer RC on other R8C/2x devices]

By connecting pin TRA0 to one of the TRCIO pins it is therefore possible to accurately determine the frequency of the LSOCO using an accurate external oscillator input component.

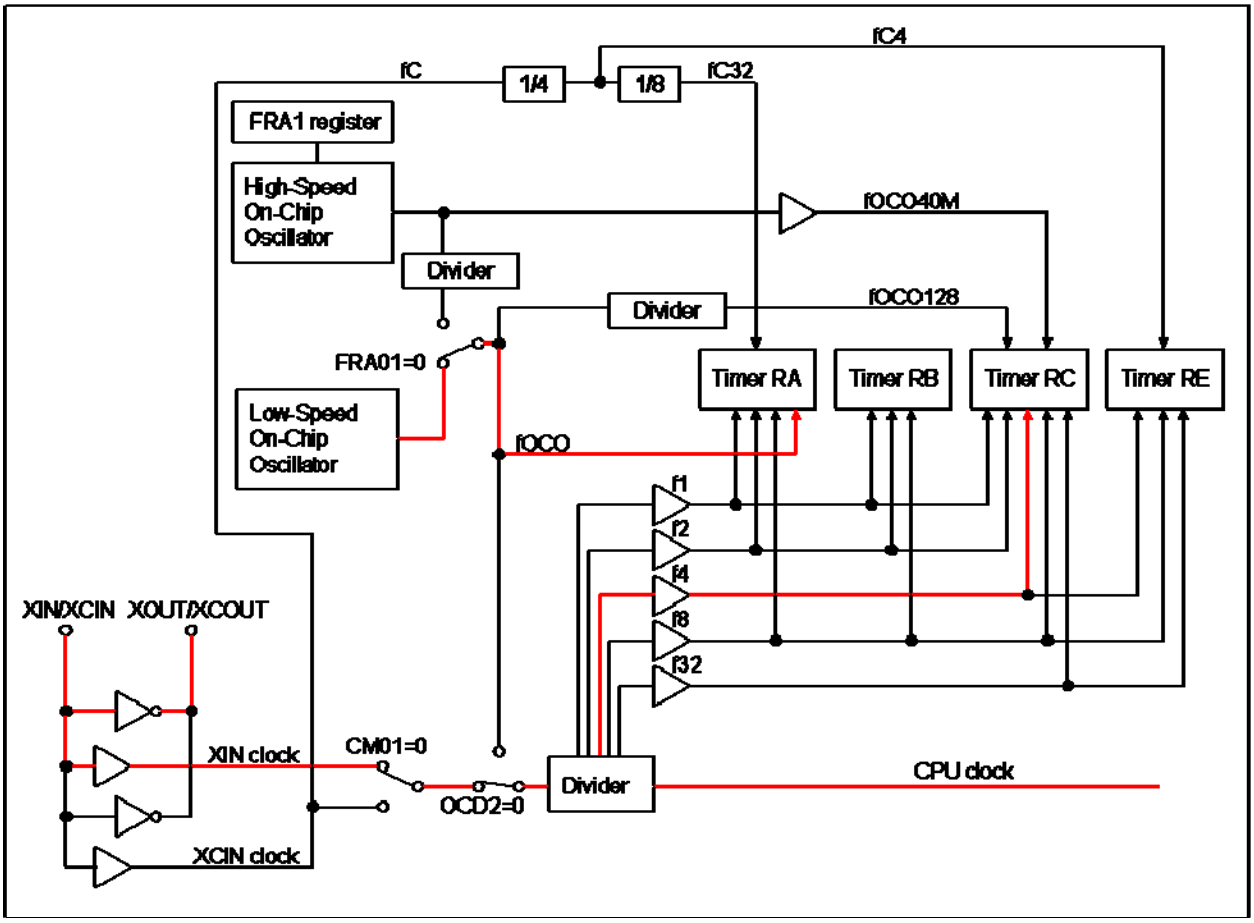


Figure 3: Clock Generation Circuit, simplified to show Timer clock source options. Measuring the LSOCO using an external clock oscillator as a reference.

In the example code, Timer RA is set up in Pulse Output Mode with no prescaler. With the LSOCO as the clock source, this will generate a pulse with period:

$$(1/f_{LSOCO}) * 256$$

With an LSOCO frequency of 125 kHz this will give a pulse period of 2.048 ms, however as the LSOCO can vary in frequency between 30 kHz and 250 kHz, the pulse could be as long as 8.533 ms or as short as 1.024 ms.

To measure the period of the pulse generated by Timer RA we configure Timer RC in Input Capture Mode. Using a clock source of f1, equal to the CPU clock without a divider, the maximum period that can be measured without an overflow is as follows:

$$(1/f_{XIN}) * 65{,}536$$

With a clock oscillator frequency f_{XIN} of 20 MHz, the free-running 16-bit Timer RC will overflow in approximately 3.277 ms. To ensure Timer RC will not overflow during the longest possible pulse period from Timer RA we need to use a clock source of f4 (f_{XIN} divided by 4).

We can calculate the actual frequency of the LSOCO from the Input Capture value as follows:

f_{LSOCO} = (IC value for f_{LSOCO} at 125 kHz) / (Actual IC value) * 125 kHz

To calculate the Input Capture value for f_{LSOCO} at 125 kHz with a clock oscillator of 20 MHz:

Timer RA pulse period = (1/125000) * 256 = 2.048 ms

Timer RC count period = (1/f4) = (1/5000000) = 200 ns

Input Capture value for f_{LSOCO} at 125 kHz

$= (2.048 * 10^{-3}) / (200 * 10^{-9}) = 10{,}240$

For example, for an actual input capture value of 8,000 the LSOCO frequency will be as follows:

10,240 / 8,000 * 125,000 = 160 kHz

The first software example LSOCO_XIN, documented in Appendix A and available to download from the Renesas website as a High-performance Embedded Workshop (HEW) project, shows measurement of the LSOCO using this method.

Program variables 'TmrC_IC_XIN' and 'LSOCOfreq_XIN' can be set to 'Auto Update', and a change in frequency with temperature observed by application of a suitable heat source to the microcontroller (put your finger on it!)

Frequency measurement of the LSOCO using an external 32 kHz watch crystal

In some systems the microcontroller may use an external watch crystal for maintaining a low power real-time clock, but may wish to utilise the LSOCO for a faster clock source without enabling the HSOCO which draws much more current.

In this case the absolute LSOCO frequency can be determined accurately, but we must use another configuration of the timer clock sources.

With XCIN running it is possible to use the derived clock divided by 32 as the clock source for Timer RA. Using a standard 32.768 kHz watch crystal, this will give a Timer RA underflow period of exactly 250 ms.

If we select the LSOCO as the CPU clock source, we can also use this as the clock source for Timer RC, as shown in Figure 4 below. In this configuration LSOCO is the higher frequency clock in the system, and the measurement time for the frequency is therefore much longer.

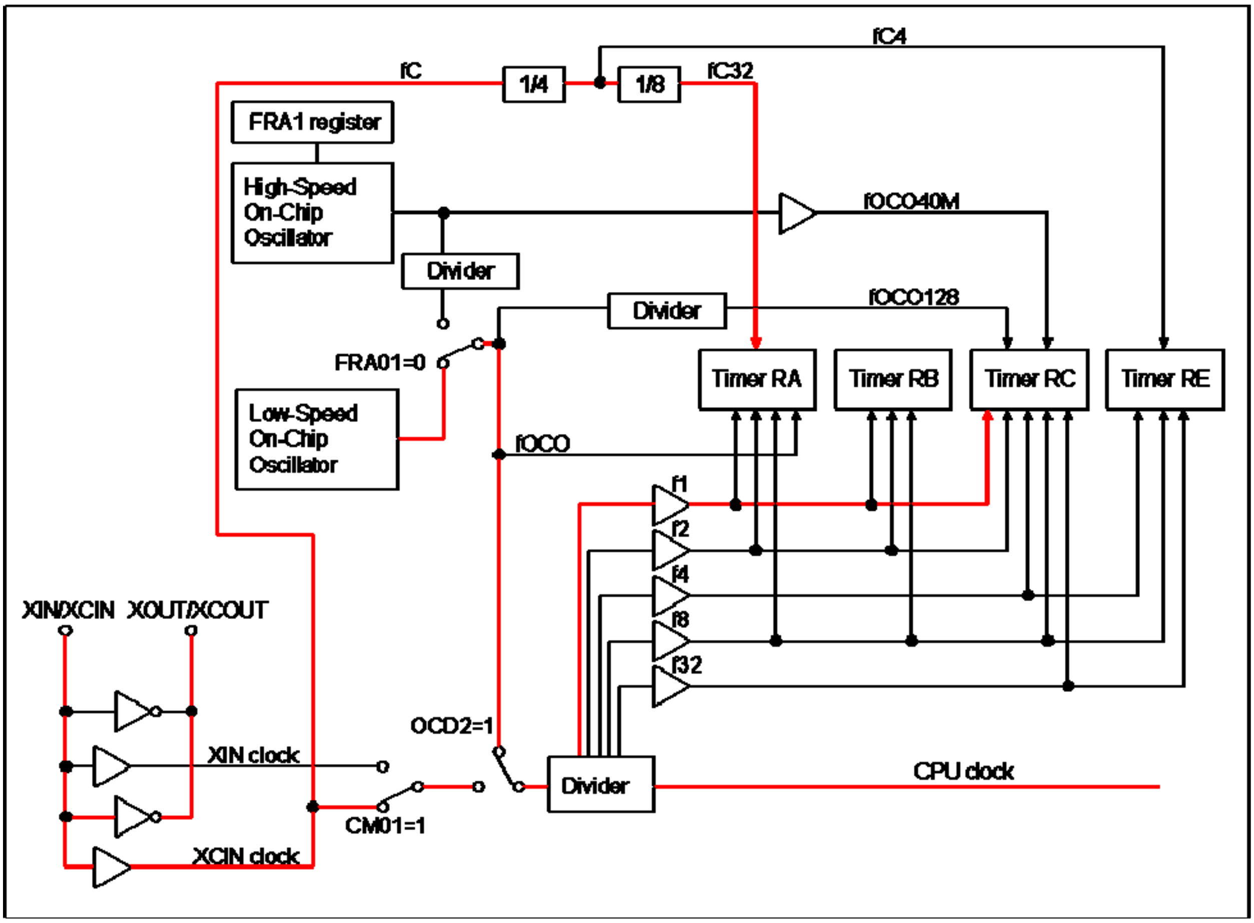


Figure 4: Clock Generation Circuit, simplified to show Timer clock source options. Measuring the LSOCO using a 32 kHz XCIN as a reference.

For an LSOCO frequency of 125 kHz we can calculate the Input Capture value:

IC value = Timer RA Pulse Output period / Timer RC clock period

= 0.25/(1/125000) = 31,250

We can then calculate the LSOCO frequency as follows:

f_{LSOCO} = (Actual IC Value / IC value for 125 kHz) * 125000

For a calculated frequency in kHz this can be simplified to the following:

f_{LSOCO} = (Actual IC Value / 250)

For example, for an Input Capture value of 32,500 the LSOCO frequency is:

f_{LSOCO} = (32,500 / 250) = 130 kHz

Measuring the frequency of the LSOCO without using input/output pins

It is usually desirable in microcontroller based systems to maximise the availability of general purpose pins and timer resources for application tasks. It is possible to avoid the use of the Timer RA Output pin (TRAO) and the Timer C Input / Output A pin (TRCIOA) utilised in the preceding methods of LSOCO frequency measurement.

We can configure Timer RA in Timer Mode to underflow in the same period as before without affecting the state of the TRAO pin. This will allow this pin to be used as a general purpose input / output pin or for Timer RA Output when the timer is reconfigured for another task.

We can then detect an underflow of Timer RA by polling the Timer RA underflow flag (TUNDF) bit in the Timer RA Control Register (TRACR). When an underflow is detected, the contents of Timer RC counter register (TRC) can be read and the frequency of the LSOCO can be calculated as before.

This method will work for any of the methods detailed above, minimising the use of valuable MCU resources for the LSOCO frequency measurement. Frequency measurement can also be done only periodically, so allowing the Timer RA and Timer RC to be reconfigured and used for other application tasks.

The second example HEW Project, ‘LSOCO_XCIN’, shows how the LSOCO frequency can be measured with a low frequency external crystal reference. Measurement with and without the use of the external pins for Pulse Output and Input Capture is demonstrated, with the SW2 push button used to switch between methods.

Care must be taken that the time taken to respond to an underflow event and execute the instructions to read the Timer RC counter register does not introduce unacceptable inaccuracy into the calculation. This is particularly the case when the CPU is being driven by the lower frequency clock, as in the method of LSOCO measurement used in the next section, using the High-Speed On-Chip Oscillator as a reference.

Frequency measurement of the LSOCO using the High-Speed On-Chip Oscillator

In many cost sensitive systems it is desirable to reduce the external component count of a microcontroller-based system to a minimum. We may therefore want to eliminate the need for all external microcontroller clock components and run the microcontroller using only the on-chip oscillators.

By careful consideration of the options available in the Clock Generation Circuit it is possible to use the factory calibrated High-Speed On-Chip Oscillator (HSOCO) to determine the frequency of the LSOCO.

As there is no external clock oscillator source, the CPU clock must be provided by one of the on-chip oscillators. Therefore, as we can see in Figure 3 above, the OCD2 bit must be set to 1. If we select the HSOCO as the CPU clock source by setting the FRA01 bit, then the LSOCO is effectively isolated and unavailable as a clock source to any of the timers.

Fortunately, if we clear the FRA01 bit and use the LSOCO as the CPU clock source, we can use the HSOCO directly as the clock source for Timer RC. In this configuration as in Figure 4 below, it is possible to calculate the frequency of the LSOCO in a similar way to before.

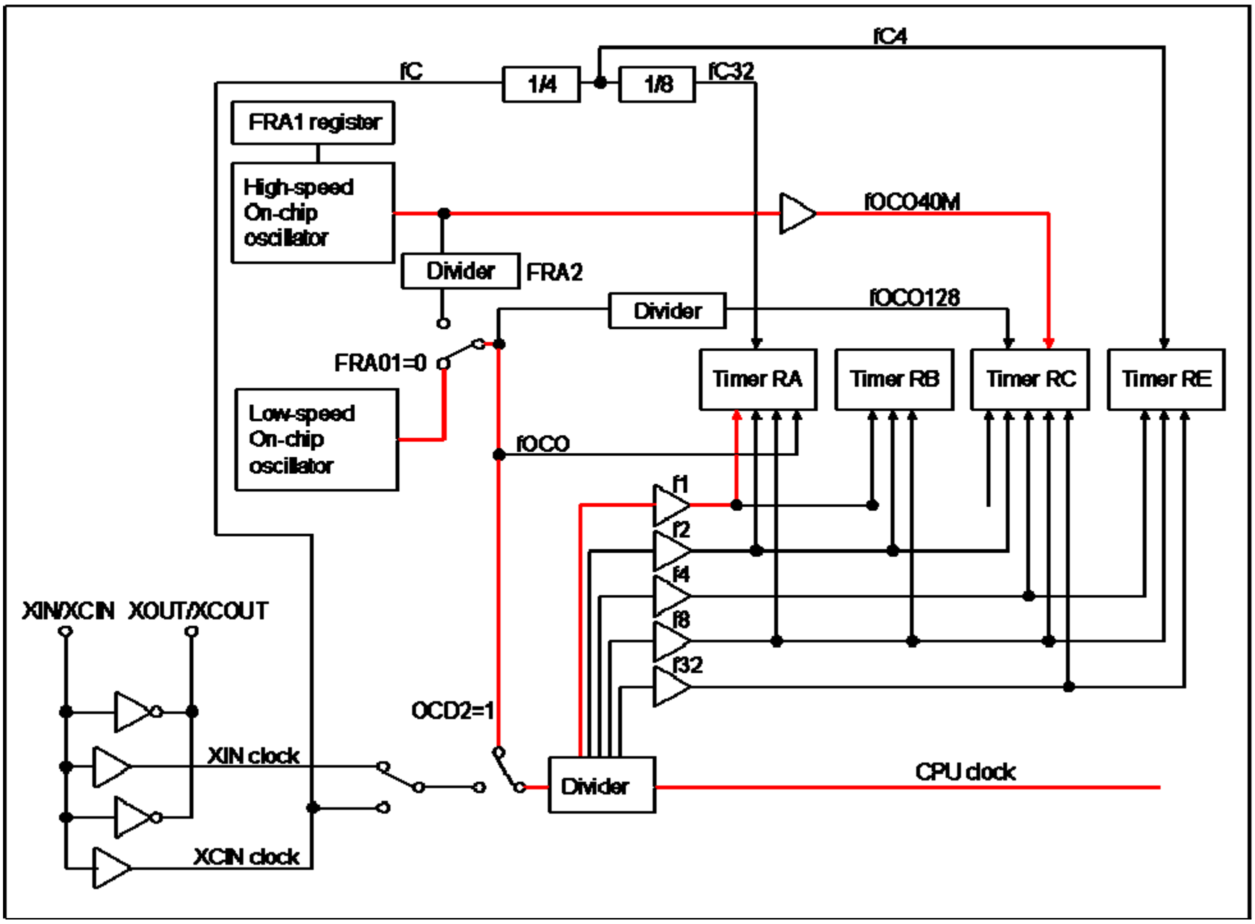


Figure 4: Clock Generation Circuit, simplified to show Timer clock source options. Measuring the LSOCO using the High-Speed On-Chip Oscillator as a reference.

As in the first method of LSOCO calculation, the period of Timer RA is given by the formula:

$$(1/f_{LSOCO}) * 256$$

An LSOCO frequency of 125 kHz will give a pulse period of 2.048 ms, however as before the underflow period could be as long as 8.533ms or as short as 1.024 ms.

The approximate overflow period for Timer RC can be calculated as follows:

$$(1/fOCO40M) * 2^{16}$$

$$= (25 * 10\text{-}9) * 65{,}536 = 1.64 \text{ ms}$$

As the HSOCO cannot be divided in this configuration to give a longer Timer RC overflow period, the free-running timer in Timer RC will likely overflow before a Timer RA underflow event occurs. We can handle this by detecting and counting the Timer RC overflows in software and including this in the LSOCO calculation. The third example project detailed in Appendix A shows this method of LSOCO frequency measurement implemented.

Other Methods of LSOCO measurement

It may be desirable to use another external timing source as a reference for measurement of the LSOCO frequency. This is possible using one of the above described Clock Generation Circuit configurations or using another configuration. Some examples of possible sources include:

- Stable AC mains power supply frequency
- Received serial communications clock, or data frame length of a known period
- Square wave generated by production automatic test equipment (ATE)

Using one of the first two methods above can eliminate the need for an external crystal or resonator in the application while still allowing accurate dynamic measurement of the LSOCO frequency. Using an ATE supplied clock source would allow measurement during the production phase of the system. The LSOCO frequency would be measured at a similar voltage and temperature to that expected in the application, and the measurement stored in non-volatile memory to be used later in the application software.

3. Application Design Considerations

3.1 Using Interrupts

When using the Input Capture function on Timer RC, the Timer RC interrupt can be used to avoid the need to poll the Input Capture flag in the Timer RC status register. The Input Capture value can be read during the Timer RC interrupt service routine and the LSOCO frequency calculated then or elsewhere in the application code.

Similarly, if a measurement method avoiding use of GPIO pins is used, Timer RA interrupt can be used to detect Timer RA underflow. The contents of the Timer RC counter register can then be read during the interrupt service routine and the LSOCO frequency calculated then or elsewhere in the application code.

For clarity, the example code does not use interrupts, but it is likely to be desirable to use one of the interrupts in this way in an application, as other tasks can then be undertaken during the Timer RA underflow period.

Appendix A: Example Software

1. Notes on the Example Software

The software that accompanies this Application Note was written using the Renesas High-performance Embedded Workbench (HEW) Version 4.03.00.001.

The example software is contained within 3 HEW Projects named "LSOCO_XIN", "LSOCO_XCIN" and "LSOCO_HSOCO". Each project demonstrates measurement of the Low-Speed On-Chip Oscillator using different clock sources as reference.

The first project "LSOCO_XIN" demonstrates the method of measurement of the LSOCO detailed in Section 2.1 "Frequency measurement of the LSOCO using a high-speed external crystal or resonator". The second project "LSOCO_XCIN" demonstrates the method of LSOCO measurement detailed in Section 2.2 "Frequency measurement of the LSOCO using an external 32 kHz watch crystal" and also shows the method explained in Section 2.3 "Measuring the frequency of the LSOCO without using GPIO pins". The third project "LSOCO_HSOCO" demonstrates the method explained in Section 2.4 "Frequency Measurement of the LSOCO using the High-Speed On-Chip Oscillator".

Variables used to store captured values from Timer RC and other parameters, as well as the calculated values of the frequency of the LSOCO, are declared globally to allow them to be viewed in Watch windows within the HEW debugging environment.

The target hardware used is the Renesas Starter Kit for the R8C/27 Group of microcontrollers. Modifications to the RSKR8C27 hardware were made as follows:

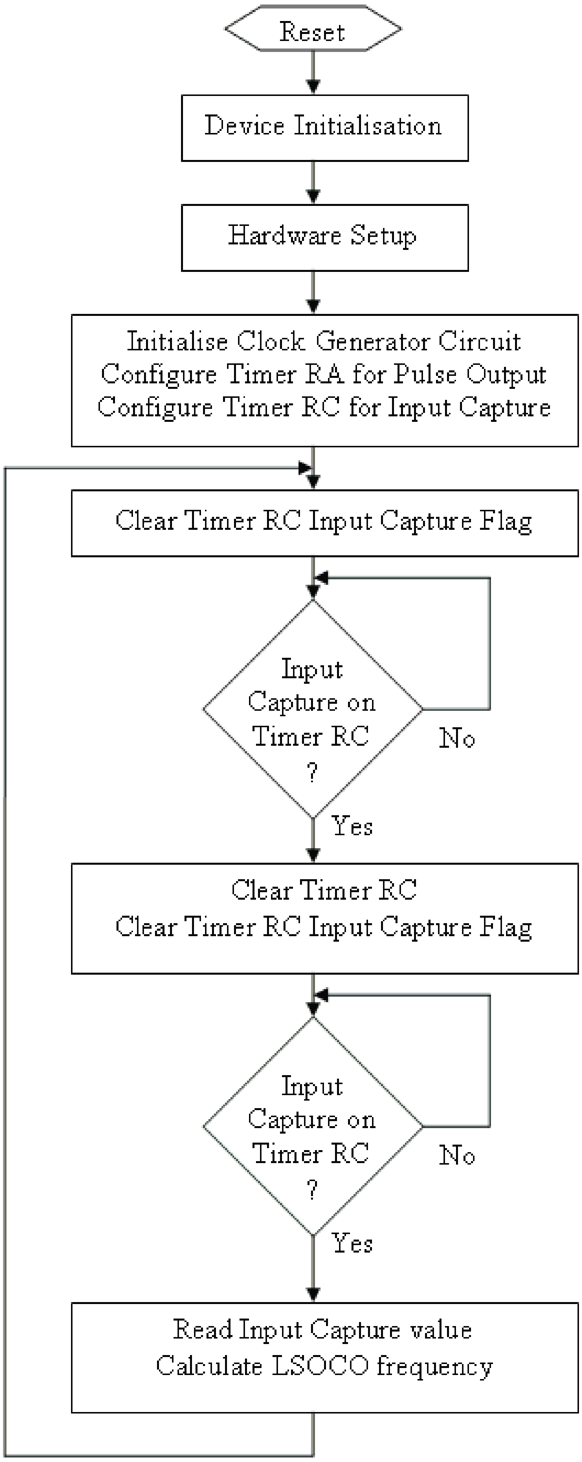
- Connect the Timer RA Output (TRAO) pin to the Timer RC Input Output A (TRCIOA) pin

- For the second project disconnect X1 20MHz XTAL by removing 0R resistors R13 and R14 and connect X2 32.768 kHz XTAL by fitting 0R resistors R11 and R12

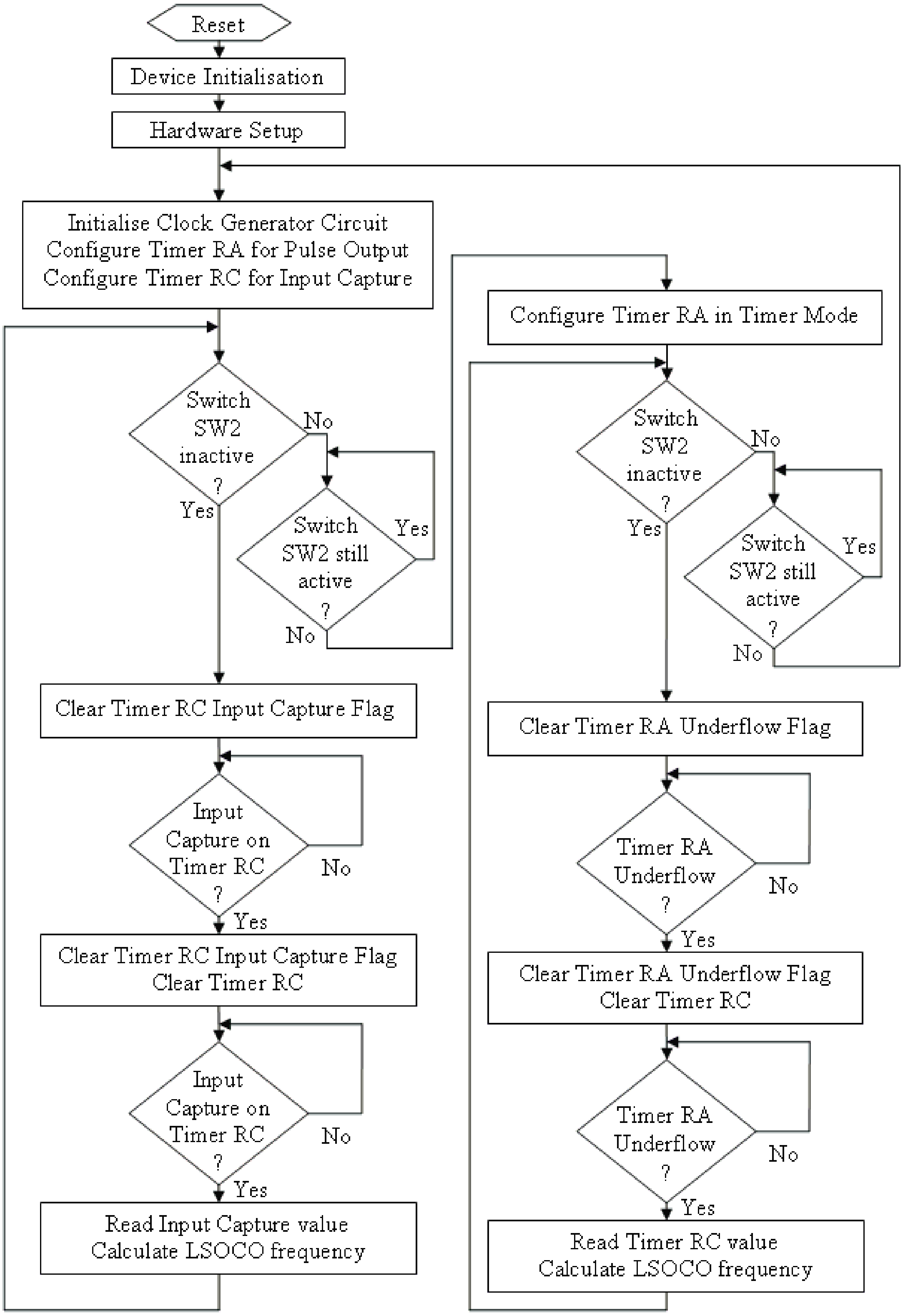
Note that although all projects were created using the standard HEW project generator for the RSKR8C27, the call to function "ConfigureOperatingFrequency( )", which enables the high-speed external oscillator in the *hwsetup.c* file, is commented out and not called in the second and third projects, as the external crystal is not used and may not be available.

2. Example Software Flow Diagrams

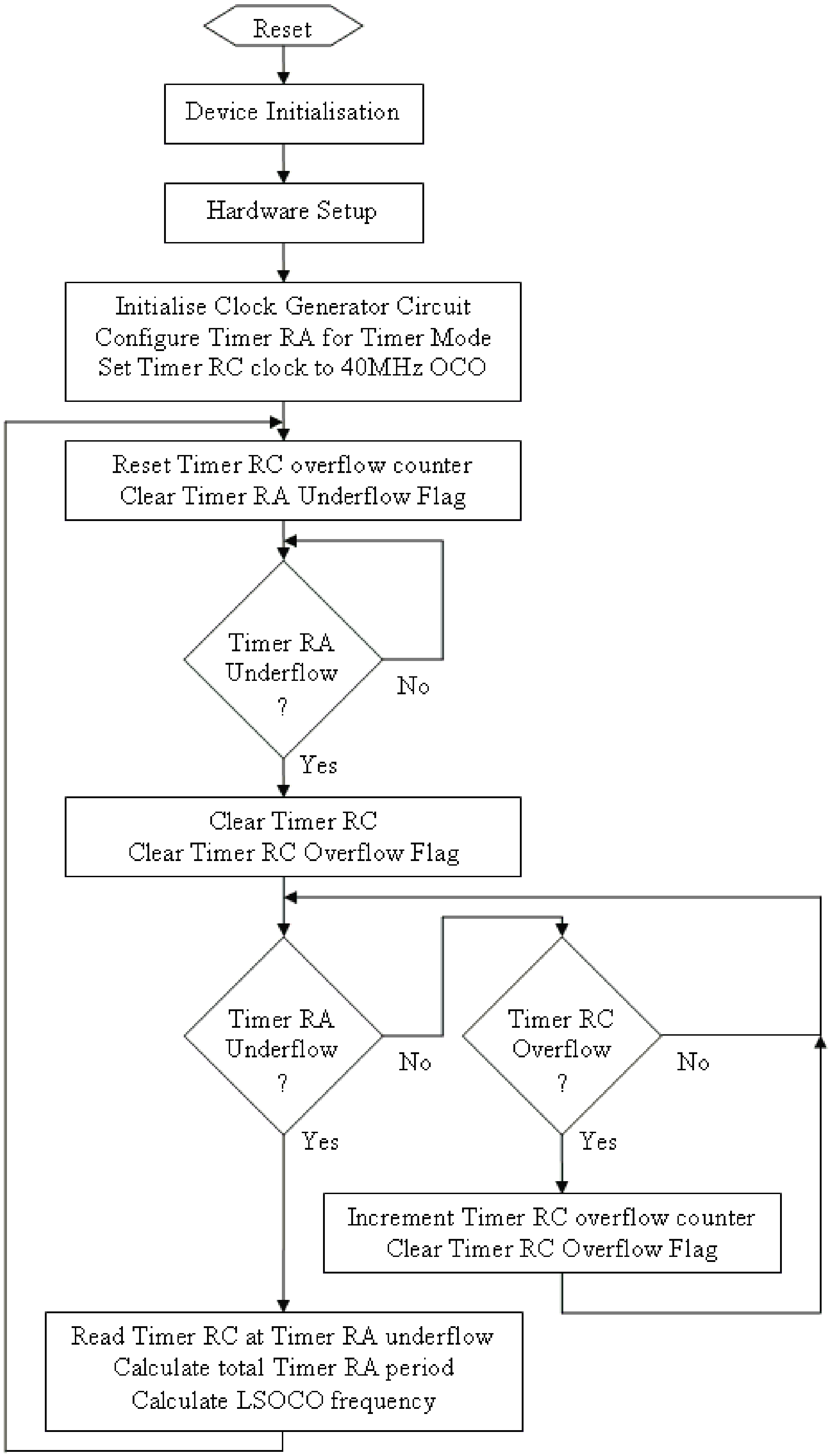
2.1 Example Project “LSOCO_XIN”



2.2 Example Project "LSOCO_XCIN"



2.3 Example Project "LSOCO_HSOCO"



<This page left blank intentionally>

Website and Support

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

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

Notes regarding these materials