Application Report
SLAA338–October 2006

Random Number Generation Using the MSP430

Lane Westlund *MSP430 Applications*

ABSTRACT

Many applications require the generation of random numbers. These random numbers are useful for applications such as communication protocols, cryptography, and device individualization.

Generating random numbers often requires the use of expensive dedicated hardware. Using the two independent clocks available on the MSP430F2xx family of devices, it is possible to generate random numbers without such hardware.

1 Introduction

The very-low-frequency oscillator (VLO) and digitally controlled oscillator (DCO) are two independent clock systems, each having its own timing source. Because the clocks are independent, the time difference between edge transitions of these two clock sources varies. The timing differences between these two clock systems can be exploited to generate a stream of random bits. In one VLO clock cycle, there will always be roughly the same number of DCO pulses. However, due to the fact that the two clock sources vary independently from each other, whether this number of pulses is even or odd is not predictable. More importantly, this number is not predictable even if the previous result is known.

Therefore, Timer_A can be configured to continuously count the number of DCO clock cycles per VLO clock cycle, and the least-significant bits (LSBs) from 16 of these results can be strung together to form a random 16-bit integer.

2 Setup

Timer_A is setup in capture mode. SMCLK is set to the DCO and set as the input clock to Timer_A. ACLK is set to the VLO, which is the trigger for the capture. Timer_A counts the number of DCO clock pulses before the next VLO low-to-high transition occurs. The number of DCO clock pulses is saved by the timer in a Capture/Compare Register (CCR). The LSB from the CCR is saved by left shifting it into a CPU register (R12). This process is repeated until 16 LSBs have been saved, forming a 16-bit result that is almost random.

3 Adding Randomness

The example software included also takes several measures that are designed to increase the randomness of the numbers measured and to make the overall system less predictable.

- Each time a CCR LSB is shifted, the BCSCTL1 register has the number five added to it. This addition changes the RSEL bits, causing the DCO speed to change relative to the VLO through each loop. Although any number could be used, it was found that five caused a large enough step change to significantly vary the DCO with respect to the VLO.
- Each time a CCR LSB is shifted, the two LSBs from the R12 register are XORed into the DIVA bits of the BCSCTL1. The DIVA bits control the divider used for the VLO before it reaches the timer. This also changes the relationship between the VLO and DCO as measured by the timer.

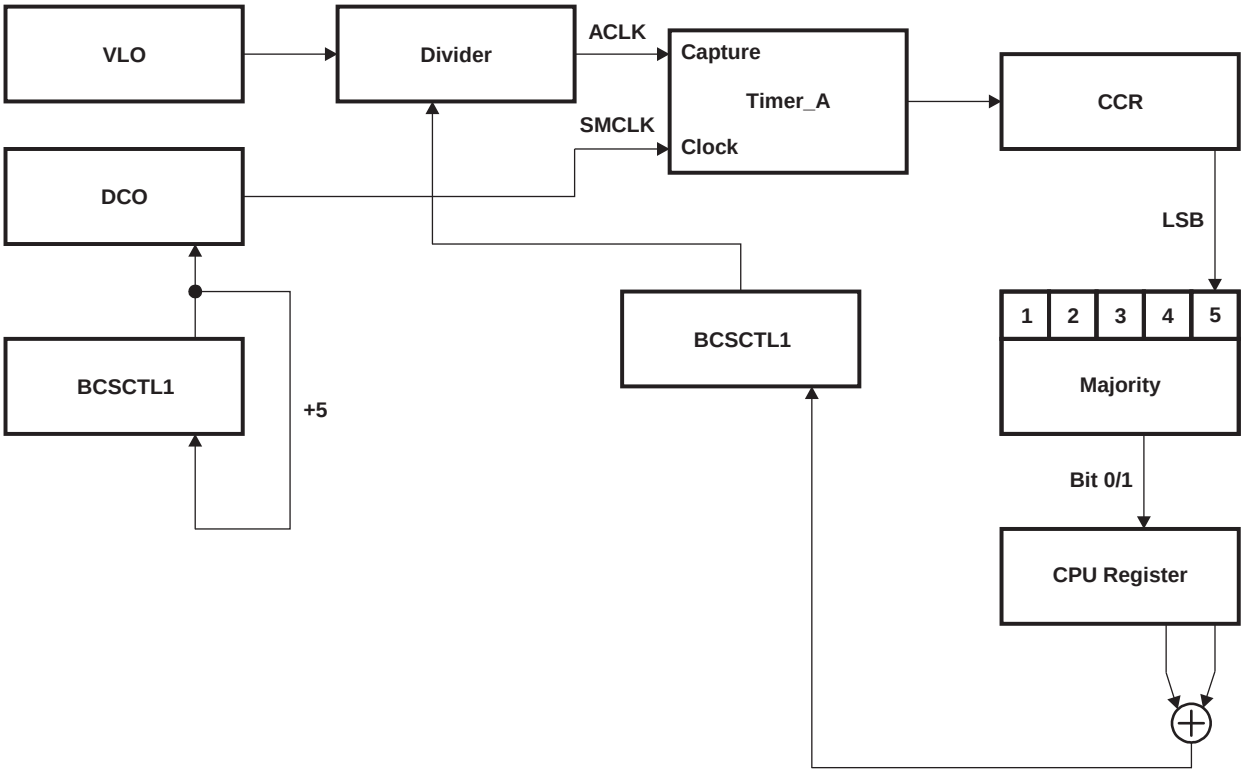
- Each result bit is actually the result of a majority vote of five loops. Each loop generates its own LSB from the CCR as described earlier, but the majority vote of five is used to select the final resultant bit. This majority vote system more evenly distributes the results and causes them to pass the poker test for randomness as described in Section 6.

4 Usage

The methods described in Section 3 that add randomness have been shown through the testing described in Section 6 to more evenly distribute the resultant data. They do, however, make changes to the clock system of the MSP430, which could interfere with other running processes. Such considerations should be made when designing a system.

If such clock-system changes are not acceptable for a running application, it still may be possible to make use of the methods presented in this application report. Instead of using these methods each time a random number is desired, an initial seed could be generated. This seed value would use these methods, and do so at device startup, before any other processes that rely on the clock system have begun. This seed could be used as a seed for a pseudo-random number generation (PRNG) algorithm such as a steam cipher.[1] This method, although more CPU intensive, could also yield numbers with good random properties, depending on the PRNG used.

5 Overview



6 Testing for Randomness

A series of statistical tests for randomness is described by the Federal Information Processing Standards (FIPS). Included with the source code for this random number generator is a C file called 'fips_test.c'. This source code implements the FIPS 140-2 tests for randomness. The results of these tests are saved in global variables and can be viewed with a debugger in order to verify the functionality of the random number generator.

Although the tests are included for statistical information, this application report has not undergone official FIPS certification or testing.

The methods outlined in this application report make use of timing differences between ACLK and SMCLK in the MSP430. The VLO was chosen due to its device-to-device variation as well as drift with temperature and voltage. It should be noted that this drift only adds to the observed entropy, and sufficiently random numbers are generated under constant temperature and voltage conditions. These factors also add to the randomness that are observed between devices.

Note that the FIPS 140-2 test is only a necessary requirement for the random numbers generated here. For an application that requires cryptographic secure random numbers, it is needed to create a model of the system and evaluate the entropy that is generated.[2]

7 LFXT1 Versus VLO

ACLK could also be sourced from LFXT1 with a 32-kHz crystal. This method can also be used to generate random numbers, but it is less reliable, due to the more predictable nature of the 32-kHz crystal. It is also less secure, because one of the clock sources is now sourced into the microcontroller externally. This fact could be used by an attacker looking to influence the selection of random numbers and compromise a system. These facts should be weighed against the security requirements of a system when deciding to use LFXT1 as the ACLK source for random number generation.

8 References

1. U. Kaiser, *Hermes8: A Low-Complexity Low-Power Stream Cipher*, http://eprint.iacr.org/2006/019.pdf
2. W. Schindler, *Evaluation Criteria for True (Physical) Random Number Generators Used in Cryptographic Applications*, CHES2002 workshop, http://ece.gmu.edu/crypto/ches02/talks_files/Schindler.pdf
3. *MSP430x2xx Family User's Guide*, TI literature number SLAU144
4. *MSP430x1xx Family User's Guide*, TI literature number SLAU049
5. FIPS PUB 140-2, National Institute of Standards and Technology, http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
Low Power Wireless	www.ti.com/lpw	Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265