ELectronics EXpress

LETTER

IEICE Electronics Express, Vol.11, No.15, 1–9

Low-power asynchronous digital pipeline based on mismatch-tolerant logic gates

Gonzalez-Carabarin Lizeth[a)], **Tetsuya Asai, and Masato Motomura**

Graduate School of Information, Science and Technology, Hokkaido University, Kita 14, Nishi 9, Kita-ku, Sapporo 060–0814, Japan

a) *lizeth@lalsie.ist.hokudai.ac.jp*

Abstract: We present design of an asynchronous digital pipeline based on stochastic resonance gates (SR gates). These gates allow mismatch-tolerant and low-power designs by utilizing the beneficial role of noise (stochastic resonance effect). However, unpredictable delays appear in SR gates due to their dependence on stochastic processes. Therefore, applications are found in asynchronous circuits. We demonstrate the performance of a three-stage-asynchronous pipeline with dual-rail data encoding. Electrical simulations were done for a 0.18-µm CMOS technology, and confirmed the pipeline performance regardless of transistor mismatches and consuming low power (100 picowatts). Moreover, experimental results based on a macrosystem confirmed the pipeline performance.

Keywords: asynchronous digital circuits, low power, logic circuits, stochastic resonance

Classification: Electron devices, circuits, and systems

References

[1] L. Gammaitoni, P. Hanggi, P. Jung and F. Marchesoni: Rev. Mod. Phys. **70** (1998) 223. DOI:10.1103/RevModPhys.70.223

[2] M. D. McDonnell and D. Abbott: PLOS Comput. Biol. **5** (2009) e1000348. DOI:10.1371/journal.pcbi.1000348

[3] A. R. Bulsara, A. Dari, W. L. Ditto, K. Murali and S. Sinha: Chem. Phys. **375** (2010) 424. DOI:10.1016/j.chemphys.2010.06.015

[4] L. Gonzalez-Carabarin, T. Asai and M. Motomura: NOLTA **E5-N** [4] (2014) in press.

[5] C. Mayer: *Asynchronous Circuit Design* (J. Wiley & Sons, Canada, 2001).

[6] S. Nowick and M. Singh: IEEE Des. Test Comput. **28** [5] (2011) 8. DOI:10.1109/MDT.2011.71

[7] P. A. Beerel, R. O. Ozdag and M. Ferretti: *A Designer's Guide to Asynchronous VLSI* (Cambridge University Press, 2010).

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

1 Introduction

Cutting-edge technologies for device miniaturization are reducing device size to molecular or atomic scales. However, as such dimensions are reached, side effects are inevitable, such as uncontrolled noise and parameter mismatches. Therefore it has become essential for designers to find new strategies to avoid these drawbacks. Alternative approaches include biological-inspired designs, where one of the approaches that has become popular is the utilization of noise to enhance certain tasks; this phenomenon has been well documented and it is known as stochastic resonance [1, 2]. Logic gates assisted by noise has been proposed previously [3]; however, the main disadvantage of these gates is the presence of hazards due to noise introduction, and the requirement of different bias voltage sources to select logic operations. In a previous study [4], we proposed the combination of nonlinear systems with hysteresis and the stochastic resonance effect to design mismatch-tolerant logic circuits that consume low power regardless of mismatches. The transistor-based circuit is shown in Fig. 1a, and the path connections to implement SR gates is shown in Fig. 1b. Another key feature of the SR gates is that their output is stable regardless of noise; this is due to the hysteresis characteristic that provides two thresholds (positive feedback at Fig. 1b provides the hysteresis characteristic). Moreover, no additional bias circuits are required to select logic operations. Simulations and experimental results showed that with the introduction of an intentional mismatch, SR logic gates correctly generate the four basic logic functions NAND, AND, NOR and OR with low power supply (0.35 V).

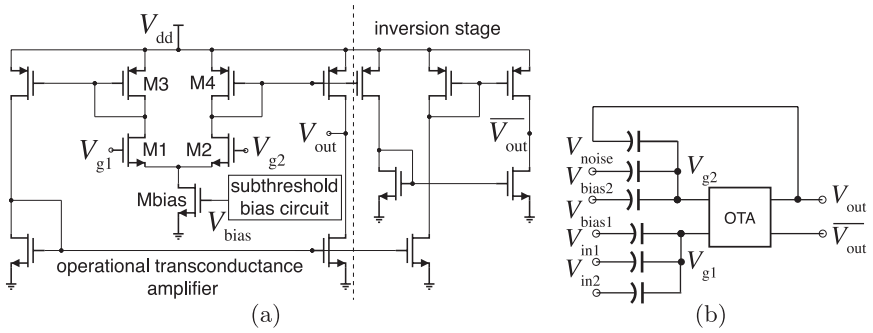

Fig. 1. (a) shows the operational transconductance amplifier (OTA) used in our previous work [4]. (b) shows the path connection of the OTA to implement SR gates, where V_{bias1} and V_{bias2} serve as selectors for the logic operations, V_{in1} and V_{in2} are logic inputs and V_{noise} is noise input; V_{g1} and V_{g2} of (a) are the weighted sum of the input capacitors of (b) (implemented with a floating-gate MOSFET)

However, because the response of the SR logic gates depends on noise, there is an unpredictable delay in their response. Consequently synchroniza-

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

tion problems are evident in synchronous circuits based on SR logic gates. Therefore, proper applications for the SR logic gates are found in the field of asynchronous circuit design, in which performance does not depend on a central clock, but on handshake protocols [5]. Asynchronous circuits allow the design of delay-insensitive circuits (DI circuits) because they rely on signals that indicate when a task is accomplished. In this study, we propose the design of an asynchronous digital pipeline to demonstrate the performance of SR logic gates regardless of the presence of mismatches and unpredictable delays in the response. Here, we demonstrate its performance through simulations and experimental results.

2 Brief overview of asynchronous circuit design

Although there is no established methodology to design asynchronous circuits, these types of circuits are based on requirement (*req*) and acknowledgment (*ack*) signals between stages. These signals indicate when new data are required (indicated by a change of state of *req* signal) to perform a certain task and when the current task is accomplished (indicated by a change of state of the *ack* signal). Execution of instructions among stages relies on the exchange of these signals.

Further, it is necessary to choose the appropriate data encoding and protocols according to the desired application. Asynchronous circuits are divided into two categories based on their data encoding: single-rail (or bounded) data and dual-rail data encoding. The former is used when the delays are known precisely, and the latter represents a more robust data encoding because it allows the design of DI circuits and that is the reason it is used for our purposes. Dual-rail data encoding refers to converting one bit to two-bit encoding (Fig. 2), where valid data are given by $\{0,1\}$ and $\{1,0\}$ which represents 0 and 1 logic respectively. $\{0,0\}$ is a spacer between valid data and $\{1,1\}$ is not used. Another advantage of the dual-rail data encoding is that it contains the *req* signal implicit in the data (where $\{0,1\}$ or $\{1,0\}$ represents a *req* signal equal to one); therefore, an additional channel for *req* signal is not necessary.

	Codeword
Data 0	{0, 1}
Data 1	{1, 0}
Spacer	{0, 0}
Not used	{1, 1}

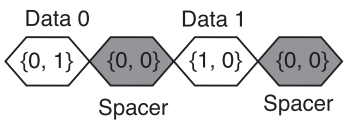

Fig. 2. Dual-rail data encoding

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

Dual-rail data encoding requires a four-phase protocol for data transmission. The four-phase protocol resets the signals *req* and *ack* when the current task has been accomplished, in contrast with the two-phase protocol, where a reset it is not necessary. To implement dual-rail encoding, data are necessary to design dual-rail logic gates based on their single-rail counterparts. In this framework, we show the disgn of a dual-rail NAND gate based on SR logic gates. The basic diagram is shown in Fig. 3, where the invert operation consists simply of the inversion of output bits. The circuit consists of SR NAND and SR AND gates previously proposed in [4].

On the other hand, in order to design high performance asynchronous circuits, the asynchronous pipeline is one of the main elements [6]. Thus, we present the implementation of an asynchronous pipeline based on a dual-rail data encoding. Fig. 4 shows the basic block diagram for a dual-rail based asynchronous pipeline for n stages. Here, data transmission relies on validation of input data. If $L_{0,0}$ and $L_{0,1}$ are either $\{0,1\}$ or $\{1,0\}$ (*req* equals to one), then data are valid, consequently the acknowledge signal $L_{\mathrm{ack},0}$ changes its state, resetting both channels. Once, reset has occurred, $L_{\mathrm{ack},0}$ changes again its state, sending new data to both channels. This process is repeated in the next stages to transmit data. Further, choosing the pipeline configuration depends on the desired application. In this study, we present the implementation of the Weak Condition Half Buffer pipeline (WCHB pipeline) [7]. Owing to the fact that the WCHB pipeline is a gate-based configuration with dual-rail data encoding, it can be easily implemented with the current SR logic gates to design a delay insensitive pipeline. Fig. 5 shows a basic configuration of the WCHB pipeline. In this configuration, NAND gates validate input signals. When data are valid, $L_{\mathrm{ack},0}$ is equal to zero, resetting the signals. Data capture will be done only when $L_{\mathrm{ack,n+1}}$ is equal to one (the next stage has accomplished data transmission successfully). Here, the basic function of the C-element is maintaining the current state until a reset is done; the basic performance of the C-element is as follows: only when both inputs are set to zero or one the output will be zero and one respectively, otherwise, the output will be held with its previous state. We have already proposed a possible implementation of the C-element [4], where the same internal configuration of the SR logic gates is used. The next section will show the simulation results of these examples, designed with the SR logic gates.

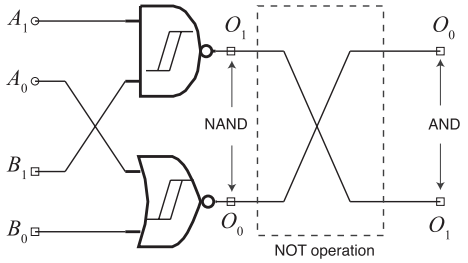

Fig. 3. Dual-rail SR AND/NAND gate

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

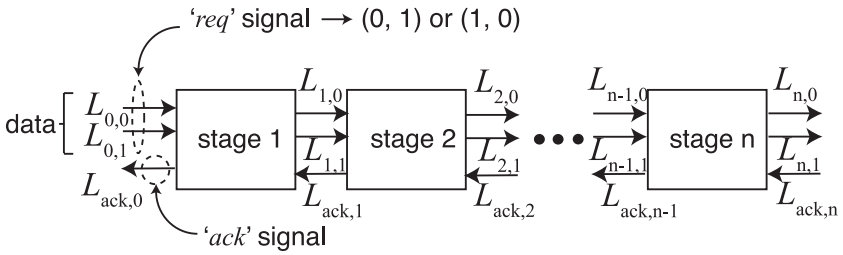

Fig. 4. Asynchronous pipeline

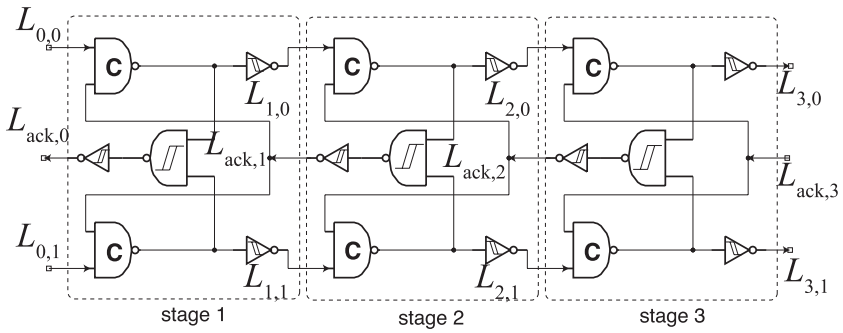

Fig. 5. Three-stage WCHB asynchronous pipeline

3 Simulations and experimental results

In this section, simulation and experimental results are presented. Simulations were performed in SPICE for a 0.18-µm CMOS technology (subsection 3.1). Experimental results for the WCHB pipeline were obtained with a macro system and they are presented in subsection 3.2. Subsection 3.3 contains an analysis of the WCHB pipeline.

3.1 Dual-rail NAND/AND gate

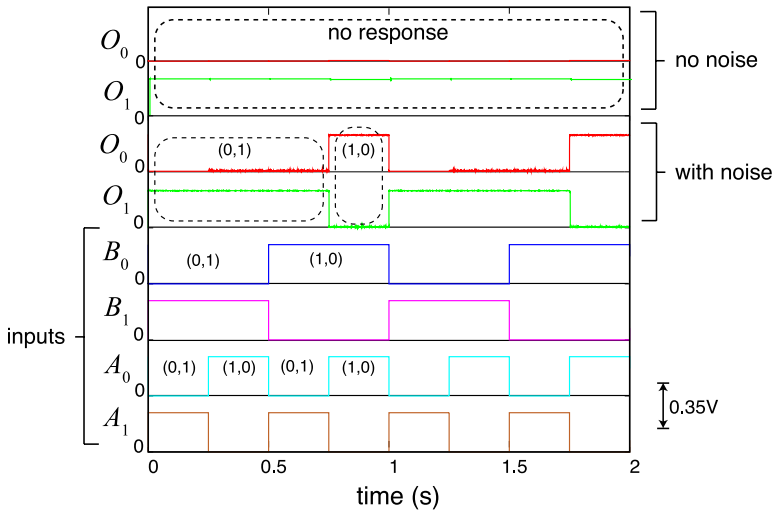

Fig. 6. SPICE simulations for the dual-rail SR AND/NAND gate (case with noise)

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

Here we show electrical simulations of the dual-rail NAND/AND gate, where the power supply was set to $V_{dd} = 0.35\,V$ and the standard deviation of noise to $\sigma_{V_{noise}} = 0.27\,mV$. Fig. 6 shows the simulation results, where two cases are present: without noise and with noise. In the first case, owing to the presence of mismatches, the logic functions are not generated. In the second case, when noise is applied, the logic functions are recovered, demonstrating the beneficial effect of noise.

3.2 WCHB pipeline based on SR logic gates

In this subsection we present the main application of this study, that is the implementation of a three-stage WCHB pipeline. The simulation parameters are the same as those used in the previous subsection. Fig. 7 shows the simulation results for the three stages.

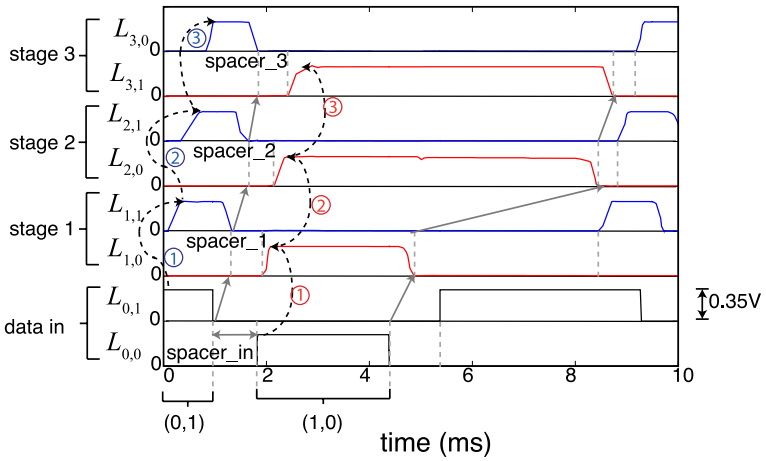

Fig. 7. SPICE simulations for the WCHB asynchronous pipeline based on SR logic gates (case with noise)

The signals at the bottom of the figure represent input data, in which every time data have been passed to the next stage (*ack* signal = 1), is generated new data, alternating between $\{0,1\}$ and $\{1,0\}$ (0 and 1) with the reset (spacer) between valid data. The red lines represent data of channel 0 ($L_{n,0}$), and the blue ones represent data of channel 1 ($L_{n,1}$). Note that data transmission is accomplished regardless different values of delays among stages. Fig. 8 shows in greater detail the control signals *ack* and *req* that help accomplish data transmission between stages 1 and 2.

In addition to the simulations, a three-stage WCHB pipeline was built using an operational amplifier OP284 connected with a positive feedback to generate a hysteresis characteristic. SR logic gates were built using the configuration shown in Fig. 9. Their performance and parameters can be reviewed in more detail in [2]. Note that the circuit used for experimental results does not correspond to the practical implementation of the circuit for SR logic gates of Fig. 1, but it is used to demonstrate how the noise can be utilized in nonlinear circuits to recover logic functions in the presence of mismatches. Therefore, parameters such as power consumption and delay

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

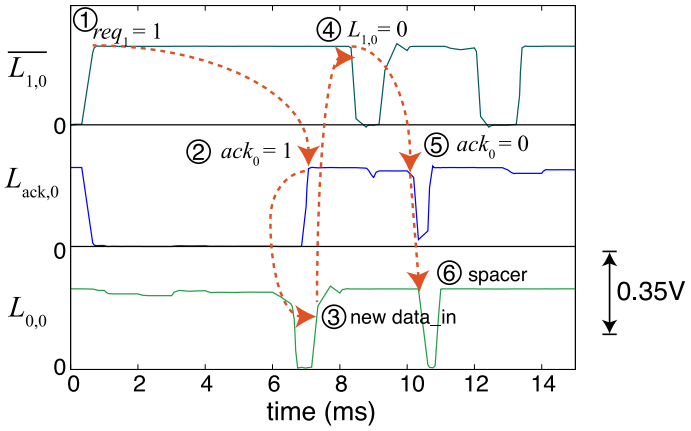

Fig. 8. Control signals between stages 1 and 2 of the WCHB asynchronous pipeline

time vary from those of simulation results. Fig. 10 shows the experimental results of data transmission between the three stages, and it is seen that data are transmitted independently of the presence of delays. Fig. 11 shows the control signals between stages 1 and 2.

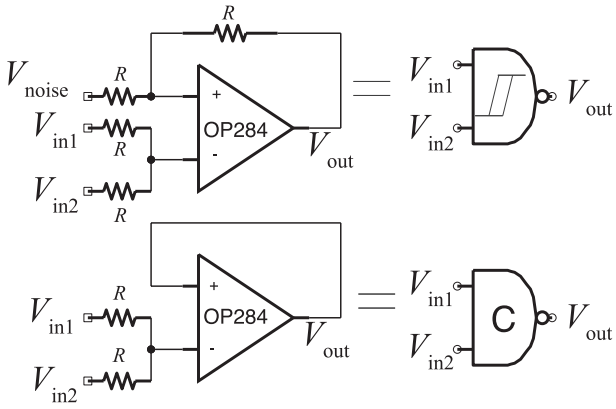

Fig. 9. OP284 configuration to implement the SR NAND gate and the C-element

3.3 Power consumption and error rate of the three-stage WCHB asynchronous pipeline

In this subsection, the general performance of the three-stage WCHB pipeline is described in greater detail in terms of the power consumption and error rate.

Fig. 12 shows the simulation results of the power consumption (P) versus standard deviation of noise ($\sigma_{V_{noise}}$), where a linear relationship between the parameters is observed. The power consumption is 105 pA for our current setting ($V_{dd} = 0.35$ V), demonstrating that the pipeline is consuming low power. Additional simulations were performed for the error rate (%) versus ($\sigma_{V_{noise}}$), where a zero tolerance is demonstrated for a $\sigma_{V_{noise}}$ value less than 0.25 V (Fig. 13).

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

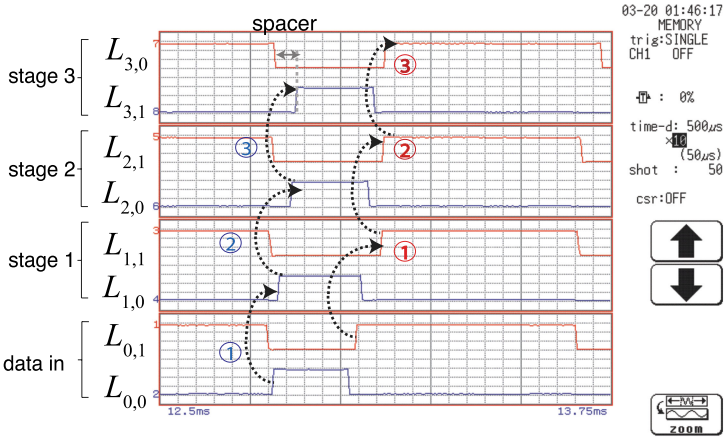

Fig. 10. Experimental results for the three-stage WCHB asynchronous pipeline (data transmission), obtained from a Memory Hicorder HIOKI 8826

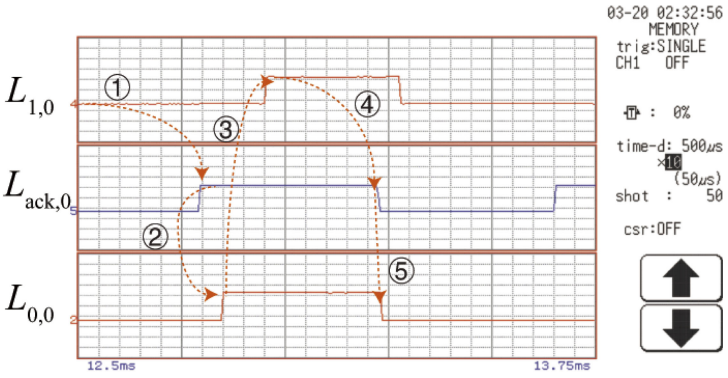

Fig. 11. Experimental results for the three-stage WCHB asynchronous pipeline (acknowledge signal between two stages)

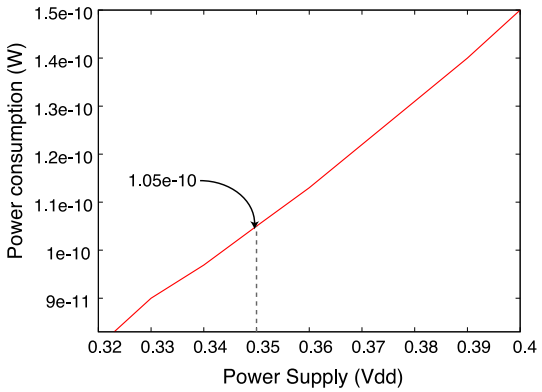

Fig. 12. Power consumption of the three stage WCHB asynchronous pipeline

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014

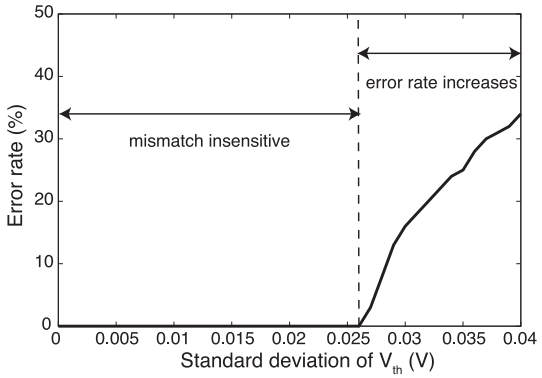

Fig. 13. Error rate versus $\sigma_{V_{\rm th}}$ (V)

4 Conclusions

SR logic gates offer three main advantages: a mismatch insensitive design over a certain range of standard deviation of the threshold voltage, a low-power design, and a stable response regardless of the noise input. However, the dependence on noise also generates unpredictable delays in their response. Although these delays represent a challenge to SR logic gate synchronization, an alternative solution is found by employing asynchronous circuit design, where the possibility of designing DI circuits allows a reliable synchronization of the SR logic gates. Although the current configuration is expensive in terms of VLSI area, it provides a solution for future devices beyond MOSFET technology which exhibits large internal fluctuations leading to system-level malfunctions.

Acknowledgments

This study was supported by a Grant-in-Aid for Scientific Research on Innovative Areas [25110015] from the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan.

DOI: 10.1587/elex.11.20140632
Received July 2, 2014
Accepted July 10, 2014
Publicized July 25, 2014
Copyedited August 10, 2014