پیاده‌سازی الگوریتم رمزنگاری A5/1 در فنّاوری اتوماتای سلولی کوانتومی

محمّدامین امیری[۱]

مژده مهدوی[۲]، ستّار میرزاکوچکی[۳]

چکیده

اتوماتای سلولی کوانتومی (QCA)، یک فنّاوری نوظهور در عرصه نانوفنّاوری می‌باشد. بدلیل ویژگی‌هایی نظیر مصرف توان اندک، ابعاد نانو و سرعت بالا، رمزنگاری یکی از کاربردهای جذاب این فنّاوری می‌باشد. الگوریتم رمزنگاری ابتدایی برای GSM ، الگوریتم رمزنگاری A5/1 بود. پیاده‌سازی الگوریتم رمزنگاری رشته‌ای A5/1 در فنّاوری اتوماتای سلولی کوانتومی، در این مقاله مورد بحث قرار گرفته است. نتایج شبیه‌سازی از نرم‌افزار QCADesigner بدست آمده‌اند.

کلید واژه

اتوماتای سلولی کوانتومی، رمزنگاری، A5/1

۱. دانش آموخته دکتری برق- دانشگاه علم و صنعت ایران amiri@ee.iust.ac.ir

۲.دانشگاه آزاد اسلامی واحد علوم تحقیقات

۳.دانشکده مهندسی برق،دانشگاه علم و صنعت ایران

تاریخ دریافت: ۸۸/۱۲/۱۵ تاریخ پذیرش: ۸۹/۱/۲۲

مقدمه

صنایع میکروالکترونیک طی دهه‌های گذشته با کاهش ابعاد ترانزیستورها، مجتمع سازی، توان مصرفی و سرعت مدارات مجتمع را بهبود داده است. اما بنظر می‌رسد علیرغم کاهش ابعاد ترانزیستورها، بعضی مشکلات نظیر توان مصرفی قابل صرفنظر نیستند. QCA که اولین بار توسط لنت و همکاران معرفی گردید [۱] ، یک فنّاوری نوظهور را در سطح نانو ارائه می‌دهد. استفاده از فنّاوری QCA برای پیاده‌سازی مدارات منطقی، یکی از روش‌هایی است که علاوه بر کاهش ابعاد مدارات منطقی و افزایش فرکانس کلاک این مدارات، توان مصرفی را کاهش می‌دهد [۱-۲]. سلول‌های QCA دارای نقطه‌های کوانتومی هستند که موقعیت الکترونها در آنها، سطح باینری صفر و یک را تعیین خواهد کرد. این مهمترین ویژگی مدارات QCA در مقایسه با مدارات معمول CMOS است که در آنها حالت‌های منطقی بوسیله سطوح ولتاژ مشخص می‌گردند.

الگوریتم رمزنگاری رشته‌ای A5/1 ، الگوریتم رمزنگاری ابتدایی برای GSM بود. این الگوریتم در سال ۱۹۸۷ ایجاد شده بود. طراحی تقریبی الگوریتم A5/1 در سال ۱۹۹۴ فاش شد و طراحی دقیق این الگوریتم در سال ۱۹۹۹ توسط Bericeno و به روش مهندسی معکوس از یک تلفن GSM واقعی بدست آمد [۸]. بعنوان یکی از کاربردهای فنّاوری QCA ، ما الگوریتم رمزنگاری A5/1 را پیاده‌سازی نمودیم.

باقیمانده مقاله بصورت زیر است. در بخش ۲، مروری کوتاه بر QCA ارائه شده است. در این بخش مروری بر الگوریتم رمزنگاری A5/1 و بلوک‌های اصلی آن نیز خواهیم داشت. در بخش ۳ نتایج پیاده سازی و شبیه سازی بلوک‌های اصلی الگوریتم مورد بحث و بررسی قرار می‌گیرند. بخش ۴ شامل نتیجه‌گیری مقاله و بحث در ارتباط با نتایج بدست آمده خواهد بود.

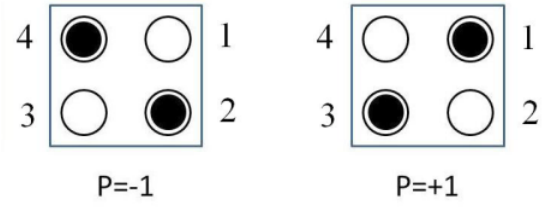
نتایج شبیه‌سازی این پیاده‌سازی از نرم‌افزار QCADesigner v2.0.3 بدست آمده‌اند. نرم‌افزار فوق در آزمایشگاه ATIPS دانشگاه Calgary کانادا تهیه شده است. این نرم‌افزار دارای موتورهای شبیه‌سازی متفاوتی می‌باشد. در این مقاله از موتور شبیه‌سازی Coherence Vector استفاده شده است زیرا ارزیابی دقیق و با جزئیات بیشتر از مدارات QCA انجام می‌دهد.

موضوعات و روش‌ها

اتوماتای سلولی کوانتومی

در اتوماتای سلولی کوانتومی، یک سلول دارای چهار نقطه کوانتومی مانند شکل ۱ خواهد بود. نقطه های کوانتومی بصورت دایره‌های توخالی نمایش داده شده‌اند. هر سلول دارای دو الکترون است که بصورت دایره‌های توپر نشان داده شده‌اند.

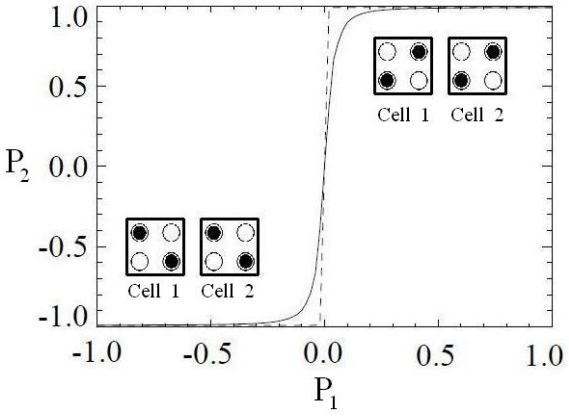
در یک سلول، الکترون‌ها با مکانیزم تونل زنی مجاز به پرش بین نقطه‌های کوانتومی مجزا هستند ولی آنها مجاز به تونل زنی بین سلول‌ها نمی‌باشند. سد پتانسیل بین سلول‌ها بقدر کافی بزرگ فرض شده است تا بطور کامل از تونل زنی بین سلولی جلوگیری کند. علیرغم اینکه دو الکترون دارای نیروی دافعه می‌باشند، آنها مجبور به اشغال نقطه‌های کوانتومی هستند.



شکل ۱) سلول QCA و حالت های پایه ای

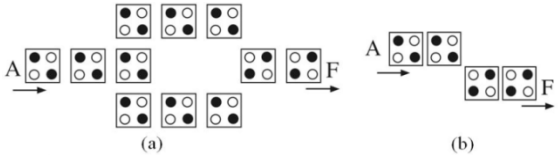
اگر آنها را بدون نیروی خارجی رها کنیم، آنها یکی از دو ترکیب پایه ای سلول را بخود می‌گیرند. واضح است که الکترون‌ها تمایل دارند که در نقطه‌های کوانتومی متمایز جای بگیرند زیرا نیروی کلمبی موجود، آنها در یک نقطه کوانتومی نخواهند ماند. با توجه به این پیش فرض‌ها، می‌توان نتیجه گرفت که حالت‌های پایه‌ای سامانه شامل دو ترکیب الکترونها در گوشه‌های مخالف سلول (شکل ۱) خواهد بود.

کوپلینگ بین دو سلول بوسیله اندرکنش کلمبی بین الکترون‌های سلولهای مختلف انجام می‌گیرد. شکل ۲ چگونگی تاثیر پذیری یک سلول از سلول همسایه اش را نشان می‌دهد [۳]. این شکل دو سلول را نشان می‌دهد که پلاریزاسیون سلول ۱ توسط پلاریزاسیون سلول همسایه اش تعیین می‌شود. فرض شده است که P_2 در یک مقدار معینی فیکس شده است و روی سلول ۱ تاثیر گذاشته و بنابراین پلاریزاسیون آن را تعیین می‌کند. نتیجه منتجه این است که کوپلینگ سلول به سلول در مدارات QCA شدیدا غیرخطی است. همانطور که مشاهده می‌شود، حتی در صورت پلاریزه شدن جزئی سلول ۲، سلول ۱ بطور کامل پلاریزه شده است [۲،۳].



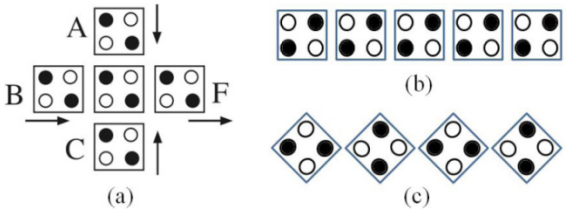
شکل ۲) کوپلینگ سلول های QCA

می‌توان از اندرکنش فیزیکی بین سلول‌ها جهت پیاده‌سازی توابع منطقی بولی پایه استفاده نمود. گیت‌های منطقی پایه در QCA شامل تابع منطقی اکثریت‌گیر و معکوس‌کننده می‌باشد که در شکل‌های ۳ و ۴ نشان داده شده‌اند. تابع منطقی اکثریت‌گیر تنها با ۵ سلول قابل پیاده‌سازی است [۴]. تابع AND منطقی را می‌توان با تابع منطقی اکثریت‌گیر و تنظیم یکی از ورودی‌هایش به منطق صفر پیاده‌سازی نمود. تابع OR منطقی را می‌توان با تابع منطقی اکثریت‌گیر و تنظیم یکی از ورودی‌هایش به منطق یک پیاده‌سازی نمود.



شکل ۳) (a) گیت معکوس کننده افزون شده، (b) گیت معکوس کننده

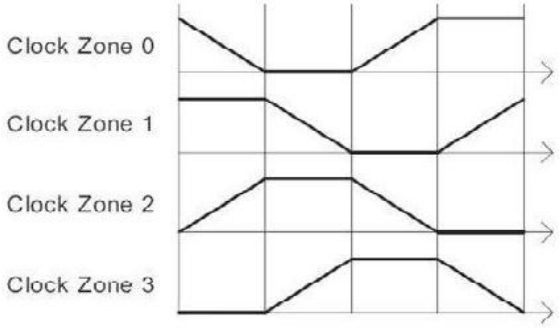
کلاکینگ در QCA سازوکاری جهت حرکت همزمان داده ها در مدار مهیا می‌کند. باید این مطلب را در نظر گرفت که کلاک همچنین جهت حرکت داده در مدارات QCA را کنترل می‌کند. کلاک همچنین توان مورد نیاز جهت عملکرد مدار را تامین می‌نماید.



شکل ۴) (a) گیت اکثریت گیر، (b) سیم باینری،(c) سیم ۴۵ درجه

بطور دقیقتر، کلاک QCA جهت کنترل ارتفاع سد تونل زنی در سلول‌ها استفاده می‌شود. هنگامی که سطح کلاک پائین است، الکترون‌ها در موقعیت‌های مربوطه‌شان به تله افتاده و نمی‌توانند به نقطه‌های کوانتومی دیگر تونل‌زنی کنند و بنابراین مقدار منطقی سلول حفظ می‌شود. این عمل با ماکزیمم نگه داشتن ارتفاع سد تونل‌زنی محقق می‌شود. هنگامی که سطح کلاک بالا است، سلول به حالت پلاریزاسیون بی اثر خواهد رفت. این عمل با مینیمم نگه داشتن ارتفاع سد تونل زنی محقق می‌شود. بین این دو حالت سلول‌ها یا در حال latch شدن و یا در حال relax شدن می‌باشند.

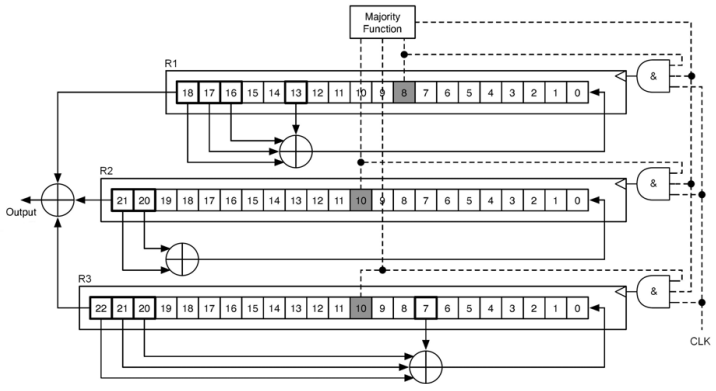
هر سلول در منطقه کلاکینگ خاص به یکی از چهار فاز کلاک QCA ممکن که در شکل ۵ نشان داده شده است، مرتبط شده است. هر سلول در ناحیه کلاک بطور همزمان با تغییرات سیگنال کلاک latch و unlatch می‌شود و به این ترتیب داده‌ها از طریق سلول‌ها منتقل می‌شوند [۵-۷].



شکل ۵) نواحی کلاک QCA

الگوریتم رمزنگاری A5/1

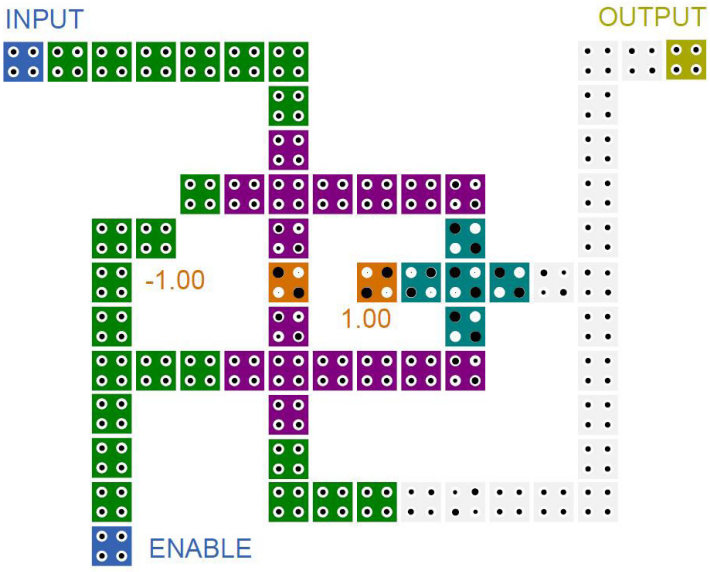
الگوریتم رمزنگاری A5/1 در سال ۱۹۸۷ ایجاد شده است. این الگوریتم از ۳ عدد جابجایی ثبات با بازخورد خطی (LFSR) کوتاه ساخته شده است. این جابجایی ثباتها با طول‌های ۱۹، ۲۲، و ۲۳ بیت به ترتیب با نام‌های R_1، R_2، و R_3 شناخته می‌شوند. بیت سمت راست در این جابجایی ثباتها بعنوان بیت صفر شناخته می‌شود. بیت‌هایی که در بازخورد R_1 نقش دارند، در موقعیت‌های ۱۳، ۱۶، ۱۷ و ۱۸ قرار دارند. بیت‌هایی که در بازخورد R_2 نقش دارند، در موقعیت‌های ۲۰ و ۲۱ قرار دارند. بیت‌هایی که در بازخورد R_3 نقش دارند، در موقعیت‌های ۷، ۲۰، ۲۱ و ۲۲ قرار دارند. هنگامی که یک جابجایی ثبات کلاک می‌خورد، بیت‌های ذکر شده با یکدیگر xor شده و نتیجه در بیت سمت راست جابجایی ثبات مربوطه که سایر بیت‌هایش به راست جابجایی داده شده‌اند، ذخیره می‌گردد. سه جابجایی ثبات ذکر شده با استفاده از یک قانون حداکثر گیری کلاک می‌خورند: هر ثبات دارای یک بیت کلاک‌زنی می‌باشد (بیت ۸ از R_1، بیت ۱۰ از R_2، و بیت ۱۰ از R_3). در هر سیکل کلاک، تابع اکثریت سه بیت فوق محاسبه می‌شود و فقط جابجایی ثباتهایی که بیت کلاک‌زنی آنها با بیت اکثریت برابر هستند، کلاک می‌خورند [۸-۱۱].



شکل ۶) الگوریتم رمزنگاری A5/1

نتایج

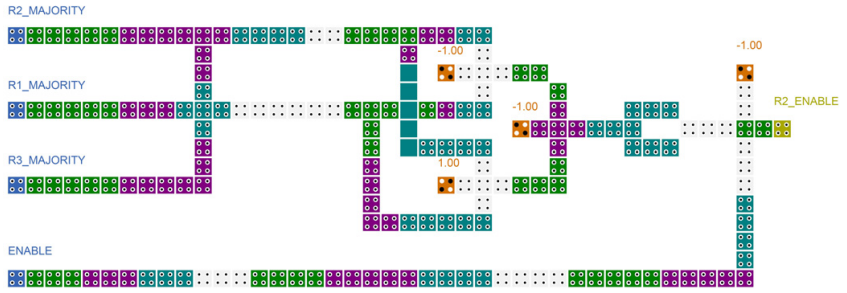
در این بخش، پیاده‌سازی الگوریتم رمزنگاری A5/1 بوسیله پیاده‌سازی و شبیه‌سازی بلوک‌های اصلی‌اش تشریح می‌گردد. یکی از این بلوک‌های اصلی، سلول حافظه می‌باشد. در این مقاله، یک سلول حافظه مبتنی بر حلقه پیاده‌سازی شده است که دارای یک پایه فعال‌ساز نوشتن، یک ورودی و یک خروجی می‌باشد. پیاده‌سازی QCA این بلوک در شکل ۷ دیده می‌شود. همانطور که در شکل ۷ دیده می‌شود، اگر پایه ENABLE دارای مقدار منطقی یک باشد، مقدار ذخیره شده در حافظه بدون تغییر باقی می‌ماند و اگر این ورودی مقدار منطقی صفر را برای یک دوره کلاک بگیرد، مقدار ورودی در حافظه ذخیره خواهد شد. خروجی OUTPUT در هر کلاک مقدار موجود در حافظه رانشان می دهد.



شکل ۷) پیاده‌سازی یک سلول حافظه

بلوک مهم دیگر از الگوریتم رمزنگاری A5/1، تابع اکثریت‌گیر می‌باشد. این تابع سیگنال فعال‌ساز جهت جابجایی خوردن ثباتهای A5/1 را تولید می‌کند. این عمل با استفاده از بیت‌های کلاک‌زنی ثباتها انجام می‌گیرد. اگر بیت کلاک‌زنی یک ثبات برابر اکثریت سه بیت کلاک‌زنی باشد، ثبات مذکور

در عملیات بعدی، کلاک خواهد خورد. پیاده‌سازی این تابع برای ثبات R_2 در شکل ۸ نمایش داده شده است. توابع اکثریت‌گیر برای ثباتهای R_1 و R_3 مشابه تابع مورد بحث می‌باشند.



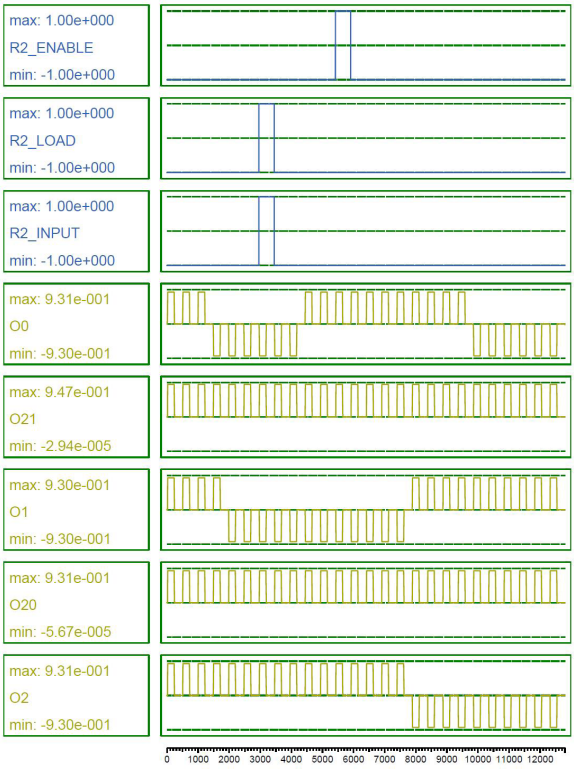
شکل ۸) پیاده‌سازی تابع اکثریت‌گیر

یک شبیه‌سازی فراگیر برای تابع اکثریت‌گیر انجام شده است. نتایج این شبیه‌سازی در شکل ۹ دیده می‌شود. می‌توان در شبیه‌سازی دید که هنگامی که بیت کلاک‌زنی جابجایی ثبات R_2 برابر اکثریت سه بیت کلاک‌زنی باشد، خروجی $ENABLE\text{-}R_2$ مقدار ورودی ENABLE را به خود خواهد گرفت. در سایر موارد، خروجی $ENABLE\text{-}R_2$ دارای مقدار منطقی صفر خواهد بود. نکته دیگری که در شبیه‌سازی دیده می‌شود، تاخیر ۳ دوره کلاک جهت معتبر بودن خروجی $ENABLE\text{-}R_2$ است که این تاخیر در نتیجه کلاک‌زنی جهت انتشار داده‌ها بوجود آمده است.

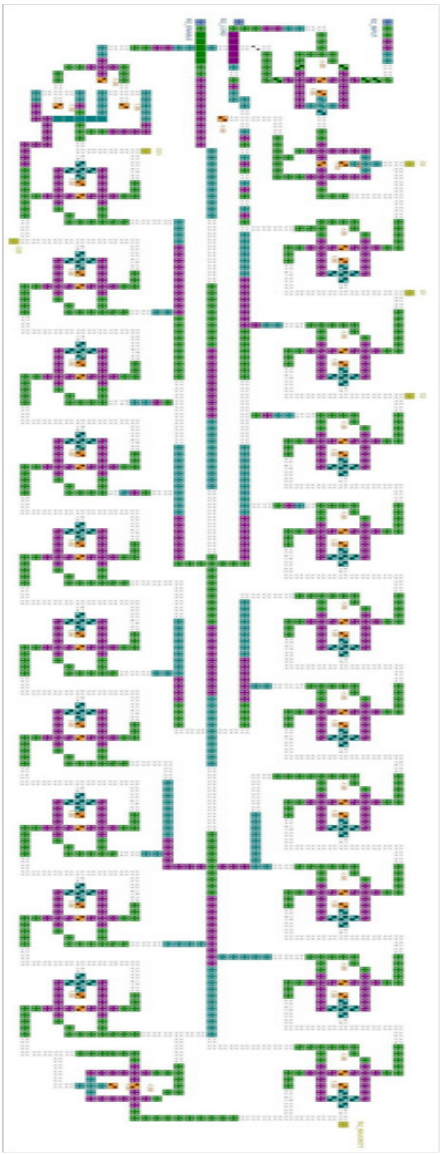
شکل ۹) شبیه‌سازی تابع اکثریت‌گیر

شکل ۱۱ پیاده‌سازی ثبات R_2 را نشان می دهد. این ثبات دارای ۲۲ سلول حافظه می‌باشد. علاوه این سلول‌ها، یک مالتی‌پلکسر و یک xor نیز همراه این ثبات وجود دارند. هنگامی که $LOAD\text{-}R_2$ فعال باشد، مالتی‌پلکسر ورودی $ENABLE\text{-}R_2$ را انتخاب می‌کند و در سایر موارد، مقدار بازخورد انتخاب می‌شود. دهمین بیت از ثبات، بیت کلاک‌زنی و بیست و یکمین بیت از ثبات، بیت خروجی است. مقدار بازخورد این ثبات حاصل xor شدن بیت‌های بیستم و بیست و یکم است. هنگامی که ورودی $LOAD\text{-}R_2$ دارای مقدار منطقی یک باشد، ورودی $INPUT\text{-}R_2$ در بیت صفرم ذخیره خواهد شد. ورودی $ENABLE\text{-}R_2$ جهت جابجایی دادن داده‌ها در ثبات R_2 استفاده شده است. هنگامی که این ورودی طی یک دوره کلاک مقدار منطقی یک را داشته باشد، تمام بیت‌های ثبات R_2 بجز بیت صفرم، پس از پنج دوره کلاک مقدار بیت قبلی را ذخیره خواهند کرد. مقدار بیت صفرم پس از چهار دوره کلاک اضافی تغییر خواهد کرد، زیرا مقدار بازخور باید توسط یک گیت xor

محاسبه گردد و تاخیر این گیت معادل چهار دوره کلاک است. می‌توان نتیجه‌گیری کرد که عملیات جابجایی در طی ۹ دوره کلاک به انجام می‌رسد. نتایج شبیه‌سازی ثبات R_2 در شکل ۱۰ نمایش داده شده است. در این شبیه‌سازی ابتدا مقدار منطقی یک در بیت صفرم ثبات R_2 ذخیره شده است. پس از آن یک عمل جابجایی انجام شده است.



شکل ۱۰) شبیه‌سازی جابجایی ثبات R_2



شکل ۱۱) پیاده‌سازی جابجایی ثبات R_2

با فرض ابعاد nm ۱۸ برای طول و عرض سلول‌های QCA و فاصله مرکز به مرکز nm ۲۰ برای دو سلول مجاور [۱۳،۱۲]، نتایج پیاده‌سازی تابع اکثریت‌گیر و ثبات R_2 در جدول ۱ نمایش داده شده است.

جدول ۱) نتایج پیاده‌سازی تابع اکثریت‌گیر و ثبات R۲

	R2 Register	Majority Function
Complexity(Cells)	1974	181
Area(μm^2)	2.4912	0.2352
Delay(Clocks)	9	3

بحث

در این مقاله، پیاده‌سازی الگوریتم رمزنگاری A5/1 مورد بحث و بررسی قرار گرفت. بلوک‌های اصلی این الگوریتم پیاده‌سازی و شبیه‌سازی شده‌اند. تعداد سلول‌ها و سطح مصرفی برای تابع اکثریت‌گیر و ثبات R_2 ارائه شده است. پیاده‌سازی‌های این مقاله در سطح سلول‌های QCA می‌باشد و بسته به نوع فنّاوری فلزی، مولکولی، نیمه‌هادی و ... که سلول‌های QCA در آن ساخته شوند، توان مصرفی و فرکانس کار متفاوت خواهد بود و در حال حاضر ابزاری برای محاسبه آنها در اختیار نداریم. از آنجایی که الکترون‌ها درون سلول حرکت می‌کنند، توان مصرفی بسیار ناچیز است.

مراجع

[1] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, ”Quantum Cellular Automata,” Nanotechnology, 1993, Vol. 4, No. 1, pp. 49-57.

[2] P. D. Tougaw and C. S. Lent, “Dynamic Behavior of Quantum Cellular Automata,” Journal of Applied Physics, 1996, Vol. 80, No. 8, pp. 4722-4735.

[3] P. D. Tougaw, C. S. Lent, and W. Porod,”Bistable Saturation in Coupled Quantum-dot Cells,” Journal of Applied Physics, 1993, Vol. 74,

No. 5, pp. 3558-3565.

[4] P.D. Tougaw and C.S. Lent,"Logical Devices Implemented Using Quantum Cellular Automata," Journal of Applied Physics, 1994, Vol. 75(3), pp. 1818-1825.

[5] K. Hennessy and C. S. Lent,"Clocking of Molecular Quantum-dot Cellular Automata," Journal of Vacuum Science and Technology B, 2001, Vol. 19, No. 5, pp. 1752-1755.

[6] C. S. Lent and Beth Isaksen, "Clocked Molecular Quantum-dot Cellular Automata," IEEE Transaction on Electron Devices, 2003, Vol. 50, No. 9,pp.100-105

[7] M. A. Amiri, M. Mahdavi, S. Mirzakuchaki,"QCA Implementation of a Mux-Based FPGA CLB," Proc. of International Conf. On Nanoscience and Nanotechnology, Melbourne, Australia, Feb. 2008, pp. 141-144.

[8] M. Briceno, I. Goldberg, D. Wagner,"A pedagogical implementation of A5/1," 1999, http://www.scard.org/gsm/a51.html.

[9] E. Biham, O. Dunkelman,"Cryptanalysis of the A5/1 GSM Stream Cipher," Progress in Cryptology, proceedings of Indocrypt00, Springer-Verlag, 2000, pp. 4351.

[10] A. Biryukov, A. Shamir, D. Wagner,"Real Time Cryptanalysis of A5/1 on a PC," Advances in Cryptology, proceedings of Fast Software

Encryption00, 2001, Springer-Verlag, pp. 118.

[11] J. Golic,"Cryptanalysis of Alleged A5 Stream Cipher," Advances in Cryptology, proceedings of Eurocrypt97, 1997, Springer-Verlag, pp. 239-255.

[12] Heumpil Cho, Earl E. Swartzlander, Adder Designs and Analyses for Quantum-Dot Cellular Automata, IEEE Trans. on Nanotechnology, 2007, Vol. 6, No. 3, pp. 374–383.

[13] Heumpil Cho, Earl E. Swartzlander, Adder and Multiplier Design in Quantum-Dot Cellular Automata, IEEE Trans. on Computers, 2009, Vol. 58, No. 6, pp. 721–727.