# A2U2: A Stream Cipher for Printed Electronics RFID Tags

Mathieu David[1], Damith C. Ranasinghe[2], Torben Larsen[1]

[1]Department of Electronic Systems, Aalborg University, Niels Jernes Vej 12, 9220 Aalborg Øst, Denmark
[2]Auto-ID Lab, The School of Computer Science, The University of Adelaide, SA 5005, Australia
[1]{md; tl}@es.aau.dk [2]damith@cs.adelaide.edu.au

***Abstract* – The design of hardware-oriented ciphers has an increasingly important role to play with emerging ubiquitous and pervasive computing devices, such as low cost passive Radio Frequency Identification (RFID) tags. The importance of such ciphers are further highlighted by novel manufacturing technologies, such as printed ink to develop extremely low cost RFID tags. Such developments bring new challenges, especially in terms of providing security, both to protect privacy as well as to enable applications dependent on security, such as e-tickets. In this paper we present a new stream cipher, A2U2, which uses principles of stream cipher design and approaches from block cipher design. Our lightweight cryptographic primitive has taken into consideration the extremely resource limited environment of printed ink tags, to develop a cipher that can be implemented with less than 300 gates, with the added benefit of high throughput provided by stream ciphers.**

**Keywords -** Stream Cipher, Lightweight Cryptography, RFID, Printed Electronics, Security, Privacy.

## I. INTRODUCTION

RFID is one of the most promising technologies of the coming decade because of its ability to automatically and uniquely identify objects wirelessly [1]. It is also a key enabling technology of the 'Internet of Things' [2]. The range of applications enabled by RFID technology is so wide that it will soon become ubiquitous. However, the multiple advantages offered by RFID are linked to numerous challenges, which need to be overcome to realize the full potential of the technology. Providing security services, such as authentication necessary for e-ticketing applications and counterfeit detection and prevention, and ensuring privacy of both consumers and corporations are among the primary concerns. These issues must be addressed to facilitate the global adoption of RFID technology with confidence.

With the emergence of printed electronics technology, RFID systems will soon reach a horizon where the cost of an RFID tag is no longer an impediment to deployments. In fact, printed electronics technology is believed to realize electronic systems at a substantially lower cost, unachievable with conventional single-crystal based integrated circuit (IC) fabrication [3]. However, thus far printed ink technologies are only capable of manufacturing RFID tags that operate at 13.56 MHz (High Frequency range). Nevertheless, such low cost tags will see the use of RFID technology grow in yet underexploited areas, such as postal items, books, e-tickets and airline baggage handling [2].

However, despite recent advances in printing processes and material science, integrated circuits on printed RFID tags are limited to a few thousand transistors [3]. Consequently printed ink based RFID tags are extremely resource-limited devices. Due to previous constraint and given the more recent advances in printed electronics, implementing a cryptographic primitive on printed RFID tags is a challenge that is largely unexplored.

In the case of printed electronics RFID tags, the limitations are threefold:

- Cost (area) has to be low in order to be integrated on the chip. Current printed RFID tags support approximately 2,000 transistors (500 gates). Around 200 gates or less are expected to be available for implementing a security mechanism [3].
- Power consumption must be low to overcome the lack of a battery and to allow a tag to operate at a minimum read range. Consequently, a tag's power consumption is limited to tens of µWs. For example ISO 14443 specifications for tags operating at 13.56 MHz must provide a read range of 100 mm [8].
- Throughput of a security primitive should be reasonable to allow real time interaction for a large number of tags (generally in the order of hundreds of tags). For example ISO 14443 specifications for tags must transmit at 106 kbps [8].

Over the last decade, various research efforts have achieved lightweight security primitives for resource constrained devices [4]-[7]. However, none of the proposed primitives are suitable for printed electronic RFID tags. Although the PRINT cipher published very recently has sought to address the challenges posed by printed electronic RFID tags, the cost of the block cipher still requires using nearly all available 2000 transistors on a printed RFID tag [15].

In this paper we propose a new hardware-oriented stream cipher, A2U2, conceptualized specifically to meet the extremely resource limited environment of printed electronic RFID tags. The central design blocks of A2U2 are based on: i) learning from vulnerabilities in previously published stream and block ciphers, ii) key cryptographic ideas introduced by the work of Rueppel on stream ciphers [10], and iii) Shannon's ideas on confusion and diffusion [11]. In particular, A2U2 overcomes significant issues such as identical initialisation values resulting in predictable bit streams on power-up, specific to the use of both block and stream ciphers in RFID-related applications. Such issues have not been dealt with in previously published designs. The result is a novel stream cipher that can be implemented on a small area to provide security services on printed ink RFID tags.

The rest of the paper is organised as follows. Section II provides an overview of recent research carried on lightweight cryptographic primitives. Section III presents the key design principles that we employed to design A2U2. Section IV details the building blocks of our stream cipher, and Section V presents an analysis of A2U2. Finally, we conclude the paper in Section VI and provide details of further work we are currently undertaking.

## II. Related Works

Early candidates for resources constrained devices were based on hardware optimisations of well-known block ciphers. Feldhofer proposed an optimised version of AES [12], while Leander developed a low-cost version of DES [13]. XTEA [4] and SEA [5] are two other block ciphers, designed specifically for embedded devices. However, none of these ciphers were designed with RFID applications as a specific target platform and, as a result, are either too slow (AES, SEA, and TEA) or too expensive in terms of implementation costs (AES, DESL and TEA).

PRESENT is the first block cipher designed specifically for resource constrained devices [6] such as RFID tags. It is the result of work carried out in the EU Project UbiSec&Sens [14]. PRESENT is based on key ideas articulated by both Shannon and Rueppel, and used in the design of two modern block ciphers: DES and AES. PRESENT's architecture was developed to support two different but competing design goals: i) a low cost implementation (1000 gates), and ii) a high throughput implementation (200 kbps) [6].

A recent block cipher, KATAN [7], has reached a further milestone in terms of area minimisation with a design tailor-made for low-cost RFID tags. A rigorous analysis of power consumption (< 1μW), area minimisation (down to 480 gates), and throughput optimisation (12.5 kbps when clocked at 100 kHz) has been achieved in its design. Finally, PRINT Cipher [15] is a more recent block cipher design that, for the first time, has targeted printed ink RFID tags. However, PRINT still requires at least 402 gates and the cost of the cipher is still well beyond the expected number of gates (less than 200 gates) available for a security primitive.

More interestingly, there are only a limited number of stream ciphers suitable for small embedded devices. In fact, none of the six stream ciphers resulting from the EU Project NESSIE (concluded in 2003) were satisfactory [16], leading to a new EU project called eSTREAM [17], to address this gap. Among the selected candidates of eSTREAM, two are of particular interest for RFID applications. GRAIN and TRIVIUM have implementation results with either low area (1294 gates for GRAIN), or low power (1.2 μW for GRAIN, 1.02 μW for TRIVIUM) [19, 20]. Again, these stream ciphers are well beyond the cost limitations of printed ink tags.

Finally, an interesting alternative to block and stream ciphers was proposed by Yüksel, with a scalable universal hash function called WH-16 [21], for RFID tags at a very low cost (460 gates).

On one hand, it is evident that KTANTAN [7] and, more recently, the PRINT Cipher [15] have advanced block cipher designs to new levels of compactness, while WH-16 achieved a similar goal with hash functions. Further reducing cost of implementation of one of these ciphers would be at the expense of its security, which is not a desirable outcome. On the other hand, the smallest stream cipher published so far is GRAIN and, with its 1294 gates, leaves a vast margin for improvement. Consequently, our approach has been to focus on developing a hardware-efficient stream cipher. Furthermore, during a conference in 2004, Adi Shamir pointed out the slow and continuous decline of stream ciphers to the benefit of block ciphers [18]. According to him, stream ciphers will be useful for only two kinds of applications in the long term: i) applications that require extremely high encryption speed (beyond Gbps), and ii) applications in resource constrained devices such as RFID.

Like Shamir, we believe that stream ciphers have been underestimated, mainly due to the difficulty of analyzing them. However, as we have shown with A2U2, stream ciphers can offer comparable, if not superior, alternatives to the current suite of low-cost block cipher designs.. In addition, stream ciphers also have the clear advantage of being much faster than block ciphers.

## III. Design Principles

Block ciphers have been the most widely studied symmetric encryption algorithms so far. The research in the past decades has led to a relatively good understanding of the security of block ciphers [35]. The security of stream ciphers has only recently received increasing attention and the European projects, NESSIE and ECRYPT, have played a key role to this end. In NESSIE, no stream cipher made it to the final portfolio, as weaknesses had been discovered in all candidate stream ciphers. This illustrates how the study of stream ciphers is not as mature as the study of block ciphers. For example, common building blocks of block ciphers such as S-box constructions (used in PRESENT and AES) can be easily analysed by existing tools [36]. However, until recently, the study of stream ciphers with a nonlinear update function was little studied and there are very diverse strategies for analysing such stream ciphers [19].

Nevertheless, there is a large body of established guidelines and design principles for building stream ciphers such as those elaborated by Rueppel [10]. We have chosen to focus on synchronous stream ciphers with a nonlinear update function as these appear to offer the best combined security and performance. Our design methodology is based on a system-theoretic approach first elaborated by Ruppel in [10]. Furthermore, it should be noted that cryptanalytic attacks against stream ciphers have often been based on exploiting flaws in their design (e.g. [22], [35] and [50]). As a result, every aspect of a stream cipher needs to be carefully designed. The following sections describe: i) the methodologies we employed to achieve a high level of complexity and security; and ii) the principles we used to reduce the implementation cost of A2U2, while balancing the need for an adequate level of security.

### *A. Use of Primitive Polynomial Function for LFSR*

The use of a primitive polynomial is perhaps the most well known guideline regarding the use of LFSR. Nonetheless, it remains a key criterion to guarantee that the LFSR is of

maximal length and has a period of $2^L$-1, where $L$ is the length of the LFSR. Furthermore, the primitive polynomial should not be sparse (combination of a small number of connections) to avoid correlation attacks [23].

### B. Use of Good Nonlinear Boolean Function for NFSR

The Nonlinear Feedback Shift Register (NFSR) is a more secure alternative to LFSR, since its nonlinear feedback function makes it cryptographically stronger against several attacks, such as correlation attacks and algebraic attacks [24]. However, good NFSRs are difficult to design and can easily be weaker than LFSR, if their nonlinear Boolean function is not carefully selected. The design aspects of nonlinear Boolean functions are generally omitted from publications on stream and block ciphers, since no simple and precise guidelines on how to build a strong function are available. Although several papers ([25]-[29]) present some construction guidelines, their results remain too complex and difficult to be implemented in practice for stream cipher design.

Nonlinear Boolean functions are characterised by the following four properties:

- *Balancedness:* A Boolean function is said to be balanced if the output probabilities of '1' and '0' are equal.
- *Nonlinearity:* The nonlinearity of a Boolean function $f$ with $n$-variables is the Hamming distance [30] of $f$ from the set of all affine functions with $n$ variables.
- *Algebraic Degree:* The algebraic degree of an $n$-variable Boolean function is defined by the highest degree of its terms. The maximum algebraic degree of an $n$-variables Boolean function is $n$-1.
- *Correlation Immunity:* A Boolean function is said to be correlation immune of order $m$, if the distribution of their truth table is unaltered while fixing any $m$ inputs [31].

By definition, it is impossible to design a Boolean function that is perfectly balanced, has highest degree of non-linearity, highest algebraic degree, and highest correlation immunity [28]. Siegenthaler [31] proved the following fundamental relation between the number of variables $n$, the degree $d$, and order of correlation immunity $m$ of a balanced Boolean function.

$$m + d \leq n - 1 \quad (1)$$

Therefore, our task of building a good nonlinear Boolean function was based on constructing a function with the best possible combination of the properties we have discussed above, while considering the key requirement of minimising the cost of implementing the function.

### C. Exploit the Confusion and Diffusion Concepts

Introduced by Shannon in 1949 [11], the concept of confusion and diffusion is often applied in block ciphers with the use of substitution-permutation networks. In stream ciphers, the relation between plaintext, ciphertext and the key is different since the plaintext is not an input to the cipher. Therefore, diffusion (dissipation of plaintext statistics in the ciphertext) is not obvious; however, it is achieved by the filter at the output of the NFSRs. The filter function randomly distributes the ciphertext in the transmitted message. We have used Shannon's ideas of confusion to ensure that the secret key is used in a complex manner, to modify the feedback function as well as the inclusion of other irregularities in the cipher design.

In addition to the nonlinear feedback functions, we introduce four sources of nonlinearity and pseudorandom irregularities in the cipher:

- Irregular change in the feedback function
- Randomised initialisation value
- Randomised number of initialisation rounds
- Irregular length of the ciphertext

These design features not only contribute to increase confusion, but also ensure that the ciphertext generated for a given plaintext is uncorrelated, each time the cipher is used.

### D. Learning from Previous Designs

One of the advantages of public scrutiny of published ciphers is the lessons learnt. In general, any specific type of attack uses and/or reveals a particular flaw or weakness in a cipher design. Experience has proven that trying to fix a broken cipher will ultimately result in a new cipher that can still be easily broken (e.g. TEA [32]). However, even if the overall design of a given cipher is weak, some concepts might be worth reusing (the S-Box concept introduced in the DES is used in most contemporary block ciphers).

In constructing A2U2, we have built on the approach taken by Coppersmith in designing the Shrinking Generator (SG) [33]. However, the Shrinking Generator in its original published version has been broken with a few thousand bits of chosen ciphertext in a recent work [34], which has also highlighted the danger of using interleaved sequences, as in the SG. The weaknesses of the SG arise due to: i) the connection polynomials of the LFSRs being known, ii) the use of LFSRs, and iii) it uses identical initial values in the LFSRs at each session. In A2U2, we solve these issues by: i) modifying the feedback values irregularly, ii) using NFSRs, and iii) using pseudo-random numbers to initialise the registers, instead of fixed initial values. It is important to note here that A2U2 does not rely on the security of SG, since we only expound upon clock controlled generator architectural design concepts of the SG. From the clock controlled generators such as the SG, we have found a simple mechanism for achieving a nonlinear keystream. However, the practical use of shrinking generators designs is limited because they are unable to generate a bit stream at a constant rate. We have addressed this issue without compromising security as discussed in Section IV. E.

Furthermore, the study of previous cipher design principles is also useful to extract key concepts regarding area optimisation. These ideas are presented in the following subsection.

### E. Area Optimisation

Since our goal is to develop a primitive for printed ink based RFID tags, reducing the number of gates is a critical design criterion. We employ two techniques to achieve a compact cipher: i) a design with short-length registers, supported by compact functional blocks, to increase non-linearity in the design, and ii) the reuse of existing capabilities,

such as the 32 random bits a tag is required to generate, partly for use as the random handle. The random handle uses 16 bits of the pseudo-random number generated by the tag, as defined in the EPCglobal standards for Class 1 Generation 2 (C1G2) RFID tags [9], to identify itself to a reader during a communication session. Note that we assume such a feature is available in tags, or such a generator can be implemented with a modest cost.

Area optimisation is often achieved through: i) processing shorter word lengths (8 bits as opposed to 32 bits at a time), ii) repeated use of hardware components (such as S-boxes) or increasing the number of rounds before ciphertext is generated to maintain security at the compromise of speed and iii) using shorter keys, block sizes (in block ciphers), feedback functions, register lengths, and permutations and substitutions boxes (P-box and S-box, see Section V. A).

Reusing design concepts of previous ciphers is a common practice, illustrated by block ciphers such as PRESENT [6] and PRINT Cipher [15]. In fact, to achieve the estimated 402 gates, the PRINT Cipher reuses a number of optimisation and design principles. For example, the 3-bit S-box layer from SEA [5], hardwired permutations layer from PRESENT [6] (thus no logic gates are needed), and the counter, as used in KATAN [7], has been replaced by an LFSR to save additional gates.

A2U2 reuses an LFSR-based counter design as in KATAN, since it has been designed to minimise the number of gates. A2U2 also uses short-length registers, hardware-efficient nonlinear Boolean functions, and irregularities coupled with nonlinear functional blocks, to overcome the weaknesses posed by shorter register lengths. This is significant, since each shift register adds 6.25 to 12 additional gates to a design. The following section describes the detailed design of the cipher, based on the concepts we have discussed above.

## IV. Cipher Design

The A2U2 stream cipher is composed of four distinct building blocks. The four elements include: i) a counter, ii) a combination of two nonlinear registers, iii) an irregular change in the feedback function through a key-bit mixing mechanism, and iv) a filter function. An overview of the cipher design is illustrated in Fig. 1.

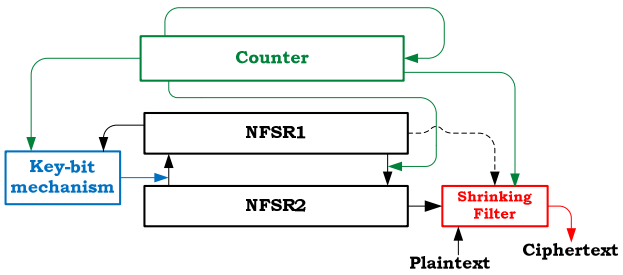


Fig. 1. Overview of the A2U2 architecture.

### A. The Counter

The counter is a 7-bit Linear Feedback Shift Register (LFSR), as represented in Fig. 2. Its feedback function is a maximal length polynomial function $F_C$ (whose period is $2^7$-1) defined by:

$$F_C = X^7 + X^4 + 1 \quad (2)$$

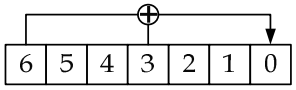


Fig. 2. The Counter used in A2U2.

The counter is initialised with an XOR operation of three strings:

- The 5 Least Significant Bits (LSBs) of a 32-bit (two 16-bit) pseudo-random numbers generated by the tag.
- The 5 LSBs of a 32-bit random number generated by the reader.
- The 5 last bits of the secret key.

The 5 bits resulting from this operation are input to the 5 Most Significant Bits (MSBs) of the LFSR (positions 6 to 2 in Fig. 2). The second LSB of the LFSR (position 1 in Fig. 2) is set to 1 to avoid an all-zeros string. Finally, the LSB of the LFSR is set to 0 in order to avoid an all-ones string, which is the condition to end the initialisation process. During the initialisation process, each clock cycle updates the counter until it reaches the all-ones state. The counter is clocked an irregular and secret number of times (ranging from 9 to 126), depending on the randomly selected initialisation value of the counter.

After initialisation, the counter simply works as an LFSR, where the bits are shifted clockwise, and the feedback is input in the LSB position. The counter also plays a role in the other building blocks of the cipher, as described in the following subsections.

### B. The Two Nonlinear Registers

This part of the cipher (as well as the counter) has been freely inspired by the block cipher KATAN [7], which introduces a new combination of two NFSRs, where the feedback function of each NFSR provides the feedback to the other NFSR, as shown in Fig. 3. The feedback functions are defined by the nonlinear Boolean functions given in (3) and (4).

$$F_1 = N_1[16] \oplus (N_1[14] \bar{\bullet} N_1[13]) \oplus N_1[11] \oplus (N_1[9] \bar{\bullet} C[6]) \oplus (N_1[6] \bar{\bullet} N_1[5] \bar{\bullet} N_1[4]) \oplus (N_1[3] \bar{\bullet} N_1[1]) \quad (3)$$

$$F_2 = N_2[8] \oplus (N_2[7] \bar{\bullet} N_2[6]) \oplus N_2[5] \oplus N_2[2] \oplus k_1 \quad (4)$$

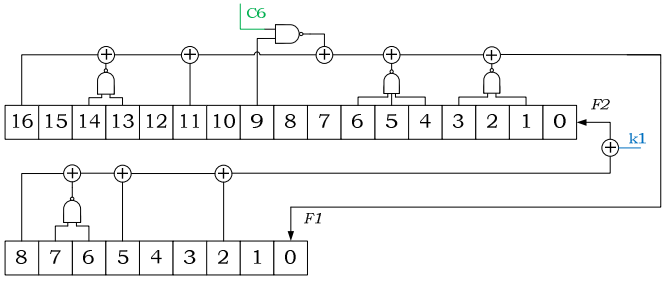


Fig. 3. The 2 NFSRs of the A2U2 cipher.

In (3) and (4), $N_y[x]$ represents the $x^{th}$ position bit of the $y^{th}$ NFSR, $\oplus$ represents an XOR operation, $\bar{\bullet}$ represents a NAND operation, $C[x]$ represents the $x^{th}$ position bit of the counter, and $k_1$ represents the bit generated by the key-bit mechanism.

Both NFSRs are initialised using the same process as the counter, with the following 26 bits of the random numbers (two 16 bit numbers generated by the tag) and the secret key. In the unlikely event (probability of $2^{-26}$) of an all-zeros initialisation value (IV), the bits introduced by the counter and the key-bit mechanism prevent the stream cipher from generating a series of zeros. Once initialised with IVs, the NFSRs are updated at each clock cycle, and the bits are shifted clockwise. The feedback function of NFSR1 provides the input for the LSB of NFSR2, and reciprocally for NFSR1.

### C. Nonlinear Boolean Function Design

As discussed in Section III, there is not a best design for a nonlinear Boolean function, but a good enough design based on a combination of the different properties outlined in Section III. Further complicating their design is the requirement to ensure that the implementation of the functions in hardware results in a small circuit footprint, to ensure that costs are contained. Hence, the number of terms of the equation needs to remain small. Functions with high algebraic degree and high correlation-immunity are composed of a large number of terms, and a perfect nonlinear function is not balanced [28]. The upper bound on nonlinearity of balanced Boolean functions with $n$ variables is theoretically $2^{n-1} - 2^{n/2}$ [29]. An example is a function of 41 terms with 7 variables [27]. It is clear that we cannot implement such a large function in A2U2. Our approach to construct a nonlinear Boolean Functions was to start with the function $f(x) = x_1x_2 \oplus x_3x_4 \oplus ... \oplus x_{n-1}x_n$, which is a perfect nonlinear function, and then to remove terms to achieve a balanced function. Finally, we increased the algebraic degree to three, by combining the variables $N_1[6]$, $N_1[5]$ and $N_1[4]$ in $F_1$.

### D. The Irregular Key-bit Mechanism

The third component of A2U2 is a function that utilises the tag's key. An extra bit ($k_1$) is XORed in the nonlinear Boolean function $F_2$ given by (4), as shown in Fig. 3. It increases the complexity of the cipher and modifies its feedback function, using the securely stored 56-bit private key. This extra bit generated is obtained with the nonlinear functional block, presented in Fig. 4.

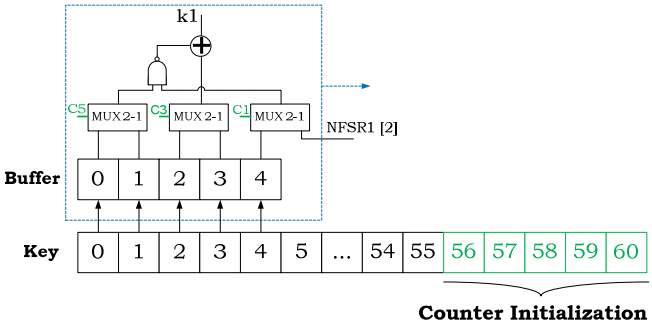


Fig. 4. The key-bit mechanism

As shown in Fig. 4, the last five bits of the key are reserved for the counter initialisation; they are not used in the process of key-bit generation. Every round, five bits of the key are loaded into a buffer. Those bits are then combined with three bits of the counter and one bit of NFSR1, as given by (5).

$$k_1 = \Big( MUX\ 2to1(B[0]B[1], C[5]) \bullet MUX\ 2to1(B[4]N_1[2], C[1]) \Big) \oplus MUX\ 2to1(B[2]B[3], C[3]) \quad (5)$$

MUX 2to1 ($x_1x_2$, $y$) is defined as a multiplexer that uses two bits ($x_1$ and $x_2$) as input, a selector bit ($y$), and output a single bit. When $y$ is equal to '0', $x_1$ is output, whereas $x_2$ is output when $y$ is equal to '1'. After this operation, the buffer is shifted by five bits in a counter-clockwise direction to generate the next value of $k_1$. The use of $N_1[2]$ considerably increases the period of $k_1$. Overall, the key-bit mechanism exploits the confusion principle described in Section III.

### E. The Filter Function

The final building block of A2U2 is the filter function, named the "shrinking filter" in reference to the clock controlled generator design of the Shrinking Generator [33]. The filter is represented in Fig. 5 and defined by (6).

$$C[x] = Mux\ 2to1((N_2[0] \oplus C[0])(N_2[0] \oplus P[x]), N_1[0]) \quad (6)$$

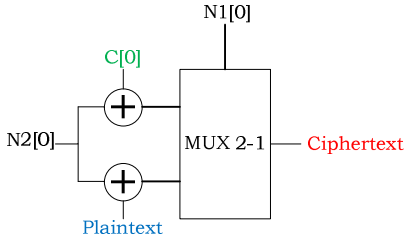


Fig. 5. The "shrinking" filter

This filter ensures that only part of the input string (NFSR2) will be XORed with the plaintext, depending on a *selector* string provided by NFSR1. We have overcome a significant drawback in the SG resulting from the irregular key stream (as a result of discarding the bits of the input string when not selected by the *selector* bit) by XORing the otherwise discarded bits of the input string with the LSB of the counter. This particular process presents several advantages:

- The "buffer" problem[1] at the output of the filter is solved, without additional hardware.
- For a series of plaintexts with a given fixed length, the resulting ciphertexts will likely have different lengths.
- The bits of ciphertext actually containing plaintext information are uniformly and randomly distributed within the ciphertext.

The output of the shrinking filter is the ciphertext of A2U2. The filter function starts operating once the initialisation phase is complete. Then, A2U2 has a throughput of 1 bit of ciphertext per clock cycle.

[1] Due to the selectivity of the shrinking generator it may not output one bit per clock cycle, since some of the input bits are discarded. To solve this issue, the solution proposed in [33] adds a buffer of a few bits (16 to 24) at the output of the filter. Then the LFSRs are clocked twice as fast as the required output of the generator.

## V. Cipher Analysis

### A. Cost Evaluation

In lightweight cryptography, every additional gate is an added cost that must be carefully considered. The additional cost of a gate is even more significant for printed ink RFID tags. Block cipher designers have reduced implementation costs by reducing the key and block sizes, altering P-box and S-box designs and using serial architectures. In the latest hardware-optimised implementation of AES [37], substitutions and permutations represent more than 48% of the 3100 gates. To reduce this area, PRESENT [6] and PRINT [15] used shorter keys (80 bits compared to 128 bits), and block sizes (64 bits and 48 bits respectively, compared to 128 bits). Recent stream cipher designers have achieved area reductions by reducing the size of registers and the complexity of Boolean functions and filter function. Grain [19] uses two 80-bit registers, and hardware-expensive nonlinear Boolean and filter functions that requires 315 gates. As a comparison, the functions of A2U2 require only 42 gates, while its registers are only 17-bit and 9-bit long.

In general, reducing the implementation cost of stream ciphers is relatively more difficult than block ciphers. Block cipher designers have the advantage of maintaining a sufficient level of security by increasing the number of rounds (547 for PRESENT and 768 for PRINT, compared to 160 for AES). However, increasing the number of rounds results lower throughput (see Fig. 7).

A thorough analysis and design of each individual building block of A2U2 has ensured that it can be implemented in less than 300 gates. At the time of writing this paper, the authors have not yet implemented the cipher in hardware. Therefore, all the evaluation details given are estimates. The details of implementation costs are presented in Table I.

TABLE I
GATE ESTIMATE OF A2U2 IMPLEMENTATION

| | Sequential Logic | Combinational Logic | Unit size (in GE) [49] | |
|---|---|---|---|---|
| Counter | 43.75 | 2.25 | 2-input MUX | 2.5 |
| Register 1 | 106.25 | 15.75 | 2-input XOR | 2.25 |
| Register 2 | 56.25 | 10 | 2-input NAND | 1 |
| Key-bit mechanism | 31.25 | 10.75 | 3-input NAND | 1.5 |
| Shrinking filter | - | 7 | D Flip-Flop [7] | 6.25 |
| ***Sub-total*** | 237.5 | 45.75 | | |
| | **TOTAL** | **283.25** | | |

To measure the impact of this result, we compare A2U2 to some of the most well-known or most recent lightweight cryptographic primitives, as well as the most hardware optimised version of AES in Fig. 6. To better reflect the difficulty of reducing the gate count, values in Fig. 6 are represented in a logarithmic scale. In fact, the effort to reduce the cost of a cipher implementation by 50% from 1000 to 500 gates is as difficult (if not more) as downscaling by 50% from 2000 to 1000 gates, while maintaining the same level of security. Such optimisation often comes at the cost of trading-off throughput or power, or both. Furthermore, the number of published block and stream ciphers increase more rapidly with increasing cost thresholds, and rapidly diminish in number below the 1000 gate boundary. An overview of the most well-known ciphers can be found in [48]. To the best of the authors' knowledge, A2U2 is the only stream cipher below the cost threshold of 900 gates[2].

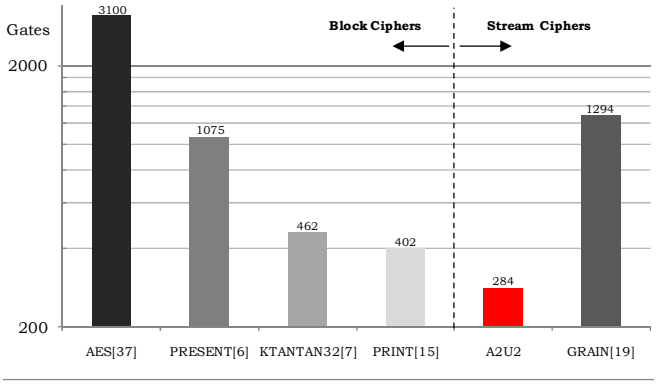


Fig. 6. Comparison of lightweight cryptographic primitives' implementation costs in their area-optimised versions.

The current area available for security in printed ink tags being close to 200 gates, we believe that A2U2 is thus far the most suitable cipher for printed ink RFID technologies. However, printed ink technology is still within its early years of production, and the 200 gates constraint may be increased with further developments in printing techniques and research into material science.

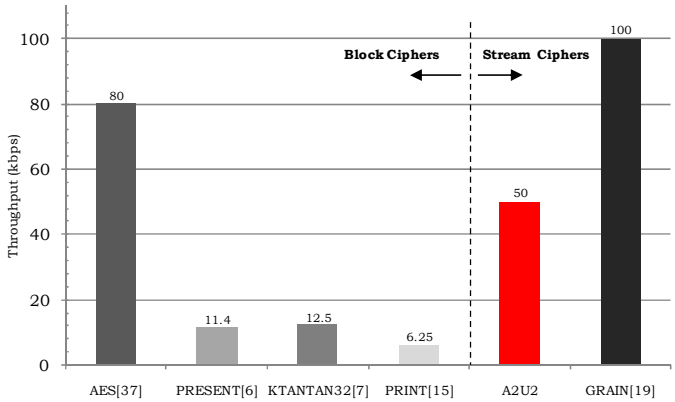


Fig. 7. Comparison of effective throughput when clocked at 100 kHz.

### B. Throughput Evaluation

As a stream cipher, once the initialisation process is complete, one bit of ciphertext is produced per clock cycle. With the particular design of A2U2, based on the use of a shrinking filter, the effective encryption throughput will be approximately half the ciphertext generation throughput. Therefore, it is estimated to be close to 50 kbps when clocked at 100 kHz. We compare the throughput of A2U2 with other ciphers in Fig. 7. The first observation we can formulate, based on Fig. 7, is the clear advantage of stream ciphers over block ciphers in terms of throughput. Despite its effective throughput, which is half its actual throughput, A2U2 outperforms all the compact block ciphers compared by at

[2] A5/1 stream cipher used in GSM networks can be implemented with less than 1000 gates (932 gates). This cipher relied on security through obscurity, but its mechanisms have been discovered and the cipher has been broken [50]. Therefore we did not include it in our comparison charts.

least 400%. Note that in Fig. 6 and Fig. 7, we use AES as a benchmark but, because of its high cost, we do not consider it as a potential candidate cipher for resource constrained devices. We also remind the reader that we compare the same cipher implementations for cost and throughput. Throughput-optimised versions of the block ciphers we compared do exist (see [6], [7], [15]); however, these designs are achieved at the expense of higher implementation costs. Therefore, we have decided to only consider hardware-optimised versions.

The throughput of A2U2 meets the highest transmit requirements of the ISO 14443 Standard [9], and is expected to be more than sufficient for applications envisioned with printed electronics RFID tags, which operate in HF (High Frequency) ISM (Industrial-Scientific-Medical) band.

### C. Security Evaluation

The security of a stream cipher relies on two different aspects: i) the randomness and complexity of its bit stream sequence (its cryptographic strength), and ii) the design criteria followed.

A2U2 has been implemented in software, using the C language. In order to evaluate its output, we used the Statistical Test Suite developed by the National Institute of Standards and Technology (NIST) [38]. This suite consists of a series of 15 tests that evaluate the randomness of a sequence. The various tests include basic tests such as the frequency test, which calculates the number of 0's and 1's, and more elaborate tests such as a linear complexity test, which evaluates if a sequence is complex enough to be considered as random. All the tests return a so called "p-value", which is a condition of passing or failing the test. The p-values are numbers ranging from 0 to 1, while $p > 0.01$ is the condition of success of any given test. The resulting p-value itself does not present much interest, since different series generated will have completely different results for the various tests. The important result in these tests is the "pass or fail" condition. It is recommended to provide as input a sequence of at least 1 million bits to test its randomness. We ran the tests with sequences of 10 million bits, and all the tests returned a "success" value.

The ultimate goal of any cipher design is to be provably secure (e.g. one-time pads). However, in general, the aim is to provide a cipher for which no attack is better than a brute force attack.

It was conjectured by Rueppel in [10] and [44], and confirmed by Dai and Yang in [45] that the linear complexity of a periodic random sequence is close to the period length [46]. Empirical test results on the two registers of A2U2 (tested separately) demonstrate a period in the order of $2^{25.3}$. With A2U2's particular design, we estimate the overall period length of A2U2 (including the counter, the key-bit mechanism and the filter) to be $2^{70}$. Therefore we estimate the linear complexity of A2U2 to be close to $2^{70}$, which currently guarantees lifetime secrecy (a brute force attack on RC5-72 would take an estimate 200 years with a few thousand computers [43]).

Furthermore, a shrunken sequence of a SG has a period of $(2^{N_2}-1)\times(2^{N_1-1})$ [33], which would be slightly below $2^{25}$ if applied to the A2U2 cipher. Therefore our period length of $2^{70}$ has significantly improved on the period of a comparable length SG.

We expect that it will not be possible to improve significantly on a brute force attack, as a result of the nonlinear function blocks, the filter function, and the large linear complexity of the cipher.

Some of the design choices in A2U2 have been made to avoid possible attacks. Firstly, the XOR operation of random numbers, generated both by the tag and the reader to initialise the registers, is a simple yet efficient mechanism to eliminate vulnerabilities arising from attacks such as replay attack [39], tag or reader impersonation, mafia fraud attack [40], man-in-the-middle attack [41] and disclosure attack [42]. The random numbers generated are assumed to be known by eavesdroppers of both the forward and the backward link (i.e. both numbers are assumed to be public information), but the XOR operation with the private key ensures the secrecy of the initialisation value. Secondly, the variable number of rounds during initialisation and the shrinking filter ensure that the cipher outputs a different ciphertext for an identical plaintext message each time the tag is powered. This feature prevents attacks such as replay and tag impersonation attacks. Finally, the filter function diffuses the plaintext bits within the ciphertext in a pseudo-random manner, making it impossible for an attacker to know which bits of the ciphertext contain relevant information.

## VI. Conclusions & Future Work

In this paper, we have presented a novel stream cipher, A2U2, which can be implemented in less than 300 gates. Our approach was based on following a path less travelled, which is the design of a stream cipher as opposed to the various developments in lightweight block ciphers such as KATAN and PRINT Cipher. A2U2 is a worthy candidate for a future implementation in printed electronics RFID tags due to its compactness, simple computational operations, and a throughput of 1 bit per clock cycle (after initialisation). Naturally, since the strength of a cipher is judged overtime and after a lengthy time of public scrutiny, A2U2 is too recent to be envisioned as a ready-to-use solution. In this regard, further work is being carried on A2U2, including further cryptanalysis, a hardware implementation (to obtain more accurate implementation cost and power consumption estimates) and the development of a family of cipher designs.

*Note regarding the cipher name:* A2U2 is the result of merging the two university acronyms, AAU and AUU, that participate in this paper. It is also a wink to the book H2G2 [47], which was a source of inspiration and distraction during the first author's research.

## References


[1] K. Finkenzeller, "Introduction," in *RFID Handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication*, 3rd ed., Ed. Wiley & Sons, 2010, ch. 1, p.1.

[2] R. Das and P. Harrop, "Introduction," in Printed, organic & flexible electronics forecasts, players & opportunities 2009-2029, Ed. IdTechEx, 2009, ch.1, p.24.

[3] P. H. Cole *et al.*, "The next generation of RFID technology," in *Unique Radio Innovation for the 21st Century: Building Scalable and Global RFID Networks*, *Unique Radio Innovation for the 21st Century: Building Scalable and Global RFID Networks* (ed.), Ed. Springer-Verlag, 2010, pp. 3-24.

[4] R. M. Needham and D. J. Wheeler, "Tea extensions," Comp. Lab., Univ. Cambridge, MA, Tech. Rep., Oct. 1997.

[5] F. Stnadaert *et al.*, "SEA: a scalable encryption algorithm for small embedded applications", in *Proc. CARDIS 2006 Conf.*, Tarragona, Spain, Apr. 19-21, 2006, LNCS, vol. 3928, pp. 222-236, 2006.

[6] A. Bogdanov *et al.*, "PRESENT: an ultra-lightweight block cipher", in *Proc. 9th Int. Workshop on Cryptographic Hardware and Embedded Systems – CHES 2007*, Vienna, Austria, Sept. 10-13, 2007, LNCS, vol. 4727, pp. 450-466, 2007.

[7] C. de Cannière, O. Dunkelman and M. Knezevic, "KATAN & KTANTAN – a family of small and efficient hardware-oriented block ciphers", in *Proc. 11th Int. Workshop on Cryptographic Hardware and Embedded Systems – CHES 2009,* Lausanne, Switzerland, Sept. 6-9, 2009, LNCS, vol. 5747, pp. 272-288, 2009.

[8] *Identification cards - Contactless integrated circuit cards - Proximity cards - Part 2: Radio frequency power and signal interface,* ISO/IEC 14443-2 Standard, 2010.

[9] *EPC Class 1 Generation 2 UHF Air Interface Protocol Standard "Gen 2"*, 2008.

[10] R. A. Rueppel, "Linear complexity and random sequences," in *Proc. Advances in Cryptology - EUROCRYPT '85*, Linz, Austria, April 9-12, 1885, LNCS, vol. 219, pp. 167–188, 1985.

[11] C. Shannon, "Communication theory of secrecy systems," Bell System Tech. Jour. 28, issue 4, pp. 656–715, 1949.

[12] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen, "AES implementation on a grain of sand," *IEE Information Security*, vol. 152, issue 1, pp. 13-20, 2005.

[13] G. Leander *et al.*, "New lightweight DES variants", *LNCS*, vol. 4593, pp. 196-210, 2007.

[14] UbiSec&Sens project website [Online]. Available: http://www.ist-ubisecsens.org/

[15] L. Knudsen *et al.*, "PRINT cipher: a block cipher for IC-printing," in *Proc. 12th Int. Workshop on Cryptographic Hardware and Embedded Systems - CHES 2010,* Santa Barbara, CA, Aug. 17-20, 2010, LNCS, vol. 6225, pp. 16-32, 2010.

[16] NESSIE project website (2004) [Online]. Available: https://www.cosic.esat.kuleuven.be/nessie

[17] eSTREAM project website (2008) [Online]. Available: http://www.ecrypt.eu.org/stream/endofphase3.html

[18] A. Shamir, "Stream ciphers: dead or alive?," in *Proc.10th Int. Conf. Theory and Applications of Cryptology and Information Security - ASIACRYPT '04*, Jeju Island, South Korea, December 5-9, 2004, LNCS, vol. 3329, p. 78, 2004.

[19] M. Feldhofer. (2007, January). *Comparison of low-power implementations of trivium and grain*. eSTREAM [Online]. Available: http://www.ecrypt.eu.org/stream/papersdir/2007/027.pdf

[20] T. Good and M. Benaiss. (2007, January). *Hardware results for selected stream cipher candidates*. eSTREAM [Online]. Available: http://www.ecrypt.eu.org/stream/papersdir/2007/023.pdf

[21] K. Yüksel, "Universal hashing for ultra-low-power cryptographic hardware applications," Dept. Elect. Eng., WPI, MA, Master Thesis, 2004.

[22] A. Maximov and A. Biryukov, "Two trivial attacks on Trivium," Cryptology ePrint Archive, Rep. 021, 2007.

[23] W. Meier and O. Staffelbach, "Fast correlation attacks on stream ciphers," in *Proc. Workshop Theory and Application of Cryptographic Techniques, Advances in Cryptology – EUROCRYPT '88*, Davos, Switzerland, May 25-27, 1998, LNCS, vol. 330, pp. 301–314, 1988.

[24] N. Courtois and W. Meier, "Algebraic attacks on stream ciphers with linear feedback," in *Proc. Workshop Theory and Application of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT '03*, Warsaw, Poland, May 4-8, 2003, LNCS, vol. 2656, pp. 346–359, 2003.

[25] K. Khoo and G. Gong, "New constructions for resilient and highly nonlinear Boolean functions," in *Proc. 8th Australasian Conference on Information Security and Privacy*, Wollongong, Australia, July 9-11, 2003, pp. 498-509.

[26] P. Sarkar and S. Maitra, "Construction of nonlinear Boolean functions with important cryptographic properties," in *Proc. 19th international conference on Theory and Application of Cryptographic Techniques*, Bruges, Belgium, May 14-18, 2000, pp. 485-506.

[27] E. Pasalic, S. Maitra, T. Johansson, and P. Sarkar, "New constructions of resilient and correlation immune Boolean functions achieving upper bound on nonlinearity," in *Proc. Int. Workshop on Coding and Cryptography, WCC2001*, Paris, France, Jan. 8-12, pp 158-167, 2001.

[28] W. Meier and O. Staffelbach, "Nonlinearity criteria for cryptographic functions," in *Proc. Workshop Theory and Application of Cryptographic Techniques, Advances in Cryptology – EUROCRYPT '89*, Houthalen, Belgium, Apr. 10-13, 1989, LNCS, vol. 434, pp. 549-562, 1989.

[29] A. Canteaut *et al.*, "Propagation characteristics and correlation immunity of highly nonlinear Boolean functions," in *Proc. Workshop Theory and Application of Cryptographic Techniques, Advances in Cryptology – EUROCRYPT '00*, Bruges, Belgium, May 14-18, 2000, LNCS, vol. 1807, pp. 507-522, 2000.

[30] R.W. Hamming, "Error Detecting and Error Correcting Codes," the Bell System Tech. Jour., Apr. 1950, vol. 29, pp. 147-160.

[31] T. Siegenthaler, "Correlation-immunity of nonlinear combining functions for cryptographic applications," *IEEE Trans. Inf. Theory*, vol. 5, pp. 776–780, 1984.

[32] D. Wheeler and R. Needham, "TEA, a tiny encryption algorithm," Comp. Lab., Cambridge Univ., UK, 1994.

[33] D. Coppersmith, H. Krawczyk, and Y. Mansour, "The Shrinking Generator," in *Proc. 13th Ann. Int. Cryptography Conf., Advances in Cryptology - CRYPTO '93*, Santa Barbara, CA, Aug. 22-26, 1993, LNCS, vol. 773, pp. 22-39, 1994.

[34] P. Caballero-Gil, A. Fúster-Sabater and M. E. Pazo-Robles, "New attack strategy for the Shrinking Generator," *Journal of Research and Practice in Information Technology*, vol. 41, no. 2, May 2009.

[35] B. Schneier, "Self-Study Course in Block Cipher Cryptanalysis", in *Cryptologia*, vol. 24, n. 1, pp. 18-34, Jan 2000.

[36] G. Leander, A. Poschmann, "On the Classification of 4-Bit S-Boxes", in *Proc. Arithmetic of Finite Fields,* 1st International Workshop, WAIFI 2007, LNCS, vol. 4547, pp. 159-176, 2007.

[37] P. Hämäläinen *et al.*, "Design and implementation of low-area and low-power AES encryption hardware core," in *Proc. 9th EUROMICRO Conf. on Digital System Design: Architectures, Methods and Tools*, Dubrovmik, Croatia, Aug. 30 – Sept. 1, 2006, pp. 577–583.

[38] A. Rukhin *et al.*, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," National Institute of Standards and Technology, NIST, Spe. Publi. 800-22, Apr. 2010.

[39] T. Aura, "Strategies against replay attacks," in *Proc. 10th Comp. Security Foundations Workshop*, Rockport, MA, Jun. 10-12, 1997, pp. 59-68.

[40] A. Alkassar and C. Stfible, "Towards secure IFF: preventing mafia fraud attacks", in *Proc. 21st Century Military Communications Conf. - MILCOM 2002*, Anaheim, USA, Oct. 2002, vol. 2, pp. 1139-1144.

[41] R. Cramer and I. Damgård, "Fast and secure immunization against adaptive man-in-the-middle impersonations," in *Proc. Workshop Theory and Application of Cryptographic Techniques, Advances in Cryptology – EUROCRYPT '97*, Konstanz, Germany, May 11-15, 1997, LNCS, vol. 1233, pp. 75–87, 1997.

[42] D. Agrawal and D. Kesdogan, "Measuring anonymity: the disclosure attack," *IEEE Security & Privacy*, vol. 1, issue 6, pp. 27-34, 2003.

[43] RC5-72 Project, [Online]. Available: http://www.distributed.net/RC5

[44] R. A. Rueppel, "New Approaches to Stream Ciphers," Swiss Federal Institute of Technology, Zürich, Ph.D. Thesis, 1984.

[45] Z.D. Dai and J.H. Yang, "Linear complexity of periodically repeated random sequences," in *Proc. Workshop Theory and Application of Cryptographic Techniques, Advances in Cryptology – EUROCRYPT '91*, Brighton, UK, Apr. 8-11, 1991, LNCS, vol. 547, pp. 168–175, 1991.

[46] B. M. Gammel and R. Göttfert, "Combining Certain Nonlinear Feedback Shift Registers," in *Proc. Workshop Record of SASC – The State of the Art of Stream Ciphers*, Bruges, Belgium, 2004, pp. 234–248.

[47] D. Adams, "H2G2: the hitchhikers guide to the galaxy," Ed. Pan Books, Oct. 1979.

[48] Selected Block Cipher Listing, ECRYPT website: http://www.ecrypt.eu.org/lightweight/index.php/Block_ciphers

[49] B. M. Gammel, R. Gottfert, and O. Kniffler, "The Achterbahn stream cipher," eSTREAM, ECRYPT Stream Cipher Project, Tech. Rep. 002, 2005.

[50] A. Biryukov, A. Shamir, and D. Wagner, "Real Time Cryptanalysis of A5/1 on a PC," in *Proc. Fast Software Encryption Workshop 2000,* New York City, April 10-12, 2000.