Secure Dynamic Authentication of Passive Assets and Passive IoTs using Self-powered Timers

Liang Zhou and Shantanu Chakrabartty
Department of Electrical and Systems Engineering
Washington University in St. Louis
St. Louis, MO, U.S.A.
{liang.zhou, shantanu}@wustl.edu

Abstract—A major limitation in authenticating passive and remotely powered sensors, tags and cards (for e.g. radio-frequency identification tags or credit cards) is that these devices do not have access to a continuously running system clock. This obviates the use of SecureID type authentication techniques involving random keys and tokens that need to be periodically generated and synchronized. In this paper we present a dynamic hardware-software authentication approach for passive assets using zero-power timers and synchronization circuits. The timers are shown to achieve robust temporal synchronization due to the self-powering and self-compensating physics of Fowler-Nordheim (FN) quantum transport of electrons tunneling onto a floating-gate. The output of the timers are then used to seed a pseudo-random number generator which produce random and synchronized authentication tokens. We validate the proposed approach using prototypes fabricated in a standard 0.5-μm CMOS process where we demonstrate synchronization accuracy greater than 40dB. Compared to conventional static authentication methods that are currently used for passive sensors, tags and cards, the proposed dynamic approach should provide enhanced security and make it more immune to counterfeiting and data theft.

I. INTRODUCTION

One of the major challenges for passive assets like credit cards and passive devices used in Internet-of-Things (IoTs) such as radio-frequency tags is to ensure secure authentication and secure access to critical data. These assets have very limited computational capabilities which obviates the use of high-performance encryption techniques, use of strong hash functions and embedding of complex pseudo-random number generators (pRNG) [1], [2]. Also, the need for fast and real-time authentication obviates the use of multi-level authentication and challenge-response protocols in these assets. In this regard, dynamic techniques like secureID type authentication [3] could be an alternative mechanism to conventional static approaches (using barcodes or product IDs [1]) by providing immunity to data theft and data counterfeiting. However, secureID type authentication requires access to a continuous and a synchronized clock which is not available on passive devices. Zero-power and self-powered timers [4], [5] can overcome this limitation by providing a synchronization mechanism without the need for any external powering. Thus, a secureID type dynamic authentication protocol can be implemented for passive assets, as summarized in Fig. 1. Each of the passive asset, shown in Fig. 1, is integrated with a self-powered timer, which is synchronized to a “Gold-standard timer” located on an authentication server. In this paper the tokens are generated by seeding the input of a pseudo-random number generator using the output of the zero-power timers.

Rapid dynamic authentication is then achieved by comparing the synchronized random tokens that are generated locally on the asset with the ones that are received from the authentication server. Because the output of the timer varies with time, the proposed approach produces a sequence of random number tokens that will be very difficult to predict and hence model using reverse-engineering techniques.

However, the success of the proposed dynamic authentication approach depends on three key aspects: (a) the timers need to continuously operate over a time-period that matches the life-cycle or the shelf-life of the passive asset (at least 3 years); (b) different timers can be accurately synchronized with respect to each other and with the “gold-standard” timer, as illustrated in Fig. 1 and (c) synchronized random numbers can be generated when seeded with the output of the timers. In [5] we proposed a zero-power timer structure based on Fowler-Nordheim (FN) tunneling of electrons through thin gate-oxide. It was shown that the physics of FN tunneling is reliable and can be accurately described using a log-linear model which can be used for implementing the ”gold-standard” timer on the authentication server. In this paper we validate the robustness of FN timers (to mismatch and fabrication errors) using prototypes fabricated in a standard 0.5-μm CMOS process. We then use the FN timer output to generate the seed for a linear feedback shift register (LFSR) type random number generator (RNG) and we show that the proposed scheme can
generate synchronized random tokens that is necessary for authentication.

II. FN TIMER SEEDED RANDOM NUMBER GENERATION

The structure of a generic FN timer comprises of a strip of poly-crystalline silicon (polysilicon) that is completely insulated by high-quality, thermally-grown silicon-di-oxide. This polysilicon strip, also referred to as a floating-gate (FG), serves as a reservoir of electrons and the surrounding silicon-di-oxide will serve as an energy barrier (shown in Fig. 2(a)) that prevents the electrons to leak out (by thermal excitation or quantum tunneling). Electrons can leak through the oxide using three mechanisms of quantum tunneling [6]: (a) Trap-assisted tunneling (TAT) where carriers tunnel through the dielectric layer with the assistance of trap states, (b) direct tunneling (DT) where carriers directly tunnel through the rectangular barrier formed by the dielectric layer, and (c) Fowler-Nordheim tunneling (FNT) where carriers tunnel through a triangular-shape barrier resulted from a strong electric field. For thermally-grown gate oxide with thickness larger than 10 nm (which is usual in 0.5-μm process), TAT and DT are negligible. FN tunneling, on the other hand, depends on the shape of the energy barrier across the dielectric layer which can be controlled and modulated by changing the electric field across the dielectric layer. The physics of FN tunneling, as illustrated in Fig. 2 is a two step process. Electrons are first thermally excited to an energy level $E$ which then tunnel through the triangular barrier into the floating-gate. Note that at the bottom of the energy barrier, the oxide thickness is large enough (greater than 10nm in a 0.5μm CMOS process) that the probability of electrons directly tunneling through is negligible. Thus, FN tunneling can be modeled by an equivalent energy scavenging circuit shown in Fig. 2(a) where the input energy source is the ambient thermal-activation or thermal-noise and rectification diode is formed by the tunneling barrier whose output is the floating-gate capacitance. By using the expression of FN tunneling current and the electric field across the gate-oxide, the dependence of floating-gate voltage on time $t$ can be expressed as [5]:

$$V_{fg}(t) = \frac{k_2}{\ln(k_1t + k_0)} + V_{sub} \quad (1)$$

where

$$k_0 = \exp\left(\frac{\beta}{E_0}\right), \quad k_1 = \frac{A_0}\beta C_T C_{ox}, \quad k_2 = \beta t_{ox} \quad (2)$$

are the model parameters. $A_0$, $E_0$, $C_T$, $C_{ox}$ and $t_{ox}$ are a function of the material properties. $h$ and $q$ correspond to the Plank’s constant and charge of free electrons. $m^*$ is the effective mass of electrons in the forbidden gap of the silicon-di-oxide, while $m$ is the mass of a free electron. $\phi$ is the barrier height at the interface. $E_0$ in $k_0$ is the initial electric field across the gate oxide, and $t_{ox}$ is the effective thickness of the gate oxide. $A$ and $C_T$ are the tunneling junction area and gate capacitance respectively. The model represented by equation 1 can achieve an accuracy of 40 dB [5].

While the initial timer response is a function of the parameters $k_0$, $k_1$ and $k_2$, for the time intervals $t > t_0 \gg k_0/k_1$ the response becomes

$$\Delta V_{fg}(t) \approx \frac{k_2(\ln t_0 - \ln t)}{\ln t_0 \ln t} \quad (3)$$

This regime given by the equation 3 is labeled as the equilibrium region and reveals two important attributes which are useful for designing robust long-term timers. First, the floating-gate voltage monotonically decreases and is inversely proportional to a logarithmic function of time. Thus the rate of decrease is slower than a linear timer and faster than a saturating response of a RC-type filter. This implies that the timer could be functional for a long-term duration relevant to the proposed application. The second attribute shown in equation 3 is that the response is only a function of the parameter $k_2$ which depends only on the material properties and the oxide-thickness, as described by equations 2. Thus, asymptotically, the response of the timer becomes theoretically independent of device sizes (cross-sectional area and floating-gate capacitance). Timers based on equation 1 should be robust to device mismatch and can be used to synchronize the seed for a random number generation as shown in Fig. 2(b).

The system architecture of a generic random number generator seeded by the self-powered timer is illustrated in Fig. 2(b). The system comprises of two components: (a) the timer which is self-powered and continuously keeps track of time; and (b) an analog-to-digital converter (ADC) and an RNG which is...
powered externally when an authentication token is requested from the system. Note that the authentication token is only requested when the passive asset is connected to an external power source (card inserted into a card reader or an remote reader accessing the RFID tag). Thus, during the read-out phase there is no constraint on the availability of power. When a request signal is sent to the system, the timer value is readout and digitized using an ADC. The digitized value is then used to feed a linear feedback shift register (LFSR) as the initial value. An exclusive "OR" operator generates the input bit for the shift register based on a feedback connection from different stages of the shift register. The choice of different topologies for maximum-length period LFSRs have been discussed in [7] and is omitted in this paper for the sake of brevity. After a certain number of cycles of shift operation, the generated random number can be used for further operation. Ideally, if two timers are completely synchronized, the seeds generated by them are identical, therefore the random number generated by them should be also identical and hence can be used for authentication.

III. CIRCUIT IMPLEMENTATION AND MEASUREMENT RESULTS

The die photograph of the timer is shown in Fig. 3, where the floating-gate is formed by the gate of a pMOS transistor which is also used for programming the initial charge onto the floating-gate [8]. The floating-gates were programmed using a combination of Fowler-Nordheim (FN) tunneling and hot-electron injection. FN tunneling removes the electrons from the floating-gate node by applying a high-voltage (15 V in 0.5 μm CMOS process) across a parasitic nMOS capacitor $C_{tun}$ (as shown in Fig. 3). Hot-electron injection, however, requires lower voltage (4.2 V in 0.5 μm CMOS process) than tunneling and hence is the primary mechanism for precise programming of floating-gates. The hot-electron programming procedure involves applying greater than 4.2 V across the source and drain terminals of the transistor M. The large electric field near the drain of the pMOS transistor creates impact-ionized hot-electrons whose energy when exceeds the gate-oxide potential barrier (3.2 eV) can get injected onto the floating-gate. Because the hot-electron injection in a pMOS transistor is a positive feedback process and can only be used to add electrons to the floating gate, the process needs to be carefully controlled and periodically monitored to ensure the floating-gate voltage is programmed to a desired precision. The methods proposed in literature achieve the desired precision either by adjusting the duration for which the FG transistor is injected or by adjusting the magnitude of the injection pulses [8]. The tunneling junction area is chosen as 54 μm² and the gate capacitance is 4 pF. To characterize the timer device, a unity gain buffer is employed to readout the floating-gate voltage as shown in Fig. 3.

The measured response of the timer is shown in Fig. 4 which is compared against the behavioral model summarized by equations (1). The measured results and the behavioral model show a close match with respect to each other and the behavioral model was used to determine the long-term response of the timer, as shown in Fig. 4. The results show that the timers can continuously operate for durations greater than 3 years. Data points measured from the fabricated timer is overlayed on the behavioral response and is shown to exhibit an accurate fit up to $2 \times 10^6$ seconds or 550 hours. This experiment validates that the behavioral model of the timer could be used to implement a software version of the timer on the authentication server shown in Fig. 1.

The second group of experiments are designed to verify the equilibrium response of the timer implied by equation 3. All the timers fabricated on different dies were initially programmed to the “approximately” same voltage of 8.5 V. Due to mismatch, timers fabricated on different dies should display a distinct response which is validated by the measurement results from three different dies shown in Fig. 5(a). However, after a reference of $6 \times 10^4$s, the timers operate in the "equilibrium region" where the voltage change of the timers were plotted in Fig. 5(b). The timers show almost identical responses, the maximum deviation of which is shown in Fig. 5(c). The maximum deviation across the whole equilibrium region is less than 1.8 mV for a dynamic range of 180 mV, which implies an accuracy of synchronization more than 40 dB. Therefore, the timers response in this region can be used for authentication purpose by checking the sychronicity between the device timers and the server timers.

The last set of experiments were conducted to verify the generation of authentication tokens using the RNGs seed by the timer outputs. The change in the timer outputs were first digitized using a 8-bit ADC with 0.25V reference voltage. The generated bit sequence is then used to seed a 8-bit LFSR random number generator. The random number was collected after running 10000 cycles shift operation. The outputs of the timers were sampled for seeding operation every 50000
Fig. 5. Comparison of variations of timers across different dies: (a) measured temporal response, (b) timer value reduction after a reference time, (c) maximum deviation among the three dies.

Fig. 6. Generated random numbers distribution using a 8-bit LFSR seeded by outputs from the timer: (a) seed without patching code and (b) seed with 3-LSB patching code.

accurately synchronized with respect to each other with an accuracy greater than 40dB and the output of the timer could be used to generate synchronized and reliable authentication tokens. The future work would be focusing on designing robust authentication algorithms and integration of the system.

V. ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under Grant CNS 1525476 and Grant ECCS 1550096, and in part by Semiconductor Research Corporation under Contract 2015-TS-2640.

REFERENCES


