Range-Controlled Floating-Gate Transistors: A Unified Solution for Unlocking and Calibrating Analog ICs

Sai Govinda Rao Nimmalapudi, Georgios Volanis, Yichuan Lu,
Angelos Antonopoulos, Andrew Marshall, Yiorgos Makris
ECE Department, The University of Texas at Dallas

***Abstract*—Analog Floating-Gate Transistors (AFGTs) are commonly used to fine-tune the performance of analog integrated circuits (ICs) after fabrication, thereby enabling high yield despite component mismatch and variability in semiconductor manufacturing. In this work, we propose a methodology that leverages such AFGTs to also prevent unauthorized use of analog ICs. Specifically, we introduce a locking mechanism that limits programming of AFGTs to a range which is inadequate for achieving the desired analog performance. Accordingly, our solution entails a two-step unlock-&-calibrate process. In the first step, AFGTs must be programmed through a secret sequence of voltages within that range, called *waypoints*. Successfully following the waypoints unlocks the ability to program the AFGTs over their entire range. Thereby, in the second step, the typical AFGT-based post-silicon calibration process can be applied to adjust the performance of the IC within its specifications. Protection against brute-force or intelligent attacks attempting to guess the unlocking sequence is ensured through the vast space of possible waypoints in the continuous (analog) domain. Feasibility and effectiveness of the proposed solution is demonstrated and evaluated on an Operational Transconductance Amplifier (OTA). To our knowledge, this is the first solution which leverages the power of analog keys and addresses both unlocking and calibration needs of analog ICs in a unified manner.**

I. Introduction

Due to the increased variability of advanced semiconductor manufacturing processes and the cutting-edge performance expectations of contemporary electronics, analog IC designers are faced with a dilemma: design conservatively, essentially leaving performance on the table but ensuring high yield, or design aggressively, essentially pushing performance but risking low yield. In an effort to break this stalemate, *post-production calibration* has become standard practice of most modern analog IC designs [1]. Specifically, tunable elements are included in the design and are used to fine-tune the performance of each analog IC after fabrication. Among the various technologies used for post-production calibration of analog ICs, the use of AFGTs as tunable design elements has become a popular, intuitive and efficient way to permanently counteract the impact of process variations [2], [3].

Simultaneously, the globalization of semiconductor manufacturing through the prevalence of the fabless business model has led to significant concerns regarding IC piracy and unauthorized use of intellectual property. As a result, various analog and mixed-signal IC locking methodologies have been developed recently [4]–[8]. All these methods require the use of a secret key in order to enable specification-compliant operation. In [4], multiple parallel transistors are used to produce the bias current of the analog IC. A digital key is, then, used to select which transistors are turned on, with the correct key producing the desired bias. The main limitation of this solution is that the performance degradation could be very small when incorrect keys are applied. To address this limitation, the method proposed in [5] used satisfiability modulo theory to design a configurable current mirror. In the context of mixed-signal designs, the techniques developed in [6], [7] used a provably-secure logic locking scheme to lock the digital part of the design. Along a different direction, in [8], an analog neural network was trained to implement a point function for biasing a low noise amplifier. In this approach, the weights stored in the analog synapses constitute the key.

In this work, we propose a method for preventing manufactured analog ICs, which have been illegitimately acquired, from being calibrated and operated within their specifications. Conceptually, our solution is to extend the operation of existing AFGTs, which are used for calibrating analog portions of the IC, so that they also serve the purpose of unlocking the IC prior to enabling calibration. To achieve this, we introduce a mechanism for initially limiting the range in which the AFGTs can be programmed, such that no value within this limited range can bring the performance of the analog IC within its expected specifications. In order to break out of this limited AFGT programming range, a sequence of 'secret' analog values, termed *waypoints*, must be programmed into the AFGTs. The number, sequence, and analog values of these waypoints constitute the key required for unlocking the IC and enabling full programmability of the AFGTs, in order to calibrate it within specifications. Considering the analog nature of the key and the multiple waypoints that must be programmed prior to unlocking the IC, an adversary who seeks to break this locking mechanism is faced with the futile task of a blind search in a vast space of options. The proposed method is demonstrated through detailed simulations on an OTA.

We emphasize that the proposed method is the first analog IC locking solution which uses *analog keys* and considers the need for *post-production calibration of manufactured analog ICs* by leveraging the underlying mechanism to facilitate both unlocking and calibration using the same hardware. The remainder of this paper is structured as follows. In Section II, we explain the use of AFGTs for the purpose of post-silicon calibration by using an OTA as an example. In Section III, the proposed method is introduced and demonstrated on the OTA in Section IV. Simulations results are shown in Section V, and conclusions are drawn in Section VI.

II. Post-Silicon AFGT-Based calibration

The AFGT includes a standard MOS transistor with an electrically isolated secondary gate, formed by using a control gate capacitor (*Ccg*) and a tunneling capacitor (*Ctun*), as shown in Figure 1(a). A resultant floating gate node (*Fg*) is

978-3-9819263-4-7/DATE20/©2020 EDAA

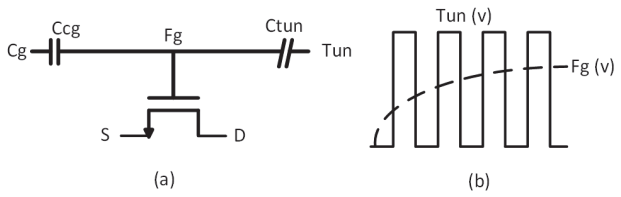

Fig. 1. AFGT: (a) Basic structure (b) Programming pulses and *Fg* voltage

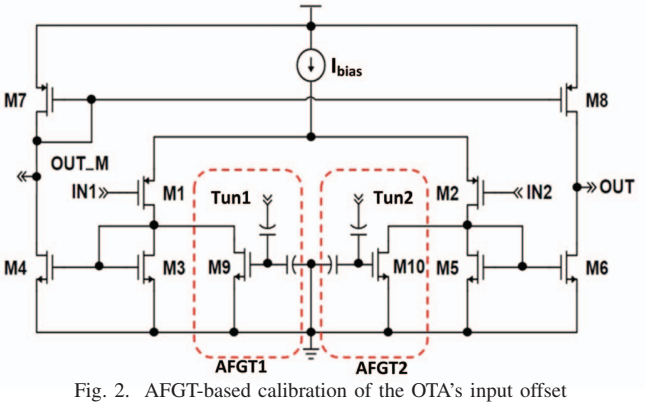

Fig. 2. AFGT-based calibration of the OTA's input offset

formed, which is surrounded by an electrically insulating layer. Charge trapped at the floating gate is, thus, stored permanently.

Adding or removing charge to or from the floating gate resembles adjusting the threshold voltage of the MOS transistor. The programmability of AFGTs, along with their ability to serve as integral parts of an analog IC design, makes them particularly appealing as tunable components for calibrating an IC to its desired operating point. A detailed description of the AFGT, as well as its programming mechanisms, can be found in [9]. Here, we use Fowler Nordheim (FN) tunneling to program the AFGTs by applying voltage pulses at the *Tun* terminal while the *Cg* terminal is grounded. These pulses are shown in Figure 1(b), along with the floating gate voltage.

As an example of AFGT-based calibration, we use the OTA shown in Figure 2, where AFGTs are used for input offset cancellation, as detailed in [10]. Input offset is an undesired feature for OTAs, which is caused by mismatches in the parameters of the differential input pair transistors due to process variations. Such mismatches make the operating conditions of these transistors deviate from the conditions they were designed for, thereby affecting the overall OTA performance. Our OTA consists of the differential input pair (*M1*, *M2*) and three current mirrors (*M3* & *M4*, *M5* & *M6*, *M7* & *M8*) which reflect the current of the input pair to the output (OUT). A current source (*Ibias*) provides biasing for the differential input pair. The calibration scheme comprises two AFGTs (*M9* & *M10*) which are enclosed in dashed boxes.

After fabrication, the input transistors are generally not perfectly matched and the threshold voltage of one (say *M1*) can be lower than the threshold voltage of the other (say *M2*). This results in higher drain-to-source current through *M1* compared to *M2*. For zero input offset, however, these currents must be equal, each being half of the current provided by the *Ibias*. To this end, the AFGT1 can be programmed such that

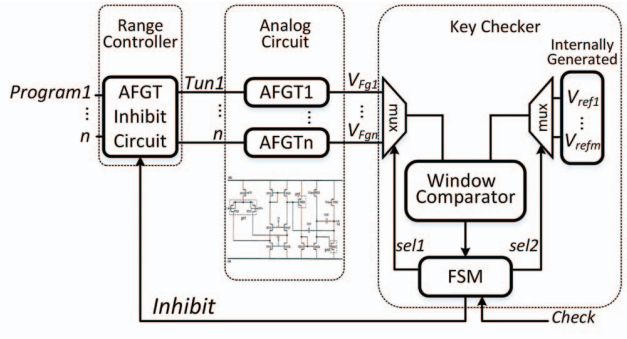

Fig. 3. Block diagram of the proposed method

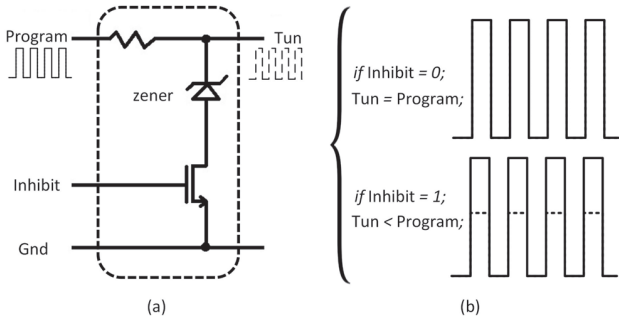

Fig. 4. AFGT inhibit circuit: (a) Circuit diagram (b) Limited voltage pulses applied at the *Tun* terminal of the AFGT

the excessive current of *M1* flows through it, thereby matching the currents and eliminating the input offset.

III. Proposed Method

Overview: Our method relies on locking the ability to successfully program the AFGTs which are already available in an analog IC design for enabling post-manufacturing performance calibration. To achieve this, we introduce a mechanism that allows us to control the programming range of the AFGTs. A block diagram of the proposed method is presented in Figure 3, where an analog circuit with n AFGTs and m waypoints (i.e., V_{ref1} to V_{refm}) is considered. The Key Checker controls the *Inhibit* signal which unlocks full-range AFGT programming through the Range Controller only when the correct sequence of waypoints has been programmed in the AFGTs.

AFGT Inhibit Circuit: The implementation of the AFGT Inhibit Circuit, which controls the programming range of the AFGTs, is shown in Figure 4(a). It consists of a Zener diode, a resistor and an NMOS switch connected in series. Zener diodes are widely used as voltage regulators when they are reverse-biased and the applied reverse bias voltage reaches a predetermined value called reverse-breakdown voltage. From that point on, the Zener diode has low impedance and acts as a sink so that the current through the load, which is connected in parallel to the diode, remains constant. Thus, the voltage across the *Tun* terminal to *Gnd* in Figure 4(a) is also kept constant. The purpose of the resistor is to limit the current flowing into the Zener diode and the load, whereas the purpose of the switch is explained in Figure 4(b). If the *Inhibit* signal is low, the switch is off and the Zener diode has no impact on

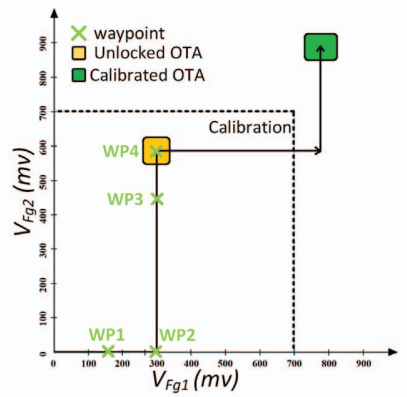

Fig. 5. Waypoint-based OTA unlocking and calibration using two AFGTs

the amplitude of the pulses applied to the *Program* terminal while FN tunneling is performed. In other words, the exact same pulses that are applied to the *Program* terminal will be applied to the *Tun* terminal of the AFGT transistor. On the other hand, if the *Inhibit* signal is high, the switch is on and the Zener diode will limit the amplitude of the pulses provided to the *Program* terminal, which in turn will limit the amount of charge (i.e., floating gate voltage, V_{Fg}) that will be trapped at the floating gate during FN tunneling. The *Inhibit* signal is controlled by the Key Checker.

Key Checker: The Key Checker shown in Figure 3 consists of a window comparator, a Finite State Machine (FSM) and two multiplexers which are controlled by the FSM. The output of the FSM is the *Inhibit* signal, which is initially set to high, resulting in a limited programming range for the AFGTs. The FSM starts by selecting the voltage at the floating gate of the AFGT that we want to monitor (i.e., V_{Fg1}), as well as the first waypoint (i.e., V_{ref1}), which is generated internally. These two voltages are provided at the inputs of the window comparator and, in case they are equal, the FSM proceeds with the next pair of floating gate voltage and waypoint. This process continues until all waypoints are considered. If the AFGTs are programmed such that their floating gate voltages match the waypoints, the FSM produces a low inhibit signal which allows programming of the AFGTs in their entire range. On the other hand, if at any point the compared voltages are not equal, the process will continue but the FSM will never lower the *Inhibit* signal. Hence, the only way to calibrate the IC is to power-cycle it and re-attempt the unlocking procedure.

IV. Demonstration using OTA Example

We applied the proposed method to the OTA described in Section II, using the pre-existing $n = 2$ AFGTs and a sequence of $m = 4$ waypoints. Figure 5 visualizes the process required for unlocking the ability to calibrate the OTA by programming the two AFGTs. Initially, the *Inhibit* signal is high, limiting the maximum voltage that can be programmed into the two AFGTs to 700mV (dashed line). Values within this voltage range are insufficient to eliminate the input offset of the OTA. Let us assume that, at first, the two AFGTs have zero charge trapped at their floating gates. This results in 0V at their floating gates, which corresponds to the origin in Figure 5. The FSM will produce a low *Inhibit* signal to remove the 700mV

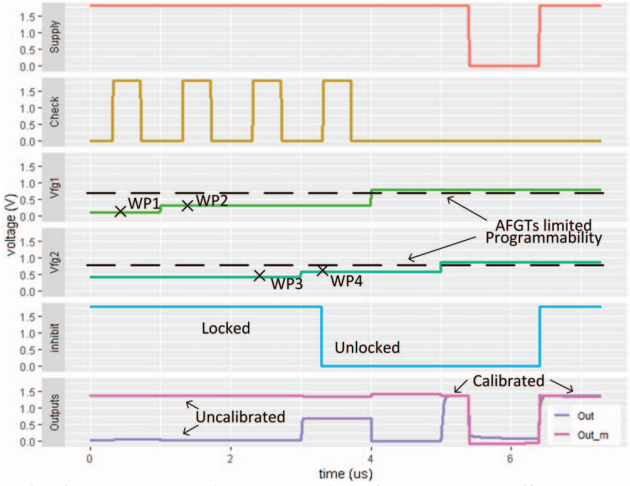

Fig. 6. Correct waypoint programming of AFGTs: OTA offset corrected

limitation only if the predefined sequence of waypoints, shown through green X symbols, is programmed into the two AFGTs. This sequence starts with the first waypoint, which means that AFGT1 needs to be programmed such that its floating gate voltage is equal to 150mV. The second waypoint is, again, related to AFGT1, which this time should be programmed to a 300mV floating gate voltage. The next two waypoints are associated with AFGT2, whose floating gate voltage must be programmed first to 450mV and then to 600mV, to reach the third and fourth waypoint, respectively. At this point, the FSM lowers the *Inhibit* signal so that the two AFGTs can be programmed in their full range (i.e., green square) where the input offset is corrected, thus calibrating the OTA.

V. Simulation Results

We designed the OTA and the appropriate Range Controller and Key Checker for our method in GlobalFoundries' 130nm CMOS process. To simulate the impact of input offset on the OTA performance, we added a DC voltage of 15mV to the input *IN1* of the OTA shown in Figure 2, which results in the outputs (i.e., *OUT* and *OUT_M*) being imbalanced. Below, we elucidate the effectiveness of the proposed method in locking the ability to calibrate the OTA, i.e., to make the two output values equal to the desired DC voltage.

Successful Unlocking and Calibration: Figure 6 shows a successful unlocking of the OTA. The first two rows show the supply voltage and the *Check* signal which signifies when to check validity of a waypoint. The third and fourth rows show the programmed floating gate voltages of AFGT1 and AFGT2, respectively. The next row shows the *Inhibit* signal and the last row plots the two OTA outputs. Before the AFGTs have been programmed to the required sequence of waypoints (i.e., WP1-WP4), the *Inhibit* signal remains high, limiting the programming range of the AFGTs to 0.7V, as highlighted by the two dashed lines. Within this range, the input offset cannot be calibrated and the two outputs *OUT* and *OUT_m* differ significantly. After the waypoint sequence has been successfully programmed in the two AFGTs, the *Inhibit* signal becomes low, unlocking the ability to program the AFGTs to voltages higher than 0.7V, which are needed for calibration of

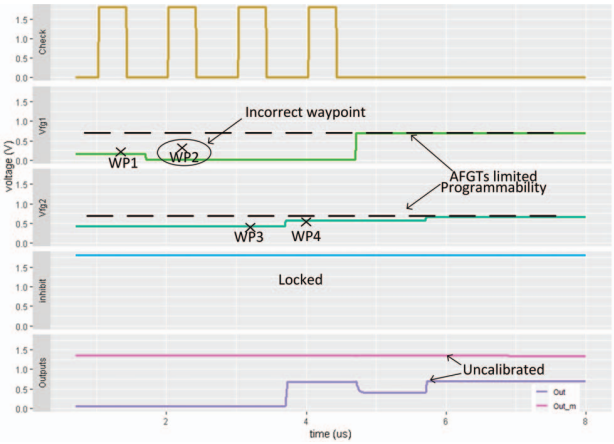

Fig. 7. Incorrect WP2 programmed in AFGT1: OTA offset remains

the OTA. As shown in the plot, at around 5us, when the AFGTs are programmed to the appropriate calibration values, the two outputs become balanced. Once the OTA is calibrated, we remove power from the IC (at around 5.4us). Since AFGTs are non-volatile storage, when we power the circuit on again (at 6.4us), they retain their programmed voltages and the circuit remains calibrated. However, the *Inhibit* signal returns to the high value, which implies that the AFGTs cannot be further programmed to values above 0.7V without having to first pass again through the correct waypoint sequence.

Failed Unlocking: Figure 7 shows an unsuccessful attempt to unlock the OTA. Although, the first programmed floating gate voltage of AFGT1 matches the first waypoint (i.e., WP1) the second voltage programmed into AFGT1 is less than the expected value for the second waypoint (i.e., WP2), which is 300mV. As a result, the *Inhibit* signal remains high despite the following two waypoints being correctly programmed into AFGT2. Calibration of the OTA is now impossible; as shown in the bottom part of the figure, the limited programming range of the AFGTs is insufficient to compensate the input offset, so the two OTA outputs cannot converge.

Programming Range Control: Figure 8 shows the operation of the AFGT Inhibit Circuit of Figure 4, which controls the range in which the AFGTs can be programmed. The first row shows the repetitive programming pulses that are applied to the *Program* terminal. The number and, most importantly, the amplitude of these pulses affects the voltage that is stored in the floating gate of the AFGT. When the *Inhibit* signal (shown in the second row) is high, the actual amplitude that passes to the *Tun* terminal of the AFGT (shown in the third row) is limited to a lower value (i.e., 7V) instead of the full value (i.e., 8V) that is applied to the *Program* terminal. This lower value restricts the floating gate voltage of the AFGT (shown in the last row) to 0.7V, which is inadequate to calibrate the input offset of the OTA. Once all the waypoints have been successfully negotiated, the *Inhibit* signal becomes low and the programming pulses that reach the *Tun* terminal are equal to the ones applied to the *Program* terminal and can reach the amplitude of 8V, which is sufficient for programming the floating gates of the AFGTs in their entire range.

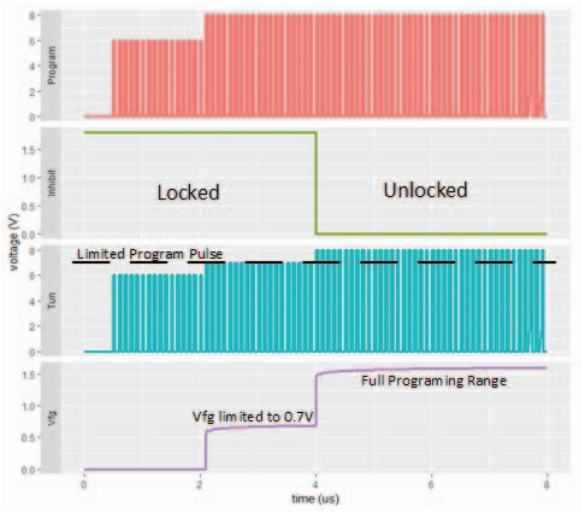

Fig. 8. Programming voltage control through *Inhibit* signal

VI. Conclusion

AFGTs are often used during post-manufacturing calibration to counteract the impact of process variations on analog IC performance. As demonstrated herein, the same AFGTs can also be used to address another contemporary problem of semiconductor manufacturing, namely to prevent the use of illegitimately-acquired, fabricated ICs. Specifically, we introduced a mechanism which prevents successful calibration of an analog IC unless the AFGTs are first programmed through a secret sequence of analog voltages, named waypoints, after which full-range programming of the AFGTs is enabled. Our method was applied and demonstrated on an OTA, corroborating its effectiveness as the first analog key-based, unified solution for both unlocking and calibrating analog ICs.

References

[1] Y. Lu, K. S. Subramani, H. Huang, N. Kupp, K. Huang, and Y. Makris, "A comparative study of one-shot statistical calibration methods for analog/RF ICs," in *International Test Conference*, 2015, pp. 1–10.

[2] V. Srinivasan, G. J. Serrano, J. Gray, and P. Hasler, "A precision CMOS amplifier using floating-gate transistors for offset cancellation," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 2, pp. 280–291, 2007.

[3] V. Srinivasan, G. Serrano, C. M. Twigg, and P. Hasler, "A floating-gate-based programmable CMOS reference," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 11, pp. 3448–3456, 2008.

[4] V. V. Rao and I. Savidis, "Protecting analog circuits with parameter biasing obfuscation," in *Latin American Test Symposium*, 2017, pp. 1–6.

[5] J. Wang, C. Shi, A. Sanabria-Borbon, E. Sánchez-Sinencio, and J. Hu, "Thwarting analog IC piracy via combinational locking," in *International Test Conference*, 2017, pp. 1–10.

[6] N. G. Jayasankaran, A. S. Borbon, E. Sanchez-Sinencio, J. Hu, and J. Rajendran, "Towards provably-secure analog and mixed-signal locking against overproduction," in *International Conference on Computer-Aided Design*, 2018, pp. 1–7.

[7] J. Leonhard, M. Yasin, S. Turk, M. T. Nabeel, M. M. Louërat, R. Chotin Avot, H. Aboushady, O. Sinanoglu, and H. G. Stratigopoulos, "Mixlock: Securing mixed-signal circuits via logic locking," in *Design, Automation & Test in Europe*, 2019, pp. 84–89.

[8] G. Volanis, Y. Lu, S. G. Rao Nimmalapudi, A. Antonopoulos, A. Marshall, and Y. Makris, "Analog performance locking through neural network-based biasing," in *VLSI Test Symposium*, 2019, pp. 1–6.

[9] P. Hasler and J. Dugger, "Correlation learning rule in floating-gate pFET synapses," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 65–73, 2001.

[10] U. Patel, S. Nimmalapudi, H. Stiegler, A. Marshall, and K. Jarreau, "Enhancing circuit operation using analog floating gates," in *International Symposium on Quality Electronic Design*, 2018, pp. 221–226.