



HAL
open science

A High-Level Design Flow for Locally Body-Biased Asynchronous Circuits

Yoan Decoudu, Katell Morin-Allory, Laurent Fesquet

► **To cite this version:**

Yoan Decoudu, Katell Morin-Allory, Laurent Fesquet. A High-Level Design Flow for Locally Body-Biased Asynchronous Circuits. 29th IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC 2021), Oct 2021, Singapore, Singapore. 10.1109/VLSI-SoC53125.2021.9606977 . hal-03662244

HAL Id: hal-03662244

<https://hal.science/hal-03662244v1>

Submitted on 9 May 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

A High-Level Design Flow for Locally Body Biased Asynchronous Circuits

Yoan Decoudu[†], Katell Morin-Allory[†] and Laurent Fesquet[†]

[†] *Univ. Grenoble Alpes, CNRS, Grenoble INP*, TIMA*

38000 Grenoble, France

firstname.name@univ-grenoble-alpes.fr

* Institute of Engineering Univ. Grenoble Alpes

Abstract—Fully Depleted Silicon on Insulator (FDSOI) technologies offer new possibilities for power management, especially with dynamic body biasing. Traditional strategies are based on a large body bias generator, which drives the IP back-gates thanks to a dedicated and often complex power management system. As asynchronous circuits use local synchronizations with handshake components, which activate only the processing parts, it is possible to take advantage of these handshake signals to implement a simple and fine-grain body biasing strategy. Instead of driving IPs with a large body bias generator, we use tiny distributed generators, locally activating small body bias regions when the circuit is processing data. These latter are based on level-shifters and implemented as standard cells. Thus, we propose a high-level design flow associating asynchronous circuits and a local body biasing strategy, which does not require complex body bias management. Indeed, the local handshake signals directly control our dedicated body bias generators. Moreover, Place and Route operations are facilitated by the use of standard cell generators. The simulations show the flow efficiency, a finer grain body biasing and a significant power reduction.

Index Terms—FDSOI, Bundled-data Circuits, High-Level Synthesis, Power Management, Fine-grain Body Biasing

I. INTRODUCTION

With the rising demand for energy-efficient systems, targeting portable applications or the Internet of Things, power consumption has gained an increased consideration for the past years. Traditional techniques to overcome this issue are voltage scaling or power gating. The first method reduces power consumption at the cost of performance degradation. The second one eliminates most of the consumption during the circuit idle period at the expense of a complex implementation and a high latency for its activation and deactivation.

Fortunately, the Fully Depleted Silicon on Insulator (FDSOI) process [1] offers new design possibilities in recent technology nodes. In addition to limiting leakage with its isolation between transistors and the substrate, it allows extensive use of body biasing. The body biasing technique, which changes the transistor threshold voltage (V_{th}), was marginal due to its limited range in standard bulk CMOS technologies. Nevertheless, FDSOI technology significantly improves the body biasing range and enables its use as a replacement or in addition to conventional techniques.

Many techniques [2], [3] have been proposed taking advantage of body biasing capabilities. The circuit is split into distinct large Body Bias Domains (BBD) as IPs have their own body bias voltage. These strategies either compensate circuit variations or address a trade-off between performance and power consumption. All these power management techniques rely on a Body Bias Generator (BBG) [4], [5]. This system can precisely bias the substrate of a large BBD at a given voltage. However, such BBGs are large and complex.

In order to minimize power consumption without loss of performance, it is possible to bias small BBDs based on data activity. Nevertheless, common BBGs and power management units are not well-suited for such a strategy as they are too large, slow and complex. Hence, we propose a simple ON-OFF biasing strategy, also called a V_{th} -Hopping scheme, relying on small BBGs. Moreover, we leverage the robustness of asynchronous circuits to quickly adapt the circuit to the changing gate delays. This idea has been first proposed in [6] and investigated in [7], but all these works are based on Quasi-Delay Insensitive (QDI) circuits [8]. Although QDI circuits are very robust, they are also very large compared to their synchronous counterparts and not supported by conventional EDA tools.

To solve these issues, we decide to apply this body biasing strategy to another class of asynchronous circuits: bundled-data circuits. These circuits are quite similar to synchronous circuits. It allows us to take advantage of the improved robustness of the asynchronous paradigm with a modest area overhead. Therefore, this paper proposes a complete design flow (from high-level description to the transistor level) to set up our power management method. This flow is entirely compliant with standard commercial EDA tools and eases the critical step as the BBD partition. In addition, since asynchronous design can be challenging for many designers, we use high-level synthesis (HLS) to quickly obtain a bundled-data circuit from a high-level description.

Section II presents the FDSOI technology and one of its main features, the body biasing. Our flow is presented in Section III. Section IV and Section V respectively detail the asynchronous architecture and the body biasing strategy used in our work. Finally, our flow is applied in Section VI.

II. FULLY DEPLETED SILICON ON INSULATOR

FDSOI [1] differs from classical bulk technology by adding a buried oxide (BOX) between the substrate and the transistor layer. It isolates drain and source pins from the substrate and, thus reduces the transistor leakage. Figure 1 shows the difference between bulk and FDSOI transistors. Moreover, the channel is now undoped, removing the random dopant fluctuation.

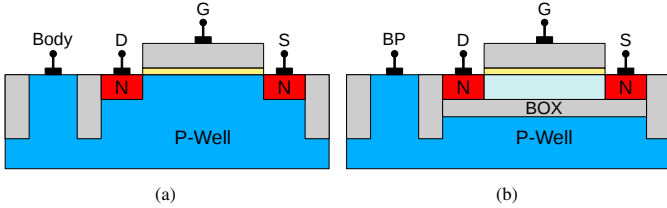


Fig. 1. NMOS transistor in classical Bulk (a) and FDSOI Technologies (b).

This structure enables new options for designers [9]. For instance, it is possible to flip the well dopant (flip-well configuration) as shown in Fig. 2. In this case, the source, the drain and the substrate have the same type of dopant. In FDSOI technology, conventional wells are usually used with Regular threshold voltage (RVT) transistors and flip-wells with Low threshold voltage (LVT) transistors. RVT transistors normally use typical initial bias: 0V for the NMOS transistor and 1V for the PMOS transistor. On the contrary, LVT transistors use 0V for both transistor types. There are two types of body biasing:

- Reverse Body Biasing (RBB) increases PMOS and decreases NMOS bias voltages from the standard body bias voltage. It results in the decrease of the leakage power while deteriorating the transistor performance.
- Forward Body Biasing (FBB) decreases PMOS and increases NMOS bias voltages from the standard body bias voltage. It results in an increase of the circuit speed at the expense of a higher leakage.

If these two kinds of body biasing can be applied to both RVT and LVT transistors, the allowed range is different. Indeed, the junction between N-well and P-well substrates limits the body biasing range. Therefore, the FBB (resp. RBB) is for the LVT (resp. RVT) transistors as the body biasing range can go up to 3.3V.

III. PROPOSED FLOW

We set up an entire digital design flow to help designers as much as possible. We leverage HLS to quickly design bundled-data circuits. With these circuits, we implement a fine-grain body biasing strategy without complex power management. This design flow facilitates the Place and Route (PnR) steps thanks to BBG standard cells.

Figure 3 presents our design flow. It is very similar to a standard design flow for a synchronous circuit, with only a few added steps (in red on Fig. 3). The HLS, the synthesis and the PnR use conventional EDA tools.

The HLS step generates several sub-systems. The desynchronization step turns each synchronous sub-system into a

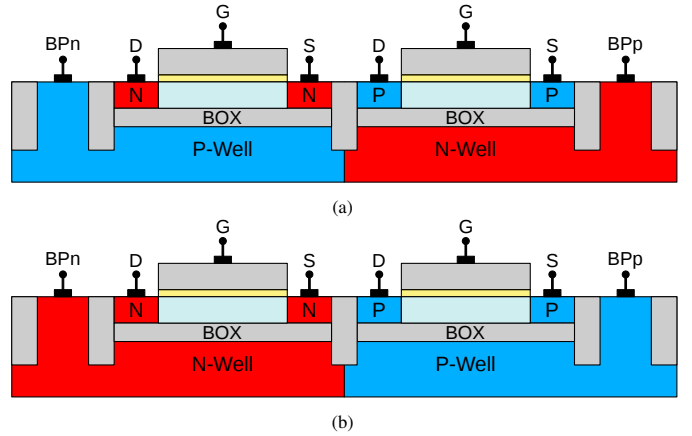


Fig. 2. Transistor with (a) Conventional well and (b) Flip-well configuration.

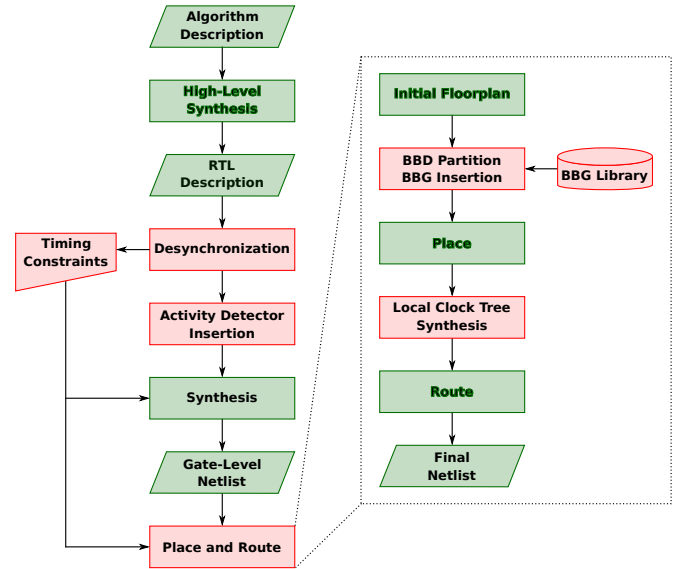


Fig. 3. Design flow steps implementing our body biasing strategy.

bundled-data one (see Section Section IV-B) in which the synchronization is done locally through control signals. The activity detectors are then added to each asynchronous sub-system in the circuit.

EDA tools rely on a Static Timing Analysis (STA) to constrain and verify circuits. Notice that the STA in bundled-data circuits can be a real issue due to the multiple local timing assumptions. Gimenez *et al.* [10] propose a methodology based on clocks and generated clocks to check each local path. We take advantage of this method to define the circuit constraints at a high level during the desynchronization step.

The PnR needs two additional sub-steps. The first one is to partition each sub-system into its own BBD and insert the BBGs in order to locally body bias the BBDs. The second one is the synthesis of local clock trees. It may seem inconsistent as there is no more clock signal in the circuit, but our desynchronized circuit has control signals that activate register banks. These signals are seen as local clock signals and they need to be buffered as any clock signal in a synchronous

circuit. This step is not different from a standard clock tree synthesis but it is locally performed.

IV. ASYNCHRONOUS ARCHITECTURE

A. Asynchronous Principle

Unlike the synchronous paradigm, there is no global synchronization signal in asynchronous circuits but many local synchronization signals respecting local timing assumptions, which ensure the correct circuit behavior. There are several classes of asynchronous implementations, depending on the timing assumptions made on the circuit model [11]. In our work, we focus on bundled-data circuits [12], that have an architecture resembling synchronous circuits. The data-path is composed of registers and combinational blocks. The synchronization block is not a clock tree like in synchronous circuits, but a control circuit composed of distributed controllers (see Fig. 4) that locally synchronize the data-path registers.

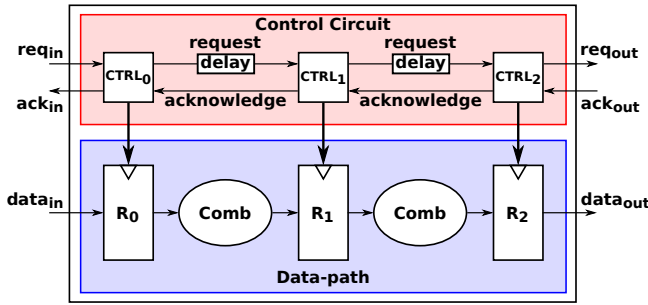


Fig. 4. Bundled-data circuit architecture.

In Figure 4, each controller $CTRL_i$ is associated to register bank R_i . The controllers communicate together thanks to handshake signals that follow a dedicated communication protocol. The emission of a new data is notified to the circuit through a handshake communication that activates the first controller. When a controller is enabled, it triggers its associated register bank (and stores a data), and it notifies the next controller. The register bank triggering must happen when the incoming data are stable to avoid any timing violation. To ensure the correct behavior of the circuit, there is a delay element inserted in the handshake communication.

Asynchronous design is challenging for most engineers due to (1) a lack of know-how and (2) support from EDA tools. By using HLS [13], we go beyond the lack of know-how to automatically generate an asynchronous circuit. The generated circuits can be handled by standard synchronous EDA tools.

The HLS tools synthesize a high-level description, usually in a C-like language, into an RTL description and have gained an increasing popularity in circuit design. Some studies [14] have been made in the field of asynchronous HLS, but these solutions rely on specific description languages and tools. Hence, we decided to use a standard synchronous HLS tool and convert its output into an asynchronous circuit.

B. Desynchronization Method

Traditional desynchronization methods [15], [16] only replace the clock signal with a control circuit. However, with

these methods we lose some benefits of the asynchronous technology: the power consumption and the speed are very similar to the original synchronous circuit with an area overhead. Furthermore, the synchronization blocks are generated for the data-path and for the control-path. It means that the data-path is controlled by two blocks: the control-path (controlled by a synchronization block) and a synchronization block. There is a redundancy in the control, we propose to merge the control-path and the synchronization block. Hence, the data-path is controlled solely thanks to an asynchronous control-path.

In practice, a synchronous circuit can be partitioned into two blocks, the control block that implements the Finite State Machine (FSM), and the data-path (see Fig. 5). In our desynchronization method [17], we desynchronize the FSM, into an Asynchronous FSM (AFSM) (see Fig. 6): each state register is replaced by a controller. The transition function and output function (Λ_{comb}^i) of the FSM remain unchanged. We only add a function that generates the local clock signals (Λ_{en}^i) of the registers in the data-path. This function depends on the enabled states.

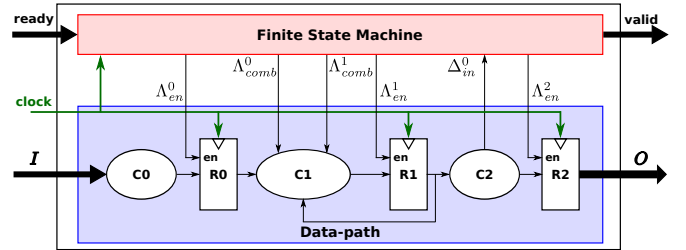


Fig. 5. Synchronous circuit architecture.

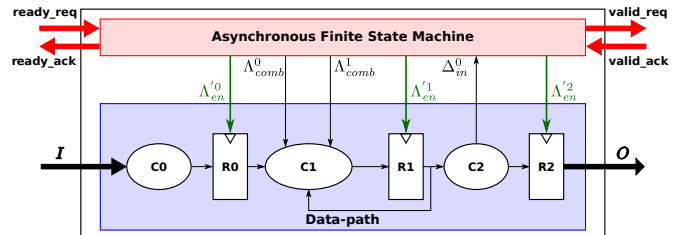


Fig. 6. Desynchronized circuit with data-path and AFSM.

This desynchronization method is particularly well-suited for circuits generated from HLS tools. Indeed, these circuits use a decomposition between a data-path and a control-path, the latter being an FSM. This method has been successfully applied to the commercial tool Catapult HLS from Siemens EDA [17].

V. BODY BIASING STRATEGY

A. V_{th} -Hopping Scheme

In order to minimize the biasing control unit complexity, we use a strategy rather simple based on V_{th} -hopping. When there is data activity in a circuit within a BBD, the transistor V_{th} is decreased to enhance the system speed. On the other hand, the transistor leakage is limited when there is no more

activity by raising the V_{th} . Thus, we use LVT transistors with an FBB scheme: the body bias voltage is applied when the system computes data and deactivated at idle time, which sets V_{th} to its baseline. With this strategy, each BBD only needs an activity detector and a BBG with an ON-OFF behavior.

Asynchronous circuits have two main advantages for a dynamic body biasing strategy. Firstly, their robustness is enhanced compared to a synchronous circuit. Therefore, it is possible to change the V_{th} parameter of the transistors within the circuit on the fly, without any timing violations. The body biasing activation and deactivation is then straightforward. Secondly, the handshake signals (the request and acknowledge signals) inherently indicate the circuit activity at a pipeline level. It gives ample freedom to choose the appropriate granularity for the BBDs, and it is only necessary to adapt the activity detector.

The combination between asynchronous circuits and a simple V_{th} -hopping body biasing scheme allows us to considerably reduce the power control complexity and the latency of the body bias activation and deactivation. It is then possible to use this dynamic body biasing strategy on smaller and faster blocks than usual.

The study in [7] shows minor gains in consumption when the BBD granularity is at a pipeline level while the area overhead strongly increases. Hence, we decide to define the BBD granularity at a system level where the start of an AFSM operation activates the body bias voltage, as shown in Fig. 7. Notice that this granularity is still much finer than the usual IP biased in conventional power management.

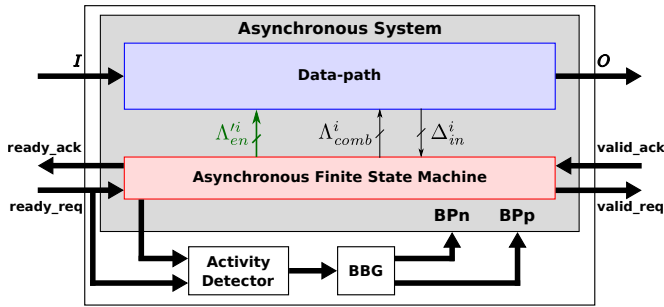


Fig. 7. Scheme of dynamic body biasing strategy.

The activity detector only corresponds to an OR logical gate between the input state request in the AFSM and the input request $ready_req$. A simple wire connecting the input state and the BBGs would be enough but taking into account $ready_req$ allows reduction of the latency of the body bias activation when new data arrive.

The BBGs, connected to the activity detector output, generate the necessary voltage for the bias of the NMOS and PMOS. As we use a simple V_{th} -hopping strategy, where the polarization is either active or inactive, the BBG can be a simple generator. The BBG is then a level-shifter as proposed in [6]. It is designed as a standard cell to ease its integration in the floorplan of digital circuits.

B. Energy Efficiency

The body biasing strategy used in our work is only effective under certain conditions. Indeed, the successive BBD activation produces an energy overhead. Hence, the energy saved during the idle period of the circuit needs to be greater than the activation of the circuit BBDs. Otherwise, there is no improvement compared to a system where the body bias is constantly active.

We analyze a linear system composed of N stages depicted in Fig. 8. The input data, represented in red, is loaded continuously in the circuit. Once data reaches a stage, its body bias voltage is activated and increments the green curve (indicating the number of biased stages). When all the data are in the pipeline, the first stages start entering idle mode. Thus, there are three phases in the scenario: a loading phase, a steady phase and an unloading phase. We define Δt_{BB} as the period where at least a stage is active and Δt_{noBB} as the idle time from which there is no more data in the circuit. We denote also Δt_{BBi} the active period of the i^{th} BBD.

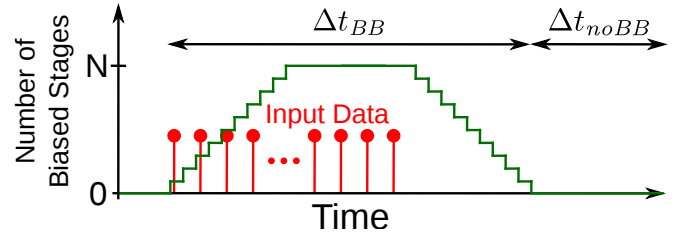


Fig. 8. Variation of the number of body biased BBDs.

The substrate of a BBD is represented as a capacitance C_b and is charged once the BBD is activated. For the i^{th} BBD in the system, the energy $E_{bias,i}$ consumed for the body bias activation or deactivation at a voltage V_{bb} is defined as:

$$E_{bias,i} = C_{b,i} V_{bb}^2 \quad (1)$$

Hence, the global energy E_{total} consumed by the circuit is:

$$E_{total} = E_{BB} + E_{noBB} + 2 \sum_{i=0}^{N-1} E_{bias,i} \quad (2)$$

During the period Δt_{noBB} , there is only circuit static power consumption without body biasing, denoted P_l^0 . The energy E_{noBB} is then given in Equation 3. Rigorously, it is necessary to include the power consumption of the activity detector and the BBGs. However, their power consumption, both dynamic and static, are negligible compared to the rest of the system.

$$E_{noBB} = \Delta t_{noBB} P_l^0 \quad (3)$$

Similarly, E_{BB} corresponds to the power consumption $P_{tot}^{V_{bb}}$ with a body bias voltage V_{bb} during Δt_{BB} . Nevertheless, the loading and unloading phases allow for a reduced power consumption. Indeed, during Δt_{BB0} for instance, only the first stage is biased, with a leakage denoted $P_{l,0}^{V_{bb}}$. Therefore, the static power consumption of the other stages is reduced to $P_{l,j}^0$ compared to a circuit completely biased. A similar analysis is done for each stage.

We denote $E_{s,pl}$ the energy saved during the loading phase. If the loading and unloading phases are balanced, both cases save the same amount of energy. It leads to the following definition:

$$E_{BB} = \Delta t_{BB} P_{tot}^{V_{bb}} - 2E_{s,pl} \quad (4)$$

where $E_{s,pl} = \sum_{i=0}^{N-1} \Delta t_{BBi} (\sum_{j=i+1}^{N-1} (P_{l,j}^{V_{bb}} - P_{l,j}^0))$

Thanks to Equations 2, 3 and 4, we precisely know the energy consumption of our method. Compared to a constantly biased circuit, we define the energy overhead E_O and the energy savings E_S of our strategy:

$$E_O = 2 \sum_{i=0}^{N-1} E_{bias,i} \quad (5)$$

$$E_S = \Delta t_{noBB} (P_l^{V_{bb}} - P_l^0) + E_{s,pl}$$

To enhance the power efficiency with the dynamic body biasing, we look for the situation where E_S is higher than E_O . Finally, we deduce from Equation 5 the minimum idle period Δt_{noBB} for an effective power reduction:

$$\Delta t_{noBB} > \frac{2V_{bb}^2 \sum_{i=0}^{N-1} (C_{b,i} - E_{s,pl})}{P_l^{V_{bb}} - P_l^0} \quad (6)$$

Equation 6 presents the lower limit of our method. This idle period, also known as the Minimum Idle Time (MIT), allows compensation for the energy overhead induced by this body biasing strategy. Therefore, the power savings strongly depend on the system operation scenarios. For this purpose, we define the activity ratio AR of the circuit:

$$AR = \frac{\Delta t_{BB}}{\Delta t_{BB} + \Delta t_{noBB}} \quad (7)$$

To compare each scenario used in the simulations, we use the activity ratio AR defined in Equation 7.

VI. TESTCASE

To ease the understanding of the results, we first apply the method on a chain of Advanced Encryption Standard (AES) designed by HLS. This system is complex enough to show the benefits of the method, but small enough to use conventional EDA tools with academic licences that limit their efficiency. The performances of the circuit have been evaluated in FDSOI 28 nm technology from STMicroelectronics.

A. System Presentation

The global circuit is a linear system of AES with a 128-bit key as shown in Fig. 9. It was hierarchically synthesized with Catapult HLS from Siemens EDA. All the synthesized AES have the same architecture.

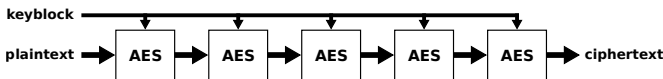


Fig. 9. Linear circuit with 5 AES.

The second step is to desynchronize the circuit. We apply the method presented in Section IV-B on each AES. Since the

BBD granularity is defined at the system level, we associate each AES to a specific BBD. We want a body bias activation in the range of 1 ns. We thus assigned 32 BBDs for each BBD.

To have some insight on the strategy benefits, we compare our power management method to a circuit with only one BBD always activated. We apply it on 3 activity ratios (0.5, 0.25 and 0.1). For each scenario, we sent the same input data burst on each circuit and we adapt the idle period Δt_{noBB} to reach the desired activity ratio. The body bias voltage applied is $V_{bb} = 1.1V$.

B. Results

The circuit is synthesized thanks to Design Compiler from Synopsys. The PnR phase uses Cadence Innovus. All the results in speed and consumption are given after the place and route. Electrical simulations at the transistor level permit to obtain an accurate power consumption evaluation. Since it is too time-consuming for digital circuits, we define the switching activity from a back-annotated simulation at the gate-level. This analysis, done with PrimeTime, is only an approximation because the energy consumed for the body bias activation is not included, but still gives a good idea of the power consumption in the circuit.

1) *Desynchronized AES*: We first compare an asynchronous AES to its original synchronous counterpart without any body biasing. Circuits are synthesized with an 800 MHz target frequency at 0.9V.

Table I
AES RESULTS AND IMPACT OF THE DESYNCHRONIZATION

	SYNC	DESYNC	Overhead
Area (μm^2)	18874	19043	+0.9%
Computation Time (ns)	38.75	39	+0.68%
Power (mW/op)	1.31	1.2	-8%

The results are reported in Table I. The last column displays the overhead between the desynchronized circuit and the synchronous one. After synthesis, the area overhead of the desynchronized AES is 0.9% which is negligible. The asynchronous AES has a latency of 39 ns, with a less than 1% difference with the synchronous circuit. The power consumption shows an improvement of 8% after the desynchronization.

We notice that our desynchronization method provides a bundled-data circuit similar to the original circuit with a slight improvement in power consumption. It has its own benefits.

2) *Global Circuit*: An AES after PnR has an area of 22440 μm^2 . The addition of the BBDs increases the area by 4.8%. Moreover, each BBD needs to be isolated from the rest of the circuit and leads to an additional increase of 12.1%. Finally, an AES with our body biasing strategy reaches an area of 26368 μm^2 . In the chained system, it leads to a global overhead of 17.5%.

A fair comparison with the other biasing strategies is not straightforward. Indeed, the conventional BBDs are larger than ours, but they are also used to bias larger systems with a precise voltage.

Regarding the speed of the system, there is no speed difference in comparison with an always biased circuit, only

negligible fluctuations. One AES with an active body bias voltage computes data in 31.9 ns, which is an improvement of about 18% compared to the AES presented before.

Table II
NORMALIZED POWER CONSUMPTION OF THE SYSTEM

AR	Normalized Power (Always)	Normalized Power (Dynamic)
0.5	1	0.58
0.25	0.89	0.36
0.1	0.83	0.23

Table II presents the power consumption of each scenario for our system and the always biased reference. We normalized the results according to the power consumption of the always biased circuit with $AR = 0.5$ activity ratio.

We observe that the two circuits consume less with the decrease of the activity ratio AR , which was expected since the dynamic power consumption is related to the circuit activity. As expected, the dynamic biasing strategy reduces the consumption compared to the always biased reference. The consumption is divided from 1.7 up to 3.6 according to the activity ratio, *i.e.* the power savings are from 42% up to 72%. The less active the circuit is, the bigger the gain. This is due to a higher leakage in the always biased circuit, even during the idle period. However, be aware that these results do not take into account the energy consumed during the body bias activation and deactivation.

The gain would even more significant if the body bias voltage was higher. We cannot show these results since the digital library models are only characterized for a few body bias values.

VII. CONCLUSION

In this work, we presented an asynchronous HLS design flow associated to a simple dynamic body biasing strategy. With a simple V_{th} -hopping scheme and the intrinsic robustness of asynchronous circuits, we achieve a very fine-grain body biasing strategy relying on data activity. According to our simulations, we demonstrate power savings from 42% up to 72% for a small data activity compared to always biased circuits.

Moreover, our design flow eases the design of fine-grain body bias asynchronous circuits, thanks to commercial HLS tools, a desynchronization script and tiny body bias generators designed as standard cells. The main design steps are as follows. First, a bundled-data circuit is obtained from a high-level description in C, which is compiled by Catapult HLS from Siemens EDA. Then, the control part resulting from the HLS is desynchronized and we extract the design constraints (mainly timing constraints). Finally, tiny BBGs, which have been embed in a standard cell format, are inserted in the floorplan during the PnR operations.

The presented simulations are only gate-level estimations. Future works will target transistor-level simulations for better accuracy. A testchip in FDSOI 28 nm from STMicroelectronics has also been designed following this design flow. It must now be tested to validate our design flow and simulation results. We also plan to investigate the power reduction

obtained with a lower supply voltage, while compensating the performance drop thanks to the body biasing. As simulations required dedicated models for each supply voltage, the best is to directly obtain these results from the testchip measurements.

ACKNOWLEDGMENT

This work has been supported by the OCEAN12 European Project (Ecsel JU, Grant Agreement N°783127).

REFERENCES

- [1] B. Pelloux-Prayer, M. Blagojevic, O. Thomas, A. Amara, A. Vladimirescu, B. Nikolic, G. Cesana, and P. Flatresse, "Planar fully depleted SOI technology: The convergence of high performance and low power towards multimedia mobile applications," in *2012 IEEE Faible Tension Faible Consommation*, Jun. 2012, pp. 1–4.
- [2] S. Kulkarni, D. Sylvester, and D. Blaauw, "A Statistical Framework for Post-Silicon Tuning through Body Bias Clustering," in *2006 IEEE/ACM International Conference on Computer Aided Design*, Nov. 2006, pp. 39–46.
- [3] H. Xu, W.-B. Jone, and R. Vemuri, "Novel Vth Hopping Techniques for Aggressive Runtime Leakage Control," in *2010 23rd International Conference on VLSI Design*, Jan. 2010, pp. 51–56.
- [4] N. Kamae, A. K. M. M. Islam, A. Tsuchiya, and H. Onodera, "A body bias generator with wide supply-range down to threshold voltage for within-die variability compensation," in *2014 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, Nov. 2014, pp. 53–56.
- [5] M. Blagojevic, M. Cochet, B. Keller, P. Flatresse, A. Vladimirescu, and B. Nikolic, "A fast, flexible, positive and negative adaptive body-bias generator in 28nm FDSOI," in *2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits)*, Jun. 2016, pp. 1–2.
- [6] J. Hamon and E. Beigne, "Automatic Leakage Control for Wide Range Performance QDI Asynchronous Circuits in FD-SOI Technology," in *2013 IEEE 19th International Symposium on Asynchronous Circuits and Systems*, May 2013, pp. 142–149.
- [7] L. Fesquet, Y. Decoudu, A. R. I. Jadue, T. F. de Paiva Leite, O. Roloff, M. Diallo, R. P. Bastos, K. Morin-Allory, and S. Engels, "A Distributed Body-Biasing Strategy for Asynchronous Circuits," in *2019 IFIP/IEEE 27th International Conference on Very Large Scale Integration (VLSI-SoC)*, Oct. 2019, pp. 27–32.
- [8] A. Martin and M. Nystrom, "Asynchronous techniques for system-on-chip design," *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1089–1120, 2006.
- [9] B. Pelloux-Prayer, A. Valentian, B. Giraud, Y. Thonnart, J.-P. Noel, P. Flatresse, and E. Beigne, "Fine grain multi-VT co-integration methodology in UTBB FD-SOI technology," in *2013 IFIP/IEEE 21st International Conference on Very Large Scale Integration (VLSI-SoC)*, Oct. 2013, pp. 168–173.
- [10] G. Gimenez, A. Cherkaoui, G. Cogniard, and L. Fesquet, "Static Timing Analysis of Asynchronous Bundled-Data Circuits," in *24th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC)*, May 2018, pp. 110–118.
- [11] J. Sparsø and S. Furber, Eds., *Principles of Asynchronous Circuit Design*. Springer US, 2001.
- [12] I. E. Sutherland, "Micropipelines," *Communications of the ACM*, vol. 32, no. 6, pp. 720–738, Jun. 1989.
- [13] P. Coussy, D. Gajski, M. Meredith, and A. Takach, "An Introduction to High-Level Synthesis," *IEEE Design & Test of Computers*, vol. 26, no. 4, pp. 8–17, Jul. 2009.
- [14] J. Sparso, "Current trends in high-level synthesis of asynchronous circuits," in *2009 16th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2009)*, Dec. 2009, pp. 347–350.
- [15] J. Cortadella, A. Kondratyev, L. Lavagno, and C. Sotiriou, "Desynchronization: Synthesis of Asynchronous Circuits From Synchronous Specifications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 10, pp. 1904–1921, Oct. 2006.
- [16] N. Srivastava and R. Manohar, "Operation-Dependent Frequency Scaling Using Desynchronization," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 4, pp. 799–809, Apr. 2019.
- [17] Y. Decoudu, J. Simatic, K. Morin-Allory, and L. Fesquet, "From High-Level Synthesis to Bundled-Data Circuits," in *Embedded Computer Systems: Architectures, Modeling, and Simulation*, 2020, vol. 12471, pp. 200–212.