SRAM Design Exploration with Integrated Application-Aware Aging Analysis

Alexandra Listl, Daniel Mueller-Gritschneder, Ulf Schlichtmann
Chair of Electronic Design Automation
Technical University of Munich, Munich, Germany
E-Mail: alexandra.listl@tum.de

Sani R. Nassif
Radyalis LLC
Austin, TX 78757
sani.nassif@gmail.com

***Abstract*—On-Chip SRAMs are an integral part of safety-critical System-on-Chips. At the same time however, they are also most susceptible to reliability threats such as Bias Temperature Instability (BTI), originating from the continuous trend of technology shrinking. BTI leads to a significant performance degradation, especially in the Sense Amplifiers (SAs) of SRAMs, where failures are fatal, since the data of a whole column is destroyed. As BTI strongly depends on the workload of an application, the aging rates of SAs in a memory array differ significantly and the incorporation of workload information into aging simulations is vital. Especially in safety-critical systems precise estimation of application specific reliability requirements to predict the memory lifetime is a key concern. In this paper we present a workload-aware aging analysis for On-Chip SRAMs that incorporates the workload of real applications executed on a processor. According to this workload, we predict the performance degradation of the SAs in the memory. We integrate this aging analysis into an aging-aware SRAM design exploration framework that generates and characterizes memories of different array granularity to select the most reliable memory architecture for the intended application. We show that this technique can mitigate SA degradation significantly depending on the environmental conditions and the application workload.**

***Index Terms*—On-Chip SRAM, BTI, Application-Specific, Reliability, SRAM Design Exploration, Aging Mitigation**

I. Introduction

The continuous trend of technology scaling has invoked major reliability challenges due to an increased susceptibility to process variations and accelerated transistor aging. Static Random Access Memories (SRAMs) constitute a significant portion of most digital system's chip area and are especially vulnerable to wear-out mechanisms since they continue to lead the migration to new technology nodes [1]. Bias Temperature Instability (BTI) has been identified as the major reliability issue because it gradually increases the threshold voltage (V_{th}) of a transistor and degrades the drain current. Especially Sense Amplifiers (SAs) are very sensitive to variations and aging and are very critical for high performance [2]. However, the failure of an SA is particularly critical since it destroys the read-out of a whole column and renders the data of every cell in that column useless.

To compensate for variability, designers usually introduce guardbands, where they add extra margins to the circuit to guarantee proper functionality throughout its entire lifetime. These margins are typically based on worst-case workload scenarios, but the workload of real applications usually does not match the worst-case and different applications in general show very different workloads [3]. In combination with an increasing impact of process variations, this leads to larger margins at the expense of more area, power and lower speed. This is especially the case for safety-critical systems where SRAM reliability has to be guaranteed for the entire lifetime of the memory.

Therefore, we propose a workload-aware aging analysis for On-Chip SRAMs that incorporates the workload of real applications running on a processor to enable a precise characterization of the memory reliability. The focus of this paper lies thereby in the characterization of the read path. We incorporate this aging analysis into an SRAM design exploration framework (SDE) that generates and characterizes memories of different array granularity (e.g. number of banks/rows/words) with detailed simulations to find the most reliable configuration in terms of aging for the intended set of applications. The presented results show that the array granularity has a significant impact on the aging behavior of the memory. The penalty in terms of area is thereby restricted to a new set of SAs for each additional bank. Hence, this tool can be a helpful means during the design phase of safety critical systems to predict the memory lifetime and identify the most reliable design for the intended set of applications.

Not much work has been done on the workload-dependent aging characterization in SRAM peripheral circuits such as write driver [4], timing control logic [5], internal memory paths [6] and sense amplifiers [7], [8]. Most of these works have considered artificial workloads, which do not represent realistic workloads from real applications. In [9] an aging-aware coding scheme is proposed to balance the aging stress of memory cells with an architectural application simulator to obtain realistic workloads. Unfortunately, this proposed scheme does not consider SA degradation. [3] has introduced a mitigation technique for SAs that balances the SA workload by modifying the SA structure. It was analyzed with real workloads from the L1 data and instruction cache.

The remainder of this paper is organized as follows. Section II discusses the considered aging mechanisms and their impact on SAs. Our approach towards an aging-aware SDE framework that considers real applications running on a processor is introduced in detail in Sections III-A and III-B respectively. Details of the aging analysis for arrays with different granularity are given in III-C. The obtained aging results for a representative set of applications and the impact of the array granularity on the memory reliability are presented in IV. The paper is concluded in Section V.

II. Aging in Sense Amplifiers

One of the most important degradation mechanisms in modern CMOS technologies is Bias Temperature Instability, which can be modeled with the charging and discharging of defects in the gate dielectric or the interface between gate dielectric and substrate [10], [11]. Gradually trapped charges increase the threshold voltage (V_{th}) of a transistor and degrade the drain current causing a temporal performance degradation

This work was supported by the German Research Foundation (DFG) as part of the Transregional Collaborative Research Center 'Invasive Computing' (SFB/TR 89).

978-3-9819263-2-3/DATE19/©2019 EDAA

of the transistor. For pMOS, the effect, called negative bias temperature instability (NBTI), appears under large negative gate voltage and has been studied extensively. For nMOS the corresponding phenomenon is called positive bias temperature instability (PBTI) and has emerged since the introduction of high-k metals. The effect of NBTI and PBTI can be described using the following model:

$$\Delta V_{th,NBTI/PBTI}(t) = A \cdot exp(\frac{-1500}{T}) V_{DD}{}^{4} t^{\frac{1}{6}} sp_{M,n/p}^{\frac{1}{6}} \quad (1)$$

where T is the current die temperature, V_{DD} the current supply voltage and t the lifetime (age) of the SRAM in seconds. $sp_{M,n/p}$ denotes the proportion of time that a transistor $M_{n/p}$ experiences BTI stress. A is a parameter to fit the model to the results in [12].

A common design for SAs is the Latch-Type SA as shown in Fig. 1. Due to the symmetry of the SA design, asymmetric wear-out increases the effect of aging [13]. Assuming e.g. that the value '1' is constantly read from BL (corresponding to a read signal probability (SP) of '1') transistor M_{n2} in Fig. 1 experiences PBTI stress and transistor M_{p1} NBTI stress. Consequently, their threshold voltage is increased. This leads to a reduced positive feedback and slows down the read-out process. Finally, the inverter in the SA may not reach a stable operating point any more. Hence, the sense delay gradually increases until either access time is violated or the readout fails. The sense delay (SD) is thereby defined as the time between the activation of the wordline and the point when the weak differential input voltage has been converted to a full swing output signal. The best-case aging is achieved, if the probability of BL being '1' is 0.5 since in that case aging appears equally on both sides of the SA and the symmetry of the design is retained. On the other hand, if no read-out occurs, e.g. because a certain bank is idle most of the time, there is no degradation for those SAs at all. Hence, the signal probabilities of SAs can differ significantly for different applications. This makes an accurate analysis of the actual workload mandatory to estimate the correct amount of aging.

III. Aging-Aware SRAM Design Exploration Framework

In this section we present the SDE framework that incorporates an aging analysis for the SA degradation based on workload from real applications. We illustrate the methodology by examining the impact of workload on the SD of the SA, but it is possible to trivially modify the flow to handle other figures-of-merit like SA offset voltage. The design exploration and analysis flow consists of two main parts (see Fig. 3): *1) simulation of a representative set of applications with an Instruction Set Simulator (ISS)* to generate stress maps containing the SP of each SA during workload execution, and *2) aging-aware netlist generation and aging simulation of arrays with different granularity* according to the stress that has been identified in the first step.

Fig. 1. Voltage Latch Sense Amplifier

A. Generation of Stress Maps using an ISS

The effect of BTI on an SA depends on the probability that a cell of a certain column is read and on the value it holds. Both probabilities depend entirely on the application that is running on the system. In order to understand the stress that a certain SA experiences it is necessary to know which addresses and corresponding values are read from and written to the array. Therefore, memory accesses from the On-Chip SRAM of a processor need to be traced while executing a representative set of applications. A memory trace consists of the time stamp when a certain address is read (or written), the address that has been accessed as well as the value. In this paper the traces are obtained with the Instruction Set Simulator from [14], but any other high-level simulation, that is able to track memory accesses can be deployed as well.

Fig. 2 shows an overview of the underlying memory model which consists of 6T SRAM cells arranged in several sub-blocks. The SRAM core array consists of K banks with a size of IxJ, where I is the number of rows and J is the number of words. The read-out circuitry comprises bit-line pre-charge logic as well as latch-type voltage SAs as shown in Fig. 1 to read-out the memory cells. From the memory traces we firstly generate a stress map $SP_{cell,v,k}$ with $v \in \{0,1\}$ representing the value that is read from the cell and k denoting the bank number.

$$SP_{cell,v,k} = \begin{pmatrix} sp_{cell,v,0,0} & \cdots & sp_{cell,v,0,(J*32)-1} \\ \vdots & \ddots & \vdots \\ sp_{cell,v,I-1,0} & \cdots & sp_{cell,v,I,(J*32)-1} \end{pmatrix}$$

In the above matrix, J is the number of words, $(J*32)-1$ the number of bits for a word-size of 32 and I is the number of rows. The matrix entry $sp_{cell,v,i,j}$ represents the signal probability of bit j in row i of a certain bank k for reading either '0' ($v=0$) or '1' ($v=1$) from that cell respectively. To consider asymmetric aging we differ between '0' and '1' values, since the duty factors of some transistors in the SA change significantly depending on which value is read. To determine the SP that a certain SA l of bank k reads a value $v \in \{0,1\}$ a stress vector $sp_{SA,v,k}$ of size $1x32$ is created. Since in our design there are only 32 SAs per bank for a word-wise read-out, each word in the array is multiplexed to these SAs. Hence, the column multiplexing factor needs to be accounted for to obtain the correct amount of stress. An SA l experiences stress whenever one of the columns $(l+n)*32$ with $n=0,..,J-1$ is read. The SP that an SA l of bank k reads a value v is therefore defined as the sum of the read-out stress for all columns that this SA is multiplexed to: $sp_{SA,v,k}(l) = \sum_{n=0}^{J-1} SP_{cell,v,k}((l+n)*32)$. The SA stress vectors are finally arranged in an SA stress matrix $SP_{SA,v}$ with K rows and 32 columns where K is the number of banks. The overall stress map of a given granularity for reading '0' or '1' is then given as

$$SP_{SA,v} = \begin{pmatrix} sp_{SA,0,0} & \cdots & sp_{SA,0,31} \\ \vdots & \ddots & \vdots \\ sp_{SA,K-1,0} & \cdots & sp_{SA,K-1,31} \end{pmatrix}$$

B. Aged Netlist Generator and Aging Simulation

The obtained stress maps $SP_{SA,v}$ are exploited by the aging-aware netlist generator. The framework consists of three stages:

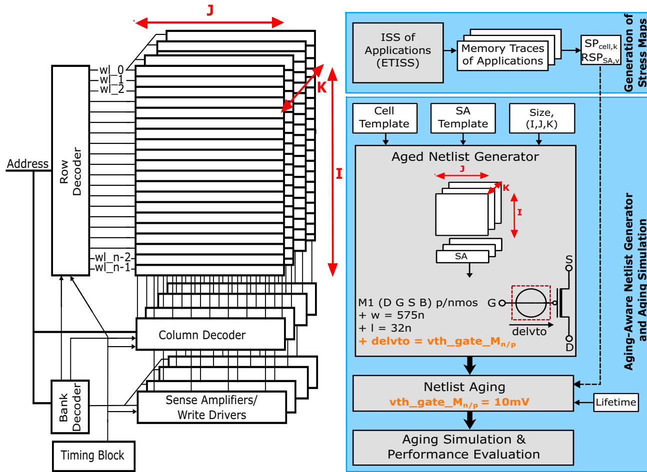

Fig. 2. Underlying Memory Model

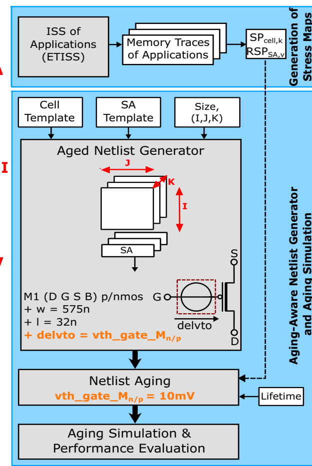

Fig. 3. Aging-aware SRAM Design Exploration Framework

1) Aged netlist generation of the SRAM for each granularity, *2) setting of aging parameters* in the generated netlists and *3) aging simulation and performance evaluation*. The inputs to the aged netlist generator are a template netlist of a 6T SRAM cell, a template netlist of the SA shown in Fig. 1, the lifetime in years, the desired memory size in kB as well as the possible granularity configurations (number of rows, columns, banks). The netlist generator first creates the netlist of each memory array according to the model in Fig. 2 and adds a constant voltage source with a variable drift parameter at the gate terminal of each transistor to incorporate the threshold voltage shift. In the second stage the threshold voltage shifts for NBTI and PBTI are calculated from the SPs, the supply voltage as well as the desired memory lifetime for each transistor. Since not all transistors are under stress for the complete read-out, it is important to consider the duty factor representing the proportion of time a transistor in the circuit actually experiences BTI stress during the read-out. The duty factor of each transistor is determined through simulation once up front. Hence, the specific V_{th}-shift that each transistor experiences is calculated with a scaled SP $sp_{M,n/p} = sp_{SA} \cdot \alpha_{M,n/p}$ with $\alpha_{M,n/p}$ being the duty factor of transistor $M_{n/p}$. Finally, the netlist is "aged" by setting the parameters of each voltage source according to the threshold voltage shift as predicted by equation (1). The third stage sets up the simulation of the array by automatically generating all necessary input waveforms and measurements for the SD depending on the array configuration and finally starts the Spectre simulations. The simulation measurements are then evaluated by determining the SD degradation for each SA and logging failed read-outs.

C. Aging Behavior Characterization of Memory Arrays with different Granularity

Especially for the reliability of safety-critical systems it can be beneficial to analyze the aging behavior of different memory architectures for the intended application before deciding on a specific design. Fig. 2 shows that the granularity of the SRAM core array has three degrees of freedom, the number of rows I, columns J and banks K. In the following, it is explored how the aging behavior of the SRAM core array changes for adjusting the array granularity using these three discrete parameters I, J, K. To explain the impact, that each of the parameters has on the memory reliability the first observation is, that although different applications generally show very different workloads, certain address ranges are accessed more frequently than others. These ranges correspond to the stack and the heap. Since stack and heap are fixed by the architecture and not the application, assuming that there is no OS running, this behavior can be observed for most applications and is independent from the processor architecture. Stack and heap are often located near the beginning and at the end of the address range. These heavily accessed addresses usually decode to only a few banks which experience exacerbated aging. Since the decoding of the addresses is dependent on the array granularity, an increase in the number of banks can spread the workload across more banks and hence mitigate aging. The penalty for a larger number of banks however is, that the area of the memory is increased since each bank needs its own 32 SAs for a word-size of 32. Furthermore, decreasing the number of rows decreases the bitline capacity, which can help to mitigate aging as well, since the bitline swing is higher for fewer rows. Adjusting the number of words however, should not have any impact on the SA aging behavior and is only necessary to retain the correct memory size. In the following section we show that investigating the aging behavior of memories with different granularity can help to find an optimal design in terms of lifetime reliability.

IV. Experimental Results

In this section we present our experimental results obtained from the aging-aware SDE for a 64kByte On-Chip memory of an OpenRisc 100 processor (OR1k) [15] in 32nm technology. We ran a representative set of applications on the processor to obtain average SPs for each SA. As our use-case we chose workloads from 15 applications including sorting algorithms (ISORT, HEAP), image processing and compression (EDGE, JDCT), encryption algorithms (AES), digital filter algorithms (IIR, FIR) and several arithmetic computations. The respective read memory accesses of this use-case can be seen in Fig. 4, which shows the number of reads for each address (in decimal). Here, the observation that most read traffic is happening upon heap and stack near the beginning and at the end of the address range is confirmed.

For the aging-aware SDE we chose the memory granularity configurations as shown in Table I. The table contains the resulting number of banks for a given combination of rows and columns for a memory size of 64kByte. Figures 6 and 7 show the stress maps $SP_{SA,v}$ for both read values and a granularity of 16x64x16 (16 banks, 64 rows and 16 words). The workload of the frequently accessed addresses is distributed to 5 out of 16 banks, reducing the SP to a maximum of 11.6%. Figures 8 and 9 show the stress maps $SP_{SA,v}$ for reading "0" and "1" respectively for a granularity of 2x256x32 (2 banks, 256 rows and 32 words). The frequently accessed addresses for stack and heap decode to 2 out of 2 banks and the SP reaches up to 20.0%. It is observed that the value '0' is read significantly more often then '1', which clearly results in asymmetric aging. To predict the SD degradation of the chosen granularity configurations we applied a use-case of 3 years of aging at $75°C$ to calculate the threshold voltage shift due to BTI aging. The simulations to measure the SD were conducted at $27°C$ room temperature and reading out the value '0' at the top cell

TABLE I
MEMORY CONFIGURATIONS

		Number of words			
		4	8	16	32
Number of rows	64	64	32	16	8
	128	32	16	8	4
	256	16	8	4	2
	512	8	4	2	1

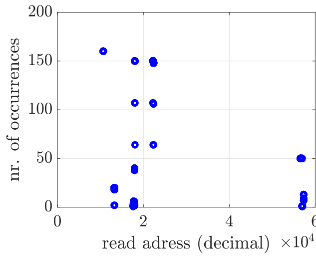

Fig. 4. Read addresses of representative set of applications

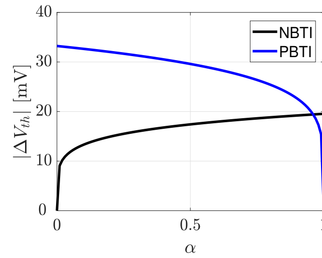

Fig. 5. Threshold voltage shift in dependance of duty factor

TABLE II
MAXIMUM AND AVERAGE SENSE DELAY DEGRADATION

		Number of Banks						
		64	32	16	8	4	2	1
Number of Rows	64	4.5/0.3	4.5/0.6	4.5/1.0	5.0/1.6	-	-	-
	128	-	5.1/0.6	5.1/1.2	5.7/1.7	5.7/3.5	-	-
	256	-	-	22.0/5.4	25.9/7.3	25.9/14.6	26.8/21.4	-
	512	-	-	-	Failed	Failed	Failed	Failed

in the array. This represents the worst-case read-out condition since the top cell drives the parasitics of the whole bitline. Table II shows the resulting maximum and average SD degradation of all SAs in percent (maximum/average) for the considered configurations from Table I. The array configurations, which have 512 rows already show some failed read-outs on some of the SAs and are therefore not considered any further. As expected, the simulation results show that the SD degradation decreases with a higher number of banks because the workload is spread across more banks. It is observed, that more rows lead to a higher degradation. This makes sense, because the longer the bitline is, the more capacitance is attached to it. Notably, this effect even has a much stronger impact on the SA reliability than the number of banks for the assumed workload. This phenomenon can be explained with the function of the threshold voltage shift over the signal probability in Fig. 5. The slope of the function changes strongly if the signal probability is close to 0 and flattens quickly for larger values. Adjusting the memory granularity to contain more banks indeed spreads the appearing stress over additional banks and reduces the stress probabilities of all banks as was shown in Figures 6, 7 and 8, 9. However, the stress probability does not decrease enough to

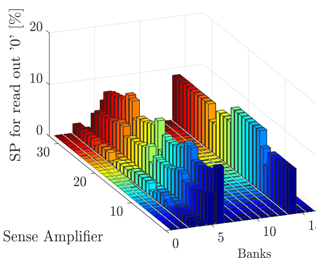

Fig. 6. SP for reading '0', 16 banks, 64 rows and 16 words

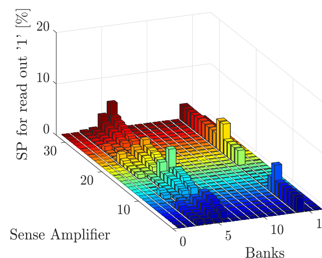

Fig. 7. SP for reading '1', 16 banks, 64 rows and 16 words

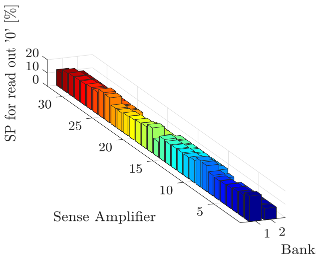

Fig. 8. SP for reading '0', 2 banks, 256 rows and 32 words

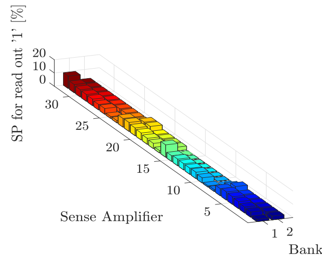

Fig. 9. SP for reading '1', 2 banks, 256 rows and 32 words

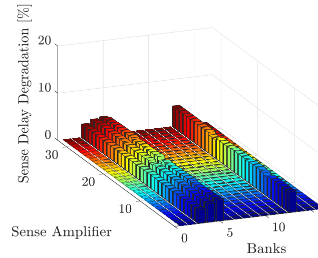

Fig. 10. Sense delay degradation, 16 banks, 64 rows and 16 words

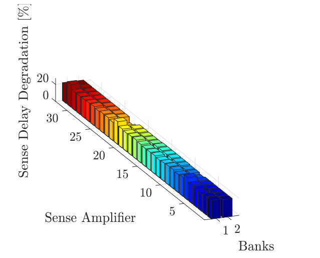

Fig. 11. Sense delay degradation, 2 banks, 256 rows and 32 words

significantly change the threshold voltage shift. Nonetheless, increasing the number of banks can prevent specific SAs from failing completely, thereby increasing the lifetime of the memory. Since aging is dependent on the workload and hence the application, an individual study is required for a different set of workloads. From Table II we chose 16x64x16 as the best-case granularity for this use-case because the maximum degradation stays the same compared to the configurations with more banks while invoking less area penalty. We compare it to 2x256x32 because it shows the worst-case degradation. Figures 10 and 11 finally show the SD degradation of the two configurations. While 2x256x32 already reaches a degradation up to 26.8%, 16x64x16 shows a maximum degradation of only 4.5%. For the chosen representative set of applications this means that choosing a granularity of 16x64x16 effectively mitigates aging and improves the lifetime of the memory array significantly. The area penalty however is an additional 14x32 SAs is compared to the worst-case granularity.

V. CONCLUSION

This paper investigates the impact of aging on the SD of SAs by employing a workload-aware aging analysis for On-Chip SRAMs that incorporates the workload of real applications. We integrate this aging analysis into an aging-aware SRAM Design Exploration framework that is able to create and analyze aged memories of different granularity. We show that SA aging can be efficiently mitigated by choosing the most favorable array granularity in terms of aging. Therefore, the proposed framework can give useful insights into the aging behavior of SRAMs and even improve the lifetime of the memory.

REFERENCES

[1] J. C. Cha and S. K. Gupta, "Characterization of granularity and redundancy for SRAMs for optimal yield-per-area," in ICCD, 2008.
[2] D. Kraak et al., "Mitigation of sense amplifier degradation using input switching," in DATE, 2017.
[3] D. Kraak et al., "Impact and Mitigation of Sense Amplifier Aging Degradation Using Realistic Workloads," in VLSI Systems, 2017.
[4] I. Agbo et al., "BTI analysis of SRAM write driver," in IDT, 2015.
[5] H. I. Yang et al., "Impacts of NBTI/PBTI on timing control circuits and degradation tolerant design in nanoscale CMOS SRAM," in CAS I, 2011.
[6] I. Agbo et al., "Read path degradation analysis in SRAM," in ETS, 2016.
[7] I. Agbo et al., "Integral impact of BTI and voltage temperature variation on SRAM sense amplifier," in VTS, 2015.
[8] R. Menchaca et al., "Impact of transistor aging effects on sense amplifier reliability in nano-scale CMOS," in ISQED, 2012.
[9] M. S. Golanbari et al., "Aging-aware coding scheme for memory arrays," in ETS, 2017.
[10] T. Grasser et al., "A unified perspective of RTN and BTI," in IRPS, 2014.
[11] V. Kleeberger et al., "A compact model for NBTI degradation and recovery under use-profile variations and its application to aging analysis of digital integrated circuits," in MR 54, 2014.
[12] S. Zafar et al., "A Comparative Study of NBTI and PBTI in SiO2/HfO2 Stacks with FUSI, TiN, Re Gates," in VLSI, 2006.
[13] E. Gunadi et al., "Combating aging with the colt duty cycle equalizer," in MICRO, 2010.
[14] D. Mueller-Gritschneder et al., "ETISS-ML: A multi-level ISS with RTL-level fault injection support for the evaluation of cross-layer resiliency techniques," in DATE, 2018.
[15] https://openrisc.io/or1k.html, "Open Risc".