54. IWK

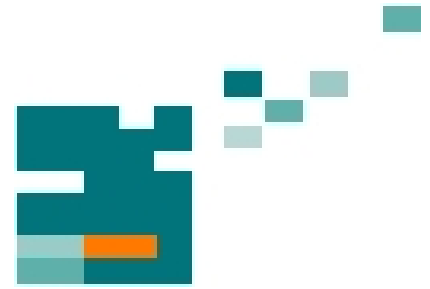## Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

Information Technology and Electrical Engineering - Devices and Systems, Materials and Technologies for the Future

Faculty of Electrical Engineering and Information Technology

Startseite / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=14089

TECHNISCHE UNIVERSITÄT ILMENAU

Development of VHDL-models for transient simulation of complex asynchronous RSFQ circuits

Tanya Stoyadinova, Ilyian Buzov, Valeri Mladenov
Department of Theoretical Electrical Engineering, Technical University of Sofia
E-Mail: tstoiadinova@abv.bg

Krasimira Filipova
Department of Automatics and Systems Engineering, Technical University of Sofia
E-Mail: KFilipova@abv.bg

Ivan Panayotov
Department of Microelectronics, Technical University of Sofia, Bulgaria
E-Mail: panayotovmail@abv.bg

Thomas Ortlepp
RSFQ design group, Institute for Information Technology, Ilmenau University of Technology
E-Mail: thomas.ortlepp@tu-ilmenau.de

ABSTRACT

This article describes the translation of a digital circuit behavior of digital superconductive circuits into high-level logical descriptions. The work is based on Rapid Single Flux Quantum (RSFQ) electronics. The basic electronic circuits, such as AND, XOR and a signal splitter are evaluated in order to extract the logical behavior as well as timing issues by analog circuit simulations. The timing condition of the adder circuit for a special input configuration in terms of their arrival time as well as their bit pattern is extracted. We use dual rail logic to analyze an RSFQ 4-bit ripple carry adder using analog and VHDL-based simulations and comparing their results in terms of total delay and simulation time.

Index Terms – Superconductive electronics RSFQ, dual-rail, asynchronous, VHDL, delay.

1. INTRODUCTION

The magnetic flux inside a superconducting ring is quantized in integer multiples of the single flux quantum $\Phi_0 = h/2e$, where h is Planck's constant and e is the elementary charge. The Rapid Single Flux Quantum (RSFQ) electronics [1] is a naturally digital electronics based on the transfer, processing and storage of these single flux quanta. Since this logic family is based on transient pulses instead of voltage levels, a dedicated processing scheme for logic circuits is required. The implementation of the synchronous signal processing scheme requires latching stages for temporary data storage in all logic cells. A logical "1" is realized by the presence of an RSFQ pulse whereat no pulse is send in case of a logical "0". The implementation of an ordinary logic gate requires the latching of each input pulse and a clock pulse, which starts the processing. As one of the main steps in developing a full-featured ultra-high speed digital signal processor (DSP) based on the technique, the gate level design and simulation of different arithmetic logic units need to be performed. In classical semiconductor design is speed usually related to a higher circuit complexity. In terms of speed, the Kogge-Stone adder [2] has been proven to be the best choice of existing hardware algorithms for a CMOS multi-bit adder implementation. From a first analysis, this looks also very promising for an implementation in RSFQ technology. The main drawback of this approach is the large complexity and the required hybrid wave-

pipelined implementation [2]. A logic level design approach of a 64 bit wave-pipelined adder yields an overall complexity of about 98,000 Josephson junctions. The strong limitation in complexity for state-of-the-art RSFQ circuits makes other approaches, like the bit-serial adder much more promising. In this case, the hardware requirement is only a single bit full adder in conjunction with a direct (one pipeline stage) feedback of the carry bit [3]. A recent implementation of such a four bit serial adder [4] in the most advanced RSFQ technology could demonstrate experimentally a maximum clock frequency of 88 GHz. This translates for a 16 adder into almost 3 GOPS (10^9 operations per Second). The bit-serial adder consist of two pipeline stages including one AND and one XOR gate each. Including the carry feedback, signal mergers and splitters, this circuit consists of only 137 Josephson junctions.

In contrast, the dual rail coding [5] of RSFQ signals is based on two signal lines where a pulse on the one represents a logical "1" and a pulse on the other one represents a logical "0". An asynchronous gate will latch input signals until all input information have arrived. The processing starts with the arrival of the last input information. Figure 1 shows the architecture of the single bit adder based on the classical textbook implementation with two half adders and an OR-gate. It is not optimized with respect to the capabilities of RSFQ circuits for reduced complexity or for speed. The total 4-bit ripple-carry implementation includes 544 Josephson junctions. Fig. 2 shows a spice-based simulation result of complete dual-rail 4 bit ripple-carry adder circuit. The cell parameters are based on circuit parameters given in [6]. The overall delay for the shown input condition is about 260 ps, which depends on the bit pattern as well as on the arrival time and order of the input bits.

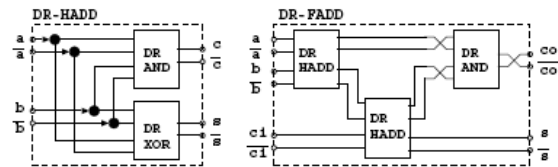The timing constrains in complex RSFQ circuits are very challenging for the circuit

Figure 1: Architecture of the dual rail half- and full adder.

design and the asynchronous approach could relax this problem, because the clock distribution network is removed.
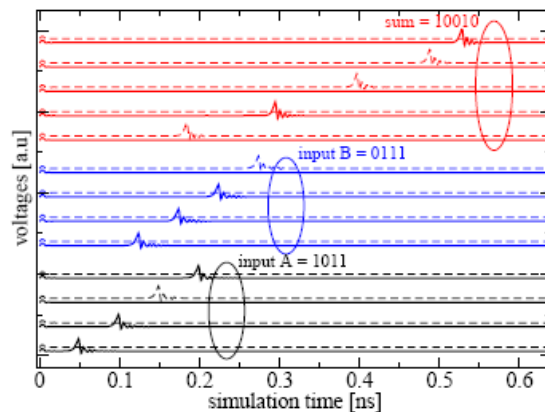

Figure 2: Transient simulation result of an asynchronous dual-rail 4 bit adder. Two lines next to each other represent one logical input, which consists of two electrical lines. The solid represents the logical 1 and the dashed line a zero.

Within this paper, use extracted timing parameters for selected asynchronous RSFQ gates. The transient logical behavior of our cell models will be translated into entities with a logical description on input and output conditions formulated in hardware description language (HDL) models. Their functionality is demonstrated by implementing an asynchronous 4-bit serial and parallel adder circuit based on dual rail signal coding. The results can be used for a practical assessment of an asynchronous dual-rail circuit realization against a classical implementation of a synchronous circuit in terms of complexity and speed.

2. VHDL DESCRIPTIONS OF BASIC RSFQ ELEMENTS

The basic elements that are used to build the half-adder and the full-adder circuits shown on Fig.1 are dual rail AND and XOR gate. The specific of the RSFQ technology

requires also signal splitters in points where the signal line branches. These basic elements will be described behaviorally in separate VHDL models. This will allow us to analyze hierarchically more complex circuits using structural descriptions such as full-adder and four-bit adder.

VHDL is one of the most popular hardware description languages (HDLs), along with Verilog-HDL, which is used in contemporary VLSI digital design [7]. It allows building behavioral, RTL (Register Transfer Level) and structural descriptions of digital systems, run simulations to verify functionality and automatically synthesize the hardware implementation corresponding to the required functionality [8]. In contrast to the classical analog circuit simulation, this allows a much faster and automated circuit analysis as well as timing investigation. The development of HDLs leads to languages capable of describing easily entire systems, such as SystemC and SystemVerilog and provides much sophisticated verification techniques. Also VHDL and Verilog are extended in order to support simulations of analog and mixed-signals designs.

The fact that RSFQ logic in its essence has the behavior of digital logic with 0 and 1 signals makes VHDL very useful to a build description of such elements in order to speed up simulations and ease the evaluation of more complex circuits. Even though this logic has its peculiarities which must be reflected correctly in language constructs. In this work, we are going to analyze dual-rail coding based on two separate lines, one for representing the logical 0 and a second for the logical 1. Signal values are represented by pulses instead of levels and these pulses may arrive in different order and in different timeslots. The value of already received signals must be memorized until the last input signal for the current computation is received. All these requirements are translated into VHDL descriptions in a specific manner.

The VHDL descriptions are behavior and built in a way that passes on the incoming pulses to the output. Internal signals are used to store the information of already processed data until the last required input pulse is received. The specific delays for every possible order of signal arrival events are provided using generics. These data needs to be extracted from various analog circuit simulations. This allows defining different delays for every element in the hierarchical designs. They are implemented using the transport delay mechanism, provided by the language syntax.

One of the simple cells is the signal splitter. It has only one delay and after the arrival of a data pulse, it directly passes this incoming trigger event to the two outputs. It is a data-driven single channel cell (not clocked and not dual rail). Such elements are used for every of the four input lines of the half adder (represented by the black dots). On fig.3 the description of the signal splitter is shown. The cell has no internal state an only one input which leads to only one timing parameter for the total delay between input and output (11 ps) for the present example.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity rsfq_splitter is
    Generic( t_out : time := 11 ps);
    Port ( signal_in : in  STD_LOGIC;
        out_a : out  STD_LOGIC;
        out_b : out  STD_LOGIC);
end rsfq_splitter;
architecture Behavioral of rsfq_splitter is
begin
process
begin
wait on signal_in;
out_a <= transport signal_in after t_out;
out_b <= transport signal_in after t_out;
end process;
end Behavioral;
```

Figure 3: VHDL description of the signal splitter.

The VHDL description of input ports for the dual-rail AND element is given in Fig.4. Here the two inputs and the output have separate lines for logical '1' and '0'. The output is a function of the input values. There are several different cases for the arrival of input signal, which are reflected in the code. A small part of the code is represented on fig.5.

```
Port ( a_1 : in STD_LOGIC;
       a_0 : in STD_LOGIC;
       b_1 : in STD_LOGIC;
       b_0 : in STD_LOGIC;
       y_1 : out STD_LOGIC;
       y_0 : out STD_LOGIC);
```

Figure 4: Ports of the dual-rail AND VHDL description.

```
if (a_0 ='1' and b_0 = '1') then
    y_0  <=  transport (a_0  and  b_0)  after
t_out_ab_e_00;      -- output 0
```

Figure 5: Assignment of the output value using transport delay, code fragment.

Using this design principles dual-rail AND, XOR and signal splitter descriptions are built. They are used to implement hierarchically larger circuits such as one-bit half-adder, full-adder and four-bit ripple-carry adder circuit.

The block diagram of the half-adder is shown on Fig. 6. It is identical to the dual-rail half-adder shown on Fig.1.
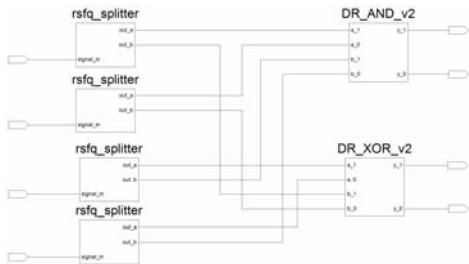

Figure 6: Half-adder structural design using behavior descriptions of DR-AND, DR-XOR and splitter.

Using two half-adders and an AND element the full-adder is built. Then four full adders are connected to form a four-bit ripple carry adder (RCA). It is shown on fig.7
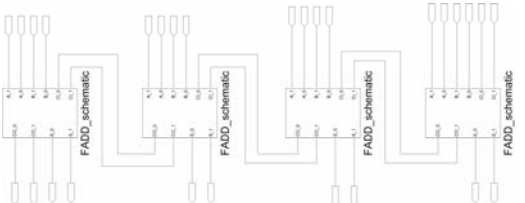

Figure 7: Four-bit RCA.

The same bit pattern, as shown on Fig.2, is applied in the VHDL simulation. The results are represented on Fig.8.
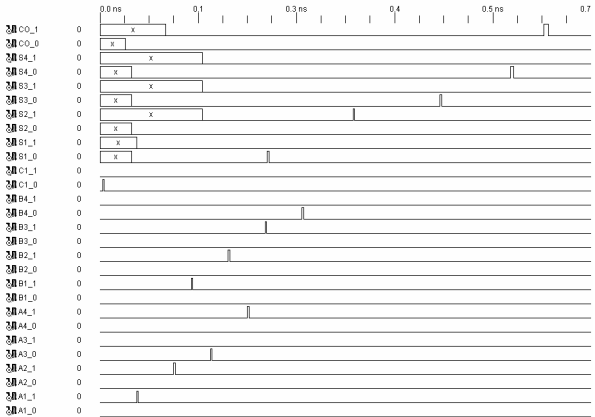

Figure 8: Simulation results for asynchronous dual-rail 4 bit adder based on a full VHDL description. Time is in 'ns'.

The input values are as follows: 1011 on input 'A'; 0111 on input 'B'; 0 on carry-in input 'C1'.

Here the digital simulator is used and the signals are represented with their levels 1 and 0. As the minimum simulator resolution time is of 'ps', the precision of the extracted delays is lowered. The overall delay for the shown input condition, obtained from VHDL simulation is about 300 ps.

The results on fig. 9 show the case when input A = 1011 and input B = 0111. Pulses are driven in the same time slots, but LSBs are received first. The delay for this case is about 480 ps. Computation time is about 4 seconds.
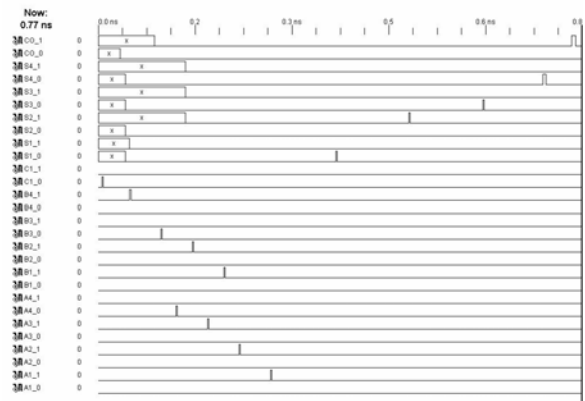

Figure 9: Simulation results for asynchronous dual-rail 4 bit adder VHDL description with reversed input signals. Time is in 'ns'.

The received simulation results, confirm the correct functionality of the VHDL description.

The overall delays for some other interesting cases are given in table 1. In all cases LSBs come first.

Table 1: Delays for different input signals.

Value on A	Value on B	Delay, ps
0111	1011	328
1010	0101	305
0000	0000	310
1111	0000	305
1111	1111	390

3. CONCLUSIONS

The represented research work confirms the ability of a modern digital hardware description language such as VHDL to be used successfully to describe basic logic units of the dual-rail RSFQ logic. Building a library of such elements based on the parameters extracted from analog simulations and real measurements will ease the verification and timing analysis, which provides a faster design process. For the present example, the analog circuit simulations require about 12 seconds and the VHDL extraction only 3 seconds. This strong improvement in simulation time is essential, because the complete timing analysis of the 4 bit adder requires the evaluation of all possible input patterns. Such a result would allow the detailed investigation of the worst and best case which is the most important information for any improvement of the circuit architecture.

In a future development step, we are going to implement parameter dependent models for an automated evaluation of the operation range versus parameter variations. We plan furthermore to extract the timing conditions of the adder circuit for all possible input configurations in terms of their arrival time as well as their bit pattern.

5. REFERENCES

[1] K. K. Likharev and V. K. Semenov, "RSFQ logic/memory family: A new Josephson-junction digital technology for sub-terahertz-clock-frequency digital systems", in IEEE Trans. on Appl. Superconductivity, vol. 1, no. 1, pp. 3-28, 1991

[2] P. Kogge and H.S. Stone, "A Parallel Algorithm for the Efficient Solution of a General Case of Recurrence Equations", in IEEE Trans. Computers, vol. C-22, no.8, pp. 786-793, 1973

[3] P. Bunyk and P. Litskevitch, "Case Study in RSFQ Design: Fast Pipelined Parallel Adder", in IEEE Trans. on Appl. Superconductivity, vol. 9, no. 2, pp. 3714-3720, 1999

[4] T. Kainuma et al., "Design and High-Speed Tests of Component Circuits of an SFQ Half-Percision Floting Point Adder using 10 kA/cm² Nb Process" presented on 12. International Superconductive Electronics Conference (ISEC), SP-P43, Yokohama, Japan, 16-19 June 2009

[5] Z. J. Deng et al., "Data-Driven Self-Timed RSFQ High-Speed Test System" in IEEE Trans. on Appl. Superconductivity, vol. 7, no. 2, pp. 3830-3833, 1997

[6] B. Dimov, Th. Ortlepp, V. Mladenov, S. Terzieva, F. H. Uhlmann, "The asynchronous rapid single-flux quantum electronics - a promising alternative for the development of high-performance digital circuits". Advances in Radio Science, Vol. 6, pp. 65-173, 2008

[7] D. Perry, VHDL, 3rd Edition, McGraw-Hill 1998

[8] K. Nancheva-Philipova, M.Hristov, I. Panayotov, V. Hristov, Use of (v)HDL for electronic devices synthesis – reference book, KING 2001, Sofia 2004