

# Test of a majority-based reversible (quantum) 4 bits ripple-carry adder in adiabatic calculation

Stéphane BURIGNAT and Alexis DE VOS

Universiteit Gent  
Vakgroep elektronika en informatiesystemen  
Sint Pietersnieuwstraat 41  
B - 9000 Gent  
Belgium

Email: [research@burignat.eu](mailto:research@burignat.eu); [alex@elis.ugent.be](mailto:alex@elis.ugent.be)

**Abstract**—Quantum computing and circuits are of growing interest and so is reversible logic as it plays an important role in the synthesis of circuits dedicated to quantum computation. Moreover, reversible logic provides an alternative to classical computing machines, that may overcome many of the power dissipation problems in the near future. As a proof of concept we designed and tested a reversible 4 bits ripple-carry adder based on a do-spy-undo structure. This paper presents some performances obtained with such a chip processed in standard  $0.35\ \mu\text{m}$  CMOS technology and used in real reversible calculation (in this study, computations are performed in both directions such that addition and subtraction are made reversibly with the same chip). We also discuss the superiority of using adiabatic signals over classical rectangular pulses when using dual-line pass-transistor logic gates. Adiabatic signals allow the signal energy stored on the various capacitances of the circuit to be redistributed rather than being dissipated as heat. Finally, we show that adiabatic signals allow to avoid calculation errors introduced by the use of conventional rectangular pulses and allow to drastically reduce the number of pulse resynchronization in large circuits.

**Index Terms**—reversible computation, design, implementation, pass-transistor logic, ripple-carry adder, Spectre simulation, quantum computation, adiabatic signal, test and measurement.

## I. INTRODUCTION

Reversible computing is useful both in lossless classical computing [1] and in quantum computing [2]. Reversible circuits also present less power consumption against classical ones [3]–[5].

The physical implementation of the adder we are presenting is reversible dual-line complementary pass-transistor CMOS logic [6] and does not make use of buffer of any sort nor level restorer. This adder has been extended and embedded in larger components such as a multiplier [7] and, more recently, in a H264/AVC encoder [8].

We present for the very first time to our knowledge, extensive electrical tests performed on a reversible CMOS chip computing in both directions: forward (adder) and backward (subtractor) and compare the efficiency of both classical rectangular pulses and adiabatic signals.

## II. SHORT HISTORY AND DESCRIPTION OF THE REVERSIBLE ADDER CHIP

### A. Short history

In 2005, Cuccaro et Al. [9] presented a new linear-depth ripple-carry quantum addition circuit making use only of controlled-NOT (CNOT or Feynman) gates and controlled-controlled-NOT (CCNOT or Toffoli) gates which was an improved version of a V-shaped reversible adder presented by [10]. In 2008, [7] presented the synthesis and design of a reversible Fourier transform making use of such a reversible adder, but using a do-spy-undo (majority-unmajority) scheme structure as firstly presented by [11]. This design was making use only of controlled-NOT (Feynman) gates and controlled-SWAP (Fredkin) gates<sup>1</sup>. Unfortunately, this 8 bits adder was embedded in a larger 8 bits multiplier making it impossible specific access to measurements.

The circuit we present in this paper is a *4 bits majority-based reversible ripple-carry adder*. As in [7], it makes use only of Feynman and Fredkin gates and presents, as a general structure, a do-spy-undo scheme [11]. Full details about synthesis discussion, structure and theoretical consumption are already presented in [12].

### B. Block structure

As a summary, let us recall the block structure of the studied adder. Let  $n$  denote the size of both numbers to be added. First, as a *do-spy-undo* structure is used, each bit addition implemented, except the most significant bit (MSB), necessitates one *do-undo* circuit. The 3 inputs majority *do* circuit and the 3 inputs unmajority *undo* circuit are presented in **Fig.1** and **Fig.2**, respectively.

The *do-undo* block constitutes a one bit adder when used forward (respectively a one bit subtractor if used backward). The numbers  $a$  and  $b$  (respectively  $S$  and  $A$ ) are the main inputs and  $C_{in}$  ( $C_{out}$ ) is the carry-in. The number  $S$  ( $b$ ) represents the computation result (sum  $a + b$  respectively

<sup>1</sup>Both Fredkin and Toffoli gates are universal gates, which means that any logical or arithmetic operation can be constructed entirely of one of those gates.

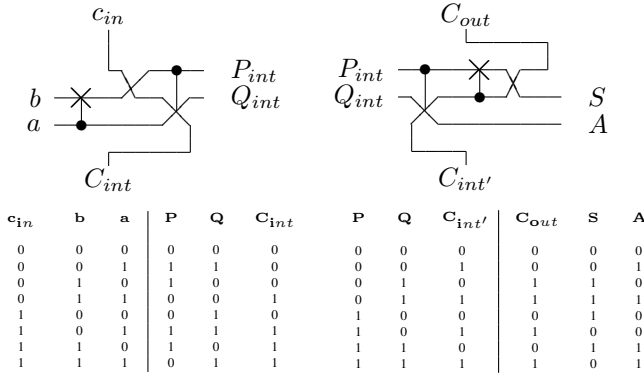


Fig. 1. Quantum diagram and truth table of a majority (do) block.

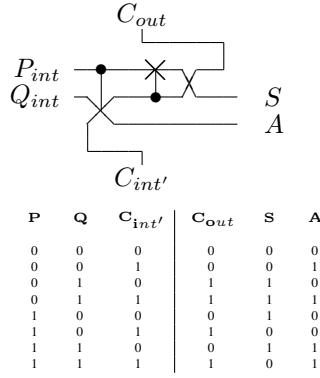


Fig. 2. Quantum diagram and truth table of an unmajority (undo) block

difference  $S - A$  and  $A$  ( $a$ ) is a copy (garbage) of the input  $a$  ( $A$ ) and so is the output  $C_{out}$  ( $C_{in}$ ) with respect to the input  $C_{in}$  ( $C_{out}$ ).

When an addition is computed (forward calculation), the internal bits  $C_{int}$  are used for carry transmission from one stage to the next during the majority operations, while the input  $C_{int'}$  of the undo block rebuild the initial input  $c_{in}$  during the undo operations ( $C_{out} = c_{in}$ ). When a difference is computed (backward calculation), the inverse process occurs leading to the calculation of the difference.

For 1 bit calculation,  $C_{int}$  would be directly connected to  $C_{int'}$  whereas if the adder size  $n \geq 2$ , the most significant bits addition is performed by using only two Feynman gates; one is used to compute the bit sum XOR ( $\oplus$ ) and the other to sum-up the carry to the final result. Each extra bit addition is realized by cascading supplementary *do-undo* blocks, linked together by connecting  $C_{int}$  to  $c_{in}$  and  $C_{out}$  to  $C_{int'}$  as presented in the full schematic of the 4 bits adder in **Fig.3**.

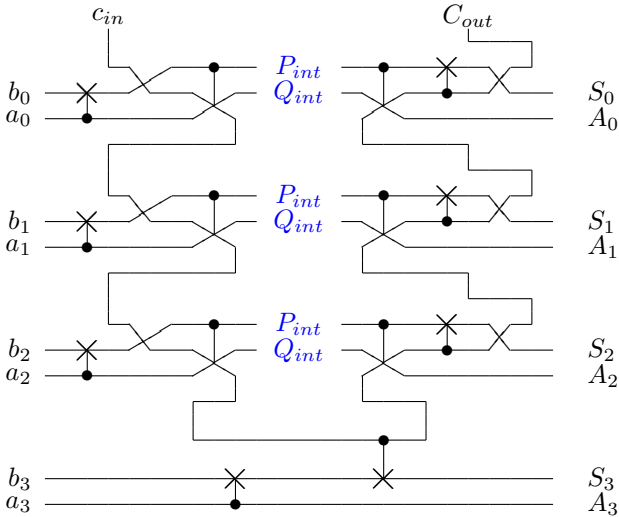


Fig. 3. Full quantum diagram of the 4 bits Cuccaro adder.

Let us notice that simplification presented in **Fig.4** has been used for the real design of the unmajority block.

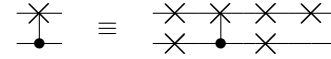


Fig. 4. Simplification sometime used for drawing of the real circuit.

### III. DESIGN AND REALIZATION

This 4 bits Cuccaro adder has been designed using the *Cadence*® computer-aided design environment software. Each electrical simulation has been performed using its *Cadence Spectre*® simulator (see **Fig.7** and **Fig.8** as examples).

The chip processing has been realized at *ON Semiconductor* through the *Europractice / IMEC* consortium in 350 nm standard CMOS technology. The transistor lengths used are  $L = 350 \text{ nm}$ , both for n-type and p-type transistors, while widths are respectively  $W_n = 500 \text{ nm}$  and  $W_p = 1500 \text{ nm}$ . The chip contains a total of 160 transistors (80 n-type and 80 p-type).

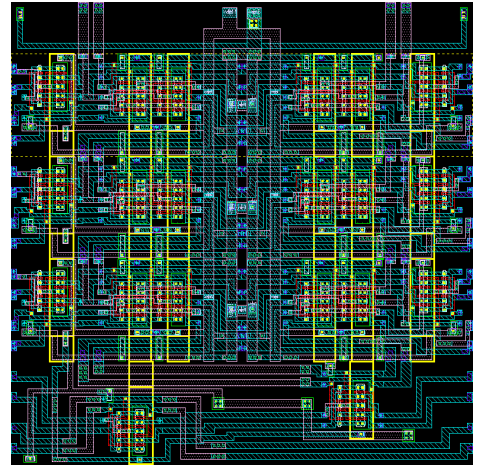


Fig. 5. Cadence Virtuoso® layout of the studied 4 bits Cuccaro adder.

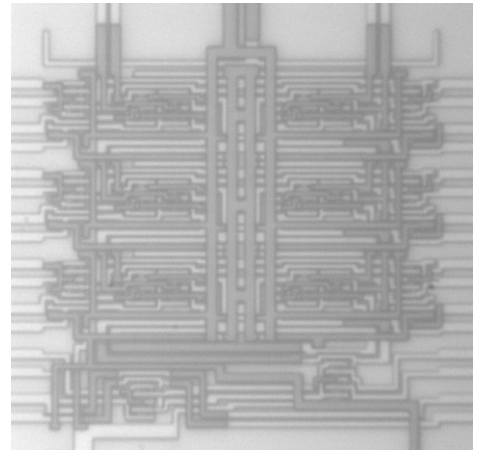


Fig. 6. Photograph of the processed 4 bits Cuccaro adder chip ( $90 \mu\text{m} \times 90 \mu\text{m}$ ).

The Cadence Layout of the studied adder is shown in **Fig.5** while **Fig.6** presents a photograph of the realized chip

(90  $\mu\text{m}$  x 90  $\mu\text{m}$ ). We can easily recognize the dual inputs and outputs at each left and right hand sides, and the two dual carries  $C_{in}$  and  $C_{out}$  at the top side while in the middle part are the wider wires used for the substrate and the N-wells biasing (typically,  $V_{SS} = -1.2\text{ V}$  and  $V_{DD} = 1.2\text{ V}$ ).

#### IV. SIMULATION RESULTS

Starting with the simulation, two different types of signal have been used: "traditional" square pulses and *quasi*-adiabatic triangle pulses.

##### A. Input signal definitions

The maximal signal voltages used, both for simulation and for measurements are  $V_+ = 1\text{ V}$  for logic "1" and  $V_- = -1\text{ V}$  for logic "0".

If the first type of signal (square pulse) is applied, for the unchanged bits, the voltages corresponding to the desired logic levels remain constant while others are fastly switched. If the second signal type (adiabatic) is applied, a triangular pulse ranges from  $\frac{V_+ + V_-}{2}$  to the desired logic level at each calculation step. Thus, in case of the second signal type, all signals (i.e. both the unchanged bits and the changing ones) temporarily come to the equilibrium voltage, half-way logic "1" and logic "0". The two signal types can be seen in **Fig.7** presenting the *minuend*  $S$  and *subtrahends*  $A$  of nine subsequent subtractions  $S - A$ .

##### B. Simulated outputs

Let us stress that the pulses of **Fig.7** are representative of the input signals previously defined. **Fig.8** presents the corresponding *difference*  $b$  and *garbage*  $a$  calculated outputs when using the Cuccaro adder in reverse mode (subtractor), for both the square and triangle adiabatic pulse shapes. In **Fig.8**, we see that rectangular pulses do not all have the same amplitude, that they are also presenting large transition peaks while for the triangle *adiabatic* signals, the pulses are regular, without any transition peak.

Let us also notice that when the size  $n$  of the numbers to be computed is larger than two, some exceptional errors may occur when square signals are used while this doesn't happen when adiabatic signals are applied (**Fig.8**).

One reason for that comes directly from the transmission gate that is build of two complementary transistors in parallel. Due to the steep slope generated during the transition of the square pulse, if a delay occurs between the control signals on the transistor gates and the signals transmitted through the transistor channels, the gates do not shut down the signal at the very moment of the transition and an opposite signal that would have been cut down is transmitted anyway, occasionally causing calculation errors.

If the adiabatic signals are used, the smooth transitions allow a delay between the signals. In effect, during the transition steps, the two transistors are working below their threshold voltages, drastically lowering the amplitude of the output signal (see also **Fig.10** and comments). As the example of **Fig.9** shows, a phase difference – expressed as a percentage of

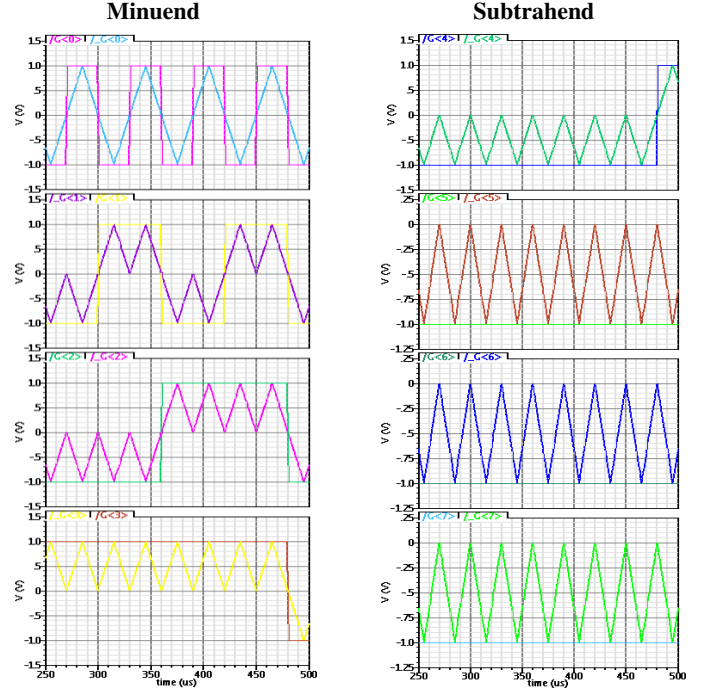


Fig. 7. Cadence Spectre<sup>©</sup> simulation: comparison of rectangular and adiabatic input pulses of same frequency. Top line is the least significant bits  $S_0$  and  $A_0$  whereas the bottom one corresponds to the most significant bits  $S_3$  and  $A_3$ .

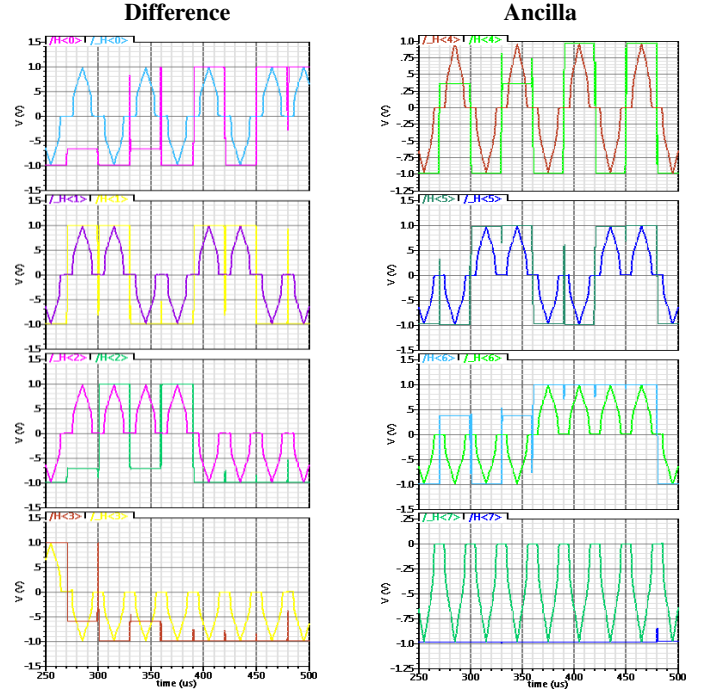


Fig. 8. Cadence Spectre<sup>©</sup> simulation: comparison of the output signals when using rectangular and adiabatic pulses, respectively. According to Spectre<sup>©</sup> electrical simulation software, using rectangular pulses introduces several errors in computation. None of these errors occur when using triangular pulses. Top line is the least significant  $b_0$  and  $a_0$  bit whereas the bottom one corresponds to the most significant ones  $b_3$  and  $a_3$ .

the period – as large as a quarter of a period  $Prct = \pm 25\%$  (plain lines) does not introduce any calculation errors, even if the final shape of the signal is deformed and if some small variations of the output potential (inferior to 40 mV) appear when the transmission gate is closed. In fact, this maximal percentage will be dependant of the ratio between the maximum pulse voltage and the pseudo-threshold voltage of the transmission-gates used. When a bigger phase appears, the output signal is no more zeroed before the command pulse.

A positive (respectively negative) phase indicates a delay of the input (respectively command) signal with respect to the command (respectively input) pulse <sup>2</sup>.

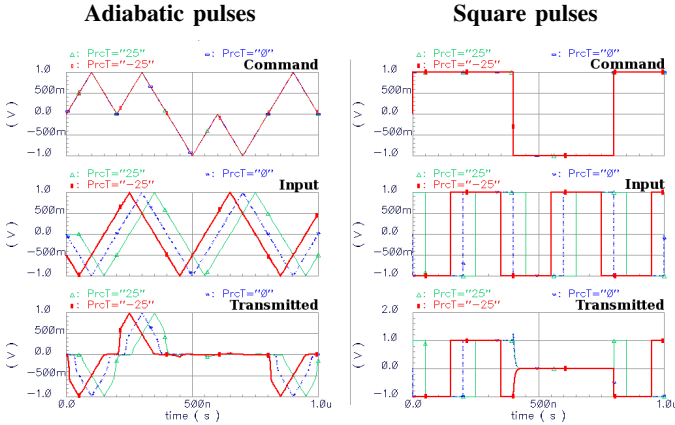


Fig. 9. Cadence Spectre<sup>®</sup> simulation of a complementary pass transistor gate: comparison between adiabatic and square pulses when a dephasing of  $Prct = -25\%$  (bold) and 25% of the period is applied between the command pulses and the signal to be transmitted. The dashed lines correspond to a phase equal to 0 %. The frequency used is 5 MHz.

Further in the calculation steps, the signals may become symmetrical again, when two phased signals are used successively as the command and the transmitted signals. This then leads to a narrowing of the final pulse width (as seen in Fig.8) but does not impact significantly its maximal amplitude that is related to the impedance of the circuits. In Fig.9 (dash lines), the simulated output signal obtained from a single transmission gate when the phase equals zero ( $Prct = 0\%$ ) is larger than the output signals obtained from a complex circuit such as a subtractor (Fig.8), where a phase is introduced by the propagation of the internal carries  $C_{int}$  and  $C_{int'}$ . In effect, in the Cuccaro adder, the  $c_{in}$  signal when going through the whole computation, has to be transmitted twice at each stage of the calculation. The greater the bit size  $n$ , the greater the number of stages, the greater the phase introduced, the narrower the pulse. When using a square signal, this may produce wrong pulses whereas when using an adiabatic shape, the signal will be deformed, narrowed, but the calculation will remain possible with very few loss in amplitude and still guaranteeing a correct logic “1” or correct logic “0”. This was checked up to  $n = 6$  bits.

<sup>2</sup>The transmitted signal is reduced when the command signal amplitude is smaller than the threshold voltage of the pass-transistor gate, causing an asymmetry of the transmitted signal.

It is also easy to verify that dual signals (e.g.  $a$  and  $\bar{a}$ ) are closely synchronized by symmetry of the dual circuits – each signal and its complementary having passed through the same number of transmission gates.

As a consequence, even if we need to pay attention to the design of the circuit in order to limit the phase introduced between the different signals used in the calculation, the triangular adiabatic shape signals allow to drastically reduce the number of level restorers and pulse resynchronizers.

## V. TESTS AND MEASUREMENTS

Several extensive measurement steps have been performed on the Cuccaro circuit, both in forward (adder) and in reverse (subtractor) calculations.

Constant voltage computations are primary presented allowing to bring information principally on impedance and voltage bias impact on calculation. In a second hand, we present the adiabatic generator used for the experimental application of the input pulses. We then present measurements of the outputs obtained both in direct and reverse calculations.

### A. Preliminary DC measurements

The voltages applied: substrate and NWELL biases, input constant biases, have been measured using a Fluke 175 multimeter in DC mode.

A large part of the truth table has been first successfully checked and some measurements have been done for some particular conditions:

- In direct calculation (adder), 32 different conditions:  $c_{in} = 0$ ,  $b = 1010$  and  $a$  varying, as well as  $c_{in} = 0$ ,  $a = 0101$  and  $b$  varying.
- In reverse calculation (subtraction), 16 different test vectors: ( $C_{out} = 0$ ,  $A = 0101$  and  $S$  varying).

From these measurements, the maximum voltage drop found for the adder is 30 mV whereas the maximum voltage drop for subtractor was found to be 6 mV over a constant input bias of 1 V. In any cases, the output drops are inferior to 3 % of the input signal biases which is the order of magnitude of the measurement incertainties.

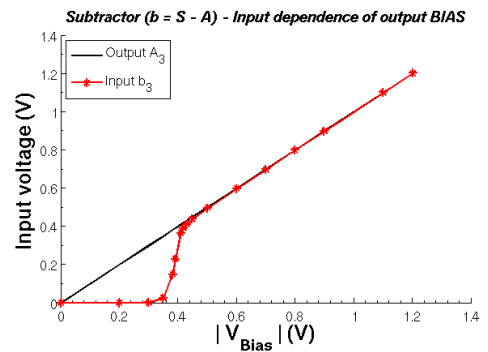


Fig. 10. Measurements of the input voltage dependence of bit  $b_2$  as a function of the output voltage bias of bit  $S_2$ , for the reverse (subtractor) calculation:  $b = S - A = 0110 - 0101 = 0001$ .

Fig.10 presents a typical variation of an input voltage as a function of the output voltage bias when the adder is used



in reverse mode (i.e. as subtractor). In fact, same curves are obtained for each output whether in direct mode (adder) or reverse one (subtractor), except for the most significant bit  $A_3$  (respectively  $a_3$ ) that is made by a direct connection to its corresponding input  $a_3$  (respectively  $A_3$ ) (see **Fig.3**). For all measurement performed, the voltages  $V_{DD}$  and  $V_{SS} = V_{GND}$  are kept constant to 1.2 V and  $-1.2$  V respectively.

We see from **Fig.10**, that a minimal voltage input as small as 420 mV can still be used (without security marge) to perform calculations with this Cuccaro chip. This is also coherent with the general shape of simulated output pulses in **Fig.8** and with the experimental measurements in **Fig.12** and **Fig.13** that present the same transition when applying a linear input variation (triangular adiabatic pulse). But in these last cases, one also needs to account for the impact of the phases introduced by the internal carry signals that not only narrow the pulse width but also slightly increase the minimal voltage input possibly used to perform a calculation. We then arbitrarily fix the limit of  $\pm 500$  mV as the minimum logical amplitude acceptable to define a logic "1" (above 500 mV) or a logic "0" (below  $-500$  mV). This choice is experimentally driven by the fact that within these two defined domains, the calculated signals obtained have always been found conform to what was expected.

The *pseudo*-threshold voltage<sup>3</sup>  $V_{pT}$  of the Cuccaro adder is found to be about 350 mV which is also close to the simulated value of the transmission-gate pseudo-threshold voltage. Between about 420 mV down to  $V_{pT} = 350$  mV a transition occurs that corresponds to the linear mode of the transmission gate transistors. Below  $V_{pT}$ , the transmission gates are blocked and, as expected, no signal is transmitted.

### B. Generation of the adiabatic signal

In order to apply the necessary adiabatic pulses, we designed an electronic module allowing to generate 4 types of triangle pulses:

- 1) A logic "1" ranging from  $\frac{V_+ + V_-}{2}$  up to +1 V and back,
- 2) A logic "0" ranging from  $\frac{V_+ + V_-}{2}$  down to  $-1$  V and back,
- 3) A switching signal ranging between +1 V and  $-1$  V,
- 4) The dual signal of the switching signal that is the symmetrical signal from the mean value  $\frac{V_+ + V_-}{2}$ .

These four signals are all build up from one single triangular pulse ranging between +1 V and  $-1$  V by using a single buffer (switching pulse), an inverter followed by a buffer (dual switching pulse) or using a precision full-wave rectifier followed either by a buffer or by an inverter and a buffer (logic "1" and logic "0" respectively). **Fig.11** presents the full schematic of the electronic module used to generate the experimental adiabatic pulses. It is a very simple thus very accurate circuit presenting very low output impedances. The operational amplifiers are polarized at  $\pm 3$  V while the input triangular signal has an amplitude of 2 V. This allows the

operational amplifiers to work in the linear part of their characteristics and gives very accurate and reproducible pulses.

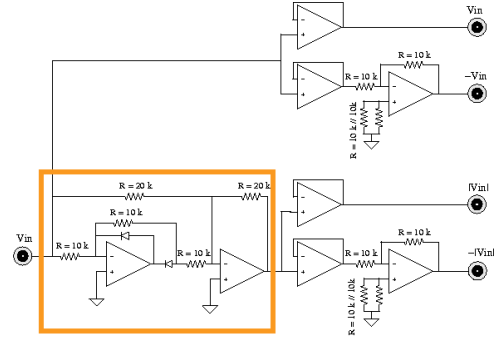


Fig. 11. Electronic schematic of the adiabatic signal generator module embedding an accurate full-wave rectifier (framed).

### C. Adiabatic AC measurements

AC measurements have been performed using a commercial Tektronics™ TDS 210 digital oscilloscope. The output charge is its  $\times 1$  probe that can be modeled by a 1 M $\Omega$  resistor in parallel with a 32 pF capacitance.

By using the voltage divider technique, the experimental output impedance of the Cuccaro adder (or subtractor) is found to be  $Z_{out} \simeq 5250 \Omega \pm 5\%$  whereas it is found to be close to 8 k $\Omega$  by Cadence Spectre© simulation.

**Fig.12** and **Fig.13** show typical experimental measurements realized respectively in forward (adder) and backward (subtractor) computation.

### Experimental Measurements in forward computation –

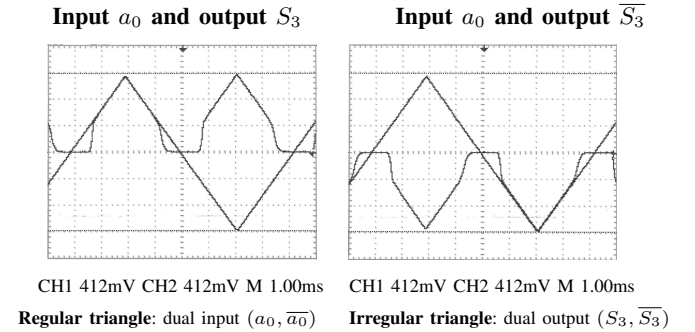


Fig. 12. Oscilloscope snap-view of the 4<sup>th</sup> dual bits ( $S_3, \overline{S_3}$ ) as a function of the dual switching input bits  $(a_0, \overline{a_0}) = (X, \overline{X})$ , obtained by direct (adder) calculation of  $S = c_{in} + a + b = 1 + 011X + 0110 = 11X\overline{X}$  (with  $c_{in} = 1$ ), measured when using adiabatic pulses.

In **Fig.12**, the dual output ( $S_3, \overline{S_3}$ ) is given as the example of a constant logic "1" adiabatic output and compared to the switching input  $a_0$ . For clarity, the dual signal  $\overline{S_3}$  that is a logic "0" is also compared to the same input  $a_0$  – the complementary  $\overline{a_0}$  being the exact opposite signal of  $a_0$  ( $\overline{a_0} = -a_0$ ).

As previously found in **Fig.10**, the impact of the threshold voltage can be seen on both dual output signals in **Fig.12**, at the rise front whereas at the descending one, its impact is superimposed to capacitance effect coming from the probe

<sup>3</sup>We will use the term *pseudo*-threshold voltage when related to a gate in order to differentiate this value from the transistor threshold voltages.

that slows down the decrease of the signal to the mean value  $\frac{V_+ + V_-}{2}$  here equal to zero, while all the transmission gates of the chip are reaching their *open circuit* functionality.

### Experimental Measurements in backward computation –

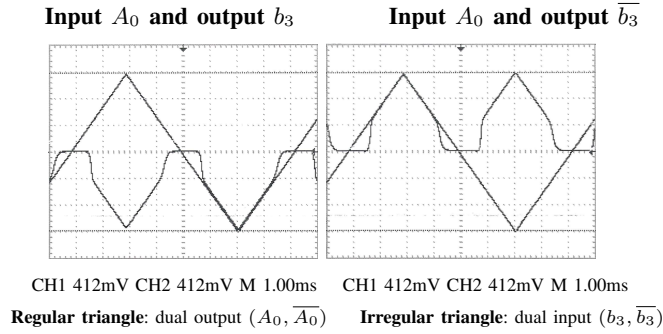


Fig. 13. Oscilloscope snap-view of the 4<sup>th</sup> dual bits ( $b_3, \bar{b}_3$ ) as a function of the dual switching bits  $(A_0, \bar{A}_0) = (X, \bar{X})$ , obtained by reverse (subtractor) calculation of  $S - A = b = 0110 - 010X = 00\bar{X}\bar{X}$  with  $C_{out} = 0$  and measured when using adiabatic pulses.

As shown in **Fig.13**, very similar results are obtained when performing a subtraction using the same chip in reverse direction. Here, the output calculated ( $b_3, \bar{b}_3$ ) signal, shown as typical example, is a constant dual zero also compared to the same input signal  $A_0$ .

As the carry propagates twice through the whole circuit, a maximal delay is obtained between the two extreme output bit pulses (e.g.  $a_0$  and  $b_3$ ). The maximum delay  $\tau = 91 \mu s$  is measured between outputs  $a_0$  and  $b_3$  during a subtraction (**Fig.14**). This delay is obtained between two pulses at a frequency 120 Hz, when probes are connected to the outputs. According to these measurements, it is possible to cascade more than 23 Cuccaro gates before the output delay reaches 25 % of the period such that calculation errors may appear.

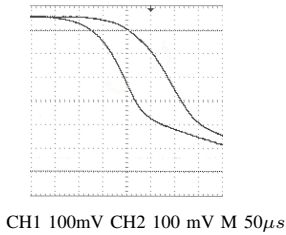


Fig. 14. Experimental maximum phase introduced during a computation step.

### Cascade Measurement –

#### Example of switching signal measured on cascaded circuits

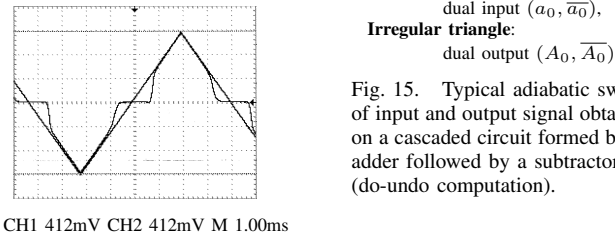


Fig. 15. Typical adiabatic switch of input and output signal obtained on a cascaded circuit formed by an adder followed by a subtractor (do-undo computation).

As a last experiment, we cascaded an adder with a subtractor. This experiment allowed to verify that two sequential calculations can be performed and that the addition can really be uncomputed giving back the initial input signal as in the example given in **Fig.15**.

A large part of the *identity* truth table (different from the first experiment) has been checked with no errors detected. Also, no significant extra phase nor attenuation occurred during this computation steps, giving the indication that a lot of reversible chips may be cascaded.

Nevertheless, the output signal having been transmitted through several transmission gates is not the exact input signal, but a calculated one and inherit of the same characteristics than another calculated output.

### VI. CONCLUSION

We presented real calculations performed in a 350 nm CMOS technology reversible (quantum) adder used in both direction: forward adder and backward subtractor.

The superiority of making use of adiabatic triangle pulses has been demonstrated and explained, showing that using adiabatic pulses on pass-transistor based reversible (quantum) chips is a suitable solution to perform linear calculations.

### ACKNOWLEDGMENT

The authors thank the *Danish Council for Strategic Research* for the support of this work in the framework of the *MicroPower* research project. They are grateful to the *Invemec* division of *Imec* v.z.w. (Leuven, Belgium) and the *Europractice* consortium for the help with the prototyping of the chips.

### REFERENCES

- [1] De Vos A., *Lossless computing*, Proceedings of the IEEE Workshop on Signal Processing, Poznań (2003) pp.7–14.
- [2] Feynman R.P., *Quantum mechanical computer*, Optics News **11** (1985) pp.11–20.
- [3] Landauer R., *Irreversibility and heat generation in the computing process*, IBM Journal of Research and Development **5**(3), (1961) pp.183–191.
- [4] Bennett C.H., *Logical reversibility of computation*, IBM Journal of Research and Development **17**(6), (1973) pp.525–532.
- [5] Toffoli T., *Reversible computing*, in: Automata, Languages and Programming, ed. by W. de Bakker, J. van Leeuwen (Springer, Berlin, 1980), Technical Memo MIT/LCS/TM-151, MIT Laboratory for Computer Science (1980) pp.632–644.
- [6] De Vos A., *Reversible computer hardware*, Electronic Notes in Theoretical Computer Science **253**, (2010) pp.17–22.
- [7] Skoneczny M., Van Rentergem Y. and De Vos A., *Reversible Fourier transform chip*, Proceedings of the 15<sup>th</sup> International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES 2008), Poznań, June 2008, (2008) pp.281–286.
- [8] De Vos A., Burignat S. and Thomsen M.K., *Reversible implementation of a discrete integer linear transformation*, Proceedings of the 2<sup>nd</sup> Reversible Computation Workshop, Bremen (2010) pp.107–110.
- [9] Cuccaro S., Draper T., Moulton D. and Kutin S., *A new quantum ripple-carry addition circuit*, Proceedings of the 8<sup>th</sup> Workshop on Quantum Information Processing, Cambridge (June 2005), arXiv:quant-ph/0410184v1 (2004) 9 pages.
- [10] Verdal V., Barenco A. and Ekert A., *Quantum networks for elementary arithmetic operations*, Physical Review A **54** (1996) pp.147–153.
- [11] Fredkin E. and Toffoli T., *Conservative logic*, International Journal of Theoretical Physics, **21** (2004) pp.219–253.
- [12] De Vos A., *Reversible Computing*, WILEY-VCH (2010) 251 pages.
- [13] Van Rentergem Y. and De Vos A., *Optimal design of a reversible full adder*, International Journal of Unconventional Computing **1** (2004) pp.339–355.