

# A Review on Performances of Reversible Ripple-Carry Adders

Stéphane Burignat and Alexis De Vos

**Abstract**—Quantum computing and circuits are of growing interest and so is reversible logic as it plays an important role in the synthesis of quantum circuits. Moreover, reversible logic provides an alternative to classical computing machines, that may overcome many of the power dissipation problems in the near future. Some ripple-carry adders based on a do-spy-undo structure have been designed and tested reversibly. This paper presents a brief overview of the performances obtained with such chips processed in standard  $0.35\ \mu\text{m}$  CMOS technology and used in true reversible calculation (computations are performed forwards and backwards such that addition and subtraction are made reversibly with the same chip). Adiabatic signals used are known to allow the signal energy stored on the various capacitances of the circuit to be redistributed rather than being dissipated as heat while allowing to avoid calculation errors introduced by the use of conventional rectangular pulses. Through the example of both simulations and experimental results, this paper aims at providing a base of knowledge and knowhow in physical implementation of reversible circuits.

**Keywords**—reversible computation, design, implementation, pass-transistor logic, ripple-carry adder, sum-difference block, Spectre simulation, quantum computation, adiabatic signal, test and measurement, error propagation.

## 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]–[6].

The physical implementation of the adder we are presenting is reversible dual-line complementary pass-transistor CMOS logic [7] 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 [8] and, more recently, in a H264/AVC encoder [9], [10].

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 evaluate the efficiency of adiabatic signals.

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

### A. Short History

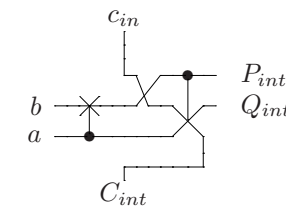
In 2005, Cuccaro et Al. [11] presented a new linear-depth ripple-carry quantum addition circuit making use only of controlled-NOT (CNOT or Feynman) gates and controlled-controlled-NOT (CCNOT or Toffoli) gates which was an improved version of a V-shaped reversible adder presented

by [12]. In 2008, [8] 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 [13]. 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.

In this paper, the performances of the circuit briefly presented in [14] are detailed. This circuit is the *majority-based reversible ripple-carry adder*. As in [8], it makes use only of Feynman and Fredkin gates and presents, as a general structure, a do-spy-undo scheme [13]. Extra details about synthesis discussion, structure and theoretical consumption can be found in [15].

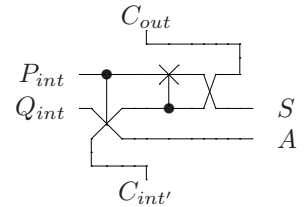
### B. Block Structure

For clarity, 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.



$c_{in}$	$b$	$a$	$P$	$Q$	$C_{int}$
0	0	0	0	0	0
0	0	1	1	1	0
0	1	0	1	0	0
0	1	1	0	0	1
1	0	0	0	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	0	1	1

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



$P$	$Q$	$C_{int'}$	$C_{out}$	$S$	$A$
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	1	1	0
0	1	1	1	1	1
1	0	0	0	1	0
1	0	1	1	0	0
1	1	0	0	1	1
1	1	1	1	0	1

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

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  $S$  ( $b$ )

S. Burignat and A. De Vos are with Universiteit Gent, Vakgroep elektronika en informatiesystemen, Sint Pietersnieuwstraat 41, B-9000 Gent, Belgium (e-mails: research@burignat.eu; alex@elis.ugent.be).

<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.

represents the computation result (sum  $a + b$  respectively difference  $S - A$ ) and  $A(a)$  is a copy (ancilla) 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 to 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 3 bits adder in Fig. 3.

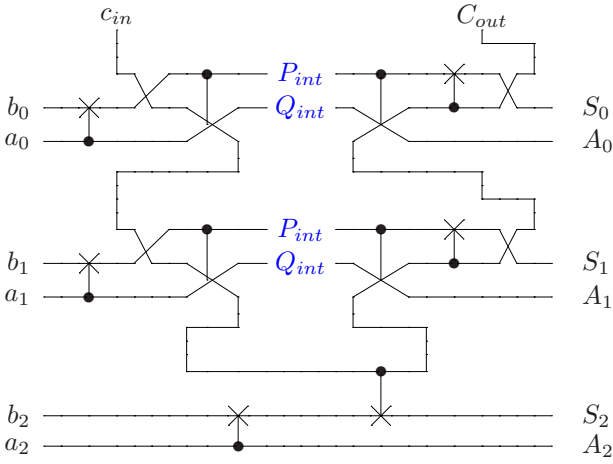


Fig. 3. Full quantum diagram of the 3 bits Cuccaro adder.

### III. DESIGN AND REALIZATION

The studied Cuccaro adders have been designed using the *Cadence*® computer-aided design environment software. Each electrical simulation has been performed using its *Cadence Spectre*® simulator (see Fig. 6 and Fig. 7 as examples).

The chip processing of the 4 bits adder has been realized at *ON Semiconductor* through the *Europractice / IMEC* consortium in 350 nm standard CMOS technology. Fig. 4 presents a photograph of the realized chip ( $90 \mu\text{m} \times 90 \mu\text{m}$ ). 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).

As an example of the extendability of the size of the Cuccaro adder, the Cadence Layout of a 6 bits adder is shown in Fig. 5.

In both the layout and the photograph of Fig. 4 and 5, 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

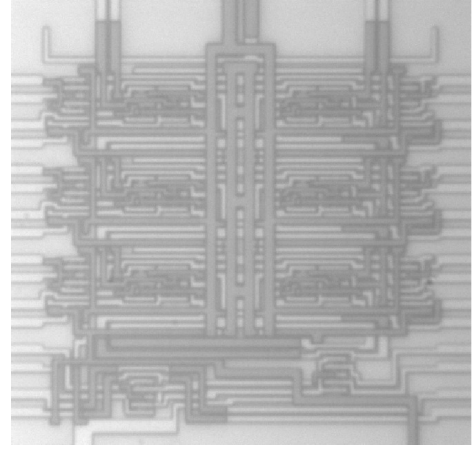


Fig. 4. Photograph of the processed 4 bits Cuccaro adder chip ( $90 \mu\text{m} \times 90 \mu\text{m}$ ).

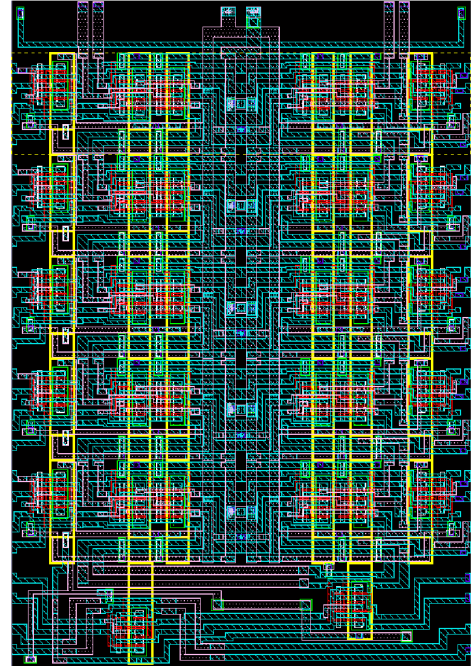


Fig. 5. Cadence *Virtuoso*® layout of the studied 6 bits Cuccaro adder.

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 simulations, two different types of signal have been used: “traditional” square pulses and *quasi-adiabatic* triangular pulses. We already proved [6], [14] that the use of adiabatic pulses allows removing the calculation errors introduced with the delay between the control and the transmitted signals at each gate. This is applicable only if the delay between the two signals remains shorter than the time the transistors of the transmission gates are working below their threshold voltage. In effect, during that time, the amplitude of the output signal is drastically lowered and even if a wrong signal appears, it has no impact on the computation result (see also Fig. 8 and Fig. 9 and comments). This is of course

valid for all signal shapes presenting smooth raising and falling edges.

In the following sections, we will focus only on the performances of the Cuccaro adder in *adiabatic* computation, starting with the known results.

### 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”.

An adiabatic triangular pulse ranges from  $(V_+ + V_-)/2$  to the desired logic level at each calculation step. Thus, between two subsequent computations, 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”. Several examples of input adiabatic signals can be seen in Fig. 6 presenting the *minuend*  $S$  and the *subtrahend*  $A$  of nine full subsequent subtractions  $S - A$ . As for example, at  $t = 259\text{ }\mu\text{s}$ , we recognize the inputs  $S = 010$  and  $A = 001$ .

### B. Simulated Outputs

Let us stress that the pulses in Fig. 6 are representatives of the adiabatic input signals previously defined. Fig. 7 presents the corresponding *difference*  $b$  and *ancilla*  $a$  calculated outputs when using a 3 bits Cuccaro adder in reverse mode. As for example, at  $t = 259\text{ }\mu\text{s}$ , we find the outputs  $b = 001$  and  $a = 001$ , confirming  $b = S - A$  and  $a = A$ . The temporary signal reduction of the output signals due to the threshold voltage  $V_T$  of the transistors is clearly visible around the equilibrium voltage.

This also can be seen on Fig. 8, where we intentionally introduced (between the two input signals) a phase difference – expressed as a percentage of the period – as large as a quarter of a period  $PrdT = \pm 25\%$  (dotted lines). Even such long delay 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  $40\text{ mV}$ ) appear when the transmission gate is closed. In fact, this maximal percentage  $PrdT$  will first depend on the ratio between the maximum pulse voltage and the pseudo-threshold voltage<sup>2</sup> of the transmission-gates used. Further in the calculation steps, the signals may become symmetrical again, however somewhat narrowed. The pulse height is not affected, as it is related to the impedance of the circuits. Neither is its position, allowing the pulse to be accurately read. After several gates, as the signals are narrowed, a large enough overlap of the defined signals has to be ensured such that the command signal can still drive the transmitted signal while the latter remains large enough to be readable. A positive (respectively negative) phase indicates a delay of the input (respectively command) signal with respect to the command (respectively input) pulse<sup>3</sup>.

As a consequence, in Fig. 8, the simulated output signal obtained from a single transmission gate when the phase

<sup>2</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.

<sup>3</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.

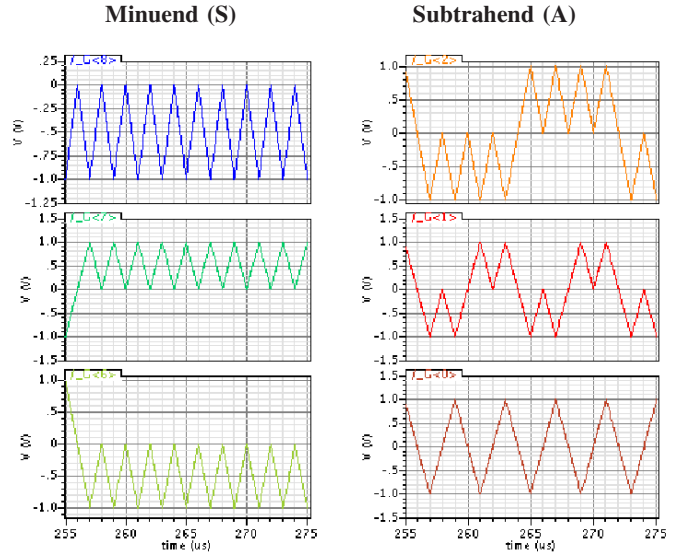


Fig. 6. Cadence Spectre<sup>®</sup> simulation for a 3 bits Cuccaro adder in adiabatic calculation.

Bottom line is the least significant **input** bits  $S_0$  and  $A_0$  whereas the top one corresponds to the most significant bits  $S_2$  and  $A_2$ .

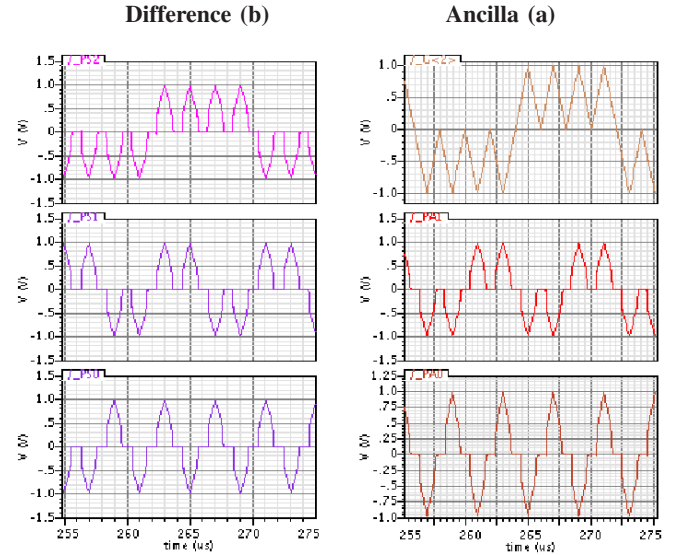


Fig. 7. Cadence Spectre<sup>®</sup> simulation for a 3 bits Cuccaro adder in adiabatic calculation.

Bottom line is the least significant **output** bits  $b_0$  and  $a_0$ , whereas the top one corresponds to the most significant bits  $b_2$  and  $a_2$ .

equals zero ( $PrdT = 0\%$ , *plain lines*) is larger than the output signals obtained from a complex circuit such as a subtractor (Fig. 7), where a phase is inevitably introduced by the propagation of the internal carries  $C_{int}$  and  $C_{int'}$ .

For the Cuccaro adder, the most extreme case is found when the signal  $c_{in}$  has to propagate through the whole computation flow from  $c_{in}$  to  $C_{out}$  (adder) or from  $C_{out}$  to  $c_{in}$  (subtractor), as it has to be transmitted twice at each stage of the calculation. Thus, the greater the bit size  $n$ , the greater the possible number of stages, the greater the phase introduced, the narrower the pulse. This also corresponds to the largest delay found between the output signals as the most significant ancilla bit  $A_{n-1}$



Impact of a small delay on adiabatic pulses

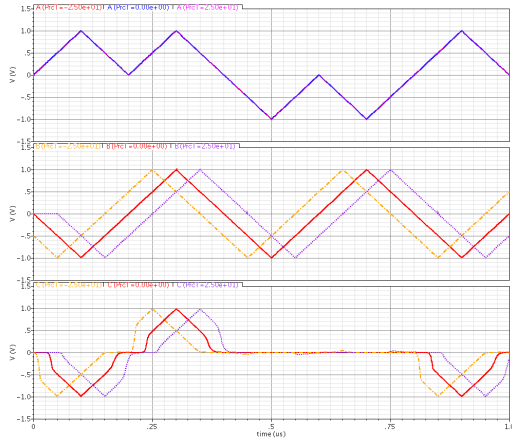


Fig. 8. Cadence Spectre<sup>®</sup> simulation of a complementary pass transistor gate: impact on a moderate dephasing of  $PrcT = -25\%$  (dashes and dots line) and  $25\%$  of the period (dotted line) is applied between the command pulses and the signal to be transmitted. The plain lines correspond to a phase equal to  $0\%$ . The frequency used is  $2\text{ MHz}$ . No error occurs in the output signal.

is a direct connection to the corresponding input bit  $a_{n-1}$  whereas the least significant bit  $S_0$  has to wait for the last step of the calculation flow for its value to be fixed. See Fig. 3, where output  $A_2$  is an instantaneous copy of input  $a_2$ , whereas output  $S_0$  results from a long sequence of computation steps, depending on all inputs  $c_{in}$ ,  $b_0$ ,  $a_0$ ,  $b_1$  and  $a_1$ .

*This argues in favor of a parallel circuit architecture instead of a ripple-carry structure for reversible circuits based on transmission gates.*

It is 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 complement having passed through the same number of transmission gates. Let us notice here that the number of transistors passed by each complementary signal is the same but not necessarily of same type. The transmission gates are optimized in such a way that their impact on signal is close to symmetry regarding the amplitude and phase introduced by the gate. But reality is more complex and p-type transistors are wider than n-type ( $W_n = 500\text{ nm}$  and  $W_p = 1500\text{ nm}$ ) even if their lengths are equal ( $L_n = L_p = 350\text{ nm}$ ). This ensures equal channel resistances, but a drawback is that the gate and bulk capacitances of p-transistors are at least three times larger than for n-type and so is their impact on the signal. Thus, a small extra delay can be introduced between complementary signals due to the difference of transistor types.

But what is the impact of a too long delay in our reversible circuits?

If a too large phase is involved, the output signal can no more be zeroed properly around the equilibrium voltage (between two subsequent computations), leading to calculation errors. Moreover, some pulses appear whereas the transmission gates should be blocked, leading to extra voltage-level errors as shown in Fig. 9.

The consequence of such large delays is exactly the same as if one failing transmission gate is only passing. A short

Impact of a large delay on adiabatic pulses

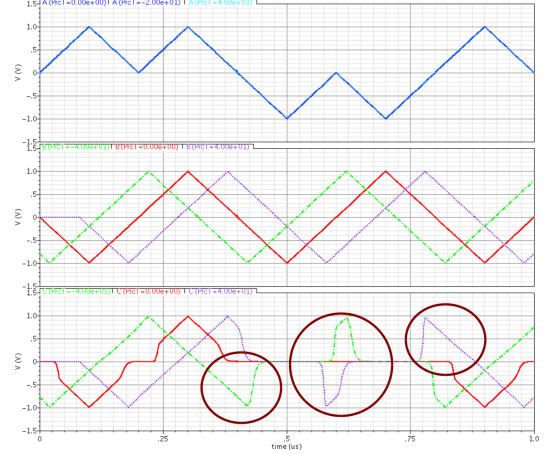


Fig. 9. Cadence Spectre<sup>®</sup> simulation of a complementary pass transistor gate: impact on a large dephasing of  $PrcT = -40\%$  (dashes and dots line) and  $40\%$  of the period (dotted line) is applied between the command pulses and the signal to be transmitted. The plain lines correspond to a phase equal to  $0\%$ . The frequency used is  $2\text{ MHz}$ . Several errors occur in the output signal (circled).

circuit appears temporarily between two signals or one signal and its complement, leading to the propagation of wrong pulses of bad amplitudes, as illustrated for a Feynman gate in Fig. 10. This error can easily be experimentally reproduced by adding a short circuit (as for example a resistor) between two input signals. The propagation of the error will impact several outputs depending on the input data, the circuit architecture and the signal propagation flow<sup>4</sup>. Fig. 11 gives an example of such a bad signal due to an internal short circuit.

The regular triangle of Fig. 11 is a typical input signal. Its left hand part with negative voltages is a logical “0”, whereas the positive part is the first half part of logic “1”. The output signal is supposed to be a logic “0”, thus a negative voltage triangle. However, due to the propagation of a wrong internal signal, this is not the case (at present computation speed) and the pulse is changing several times both in amplitude and voltage polarity over its half period, depending on the command flow received on the different intermediate gates.

Another cause for a similar error is a too high calculation frequency. In this case, the error comes from the fact that some of the gates are not closed during the recovering time due to capacitance discharge effects. The signals are then not properly zeroed at the equilibrium or kept at too large an amplitude. In effect, we have shown in [16] that a non zero amplitude at equilibrium can be acceptable only if limited to a small percentage of the gate threshold voltage. Thus, when next data are injected in the circuit, these non-closed gates cause internal short-circuits leading, by the same process as previously, to false signals and to propagation of errors.

<sup>4</sup>The signal propagation flow is of most importance as it reconfigures the circuit schematic according to the calculation to perform. In some way, reversible circuits built of transmission gates are reconfigurable circuits, reprogrammed at each calculation step by their input data themselves, the data propagation flow also being the reprogramming data flow.

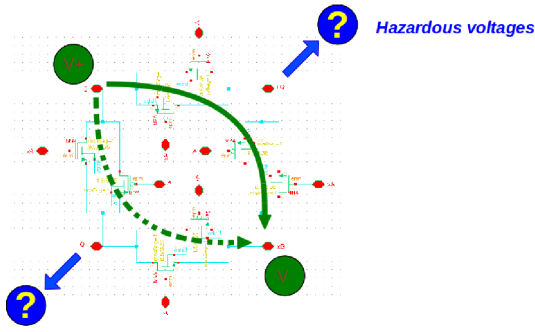
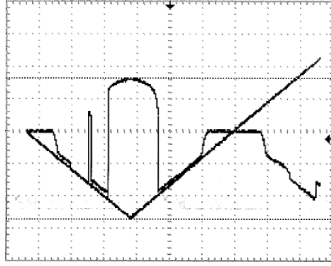


Fig. 10. Illustration of a temporary short circuit between two complementary signals in a Feynman gate, leading to hazardous output voltages.



CH1 500mV CH2 500mV M 2.5s

Fig. 11. Example of a bad output signal, triggered by a deliberately temporary short circuit at the input of the H264/AVC encoder presented in [9].

### C. Reduction of the Output Delay – the Sum-Difference Block

Even if the previous results push to prefer – in general – parallel architecture in order to reduce the delays between signals, in some particular cases, the ripple-carry structure can give very good results, in particular when the signal of shortest path is routed at the next step of calculation to the longest path of next computation block.

A good example is found with the Sum-Difference circuit. This computation block is composed of a Cuccaro adder cascaded to a Cuccaro subtractor. Its quantum diagram is presented in Fig. 12 where at each step of calculation, the data flow is expressed.

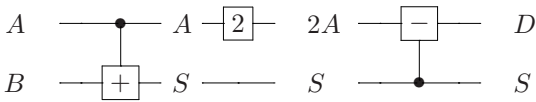


Fig. 12. Quantum diagram of the Sum-Difference block.

In this specific circuit, a first Cuccaro adder provides  $S = A + B$  and  $A$  as outputs.

If we want the computation to be done both lossless and the output to provide a full size representation, the input width has to be one bit smaller than the output width. Then, the most significant input bits (MSB) of the Cuccaro adder have to be fixed to zero. Therefore, at next calculation step,  $A$  is multiplied by two, which correspond to a bit shift – where the least significant bit (LSB) introduced is zero. Then in order to avoid extra garbage line, the  $A$ 's MSB can be used as the LSB while shifting all the other bits to the following order.

In the first computation step – i.e. the addition step –, the  $A$ 's

MSB is a direct connection from input to output whereas the  $A$ 's LSB has to wait for the last step of computation. When the data are sent to the next computation step – i.e. the subtraction step –, the MSB which was previously the LSB has the shorter path to flow whereas the LSB which was previously the MSB has now to wait until the last computation step of the subtractor for its value to be fixed. At the outputs of the Sum-Difference circuit, the delay between  $A_0$  and  $A_{n-1}$  is now drastically reduced.

We come to the surprising but fortunate conclusion that the outputs of a Sum-Difference block – although composed of two Cuccaro adders – are better synchronized than the outputs of a Cuccaro adder alone. This has been experimentally verified for such blocks ranging from 4 bits to 6 bits while testing the physically implemented H264/AVC encoder [6], [10].

A layout of a 5 bits Sum-Difference circuit is shown in Fig. 13. Between the two Cuccaro blocks, the metallic connections having the function of a permutation block are visible, thus performing the  $\times 2$  step. Two vertical wires allow connecting the output MSB of signal  $A$  coming from the adder to the LSB of the signal  $2A$  entering the subtractor.

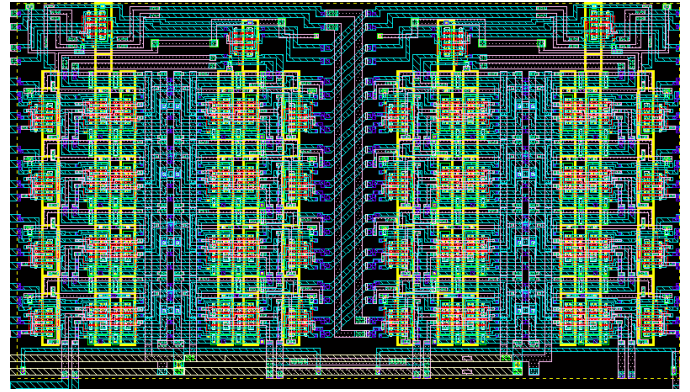


Fig. 13. Cadence Layout of the Sum-Difference circuit built from two 5 bits Cuccaro adders.

The Sum-Difference blocks allow to drastically reduce the delays between the outputs, facilitating the synchronization of data during the computation flow. As the delays are reduced, the output signals are kept in good symmetry, assuring a good positioning of the data in time.

This property of the Sum-Difference block is much appreciated, as this circuit is an important building block for more complex circuits; e.g. the reversible Hadamard transform [10] consists merely of four such blocks (Fig. 14):

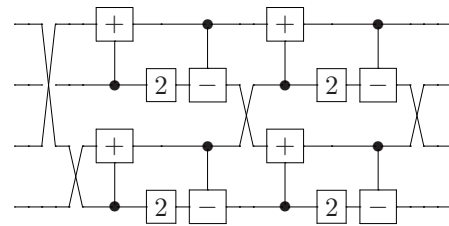


Fig. 14. Full quantum diagram of the Walsh-Hadamard  $4 \times 4$  matrix.

## 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 allow bringing information principally on impedance and voltage bias impact on calculation whereas adiabatic calculation is the normal functioning of the circuit.

In a first step, the experimental conditions will be presented. In a second step, some measurements performed at the outputs and obtained both in direct and reverse calculations will be exposed.

### A. Experimental Conditions

As far as current measurements are not concerned<sup>5</sup>, the electronics is very basic and cheap apparatus are sufficient, as the studied voltages are of the order of magnitude of 1 V and the highest reachable frequencies are of the order of megahertz for the 350 nm technology.

Then, for biasing and DC voltages a simple dual power supply Delta Elektronika has been used. All, the substrate and NWELL biases as well as input constant biases have been checked using a *Flucke 175* multimeter in DC mode.

In order to apply the necessary adiabatic pulses, an electronic module allowing the generation of 4 types of triangular pulses have been designed on purpose; its full schematic is presented in Fig. 15.

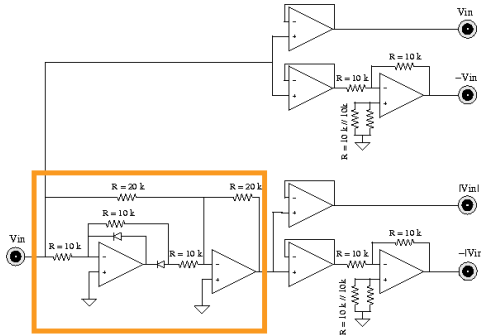


Fig. 15. Electronic schematic of the adiabatic signal generator module embedding an accurate full-wave rectifier (framed).

It is composed of four circuits altogether connected at the same input: a single buffer (switching pulse), an inverter followed by a buffer (dual switching pulse) and a precision full-wave rectifier followed either by a buffer or by an inverter and a buffer (logic “1” and logic “0” respectively). This very simple thus very accurate circuit presents very low output impedances. The used 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. The four adiabatic signals are all built up from one single triangular pulse provided at the input of the adiabatic module by a function generator TTI-TG210. Let notice that the input signal is not necessarily triangular. Sinusoidal signals have also been tested [16].

<sup>5</sup>The simulated output currents are of the order of 100 nA, which is difficult to accurately measure in dynamic.

The output signals are:

- 1) A logic “1” ranging from  $(V_+ + V_-)/2$  up to +1 V and back,
- 2) A logic “0” ranging from  $(V_+ + V_-)/2$  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  $(V_+ + V_-)/2$ .

The adiabatic signals presented below were measured using an oscilloscope Tektronix TDS 210 with a classical x1 or x10 probe. The x1 probe can be represented as a 72 pF capacitance in parallel to a 1 M $\Omega$  resistor. The x10 probe can be represented as a 22 pF capacitance in parallel to a 10 M $\Omega$  resistor.

For all measurements performed, the voltages  $V_{DD}$  and  $V_{SS} = V_{GND}$  are kept constant to 1.2 V and -1.2 V respectively.

### B. Output Characteristics

#### 1) General characteristics:

Fig. 16 shows typical experimental measurements realized in forward (adder) computation on a 4 bits Cuccaro adder.

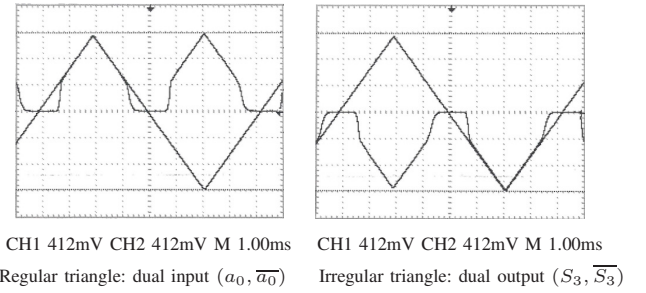


Fig. 16. Oscilloscope snap-view of the 4<sup>th</sup> dual bits ( $S_3, \overline{S_3}$ ) as a function of the dual switching input bits  $(a_0, \overline{a_0}) = (X, \overline{X})$ , obtained by direct (adder) calculation of  $S = c_{in} + a + b = 1 + 011X + 0110 = 11X\overline{X}$  (with  $c_{in} = 1$ ), measured when using adiabatic pulses on a 4 bits Cuccaro adder.

In Fig. 16, 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  $a_0$  ( $\overline{a_0} = -a_0$ ).

#### 2) Output voltage dependance on input bias:

The shape of these signals can easily be explained when considering the variation of an output voltage as a function of the input voltage bias. Fig. 17 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). The same curves are obtained for each output either in direct mode (adder) or reverse one (subtractor), except for the most significant bit of  $A$  (respectively  $a$ ) which is a direct connection between the input and the output (see Fig. 3).

First of all, the *pseudo*-threshold voltage  $V_{pT}$  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. This value corresponds to about half the threshold voltage of a single transistor. The difference is the result of



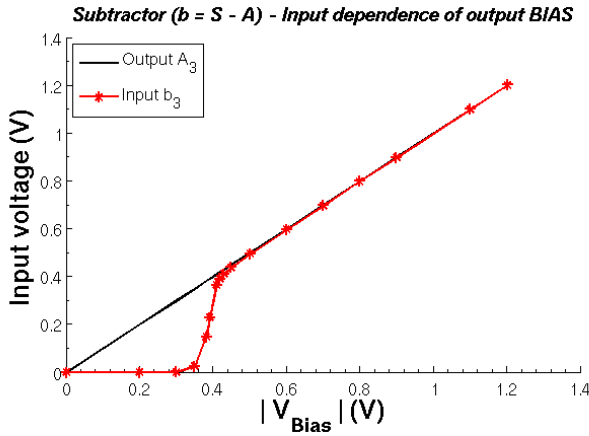


Fig. 17. Measurements of the input voltage dependence of bit  $b_2$  as a function of the output voltage bias of bit  $S_2$ , for the reverse (subtractor) calculation:  $b = S - A = 0110 - 0101 = 0001$ .

the two transconductances of the p-type and n-type transistors that are placed in parallel to form a transmission gate [15]. During the equilibrium phase, each command signal should not exceed this value otherwise, computation error will occur as previously explained.

A minimal voltage input as small as 420 mV can still be used to perform calculations with the Cuccaro adders. If so small input voltages are used, the output signals will be very steep and narrow triangles either positive (logic “1”) or negative (logic “0”), but still usable if very well synchronized.

The impact of the threshold voltage can be seen on the rising fronts of both dual output signals (Fig. 16). At the descending fronts, its impact is superimposed to capacitance effect coming from the probe that slows down the decrease of the signal to the mean value  $(V_+ + V_-)/2$  (here equal to zero), where all the transmission gates of the chip are reaching their equilibrium state (*open circuit* functionality).

### 3) Charge rebalancing:

But as shown in Fig. 18, during the transition to equilibrium, some more physical effects may occur. A first one is charge rebalancing.

Some charges stored in the circuits (wires and transistor’s channels) tend to evacuate during the falling edge of the signals. During this transition, they are supposed to flow back to the sources (charge recovering). But when the input voltages become too small, some charges are trapped locally into the circuits (parasitic capacitances), causing local non-zero potentials. Thus, some of the transmission gates may not be perfectly turned off as small voltages can remain. Nevertheless, if the calculation speed is not too fast (adiabaticity) these voltages are a lot smaller than the  $V_{pT}$  and cause no calculation error. But when comes the next signal, the trapped charges are freed up causing short peaks of extra currents and thus peaks of voltages at the very beginning of the raising edge. The higher the frequency, the larger the amplitude of the peaks.

This effect visualized by simulation is also observable experimentally. It is easy to verify either by simulation or by experiment, that the higher the probe capacitance, the larger the peak and the lower the maximum frequency experimentally reachable.

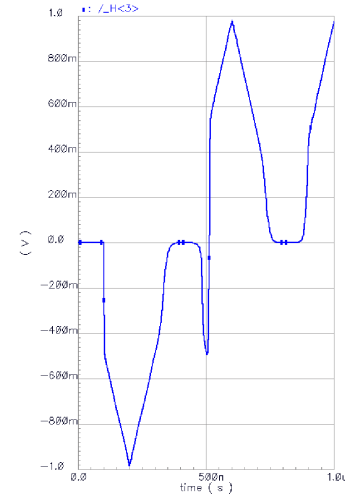


Fig. 18. Simulation of charge rebalancing effect causing parasitic voltage peaks at the very beginning of the raising edge of adiabatic signals. Frequency used is 5 MHz.

### 4) Output voltage drop and output impedance:

We successfully experimentally checked a large part of the truth table both in DC and when using adiabatic input pulses. 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 for 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 uncertainties.

By using the voltage divider technique, the output voltage without load resistor  $R$  is found to be  $V_{O0} = 1200mV$ . When adding a load resistor  $R = 11.12k\Omega$ , a modified output  $V_{O1} = 632mV$  is measured. The experimental output impedance of the Cuccaro adder both in forward and backward calculation is thus found to be  $Z_{out} = 5263\Omega \pm 5\%$ . In contrast, it is found to be close to 8 k $\Omega$  by Cadence Spectre<sup>©</sup> simulation. This is partly due to the impedance of the probe that is added in parallel to the load resistor  $R$  and also probably to differences between the Spectre<sup>©</sup> models used compared to the real devices.

### 5) Experimental delay:

As previously explained, a maximal delay is obtained between the two extreme output bits which are, for a 4 bits Cuccaro subtractor, the two outputs  $a_0$  and  $b_3$ .

The maximum measured delay, obtained for a frequency of 120 Hz is  $\tau = 91 \mu s$  (Fig. 19). Of course, this value is an upper bound, as two probes are connected at the outputs and have a negative impact on the signals by increasing the parasitic capacitance charge and discharge times.

This means that more than 23 Cuccaro adders may be cascaded before the total output delay reaches the  $PrcT$  limit of 25 %. To know the exact experimental maximal output delay of a reversible circuit is not so easy. In effect, each probe will have an impact on a specific output and the output, depending on the input data, will impact the calculation flow according to the configuration of the circuit for this specific

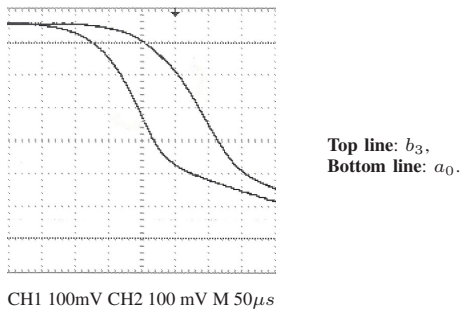


Fig. 19. Experimental maximum phase introduced during a computation step measured on a 4 bits Cuccaro subtractor. Two identical probes are used.

calculus. Placing some probes at each output will impact the whole circuit and degrade the computation performances.

Therefore, in future prototype circuits will be introduced some local measurement circuits allowing to reduce the impact of measurement.

#### 6) Test of Sum-Difference blocks:

A direct access to a 4 bits Sum-Difference blocks is provided within the H264/AVC video encoder presented in [6], [10]. The Sum-Difference block is used three times in this reversible video encoder, each one of these occurrences being of different width: one of 4 bits, one of 5 bits and one of 6 bits. Therefore, the video encoder allows validating at the same time the Sum-Difference function, its width extension and the cascade of two blocks of different depths. Fig. 20 presents a typical logic "0" output signal coming from a cascade of two Sum-Difference blocks.

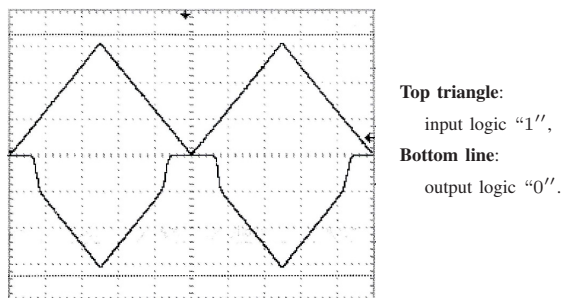


Fig. 20. Typical measured output of the H264/AVC video encoder presented in [6], [10]. The video encoder is cascading two Sum-Difference blocks. Frequency used is 200 Hz.

A better symmetry of the signal is found for the video encoder compared to the output signals coming from one Cuccaro adder alone (compare with Fig. 16). As only two Sum-Difference blocks are cascaded, the narrowing of the signal is not significant and could not be measured (difference inferior to the measurement uncertainties).

This anyway validates the use of such blocks in reversible architecture as the H264/AVC encoder is properly functioning.

## VI. CONCLUSION

This paper presented an overview of the performances of the V-shape ripple carry adder so called Cuccaro adder, for the 350 nm CMOS technology. This reversible (quantum) gate has been used in both directions: forward adder and backward

subtractor and blocks of width ranging from 3 to 6 bits have been used. Through this study, several important aspects of the adiabatic reversible computing have been discussed such as the impact of the delays.

Special attention has been paid to the application of two adders within the Sum-Difference block, a step stone to even more complex circuits.

The so far fabricated quantum-based reversible circuits give encouraging results as well as allowing building a base of knowledge and know how in physical implementation of reversible circuits.

## 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 *Invomec* division of *Imec v.z.w.* (Leuven, Belgium) and the *Europactice* consortium for the help with the prototyping of the chips.

## REFERENCES

- [1] A. De Vos, "Lossless computing," in *Proceedings of the IEEE Workshop on Signal Processing*, Poznań, Poland, 2003, pp. 7–14.
- [2] R. Feynman, "Quantum mechanical computer," *Optics News*, vol. 11, pp. 11–20, 1985.
- [3] R. Landauer, "Irreversibility and heat generation in the computing process," *IBM Journal of Research and Development*, vol. 5, no. 3, pp. 183–191, 1961.
- [4] C. Bennett, "Logical reversibility of computation," *IBM Journal of Research and Development*, vol. 17, no. 6, pp. 525–532, 1973.
- [5] T. Toffoli, "Reversible computing," in *Automata, Languages and Programming*, ser. Technical Memo MIT/LCS/TM-151, W. De Bakker and J. Van Leeuwen, Eds. Springer, Berlin: MIT Laboratory for Computer Science, 1980, pp. 632–644.
- [6] S. Burignat, "Reversible computation, a quantum-inspired low-consumption viable technology? invited tutorial," in *IEEE Conference, Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA 2011)*. Poznań, Poland: IEEE Conference, 29<sup>th</sup>–30<sup>th</sup> September 2011, pp. 9–10.
- [7] A. De Vos, "Reversible computer hardware," *Electronic Notes in Theoretical Computer Science*, vol. 253, no. 6, pp. 17–22, March 2010.
- [8] M. Skoneczny, Y. Van Rentergem, and A. De Vos, "Reversible fourier transform chip," in *Proceedings of the 15<sup>th</sup> International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES 2008)*, Poznań, Poland, June 2008 2008, pp. 281–286.
- [9] A. De Vos, S. Burignat, and M. Thomsen, "Reversible implementation of a discrete integer linear transformation," in *Proceedings of the 2<sup>nd</sup> Reversible Computation Workshop*, Bremen, 2010, pp. 107–110.
- [10] S. Burignat, K. Vermeirsch, A. De Vos, and M. Thomsen, "Garbageless reversible implementation of integer linear transformations," in *Proceedings of the 4<sup>th</sup> Reversible Computation Workshop*, Copenhagen, Denmark, July 2<sup>nd</sup>–3<sup>rd</sup> 2012, pp. 187–197.
- [11] S. Cuccaro, T. Draper, D. Moulton, and S. Kutin, "A new quantum ripple-carry addition circuit," in *Proceedings of the 8<sup>th</sup> Workshop on Quantum Information Processing*, Cambridge, June 2005, p. 9 pages.
- [12] V. Vedral, A. Barenco, and A. Ekert, "Quantum networks for elementary arithmetic operations," *Physical Review A*, vol. 54, pp. 147–153, 1996.
- [13] E. Fredkin and T. Toffoli, "Conservative logic," *International Journal of Theoretical Physics*, vol. 21, pp. 219–253, 2004.
- [14] S. Burignat and A. De Vos, "Test of a majority-based reversible (quantum) 4 bits ripple-carry adder in adiabatic calculation," in *Proceedings of the 18<sup>th</sup> International Conference "Mixed Design of Integrated Circuits and Systems" (MIXDES 2011)*, Gliwice, Poland, 16–18 June 2011, pp. 368–373.
- [15] A. De Vos, *Reversible Computing*. Wiley-VCH, Berlin, October 2010, ISBN-10: 3-527-40992-0 ISBN-13: 978-3-527-40992-1.
- [16] S. Burignat, M. Olczak, M. Klimczak, and A. De Vos, "Towards the limits of cascaded reversible (quantum-inspired) circuits," in *Reversible Computation, Proceedings*, ser. Lecture Notes in Computer Science, no. 7165. Gent, Belgium: Springer-Verlag, 2012.