Written exam for the 'Systèmes Numériques' course

2025-2026

It is better to answer some questions in depth than to address all of them superficially. The exam duration is 3.5 hours. You cannot use class material. Be clear & concise, and make sure your handwriting is legible. Number your sheets and write your name on each of them.

0 Common definitions & rules

0.1 Memoryless Circuits

All circuits in these problems are *combinatorial* (i.e. memory-free). Circuits in questions 2 and 4 will be presented explicitly in two equivalent ways, by their:

Schema A *Directed Acyclic Graph* (DAG) of its structure, properly drawn from inputs to outputs, and documented by the symbolic names from the net-list.

Net-List The *net-list* $v_1 = e_1; \ldots; v_n = e_n$ defines each variable v_i by an expression $e_i \in \{t \wedge t';\ t \vee t';\ t \oplus t'\}$ in which terms t and t' represent either: an input; a constant $0, 1$; a *previously defined* variable v_j with $j < i$.

0.2 Straight Line Programs

A *Single Static Assignment* (SSA) program is a list of instructions $v_1 = e_1, \ldots, v_n = e_n$. Each instruction defines the symbolic integer variable v_j by an expression $e_j \in \{t \wedge t';\ t \vee t';\ t \oplus t', t + t', t - t'\}$, in which t and t' represent terms which can be, either:

1. a ⟦w⟧ bits input variable x;
2. a ⟦w⟧ bits integer constant;
3. a previously defined variable v_i, with $i < j$.

An SSAS (SSA plus shifts) program is similarly defined, by adding $z^i(t)$ to our instruction set. It expresses both the logical up-shift $z^i(t) = 2^i t = (t \ll i)$ when $i \geq 0$, and the logical down-shift $z^i(t) = t/2^{-i} = (t \gg -i)$ when $i < 0$.

Unless mentioned otherwise, all SSA/SSAS programs are executed on a 16-bit machine (⟦w⟧ $= 16$), with a single input. Arithmetic operations are computed modulo 2^{16}, at the rate of one instruction per cycle. Hence, the length of the SSA/SSAS code is equal to the program time (number of execution cycles).

1 Parity

The *parity* $p = \mathcal{P}_n(x)$ of a n-bit binary word $x = x_0 \ldots x_{n-1} \in \mathbf{B}^n$ is the word $p \in \mathbf{B}^n$ whose k-th bit is the sum modulo 2 of the previous bits of x:

$$p_k = x_0 \oplus \ldots \oplus x_k = \sum_{j \leq k} x_j \pmod 2. \tag{1}$$

Question 1. (Minimal Size Parity Circuit)

Describe the minimal size parity circuit $\mathcal{P}_n$, with n bits of input $x_0 \dots x_{n-1}$, and n bits of output $p_0 \dots p_{n-1}$.

1. Show that output p_k in your circuit satisfies specification (1).
2. Show that it is the *unique minimal size* circuit for computing parity.
3. Analyze the combinatorial depth (maximal number of gates between inputs & outputs) of the circuit.

Question 2. (Minimal Depth Parity Circuit)

1. Show that the combinatorial depth q of any parity circuit $\mathcal{P}_n$ is bounded by $q \geq \lceil \log_2(n) \rceil$.
2. Construct a minimal depth parity circuit $\mathcal{P}_n$, for $n = 2^q$.
3. Analyze your circuit: number of gates, depth, area.

Question 3. (Optimal Parity Program)

1. Provide an SSA program for computing $\mathcal{P}_{16}$. Analyze its length.
2. Provide a minimal length SSAS program for computing the parity over 16 bits.
3. Give an SSA code for question 3.1 of minimal length. If you cannot prove it is of minimal length, give the best code you found.

2 Population Count

The number $\nu(x)$ of bits equal to 1 in $x = x_0 \dots x_{n-1} \in \mathbf{B}^n$ is $\nu(x) := x_0 + \dots + x_{n-1}$. In software, integer $\nu(x) \in \mathbb{N}$ is known as the population count, and also the sideways sum of x. A hardware circuit for solving the same problem is named a n bit parallel counter.

Question 4. (Parallel Counter Circuit)

A parallel counter has n bits of input $x_0 \dots x_{n-1} \in \mathbf{B}^n$ and m bits of output $s_0 \dots s_{m-1} \in \mathbf{B}^m$ which represent the population count $s = \nu(x) = \sum_{k<m} s_k 2^k$ in binary.

1. Express m in terms of n, and give the explicit values of m for $n = 15$ and $n = 16$.
2. Provide a lower bound on the depth of all n bit parallel counters.
3. Construct a n bit parallel counter, for arbitrary n.
4. Cleanly draw the schemas of your circuit for $n = 15$.
5. For arbitrary n, analyze your circuit: number of gates, depth, area.

Question 5. (Population Count Program)

1. Provide an efficient SSAS program for counting the population over 16 bits.
2. Generalize your program to an arbitrary number of bits n (with n a power of 2), assuming now that the underlying computer has n rather than just 16 bits.
3. Analyze the length of your generalized program. Is it minimal?

Question 6. (Sideways Sum in FPGA)

Our FPGA gates consist solely of 6:3 compressors ($\texttt{COMP}_6^3$), a.k.a. *lookup tables* (LUT), which map 6-bit inputs to 3-bit outputs. Wires are arbitrary.

Provide a circuit for computing the sideways sum over 36 bits. Make the circuit depth as small as possible, while keeping a reasonable number of LUTs.

Notes

- Cet examen est inspiré d'un ancien examen de Jean Vuillemin.
- Un élève a réussi à résoudre optimalement la question 6 **pendant** l'examen.