Logic Circuits and Signals
Hardware/Software Connection

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory
Dipartimento di Matematica e Informatica - Università di Catania, Italy
santoro@dmi.unict.it

L.S.M. Course

Logic Circuits

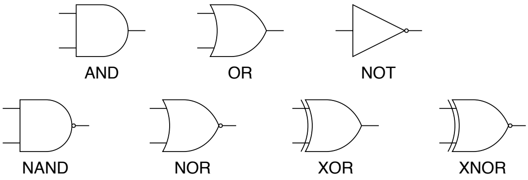
Basics of Logic Circuits

- Logic circuits are characterized by the fact that **voltage of wires** can assume only **two values**:
  - $0\ V$, corresponding to **Logic/bit-value "0"**
  - $+V_{DD}$, corresponding to **Logic/bit-value "1"**
- where V_{DD} is the power supply of the whole circuit, it may be $5\ V$, $3.3\ V$, $1.8\ V$

Logic Circuits Classification

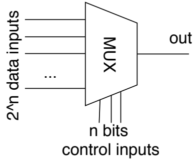
Combinatorial Circuits

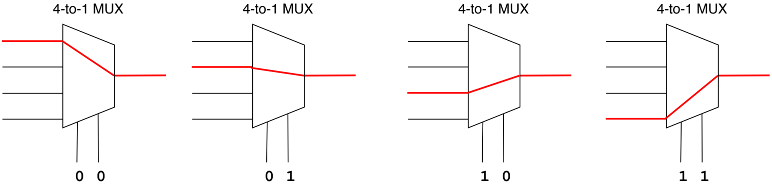
- They are logic circuits where the output is dependent only on the **current state** of the inputs
- **Logic gates** (AND, OR, NOT, etc.) are kind of combinatorial circuits
- Their behaviour is represented by a **truth table**



Multiplexers

- **Multiplexers** are combinatorial circuits that act as **signal switches**
- One **data input** at time is routed to the output on the basis of the values of **control input bits**





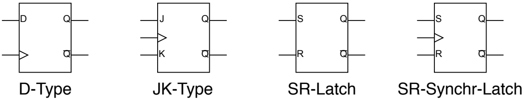
Logic Circuits Classification

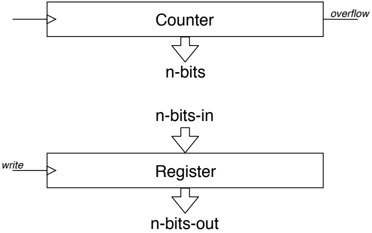
Sequential Circuits

- They are logic circuits where the output is dependent only on the **current and past state** of the inputs
- They are sensible to **variations** of the inputs
- They are logic circuits that have a **memory**
- **Flip-Flops** and their derivatives are kind of combinatorial circuits
- Their behaviour is represented by a **finite-state machine**
- In general, a **clock signal** drives their behaviour

Kind of Sequential Circuits

- Flip-flops
- Counters
- Registers





Logic Signals

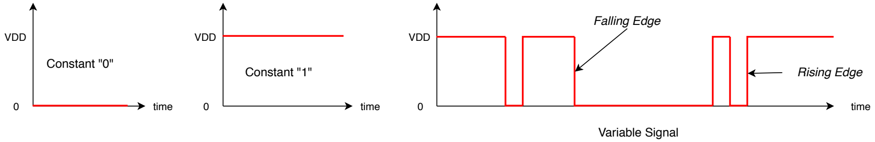
Logic Signals

Constant Signals

- They are logic states that does not vary in time

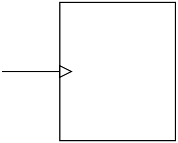
Variable Signals

- They are logic states that may vary in time
- They are featured by **edges**
  - **Falling Edge:** variation from “1” to “0”
  - **Rising Edge:** variation from “0” to “1”

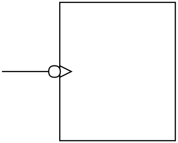


Edges and Sequential Circuits

- Sequential Circuits are sensible to **edges**
- Edge inputs are represented in circuits by “triangles”
- **simple triangle: rising edge**
- **circle + triangle: falling edge**



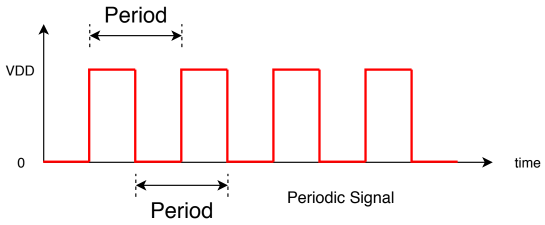
State change on **rising** edge



State change on **falling** edge

Periodic Signals

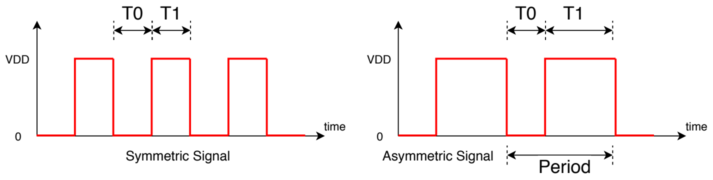
- They are kind of *variable signals* where the **time distance** between **two edges of the same time** is **constant**
- This distance is called **Period,** *P* and measured in seconds
- The **frequency**, computed as $f = \frac{1}{P}$, is the **number of "periods" per second** and is measured in **Hertz,** *Hz*



Periodic Signals

- Periodic signals can be:
  - **symmetric:** the time durations of state "0" and state "1" is the same and equal to $T_0 = T_1 = \frac{P}{2}$
  - **asymmetric:** the time durations of state "0" and state "1" is different $T_0 \neq T_1$
- The "asymmetry" is called **duty cycle** and is the **percentage of period in which the signal is "1"**

$$DC = \frac{T_1}{T_0 + T_1} 100 = \frac{T_1}{P} 100$$



Time and Frequency Measures

Time Measures

- **milliseconds:** $1\ ms = 10^{-3}\ s$
- **microseconds:** $1\ \mu s = 10^{-6}\ s$
- **nanoseconds:** $1\ ns = 10^{-9}\ s$
- **picoseconds:** $1\ ps = 10^{-12}\ s$

Frequency Measures

- **KiloHertz:** $1\ KHz = 10^{3}\ Hz$
- **MegaHertz:** $1\ MHz = 10^{6}\ Hz$
- **GigaHertz:** $1\ GHz = 10^{9}\ Hz$
- **TeraHertz:** $1\ THz = 10^{12}\ Hz$

Hardware/Software Connection

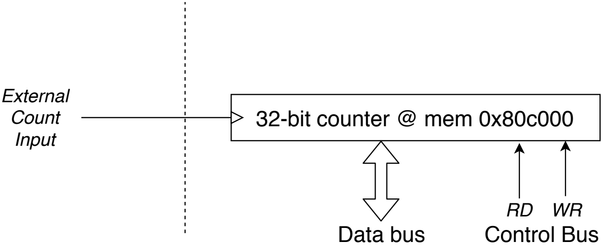
Hardware/Software Connection

Hardware/Software Connection

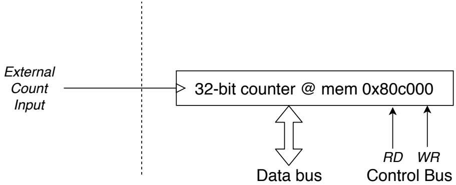
- Circuits of a MCU are “connected” to the software through **registers/counters**
- A **register** is mapped in memory at a **known memory address**
- Hardware/software interaction is performed by reading/writing at that **memory address**

An Example: Piece Counter

- A “presence sensor” generates a pulse each time a “piece” is identified
- The pulse is connected to an **external counter** MCU input
- Each time the sensor generates a pulse, the counter increments (in hardware)



Hardware/Software Connection



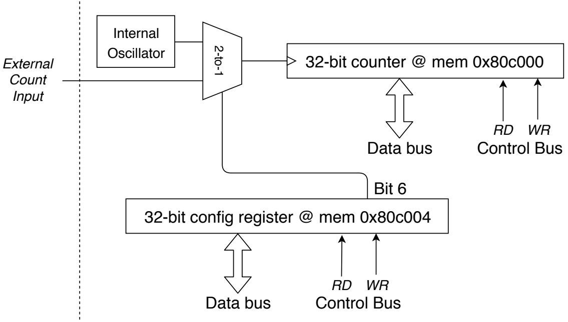
Reading/Writing the Counter

```
...
int32_t * counter_prt = (int32_t *)0x80c000;
...

/* clearing the counter */
*counter_ptr = 0;
...

/* printing the counter */
printf("Counter value %d\n", *counter_ptr);
```

Hardware/Software Connection



Configuring the Counter

```
int32_t * config_prt = (int32_t *)0x80c004;

/* counter input from Internal Oscillator */
*config_ptr = 0;

/* counter input from External Counter Input */
*config_ptr = 0x40;
```

Logic Circuits and Signals
Hardware/Software Connection

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory
Dipartimento di Matematica e Informatica - Università di Catania, Italy
santoro@dmi.unict.it

L.S.M. Course