**Marcelo Oliveira Camponez, PHD**

**@marcelocamponez**

[marcelo.camponez@hotmail.com](mailto:marcelo.camponez@hotmail.com)

**Brazil**

1. **PROCESSOR ARCHITECTURE**

Accumulates tOR (AC)

3

Program

Counter (PC

Program

Memory

ROM

(16 x 8 bits)

4

*Clock*

Address

Control

Unit

*Clock*

Instruction

Registerr (IR)

8

Instruction

*Clock*

Instruction

8

Operation

*Display*

4

4

4

MUX

Z

C

Date

Memory

(16 x 4 bits)

4

*Overflow*

Zero

4

0

1

Date

Addressing Mode

4

Addressing / Constant

Read /Write

*Chip*

*Clock*

*Reset*

*Reset*

*Chip Selection*

*Load*

4

Address

*Enable*

Zero

*Overflow*

Direction

ULA

Figure 1 - Processor Architecture

# INSTRUCTIONS DESCRIPTION

**MOVES X, E**

­­­­­­­­­­­­­­Description: Performs the move (copy) operation of a constant X, if E is equal to 0, or from the contents of the X address of the RAM, to the AC register.

Limits: 0 ≤ X ≤ 15 (4 bits)

E = 0 (OP = X) or E = 1 (OP = M(X))

Operation: AC  OP

Affected bits: Z and O

Encoding: 000E XXXX

**ADD X, E**

­­­­­­­­­­­­­­Description: Performs the add-on operation between AC and a constant X, if E is equal to 0, or the addition between AC and the contents of the X address of the RAM. The result will always be stored in AC.

Limits: 0 ≤ X ≤ 15 (4 bits)

E = 0 (OP = X) or E = 1 (OP = M(X))

Operation: AC  AC + OP

Affected bits: Z and O

Encoding: 001E XXXX

**SUB X, E**

­­­­­­­­­­­­­­Description: Performs the subtraction operation between AC and a constant X, if E is equal to 0, or the subtraction between AC and the contents of address X of RAM. The result will always be stored in AC expressed in 2´s complements.

Limits: 0 ≤ X ≤ 15 (4 bits)

E = 0 (OP = X) or E = 1 (OP = M(X))

Operation: AC  AC - OP

Affected bits: Z and O

Encoding: 010E XXXX

**OR X, E**

­­­­­­­­­­­­­­Description: Performs the logical operation 'OR' (or) between AC and a constant X, if E is equal to 0, or between AC and the contents of the X address of the RAM. The result will always be stored in AC.

Limits: 0 ≤ X ≤ 15 (4 bits)

E = 0 (OP = X) or E = 1 (OP = M(X))

Operation: AC  AC or OP

Affected bits: Z and O

Encoding: 011E XXXX

**XOR X, E**

­­­­­­­­­­­­­­Description: Performs the logical operation 'OR EXCLUSIVE' (xor) of a constant X, if E is equal to 0, or the contents of the X address of the RAM. The result will always be stored in AC.

Limits: 0 ≤ X ≤ 15 (4 bits)

E = 0 (OP = X) or E = 1 (OP = M(X))

Operation: AC  AC xor OP

Affected bits: Z and O

Encoding: 100E XXXX

**SHIFT X, D**

­­­­­­­­­­­­­­Description: Performs a shift to the right or left of the Q bits accumulator. If D is equal to 0 the offset will be left, otherwise to the right. The result will always be stored in AC.

Limits: 0 ≤ Q ≤ 15 (4 bits)

D = 0 (left shift) or D = 1 (right shift)

Operation: AC  AC << Q, for D = 0 or

AC  AC >> Q, for D = 1

Affected bits: Z

Encoding: 101 D QQQQ

**JMP X, E**

­­­­­­­­­­­­­­Description: Makes a conditional program jump so that the next executed statement is the one allocated at address X or address already pointed out by the PC, depending on the condition established. The jump is made by changing the contents of the PC by X, if the condition is true. The jump can occur if AC is equal to zero (S = 0) or if AC is non-negative (S = 1).

Limits: 0 ≤ X ≤ 15 (4 bits)

S = 0 (jumps if AC = 0) or S = 1 (jumps if AC ≥ 0)

Operation: PC  X, for S = 0 and if AC = 0 (i.e. Z = 1) or for S = 1 and O = 1 (i.e., overflow occurred)

Affected bits: none

Encoding: 110 S XXXX

**GOTO X**

­­­­­­­­­­­­­­Description: Makes an unconditional program jump so that the next executed statement is the one allocated to the X address of the ROM memory. The jump is made by changing the contents of the PC by X.

Limits: 0 ≤ X ≤ 15 (4 bits)

Operation : PC

Affected bits: none

Encoding: 1110 XXXX

**STORE X**

­­­­­­­­­­­­­­Description: Performs the storage operation of the contents of the AC register to the X address of the RAM.

Limits: 0 ≤ X ≤ 15 (4 bits)

Operation: M( X)

Affected bits: none

Encoding: 1111 XXXX