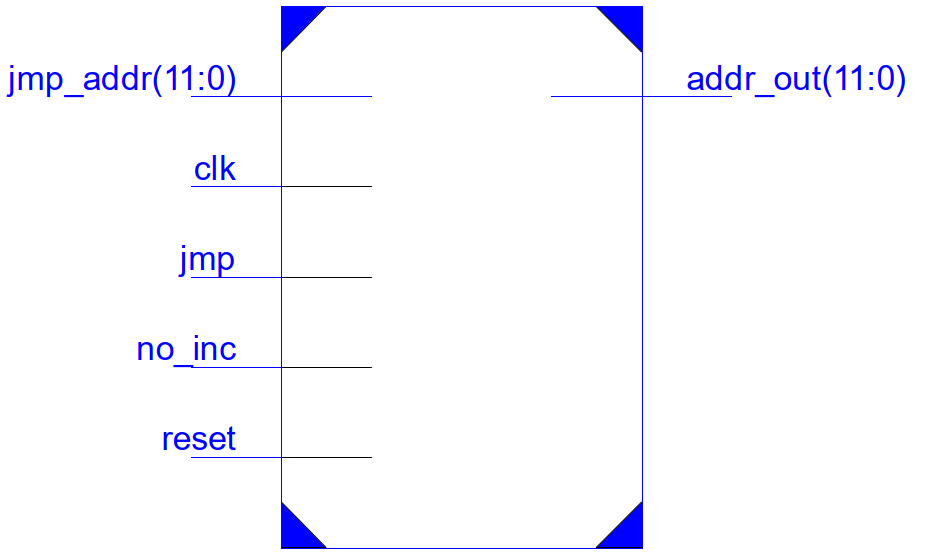
**CHAPTER 3**

# **Modules**

## **3.1 Program\_Counter**



Program Counter is a 12-bit register which is sensitive to positive edge of the clock pulse.

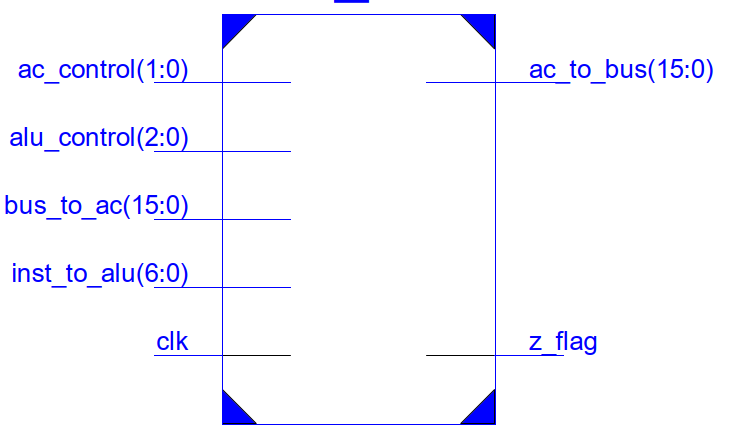
|  |  |
| --- | --- |
| **Control Signals** | **Description** |
| jmp | if ( jmp ) : addr\_out = jmp\_addr |
| no\_inc | If ( no\_inc ) : addr\_out will not increment |
| reset | If ( reset ) : addr\_out set to zero |

Table 3.1: Control signals of PC

In default configuration **jmp, no\_inc** & reset signals are set to zero. Therefore, at every positive edge of the clock pulse **addr\_out** will be incremented by 1. i.e it is pointed to next instruction to be fetched.

When type – M2 instructions use **jmp** signal to perform branching. If you want to set the processor into idle state **no\_inc** signal should be set. **reset** pin is used to restart the program.

3.2 AC\_ALU



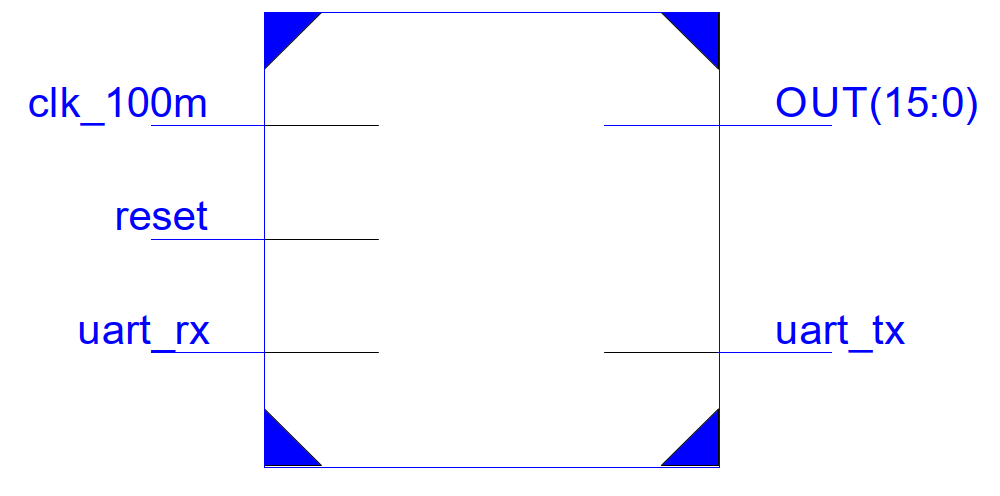
This module includes Accumulator (AC) and Arithmetic and Logic Unit (ALU). ALU can perform 4 basic arithmetic operations: Addition, Subtraction, Division and Multiplication. AC can be used as a shift register which can shift 0-15 positions left/right (But it also done through ALU) . **Z\_flag** set whenever AC is zero.

Main bus is connected to module through **bus\_to\_ac** port and constants from instruction **(CONST/N)** is connected via **inst\_to\_alu** port. AC’s value is always given out through **ac\_to\_bus** port.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **control Signal** | | **Selection** | **Operation** | **Description** |
| alu\_control (2:0) | | 000 | Addition | Data in Main bus and Constant from instruction is added to AC and stored back in AC |
| 001 | Subtraction | Data in Main bus and Constant from instruction is added together and substracted from AC and stored back in AC |
| 010 | Multiplication | Data in Main bus and Constant from instruction is added together and multiplied by AC and stored back in AC |
| 011 | Division | Data in AC is divided by the summation of Main bus value and Constant |
| 100 | Shift Left | Data in AC is shifted to N bits left (N is given as Constant) |
| 101 | Shift Right | Data in AC is shifted to N bits right ((N is given as Constant)) |
| ac\_control (1:0) | ac\_control[0] | 0 | AC = ALU output | write ALU output to AC |
| 1 | AC = Main bus input | write data in main bus to AC |
| ac\_control[1] | 0 | AC write disabled | don't write input data to AC register |
| 1 | AC write enabled | write input data to AC register |

Table 3.: control signals of AC\_ALU

3.3 Processor\_top\_module



This module wraps all sub modules (processor, RAM and UART transceiver) together. **clk\_100m** pin is connected to internal built-in clock of FPGA and internally divided into 6.25MHz clock because timing analysis indicates that maximum clock speed we can use is 8.602MHz. It is done by incrementing clkreg a 4-bit register at positive edge of clk\_100m and by taking 4th bit as divided clock signal. **reset** pin is connected to reset button in order to restart the program. Data communication between PC and DPUT Processor is done through **uart\_tx** and **uart\_rx** pins of this module. These modules are connected to UART interface of the development board. **OUT** is a 16-bit port which is connected to 8 LEDs of the development board. it can be connected to any internal wire/register and can be used to observe states of connected wire/registers. By default, it is connected to wire **ac\_to\_bus**. [Appendix I](https://github.com/damithkawshan/Image-Downsampling-Processor/blob/master/Documentation/top_module_inside.pdf) shows the internal wiring and sub modules of this module.