Our source code consists of eight packages, and each of them realizes different function:

|  |  |
| --- | --- |
| Name of Packages | Function |
| edu.gwu.computerarchitecture.alu | Realize Arithmetic Logical Unit |
| edu.gwu.computerarchitecture.cache | Realize cache operation |
| edu.gwu.computerarchitecture.cpu | Realize CPU operation |
| edu.gwu.computerarchitecture.entity | Define register and memory |
| edu.gwu.computerarchitecture.main | Entrance of the entire program |
| edu.gwu.computerarchitecture.operation | Define instructions, and Realize fetch, decode, writeback |
| edu.gwu.computerarchitecture.operation.execute | Execution of each instructions |
| edu.gwu.computerarchitecture.ui | Design the UI of the program |

1. [Class](app:ds:class) [diagram](app:ds:diagram)

ALU

The class of “ALU” realizes the function of “Arithmetic Logical Unit”, including several [attribute](app:ds:attribute)s and methods:

TX1 and TX2 are temporary registers used to store operands.

result1 and result2 compose the output of ALU.

equal\_flag is the result of TER instruction.

overflow is the flag of overflow of calculation.

divByZero is the flag for DIV instruction.

setTX1() and setTX2(): setting the value of TX1 and TX2.

setOverFlow(): setting the value of overflow.

setDivByZero():setting the value of divByZero.

add(): [add](app:ds:add) [operation](app:ds:operation).

sub(): [subtraction](app:ds:subtraction)[operation](app:ds:operation).

mul(): [multiplication](app:ds:multiplication) [operation](app:ds:operation).

div(): [division](app:ds:division) [operation](app:ds:operation).

equal(): tell if TX1 equals to TX2.

and(), or() and not() are for logical calculation(&, |, ~).