WORK IN PROGRESS

FEATURES

- Registers :

> - *16 bit register 0 - 15*
> - *Value range:(-32767 - 32767)*

- Instructions : 32 bits * 250 = 1 KB
- All instructions must be in small caps
- All labels must be in caps
- Flag register :

> **7 6 5 4 3 2 1 0**
> NA GREATER THAN EQUAL TO LESS THAN OVERFLOW SIGN CARRY ZERO

INSTRUCTION FORMAT

ldi - Load Immediate

Ex: ldi r0 8 ---> r0 = 8

opcode:

- *8 bits [31-24] ---> **00001111***

desitnation register:

- *4 bits [23-20]*

data:

- *16 bits signed [15:0]*
- *max: 32767*
- *min: -32767*

add - Add two registers and store in destination register

Ex: add r0 r1 r2 ---> r0 = r1 + r2

opcode:

- *8 bits [31:24] ---> **11110000***

desitnation register:

- *4 bits [23:20]*

source registers:

- *4 bits [19:16]*
- *4 bits [15:12]*

sub - Subtract two registers and store in destination register

Ex: sub r0 r1 r2 ---> r0 = r1 - r2

opcode:

- *8 bits [31:24] ---> **11110001***

desitnation register:

- *4 bits [23:20]*

source registers:

- *4 bits [19:16]*
- *4 bits [15:12]*

mul - Multiply two registers and store in desitnation register

Ex: mul r0 r1 r2 ---> r0 = r1*r2

opcode:

- *8 bits [31:24] ---> **11110010***

desitnation register:

- *4 bits [23:20]*

source registers:

- *4 bits [19:16]*
- *4 bits [15:12]*

shr - Shift the register value to right by given amount and store in desitnation register

> ***Ex: shr r0 r1 2 ---> r0 = r1>>2***
>
> *opcode:*
>
> - *8 bits [31:24] ---> **11110011***
>
> *desitnation register:*
>
> - *4 bits [23:20]*
>
> *source register:*
>
> - *4 bits [19:16]*
>
> *immidiate data:*
>
> - *16 bits [15:0]*

shl - Shift the register value to left by given amount and store in desitnation register

Ex: shl r0 r1 2 ---> r0 = r1<<2

opcode:

- *8 bits [31:24] ---> **11110100***

desitnation register:

- *4 bits [23:20]*

source register:

- *4 bits [19:16]*

immidiate data:

- *16 bits [15:0]*