**LAPORAN UAS Digital System Design**

**8 BIT CPU**



**Julando Omar**

**00000027458**

**2020**

**Teknik Komputer**

**Universitas Multimedia Nusantara**

**BAB I**

**Spesifikasi Sistem**

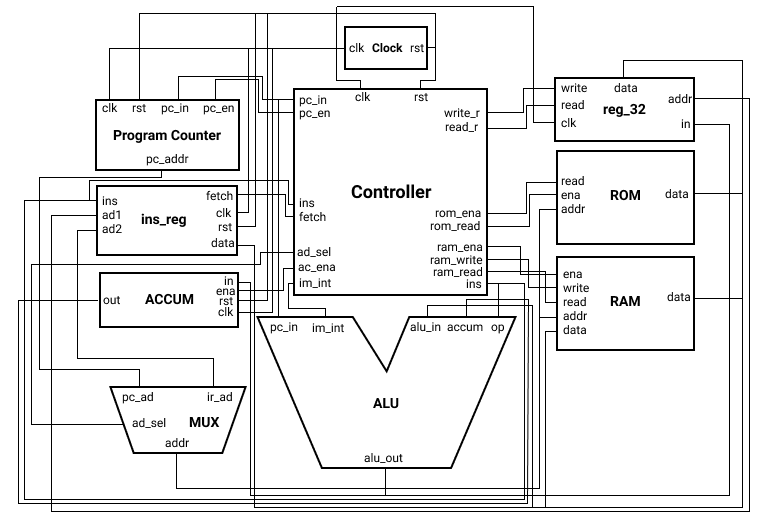
Sistem yang kami rancang memiliki spesifikasi sebagai berikut:

* CPU yang menggunakan arsitektur RISC
* Mendukung pemrosesan 8 Bit
* 8 Bit terdiri dari 4 bit Opcode dan 4 Bit Operand
* Terdapat 12 Operasi
* Register sebanyak 16 Byte
* ROM dan RAM sebanyak 256 Byte

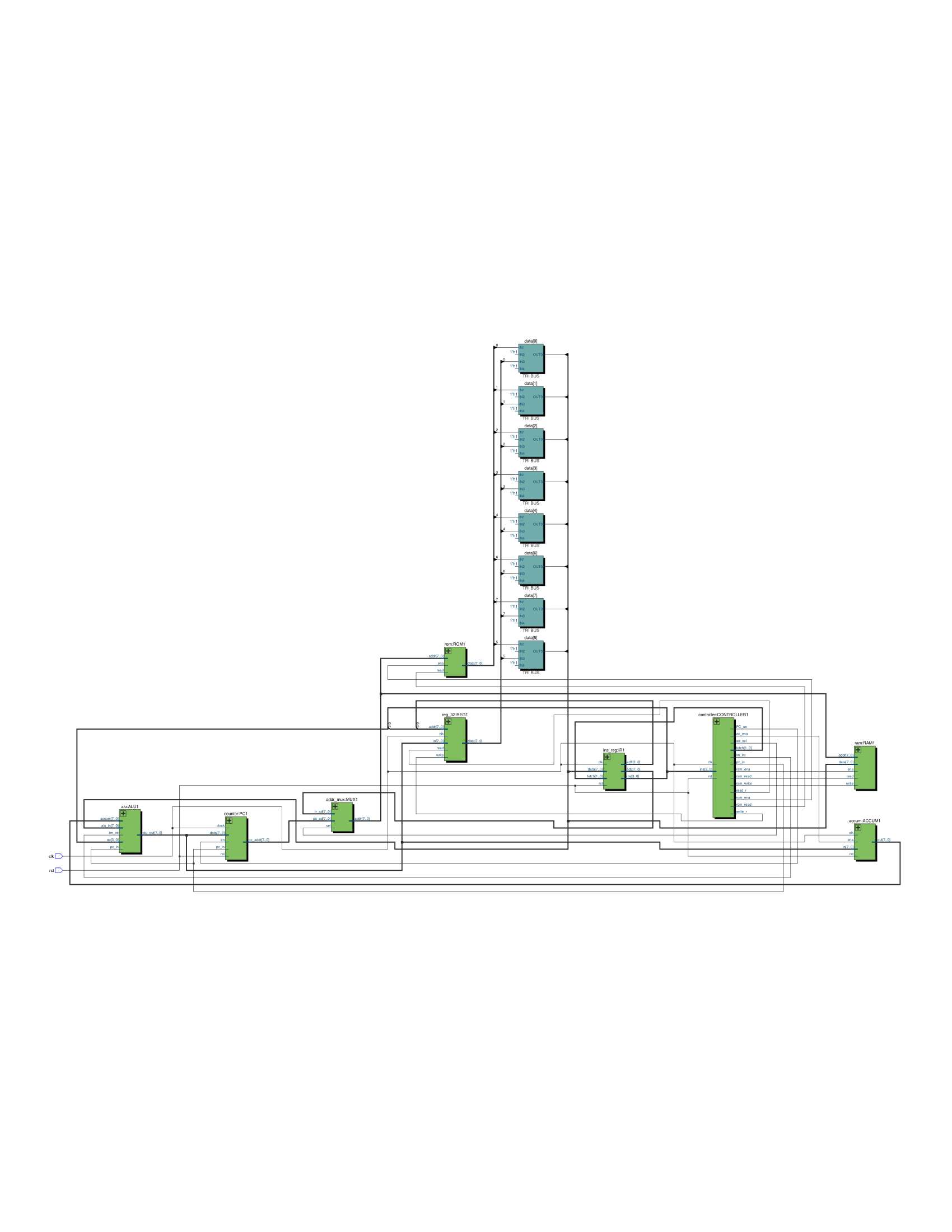
Sistem CPU yang kami rancang terdiri dari beberapa *block,* yaitu:

* Clock, yang digunakan untuk mengenerasi signal *clock* atau signal *reset*;
* Program Counter, yang digunakan untuk memberikan address untuk dieksekusi;
* ALU, digunakan untuk proses perhitungan aritmatik;
* Control Unit (Controller), sebagai otak dari CPU, digunakan untuk mentranslate instruksi dan memberikan sinyal yang tepat ke *block-block* yang lainnya.
* Register, digunakan untuk menampung data dari hasil perhitungan ataupun data yang diambil dari ROM atau RAM
* Accumulator, digunakna untuk menampung data sementara hasil dari perhitungan alu
* Instruction register, digunakan untuk *decode* instruksi, apakah instruksi diambil dari ROM atau dari RAM
* Address Multiplexer, digunakan untuk memilih antara address PC atau address dari instruksi yang digunakan
* ROM, digunakan untuk menyimpan program instruksi yang akan dijalankan
* RAM, digunakan untuk menyimpan hasil data dari register.

Sistem CPU yang kami rancang memiliki diagram skematik sebagai berikut:



Gambar 1 Skematik CPU

CPU kami juga memiliki RTL sebagai berikut

Gambar 2 RTL

**BAB II**

**Instruksi**

Pada CPU 8 Bit instruksi dibagi menjadi:

* Short Instruction

OP Code R\_ADDR

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |

* Long Instruction

OP Code R\_ADDR ROM/RAM\_ADDR

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

CPU kami memiliki daftar instruksi sebagai berikut

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instruksi | Binary | Deskripsi | Type | Operasi |
| NOP | 0000 | **N**o **Op**eration | Short |  |
| LDO | 0001 | **L**oa**d** Content From R**O**M into REG | Long | ROM [ROM\_ADDR] →  REG [R\_ADDR] |
| LDA | 0010 | **L**oa**d** Content From R**A**M into REG | Long | RAM [RAM\_ADDR] →  REG [R\_ADDR] |
| STO | 0011 | **Sto**re Register data to RAM | Long | REG [R\_ADDR] →  RAM [RAM\_ADDR] |
| PRE | 0100 | **Pre**fetch data from REG to Accumulator | Short | REG [R\_ADDR] →  ACCUM |
| ADD | 0101 | **Add** content from REG with accumulator | Short | REG [R\_ADDR] + ACCUM →  ACCUM |
| LDM | 0110 | **L**oa**d** data from ACCU**M** to REG | Short | ACCUM → REG[R\_ADDR] |
| ADN | 0111 | **Ad**d I**n**teger to ACCUM | Short | R\_ADDR → ACCUM |
| INC | 1000 | **Inc**rease ACCUM count | short | ACCUM + 1 |
| Instruksi | Binary | Deskripsi | Type | Operasi |
| DEC | 1001 | **Dec**rement ACCUM Count | Short | ACCUM - 1 |
| JMP | 1010 | **J**u**mp** to specific address | Long | ROM\_ADDR → PC |
| HLT | 1111 | **H**a**lt** the operation | Short |  |

**BAB III**

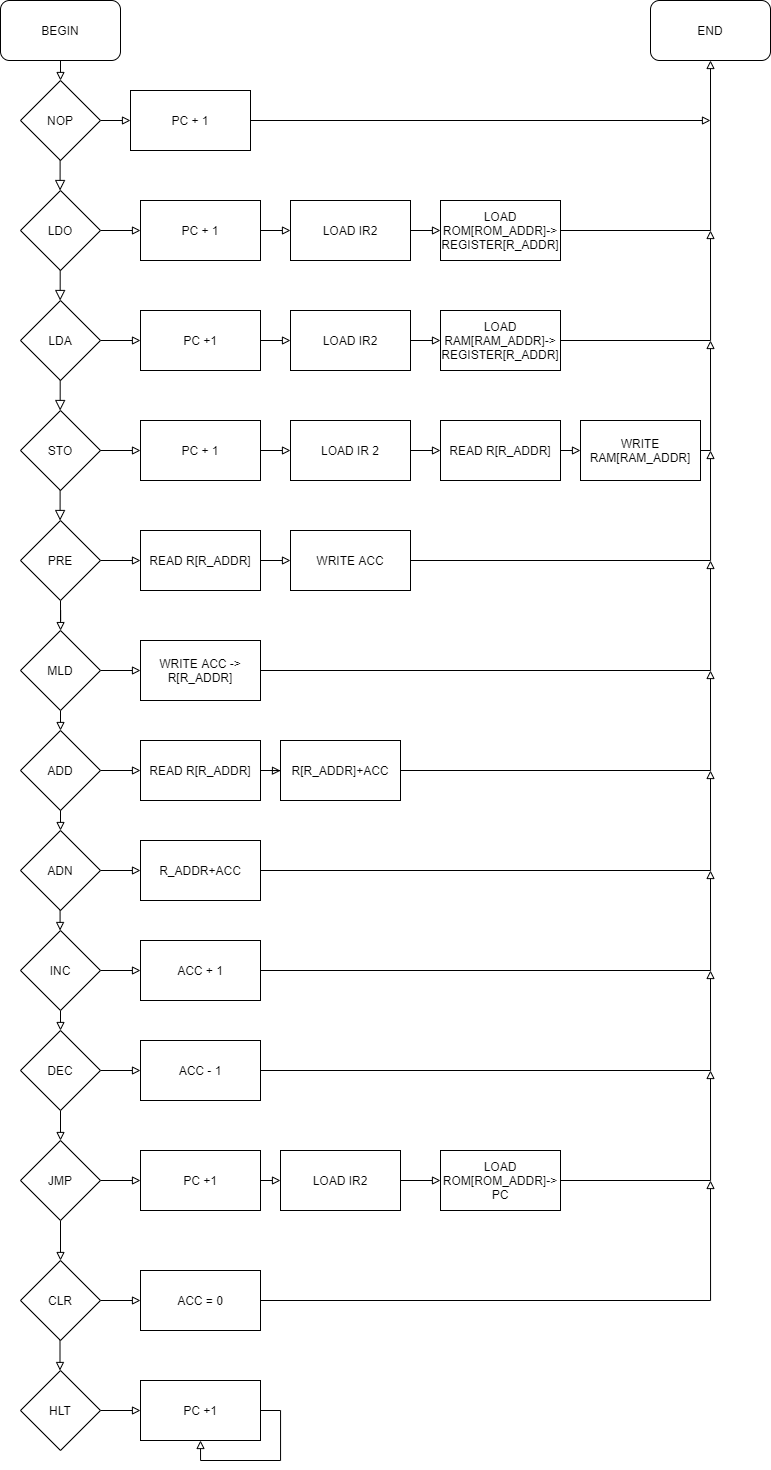
**SYSTEM FLOW**

CPU kami memiliki system flow secara singkat sebagai berikut

Diagram, schematic

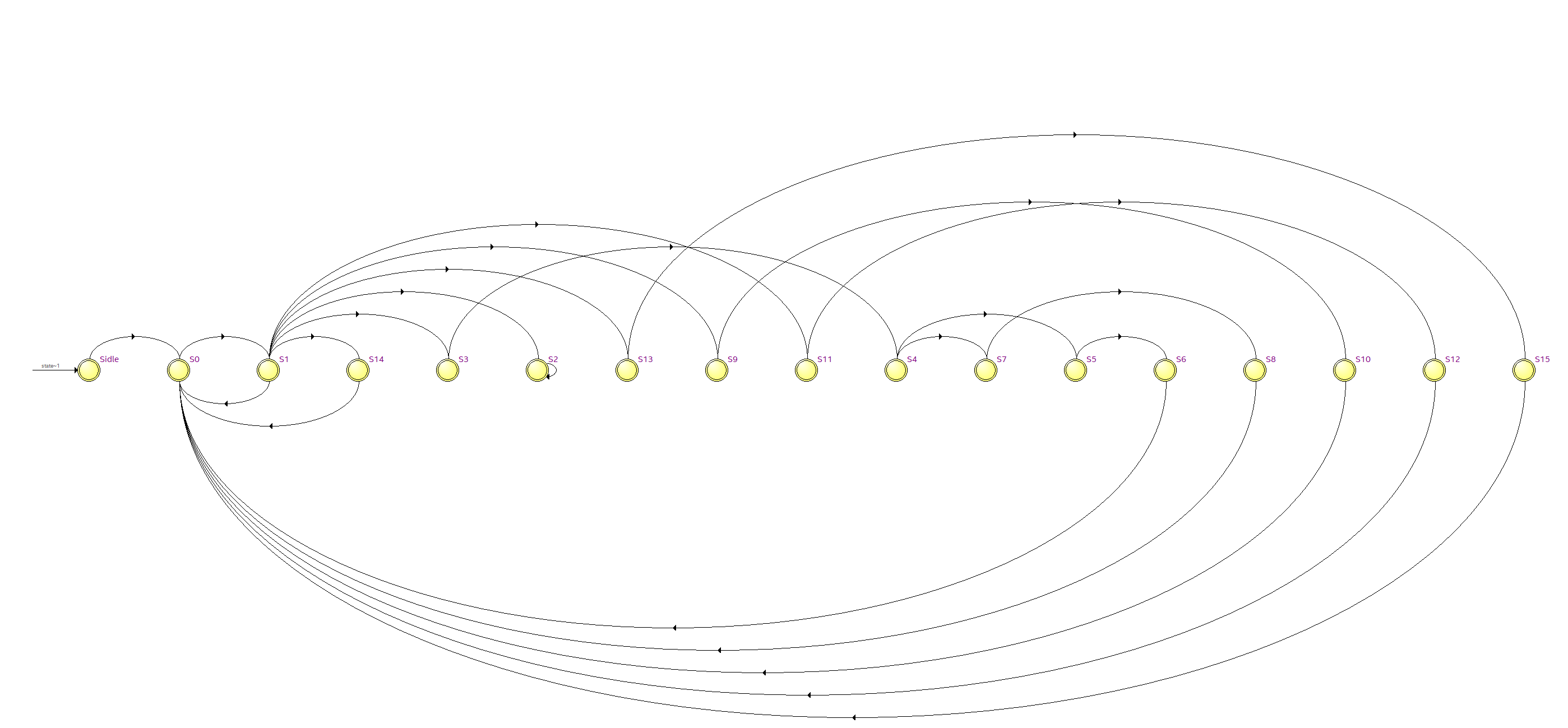
Description automatically generated

Gambar Flowchart CPU



Gambar Flowchart Instruction

CPU kami memiliki Finite State Machine Sebagai berikut



Gambar Finite State Machine CPU

State-state tersebut memiliki penjelasan sebagai berikut:

|  |  |  |
| --- | --- | --- |
| State | Deskripsi | Keterangan |
| S0 | PC + 1 | Melakukan penambahan PC sebanyak satu decimal |
| S1 | Fetch IR | Mengambil instruksi dari ROM |
| S2 | HALT | Mensetop semua operasi dan melakukan *Looping* di S2 |
| S3 | Fetch IR 2 | Mengambil instruksi Nomor 2 dari ROM |
| S4 | PC +1 | Melakukan penambahan PC sebanyak satu decimal |
| S5 | ROM/RAM to REG | Melakukan perpindahan data dari RAM atau ROM ke Register |
| S6 | Protect | Melakukan protect data agar tidak terjadi override data |
| S7 | Read Register | Fase pertama dari instruksi STO |
| S8 | Write RAM | Fase kedua dari instruksi STO |
|  |  |  |

**BAB IV**

**ARISTEKTUR MICRO**

CPU kami memiliki arsitektur mikro sebagai berikut

* ALU