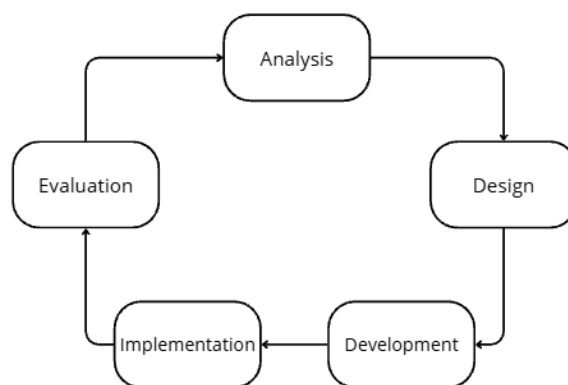


BAB III

METODE PENELITIAN

Metode penelitian yang akan digunakan dalam penelitian ini adalah *Desain and Development* (D&D). Menurut Nurafifah & Purnamasari. (2022) metode *Design and Development* (D&D) merupakan penelitian yang mendesain dan mengembangkan agar lebih baik lagi, dalam metode D&D terdapat dua kategori yaitu penelitian produk & alat dan penelitian model.

Proses *Desain and Development* (D&D) akan melibatkan beberapa tahap yang mencakup analisis, desain, pengembangan, implementasi dan Evaluasi yang akan dijelaskan pada Gambar 3.1.



Gambar 3. 1 Alur Penelitian (Rusdi., 2019)

Pada gambar 3.1 menjelaskan bahwa metode *Desain and Development* (D&D) dimulai dari studi pustaka untuk memahami teori dan konteks yang relevan, seperti arsitektur *Arithmetic Logic Unit* (ALU), cara kerja atau cara merancang pada *Field Programmable Gate Array* (FPGA), serta bahasa pemrograman VHDL. Tahapan ini sangat penting untuk memperoleh dasar pengetahuan dalam merancang *Arithmetic Logic Unit* (ALU) 8-Bit pada FPGA. Selanjutnya, melakukan pengumpulan informasi, seperti spesifikasi FPGA Altera Cyclone IV EP4CE6E22C8N, serta analisis kebutuhan *input* dan *ouput* untuk digunakan dalam implementasi ALU 8-Bit pada FPGA Altera Cyclone IV.

Setelah perancangan selesai, dilakukan proses pengujian untuk menilai keberhasilan rancangan. Pengujian dilakukan melalui uji simulasi menggunakan ModelSim, serta pengujian alat secara langsung untuk mengevaluasi seluruh operasi yang telah dirancang. Setiap jenis operasi diberikan lima kali uji coba guna memastikan konsistensi dan keakuratan hasil keluaran.

3.1 Analysis

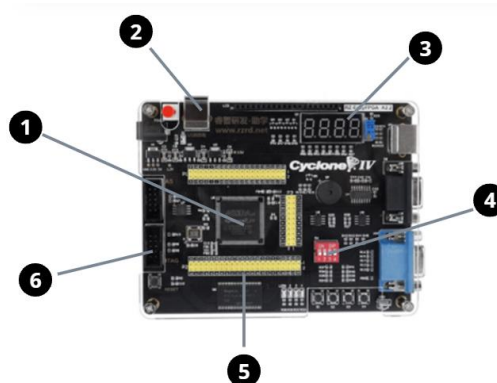
Analisis merupakan langkah awal yang penting dalam proses perancangan dan implementasi *Arithmetic Logic Unit* (ALU) 8-bit pada FPGA Altera Cyclone IV. Berikut analisis kebutuhan perancangan dan implementasi *Arithmetic Logic Unit* (ALU) 8-bit pada FPGA Altera Cyclone IV.

1. Analisis Kebutuhan Fungsional

Arithmetic Logic Unit (ALU) 8-bit dirancang untuk menjalankan operasi aritmatika dan logika dasar meliputi penjumlahan, pengurangan, perkalian dan pembagian. Sistem harus menerima dua *input* berukuran 8-Bit dan menghasilkan *output* berukuran 8-Bit sesuai dengan operasi yang dipilih melalui sinyal kontrol.

2. Analisis Kebutuhan Perangkat Keras

Pada kebutuhan perangkat keras ini menggunakan FPGA sebagai *platform* utama untuk mengimplementasikan *Arithmetics Logic Unit* (ALU). Jenis *Field Programmable Gate Array* (FPGA) yang digunakan adalah FPGA Altera Cyclone IV.



Gambar 3. 2 Spesifikasi FPGA

Pada gambar 3.2 menjelaskan spesifikasi FPGA yang akan digunakan pada penelitian ini, komponen FPGA yang akan digunakan akan dijelaskan pada tabel 3.1.

Tabel 3. 1 Spesifikasi FPGA

No	Nama komponen	Keterangan
1	Chip Altera Cyclone IV EP4CE6E22C8N	mengimplementasikan dan menjalankan berbagai fungsi logika digital.
2	USB power supply	Untuk menghubungkan <i>Field Programmable Gate Array</i> (FPGA) dengan power supply
3	Digit tube atau Layar tujuh segmen	Untuk memperlihatkan <i>Output</i> Desimal
4	Dip switch	Untuk memasukan <i>input</i>
5	I/O Pins	Untuk menyambungkan PCB <i>input</i> dan PCB <i>output</i>
6	JTAG interface	Untuk mengkonfigurasi bahasa pemrograman VHDL kedalam <i>Field Programmable Gate Array</i> (FPGA)

Sebagai bagian dari sistem *input-output*, digunakan switch dalam bentuk *Printed Circuit Board* (PCB) *input* untuk memasukkan data biner ke dalam *Arithmetic Logic Unit* (ALU), serta *Light-Emitting Diode* (LED) dalam bentuk *Printed Circuit Board* (PCB) *output* biner eksternal untuk menampilkan hasil keluaran dari operasi yang dijalankan oleh *Arithmetic Logic Unit* (ALU). Komponen eksternal ini dirancang terpisah dari board FPGA dan dihubungkan melalui kabel jumper, sehingga memudahkan dalam proses pengujian dan visualisasi data secara langsung.

3. Analisis Kebutuhan Perangkat Lunak

Pada kebutuhan perangkat lunak ini menggunakan Quartus Prime dan ModelSim untuk mendesain dan mensimulasikan *Arithmetics Logic Unit* (ALU) 8-Bit. Perangkat lunak Quartus Prime dipakai karena dirancang khusus untuk mendukung berbagai jenis FPGA terutama FPGA jenis Altera, termasuk Cyclone IV.

3.2 Desain

Pada bagian ini menjelaskan desain *Aritmatic Logic Unit* (ALU) 8-Bit pada FPGA Altera Cyclone IV.

3.2.1 Kontrol Aritmatic Logic Unit (ALU)

Kontrol *Aritmatic Logic Unit* (ALU) digunakan untuk mengatur operasi yang akan digunakan pada penelitian ini. Kontrol *Aritmatic Logic Unit* dapat dilihat pada tabel 3.2.

Tabel 3. 2 Kontrol *Aritmatic Logic Unit* (ALU)

Selector	Operasi
0000	A+B
0001	A-B
0010	A*B
0011	A/B
0100	AND
0101	OR

Seperti yang dijelaskan pada Tabel 3.1 di atas *Aritmatic Logic Unit* (ALU) memerlukan masukan untuk operasi kontrol yang berupa 2 masukan seperti pada tabel, masukan biner “0000” akan menghasilkan operasi nilai awal A + B, masukan biner “0001” akan menghasilkan operasi dari nilai awal A - B, sedangkan masukan biner bernilai “0010”

3.2.2 Diagram Blok *Arithmetic logic unit* (ALU) 8-Bit Pada FPGA Altera Cyclone IV

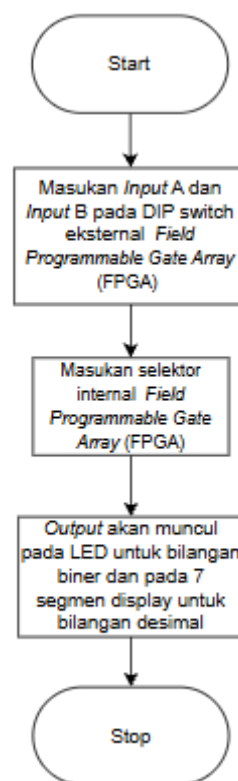
Pada gambar 3.3 diagram blok ALU 8-Bit dijelaskan dimana FPGA akan menerima dua *input* dari DIP switch A dan DIP Switch B, serta satu *input* tambahan berupa *selector* dari DIP Switch internal. *Selector* ini digunakan untuk memilih jenis operasi aritmatika yang akan dilakukan oleh FPGA Terhadap *input* A dan *input* B.

Input tersebut akan diproses oleh FPGA sesuai dengan pilihan operasi yang sudah *diinput* oleh *selector*. Lalu setelah diproses FPGA akan menampilkan *output* melalui layar tujuh segmen untuk menampilkan hasil desimal dan juga melalui *Light-Emitting Diode*

(LED) untuk menampilkan hasil biner dan Layar tujuh segmen pada FPGA untuk menampilkan hasil desimal.

3.2.3 Flowchart sistem *Arithmetic Logic Unit (ALU) 8-bit Pada Field Programmable Gate Array (FPGA)*

Pada gambar dibawah menunjukan bagaimana cara kerja sistem *Arithmetic Logic Unit (ALU) 8-Bit* pada FPGA Altera Cyclone IV. *Flowchart* sistem dapat dilihat pada gambar 3.4.



Gambar 3. 4 *Flowchart* sistem *Arithmetic Logic Unit (ALU) 8-bit* pada FPGA Altera Cyclone IV

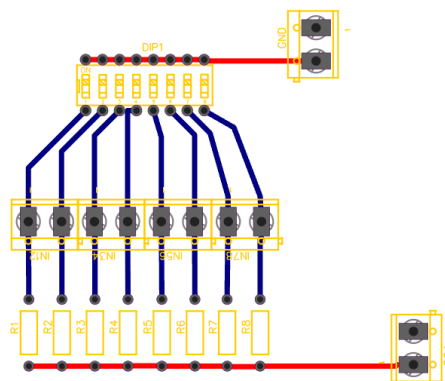
Pada gambar 3.4 menjelaskan seperti berikut:

1. Pada tahap pertama adalah memasukan dua *input A* dan *input B* melalui DIP Switch eksternal

2. Setelah itu memilih jenis operasi, seperti penjumlahan, pengurangan, perkalian, pembagian, *AND* dan *OR* yang diinginkan menggunakan *selector* melalui DIP switch internal yang sudah ada pada FPGA.
3. *Output* akan ditampilkan pada *Light Emitting Diode* (LED) untuk hasil bilangan biner dan Layar tujuh segmen untuk hasil bilangan desimal.

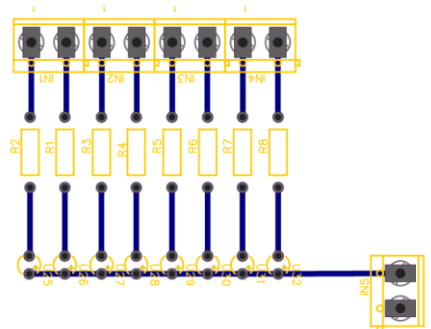
3.2.4 Desain PCB

Pada bagian ini terdapat dua desain PCB yang dibuat, diantaranya yaitu PCB *input* dan PCB *output*. Pada PCB *input* terdapat 4 komponen yaitu DIP switch, resistor, terminal PCB dan jumper. Dimana switch akan di hubungkan ke terminal dan ke resistor untuk menstabilkan sinyal *input* yang akan digunakan. Selanjutnya rangkaian ini akan dihubungkan ke VCC pada FPGA untuk menerima daya dan GND untuk *ground*, sebagaimana yang akan ditunjukkan pada gambar 3.5.



Gambar 3. 5 Rangkaian PCB *Input*

Selain PCB *input*, pada perancangan ini juga dibuat PCB *output* untuk menampilkan data *output* biner. Dimana pada output biner menggunakan 4 komponen yaitu LED, resistor, terminal PCB dan jumper. Dimana LED akan di sambungkan dengan resistor dan Terminal pada bagian anoda dan pada katoda akan dihubungkan pada GND pada FPGA, sebagaimana yang akan ditunjukkan pada gambar 3.6.

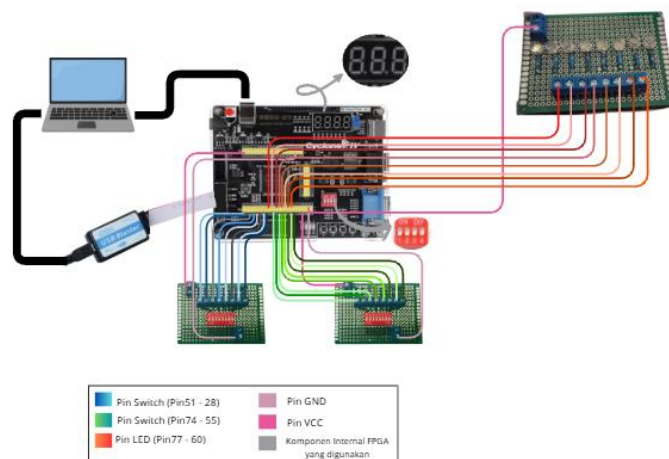


Gambar 3. 6 Rangkaian PCB *Output*

Pada kedua PCB tersebut akan dihubungkan kepada FPGA dengan menggunakan Jumper yang sebelumnya sudah terhubung pada masing masing terminal PCB yang tersedia. Dengan adanya sambungan ini, sinyal dari PCB input dapat diteruskan ke FPGA untuk diproses, sedangkan hasil pemrosesan dari FPGA akan dikirimkan kembali menuju PCB output sehingga sistem dapat bekerja secara terintegrasi.

3.2.5 Rangkaian Pin dan Wiring *Aritmatic Logic Unit (ALU) 8-bit*

Pada penelitian ini membahas bagaimana logika perancangan *Arithmetics Logic Unit (ALU) 8-bit* pada FPGA Altera Cyclone IV. Wiring *Aritmatic Logic Unit (ALU)* dapat dilihat pada gambar 3.5



Gambar 3. 7 Rangkaian Wiring

Pada gambar wiring diagram Aritmatic Logic Unit 8-bit pada FPGA diatas menjelaskan bagaimana merancang *Aritmatic Logic Unit* (ALU) 8-bit menggunakan FPGA dan komponen eksternal seperti DIP Switch sebagai *input* data biner dan juga *Light-Emitting Diode* (LED) sebagai salah satu *output*.

Pada wiring diagram tersebut juga memperlihatkan bagaimana cara koneksi FPGA ke laptop yang membutuhkan dua kabel penghubung yaitu kabel *power* untuk catu daya FPGA dan kabel USB Blaster untuk mengunggah program dari laptop ke FPGA. Seluruh koneksi dirancang agar FPGA dapat menerima *input*, proses dan *output*. Tabel rangkaian pin dapat dilihat pada tabel 3.3.

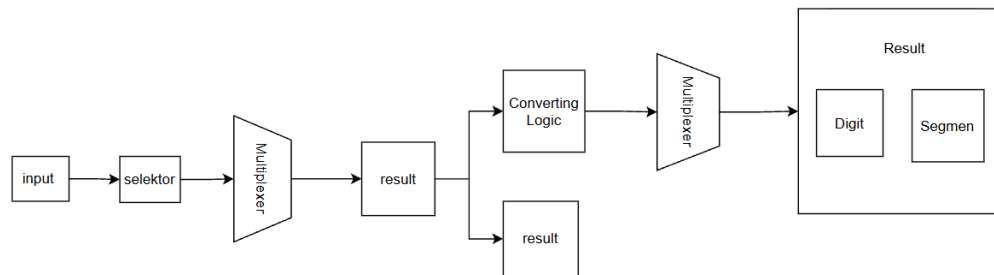
Tabel 3. 3 Konfigurasi Pin

Nama	koneksi	Pin
Clk	<i>Input</i>	PIN_23
SW1 [0]	<i>Input</i>	PIN_51
SW1 [1]	<i>Input</i>	PIN_49
SW1 [2]	<i>Input</i>	PIN_44
SW1 [3]	<i>Input</i>	PIN_42
SW1 [4]	<i>Input</i>	PIN_38
SW1 [5]	<i>Input</i>	PIN_33
SW1 [6]	<i>Input</i>	PIN_31
SW1 [7]	<i>Input</i>	PIN_28
SW2 [0]	<i>Input</i>	PIN_74
SW2 [1]	<i>Input</i>	PIN_72
SW2 [2]	<i>Input</i>	PIN_70
SW2 [3]	<i>Input</i>	PIN_68
SW2 [4]	<i>Input</i>	PIN_66
SW2 [5]	<i>Input</i>	PIN_64
SW2 [6]	<i>Input</i>	PIN_59

Nama	koneksi	Pin
SW2 [7]	<i>Input</i>	PIN_55
SEL [0]	<i>Input</i>	PIN_89
SEL [1]	<i>Input</i>	PIN_88
DIGIT [0]	<i>Output</i>	PIN_133
DIGIT [1]	<i>Output</i>	PIN_135
DIGIT [2]	<i>Output</i>	PIN_136
DIGIT [3]	<i>Output</i>	PIN_137
SEG [0]	<i>Output</i>	PIN_128
SEG [1]	<i>Output</i>	PIN_121
SEG [2]	<i>Output</i>	PIN_125
SEG [3]	<i>Output</i>	PIN_129
SEG [4]	<i>Output</i>	PIN_132
SEG [5]	<i>Output</i>	PIN_126
SEG [6]	<i>Output</i>	PIN_124
LEDR [0]	<i>Output</i>	PIN_77
LEDR [1]	<i>Output</i>	PIN_75
LEDR [2]	<i>Output</i>	PIN_73
LEDR [3]	<i>Output</i>	PIN_71
LEDR [4]	<i>Output</i>	PIN_69
LEDR [5]	<i>Output</i>	PIN_67
LEDR [6]	<i>Output</i>	PIN_65
LEDR [7]	<i>Output</i>	PIN_60

3.4.6 Struktur Sistem *Aritmatic Logic Unit* (ALU) 8-Bit

Pada bagian ini menjelaskan mengenai Struktur sistem *Arithmetic Logic Unit* (ALU) 8-Bit untuk menjelaskan bagaimana logika *Arithmetic Logic Unit* (ALU) 8-Bit. Struktur sistem *Arithmetic Logic Unit* (ALU) 8-Bit dapat dilihat pada gambar 3.6.



Gambar 3. 8 Struktur Arithmetic Logic Unit (ALU) 8-Bit

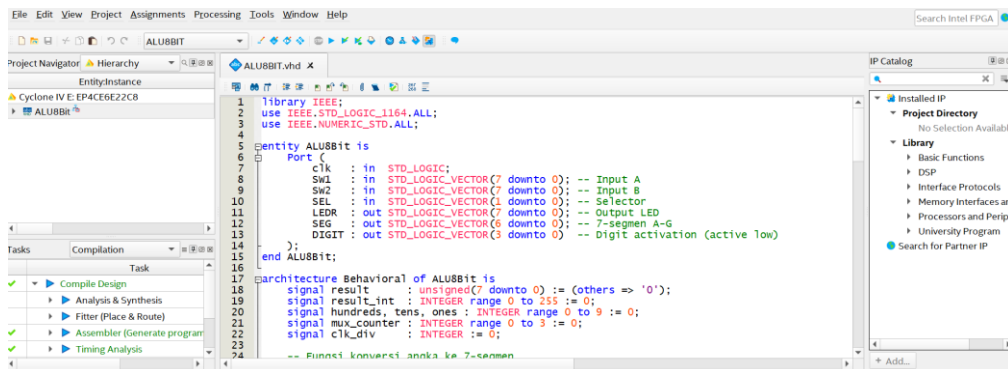
Pada gambar 3.8 menampilkan struktur logika *Arithmetic Logic Unit* (ALU) 8-Bit, dimana pada struktur terdapat dua multiplexer, pada multiplexer pertama digunakan untuk proses operasi yang dimana sebelumnya sudah diinputkan dan akan menghasilkan *output* biner. Pada multiplexer kedua digunakan untuk mengubah atau mengkonversi bilangan biner ke bilangan desimal, yang nantinya akan menghasilkan sebuah digit dan segmen sekaligus untuk *output* desimal pada layar tujuh segmen. *Output* RTL dapat dilihat pada lampiran 14.

3.3 Pengembangan

Pada penelitian *Aritmatic Logic Unit* (ALU) 8-bit pada FPGA, setelah merancang alat dan menyusun seluruh rangkaian serta konfigurasi pin, langkah selanjutnya adalah melakukan pemrograman pada perangkat lunak Quartus. Proses ini bertujuan untuk mengunggah pemrograman yang akan diimplementasikan ke dalam FPGA sehingga perangkat dapat menjalankan fungsi logika dan aritmatika sesuai desain yang telah dibuat. Berikut langkah langkah pemrograman:

1. Membuat Program menggunakan Bahasa pemograman VHDL.

Tahap pertama pada pemograman *Aritmatic Logic Unit* (ALU) 8-bit pada FPGA yaitu membuat pemograman VHDL. Pembuatan program dapat dilihat pada gambar 3.9.



Gambar 3. 9 Pembuatan pemrograman

2. Melakukan Kompilasi untuk mencegah adanya kesalahan

Setelah membuat pemrograman menggunakan pemrograman VHDL seperti pada gambar 3.9. Langkah selanjutnya yaitu melakukan kompilasi untuk melihat apakah ada *error* pada pemrograman atau tidak. *Tools* pada saat melakukan kompilasi dapat dilihat pada gambar 3.10.



Gambar 3. 10 Toolbar Compilation

3. Melakukan Konfigurasi Pin

Setelah pemrograman sudah dipastikan tidak ada error, maka langkah selanjutnya adalah melakukan Konfigurasi Pin. Hal ini bertujuan untuk menghubungkan pemrograman pada Quartus dengan pin fisik FPGA, sehingga sinyal *input* dan *output* dari kode dapat terhubung langsung ke perangkat keras seperti switch dan *Light-Emitting Diode* (LED) yang digunakan pada *board* FPGA. Berikut terdapat gambar toolbar pin planner, tampilan pin planner dan chip FPGA Altera Cyclone IV yang ditampilkan pada gambar 3.11 hingga gambar 3.13.

Atsillah Cyntiasari Hidayat Putri, 2025

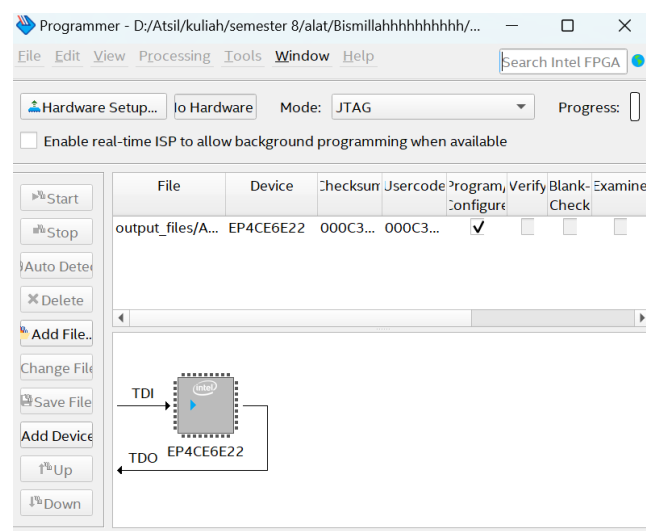
**RANCANGAN DAN IMPLEMENTASI ARITHMETIC LOGIC UNIT 8-BIT
PADA FPGA ALTERA CYCLONE IV**

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu



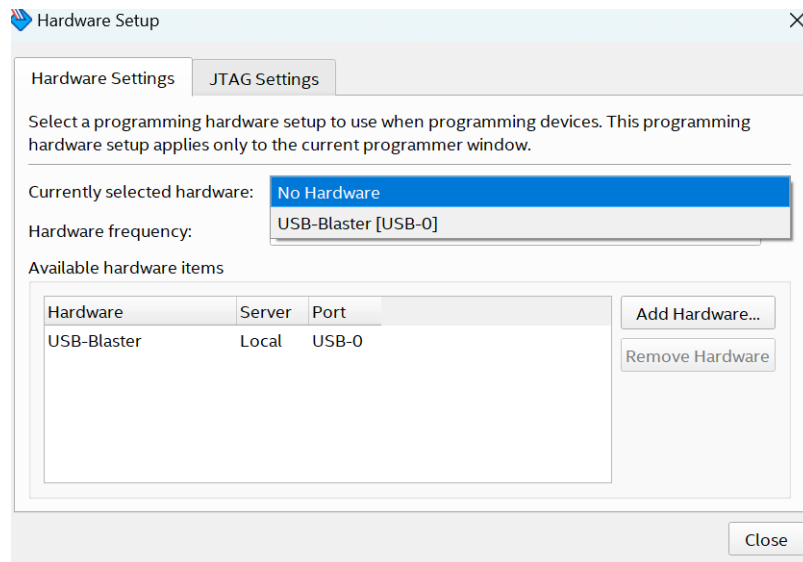
Gambar 3. 14 Toolbar Programmer

Untuk mengunggah pemograman klik ikon seperti pada gambar 3.14, setelah itu akan muncul seperti gambar 3.15.



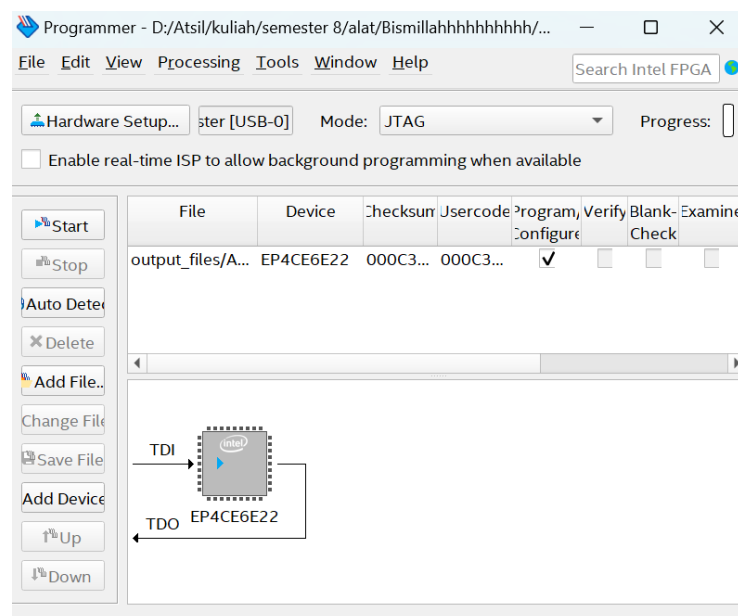
Gambar 3. 15 Tampilan dari Programmer

Setelah itu klik hardware setup untuk menghubungkan USB Blaster pada perangkat lunak Quartus. Setelah dihubungkan maka klik 'close' yang ada pada gambar 3.16.



Gambar 3. 16 Tampilan dari hardware setup

Setelah menghubungkan USB blaster maka program siap untuk di unggah ke FPGA, dengan cara klik 'start' pada bagian kiri pada gambar 3.17.



Gambar 3. 17 Tampilan programmer setelah dihubungkan USB Blaster

3.4 Implementasi

Pada bagian ini membahas proses implementasi *Arithmetic Logic Unit* (ALU) 8-Bit ke dalam FPGA menggunakan bahasa pemrograman VHDL serta menggunakan perangkat lunak Quartus. Selain itu, dijelaskan pula skenario pengujian yang dilakukan untuk memastikan bahwa seluruh operasi aritmatika seperti penjumlahan, pengurangan, perkalian, dan pembagian dapat berjalan dengan benar sesuai spesifikasi desain.

3.5 Evaluasi

Pada penelitian ini terdapat dua pengujian yang telah diperoleh. Pengujian dilakukan untuk memastikan apakah alat dapat berfungsi dengan baik dan sesuai. Setiap pengujian dilakukan secara sistematis untuk mengevaluasi kinerja dan keakuratan alat dalam kondisi nyata. Pengujian pertama dilakukan dengan mensimulasikan program menggunakan perangkat lunak ModelSim. Pengujian kedua dilakukan dengan perancangan dan mengimplementasikan program ke dalam perangkat keras (*hardware*) yang telah dirancang.

Pada pengujian pertama melakukan pengujian atau simulasi pada pemrograman yang sudah dibuat untuk memastikan keakuratan logika dan fungsi program yang telah dirancang dengan menggunakan perangkat lunak ModelSim dengan menggunakan skenario pengujian ModelSim pada tabel 3.4.

NO	Current Simulation Time (ns)	SW1		SW2		Selector	Result Biner	Result Desimal
1	10 – 20	00001111	15	00000101	5	0000	00010100	20
2	20 – 40	00011001	25	00001010	10	0001	00001111	15
3	40 – 60	00000111	7	00000011	3	0010	00010101	21
4	60 – 80	01100100	100	00000100	4	0011	00011001	25
5	80 - 100	00001100	12	00001010	10	0100	00001000	8

NO	Current Simulation Time (ns)	SW1		SW2		Selector	Result Biner	Result Desimal
6	100 - 120	00001100	12	00001010	10	0101	00001110	14

Tabel 3.4 Skenario Pengujian ModelSim

Pada tabel 3.4 ditunjukkan bahwa *Current Simulation Time* (ns) merepresentasikan rentang waktu saat proses simulasi berlangsung untuk melihat kapan input diberikan dan *output* dihasilkan. SW1 merupakan *input* pertama (operand A) yang bernilai 25 (biner: 00011001), sedangkan SW2 merupakan *input* kedua (operand B) yang bernilai 10 (biner: 00001010). Selektor berfungsi sebagai sinyal kendali untuk menentukan jenis operasi yang dijalankan ALU, Dengan demikian, skenario pengujian ini menjadi acuan dalam pengujian logika pemrograman pada perangkat lunak ModelSim.

Pada pengujian kedua melakukan pengujian terhadap rangkaian *Arithmetic Logic Unit* (ALU) 8-bit yang di implementasikan pada FPGA Altera Cyclone IV. Pengujian dilakukan dengan memberikan berbagai kombinasi nilai pada SW1 (*input* A) dan SW2 (*input* B), serta memilih jenis operasi menggunakan *selector*. Masing-masing jenis operasi aritmatika (penjumlahan, pengurangan, perkalian, dan pembagian) dan operasi logika (*AND* dan *OR*) diuji sebanyak lima kali dengan nilai bilangan yang berbeda-beda untuk memastikan seluruh logika berfungsi dengan benar dan hasilnya sesuai dengan yang diharapkan. *Output* dari *Arithmetic Logic Unit* (ALU)8-Bit ditampilkan melalui *Light-Emitting Diode* (LED) sebagai representasi biner, dan melalui tampilan layar tujuh segmen sebagai bentuk desimal.

- a. Skenario yang digunakan pada pengujian operasi penjumlahan sebagai berikut:

No	Input 1		Input 2		Selektor
1	00000011	3	00000011	3	0000
2	01111111	127	10000000	128	0000

No	Input 1		Input 2		Selektor
3	11001000	200	11001000	200	0000
4	11111111	255	00000101	5	0000
5	11111111	255	11111111	255	0000

- b. Skenario yang digunakan pada pengujian operasi pengurangan sebagai berikut:

No	Input 1		Input 2		Selektor
1	00001010	10	00001010	10	0001
2	10000010	130	10000101	133	0001
3	10010110	150	00011110	30	0001
4	00000010	2	11111111	255	0001
5	11111111	255	11111111	255	0001

- c. Skenario yang digunakan pada pengujian operasi Perkalian sebagai berikut:

No	Input 1		Input 2		Selektor
1	01100100	20	01100100	100	0010
2	00000000	203	00000000	0	0010
3	01111010	4	01111010	122	0010
4	11111111	1	11111111	255	0010
5	11111111	255	11111111	255	0010

- d. Skenario yang digunakan pada pengujian operasi pembagian sebagai berikut:

No	Input 1		Input 2		Selektor
1	10011000	152	00000110	6	0011

No	Input 1		Input 2		Selektor
2	01011111	95	00000010	2	0011
3	10011011	155	00000000	0	0011
4	11111010	250	11111111	255	0011
5	11111111	255	11111111	255	0011

- e. Skenario yang digunakan pada pengujian operasi AND sebagai berikut:

No	Input 1		Input 2		Selektor
1	11111010	150	00000100	4	0100
2	01100100	100	11001000	200	0100
3	11100000	224	00111100	60	0100
4	01010110	86	01010101	85	0100
5	11111111	255	11001111	207	0100

- f. Skenario yang digunakan pada pengujian operasi OR sebagai berikut:

No	Input 1		Input 2		Selektor
1	11111010	10	00000010	2	0101
2	01111111	127	01111110	126	0101
3	01100100	100	01010000	80	0101
4	01001011	75	00001010	10	0101
5	11111111	255	11111111	255	0101

Pengujian ini bertujuan untuk membuktikan apakah sistem *Arithmetic Logic Unit* (ALU) 8-bit yang telah diimplementasi berfungsi secara logis dan konsisten, sesuai dengan arsitektur yang telah dirancang. Pada pengujian alat terdapat 6 operasi yang dilakukan dan masing masing 5 pengujian. Sebagai salah satu contoh pada pengujian pertama pada penjumlahan pada *input* 1 akan di masukan nilai sebesar 00000011 dan *input* 2 sebesar 00000011 yang

nantinya LED akan menyala dengan biner 00000110 dan layar tujuh segmen akan memperlihatkan nilai *output* 6 untuk mempersentasikan bilangan desimal dari 00000110.

Pada kinerja sistem FPGA akan diperlihatkan pada perangkat Quartus dengan fitur *analysis & synthesis* dimana pada fitur tersebut akan menampilkan logika FPGA yang digunakan untuk *Arithmetic Logic Unit* (ALU) 8-Bit. Pengujian yang dilakukan menggunakan pendekatan dalam penelitian Jagtap dkk., (2024), penelitian tersebut menggunakan FPGA spartan dan menggunakan perangkat lunak Xilinx, sehingga terdapat beberapa perbedaan pada hasil yang ditampilkan. Dan menggunakan fitur *power analyzer* pada perangkat Quartus, mengikuti pendekatan yang digunakan dalam penelitian oleh Suhaili & Watanabe (2016).