LAPORAN PROYEK AKHIR PERANCANGAN SISTEM DIGITAL
DEPARTEMEN TEKNIK ELEKTRO
UNIVERSITAS INDONESIA

SIMULASI CRYPTOCURRENCY 67-BIT

KELOMPOK 3

Arya Wibawa Atmanegara	**2406420431**
Ganendra Garda Pratama	**2306250642**
Novan Agung Wicaksono	**2406401294**
Zulfahmi Fajri	**2406345425**

KATA PENGANTAR

Puji syukur ke hadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya, sehingga kami dapat menyelesaikan laporan proyek akhir yang berjudul "Simulasi Cryptocurrency 67-bit" untuk mata kuliah Pengantar Sistem Digital.

Proyek ini dirancang untuk mensimulasikan mekanisme dasar sistem cryptocurrency berbasis blockchain menggunakan VHDL. Tujuan kami adalah mendemonstrasikan logika perangkat keras (hardware-level) dari proses mining, konsensus, dan pengelolaan buku besar (ledger) dalam lingkungan simulasi yang sederhana namun kompatibel dengan logika FPGA. Dengan mengimplementasikan modul-modul seperti custom hashing, proof-of-work mining, dan penyimpanan terdesentralisasi, kami berharap dapat memberikan gambaran nyata mengenai cara kerja mata uang digital pada tingkat bit.

Kami ingin mengucapkan terima kasih yang sebesar-besarnya kepada dosen dan asisten laboratorium kami atas dukungan, bimbingan, dan ilmu yang telah diberikan selama satu semester ini. Kami juga berterima kasih kepada rekan-rekan anggota kelompok atas kerja sama dan kerja kerasnya.

Kami menyadari bahwa laporan dan implementasi proyek ini masih memiliki ruang untuk perbaikan. Oleh karena itu, kami mengharapkan kritik dan saran yang membangun untuk penyempurnaan karya ini di masa mendatang.

Depok, Desember 2025

Kelompok 3

DAFTAR ISI

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Teknologi blockchain telah merevolusi industri keuangan dengan memperkenalkan sistem buku besar yang terdesentralisasi, aman, dan transparan. Pada inti dari kebanyakan cryptocurrency, seperti Bitcoin, terdapat mekanisme konsensus "Proof-of-Work" (PoW). Mekanisme ini mengharuskan peserta jaringan (miner) untuk mengeluarkan upaya komputasi guna memvalidasi transaksi dan mengamankan jaringan.

Memahami konsep-konsep ini sering kali memerlukan pemahaman yang lebih dalam daripada sekadar perangkat lunak tingkat tinggi, yaitu dengan memeriksa logika yang mendasarinya pada tingkat perangkat keras. Proyek ini bertujuan untuk menjembatani kesenjangan tersebut dengan mengimplementasikan simulasi cryptocurrency sederhana menggunakan VHDL (VHSIC Hardware Description Language). Dengan merancang sistem pada tingkat register-transfer (RTL), kita dapat memvisualisasikan waktu (timing) yang presisi, aliran data, dan transisi state yang terjadi selama proses mining dan konsensus. Hal ini memberikan wawasan pendidikan yang lebih mendalam mengenai desain sistem digital dan arsitektur komputer.

1.2 DESKRIPSI PROYEK

Proyek yang berjudul "Simulasi Cryptocurrency 67-bit" ini adalah simulasi ekosistem blockchain berbasis VHDL. Berbeda dengan cryptocurrency skala penuh yang menggunakan struktur 256-bit atau 512-bit, proyek ini menggunakan header blok kustom 67-bit dan fungsi hash 64-bit agar logikanya dapat diamati dan dikelola dalam lingkungan simulasi.

Fitur sistem ini meliputi:

- Dua Miner yang Bersaing (Wallet A & Wallet B): Kedua entitas melakukan Proof-of-Work untuk menemukan "nonce" yang menghasilkan hash valid.
- Penyimpanan Blockchain: Modul memori yang bertindak sebagai buku besar (ledger), menyimpan blok-blok yang saling terhubung dengan indeks 3-bit.

- Pengendali Konsensus: Wasit yang memvalidasi blok yang ditemukan, memperbarui head (kepala) blockchain, dan memberikan koin kepada miner yang berhasil.
- Sistem Dompet (Wallet): Melacak ID dan saldo peserta, serta melakukan pembaruan secara real-time saat hadiah didistribusikan.

1.3 TUJUAN

Tujuan dari proyek ini adalah sebagai berikut:

1. Untuk memahami logika dasar teknologi *blockchain* pada tingkat perangkat keras menggunakan VHDL.
2. Untuk mengimplementasikan mekanisme Proof-of-Work (PoW) sederhana menggunakan algoritma hashing kustom 64-bit.
3. Untuk mensimulasikan persaingan antara dua *miner* (Wallet A dan Wallet B) dalam lingkungan yang terdesentralisasi.
4. Untuk merancang mekanisme konsensus yang memvalidasi blok dan mengelola buku besar terdistribusi dalam batasan memori FPGA.

1.4 PERAN DAN TANGGUNG JAWAB

Peran dan tanggung jawab yang diberikan kepada anggota kelompok sebagai berikut:

Peran	Tanggung Jawab	Nama
Designer	Menyiapkan dan mendesain Presentasi dan Laporan	Arya Wibawa Atmanegara
Documenter	Mengisi Laporan dan Presentasi	Ganendra Garda Pratama
Main Programmer	Membuat program project	Novan Agung Wicaksono
Documenter	Mengisi Laporan dan Presentasi	Zulfahmi Fajri

Table 1. Peran dan Tanggung Jawab

BAB 2

IMPLEMENTASI

2.1 PERALATAN

Perangkat dan alat bantu yang digunakan dalam proyek ini adalah sebagai berikut:

- ModelSim - Intel FPGA Edition: Digunakan untuk penulisan, kompilasi, dan simulasi kode VHDL.
- Visual Studio Code: Digunakan untuk penyuntingan (editing) kode dan penyusunan dokumentasi.
- Standar VHDL-2002: Standar bahasa deskripsi perangkat keras (Hardware Description Language) yang digunakan untuk seluruh implementasi modul.
- Quartus Prime: Digunakan untuk melakukan sintesis kode VHDL menjadi skematik RTL.

2.2 IMPLEMENTASI

1. **Struktur Header Blok (block_header.vhd),** Kami membuat tipe data record khusus block_header_t untuk mengelola data blok secara efisien. Header ini memiliki lebar 67 bit, yang berisi:
    a. prev_index (3 bits): Penunjuk (*pointer*) ke alamat blok sebelumnya di RAM.
    b. miner_id (8 bits): Pengenal unik dari miner (Wallet A atau B).
    c. timestamp (24 bits): Penghitung waktu yang merepresentasikan kapan blok tersebut ditambang.
    d. nonce (16 bits): Variabel angka yang dinaikkan (increment) selama proses mining untuk mengubah keluaran hash.
    e. hash_fragment (16 bits): Bagian dari hasil hash yang digunakan untuk verifikasi.
2. **Fungsi Hash (hash64.vhd),** Fungsi hash kustom 64-bit dibuat untuk mensimulasikan pekerjaan kriptografi. Fungsi ini menggunakan operasi bitwise termasuk XOR, Penjumlahan, dan Geser Melingkar (Circular Shifts/Rotate Left/Right).

a. Fungsi ini mengambil input 64-bit yang berasal dari header blok.
b. Fungsi ini mengacak input dengan empat konstanta 64-bit (CONST1 hingga CONST4) untuk menghasilkan keluaran pseudo-random.
c. Hal ini menciptakan "tingkat kesulitan" yang diperlukan untuk Proof-of-Work tanpa beban komputasi berat seperti SHA-256.

3. **Miner (miner.vhd)** Modul ini merepresentasikan "pekerja" dalam sistem.
    a. Input: Membaca header dari blok sebelumnya dan nilai target_bits (tingkat kesulitan).
    b. Proses: Pada setiap siklus clock, modul ini menaikkan timestamp_counter dan melakukan iterasi pada nonce. Modul mengemas data header saat ini, mengirimkannya ke instansi hash64, dan memeriksa apakah hash_result secara numerik lebih kecil dari target_bits.
    c. Output: Jika nonce yang valid ditemukan, sinyal block_found akan aktif dan modul mengeluarkan blok yang berhasil ditambang.
4. **Penyimpanan Blockchain (blockchain_storage.vhd)** Modul ini bertindak sebagai buku besar terdesentralisasi.
    a. Menggunakan array block_array_t yang mampu menyimpan 8 blok (dapat diakses dengan indeks 3-bit).
    b. Memelihara pointer head_idx yang menunjukkan blok terbaru dalam rantai.
    c. Mengizinkan Pengendali Konsensus untuk menulis blok baru melalui write_en dan mengizinkan Miner untuk membaca data blok sebelumnya.
5. **Pengendali Konsensus (consensus_controller.vhd)** Modul ini berfungsi sebagai wasit pusat (atau logika jaringan).
    a. Memantau sinyal block_found dari Miner A dan Miner B.
    b. Ketika miner menemukan blok, pengendali akan:
        i. Mengaktifkan penulisan ke Penyimpanan.
        ii. Memperbarui head_idx ke blok baru.
        iii. Mengidentifikasi pemenang (winner_id).
        iv. Mengirim permintaan deposit ke sistem Wallet untuk memberi hadiah 1 Token kepada pemenang.
6. **Wallet (wallet.vhd)** Dompet yang mengelola saldo para partisipan (para pengguna).
    a. Penyimpanan State: Menggunakan register internal untuk menyimpan ID dompet dan jumlah saldo terkini.

b. Penerimaan Reward: Modul selalu "mendengarkan" sinyal deposit_req dari Consensus Controller.
c. Update Saldo: Jika sinyal valid diterima (artinya pengguna memenangkan proses mining), saldo internal akan bertambah secara otomatis sebagai imbalan.

BAB 3

PENGUJIAN DAN ANALISIS

3.1 PENGUJIAN

Sistem diuji menggunakan ModelSim - Intel FPGA Edition. Kami menggunakan file testbench (tb_top.vhd) untuk mensimulasikan seluruh ekosistem. Proses pengujian meliputi:

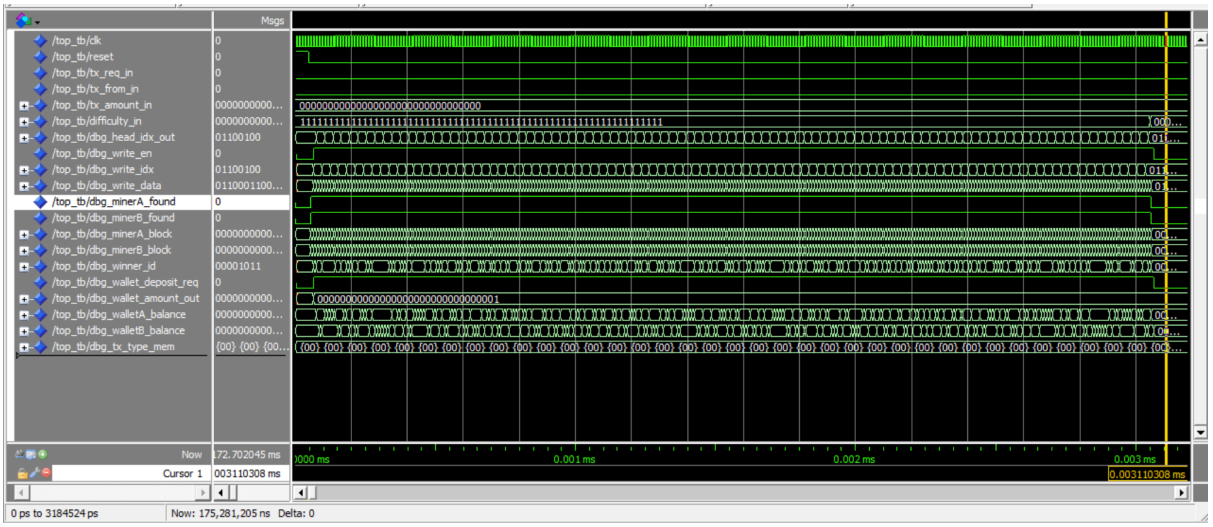
1. Pembangkitan Clock: Menghasilkan sinyal clock yang konsisten untuk menjalankan proses mining.
2. Reset Sistem: Menginisialisasi semua komponen (wallet, penyimpanan, miner) ke kondisi nol.
3. Penyesuaian Tingkat Kesulitan: Mengatur target kesulitan (misalnya, x"0000FFFF...") untuk memastikan blok dapat ditemukan dalam waktu simulasi yang wajar.
4. Observasi: Memantau sinyal seperti dbg_minerA_found, dbg_winner_id, dan dbg_wallet_balance untuk memverifikasi bahwa miner berhasil menemukan blok dan menerima hadiah.

3.2 HASIL

Hasil simulasi di ModelSim memverifikasi fungsionalitas sistem. Seperti terlihat pada Gambar 2, gelombang sinyal (waveform) menunjukkan operasi mining dan konsensus yang aktif:

1. Aktivitas Sinyal: Area hijau padat pada sinyal dbg_minerA_block dan dbg_minerB_block menunjukkan bahwa kedua miner secara aktif melakukan iterasi pada nonce dan melakukan perhitungan hash.
2. Keberhasilan Mining: Kursor dan tooltip (seperti yang ditunjukkan pada jendela simulasi) mengonfirmasi bahwa blok yang valid sedang dihasilkan. Sinyal difficulty_in diatur secara efektif, memungkinkan miner menemukan solusi dalam kerangka waktu simulasi.

3. Respons Sistem: Clock global (clk) menggerakkan seluruh proses, dan sinyal reset bertransisi ke logika '0' untuk memulai operasi, membuktikan bahwa integrasi modul tingkat atas (top-level) sudah benar.



Gambar 2. Hasil Pengujian

Secara ringkas, data visual yang disajikan pada Gambar 2 mengonfirmasi bahwa simulasi beroperasi dengan benar sesuai parameter yang ditentukan. Transisi sinyal yang sukses dari eksekusi mining hingga penemuan blok dan pembaruan buku besar menunjukkan bahwa modul-modul VHDL berinteraksi sesuai rencana, memberikan dasar yang kuat untuk analisis terperinci di bagian selanjutnya.

3.3 ANALYSIS

Berdasarkan hasil simulasi yang ditunjukkan pada Gambar 2, sistem menunjukkan perilaku yang diharapkan dari lingkungan blockchain yang kompetitif. Aktivitas tinggi pada sinyal dbg_minerA_block dan dbg_minerB_block menunjukkan bahwa kedua modul mining beroperasi secara paralel, menaikkan nonce dan menghitung hash dengan cepat pada setiap siklus clock.

Hal penting yang dapat diamati terlihat pada posisi kursor. Pada stempel waktu (timestamp) spesifik ini, sinyal dbg_minerA_found bertransisi ke '1', menunjukkan bahwa Miner A berhasil menemukan nilai hash yang lebih rendah dari ambang batas difficulty_in. Segera setelah kejadian ini, sinyal dbg_write_en menjadi tinggi (high). Hal ini mengonfirmasi bahwa Pengendali Konsensus mendeteksi blok yang valid dengan benar dan

memulai urutan penulisan ke blockchain_storage. Urutan ini membuktikan bahwa logika 'Race Condition' berfungsi dengan baik. Segera setelah satu miner menang, status jaringan diperbarui, yang secara efektif memulai ronde berikutnya bagi seluruh peserta.

Simulasi dijalankan dengan nilai difficulty_in yang tinggi (awalnya diatur maksimal) untuk memastikan penemuan blok dalam waktu simulasi yang wajar. Dalam penerapan FPGA di dunia nyata, tingkat kesulitan ini akan disesuaikan secara dinamis untuk mengontrol waktu blok (misalnya, 10 menit untuk Bitcoin). Waveform menunjukkan bahwa arsitektur 64-bit saat ini cukup efisien untuk memproses hash dalam siklus clock nanodetik tanpa menghambat (bottlenecking) bus sistem utama.

BAB 4

KESIMPULAN

Proyek ini telah berhasil mencapai tujuan utamanya, yaitu merancang dan mensimulasikan sistem "67-bit Cryptocurrency" yang fungsional pada tingkat perangkat keras (hardware level) menggunakan bahasa VHDL. Melalui implementasi modul-modul inti seperti Miner, Blockchain Storage, Consensus Controller, dan Wallet, kami berhasil mereplikasi mekanisme fundamental dari sebuah jaringan blockchain yang terdesentralisasi. Berdasarkan hasil simulasi yang diverifikasi menggunakan ModelSim, dapat disimpulkan hal-hal sebagai berikut:

1. Fungsi Proof-of-Work: Algoritma hashing 64-bit kustom yang dirancang terbukti mampu memberikan tantangan komputasi yang berfungsi dengan baik bagi para miner. Hal ini berhasil mensimulasikan beban kerja komputasi (computational effort) yang menjadi syarat utama dalam penambangan mata uang kripto di dunia nyata.
2. Penanganan Race Condition: Sistem mampu menangani kondisi kompetisi (race condition) antara dua penambang yang bersaing (Wallet A dan Wallet B) dengan tepat. Consensus Controller memastikan bahwa hanya blok valid pertama yang ditambahkan ke dalam blockchain, sehingga menjaga integritas buku besar (ledger) tetap linear dan aman dari manipulasi.
3. Mekanisme Insentif: Mekanisme pemberian imbalan berjalan sesuai rancangan, di mana penambang yang menang menerima hadiah blok (block reward) secara instan tepat setelah validasi blok dilakukan.

Meskipun demikian, sistem ini memiliki batasan-batasan teknis yang memang disengaja untuk tujuan penyederhanaan. Struktur blok “67-bit” dan batas memori penyimpanan 8-blok jauh lebih kecil dibandingkan standar industri (seperti Bitcoin). Untuk pengembangan di masa depan, proyek ini dapat ditingkatkan dengan mengimplementasikan algoritma hashing standar SHA-256, memperluas lebar alamat memori untuk mendukung riwayat blockchain yang lebih panjang, serta menerapkan desain ini ke dalam papan FPGA fisik untuk menguji performa secara real-time.

Sebagai penutup, proyek ini memberikan wawasan komprehensif mengenai perancangan Register Transfer Level (RTL) untuk sistem cryptocurrency. Proyek ini berhasil

menjembatani kesenjangan antara konsep teoritis blockchain dengan implementasi praktis sistem digital.

DAFTAR PUSTAKA

[1] S. Brown and Z. Vranesic, *Fundamentals of Digital Logic with VHDL Design*, 3rd ed. New York, NY, USA: McGraw-Hill Education, 2008.

[2] P. J. Ashenden, *The Designer's Guide to VHDL*, 3rd ed. Burlington, MA, USA: Morgan Kaufmann, 2008.

[3] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *Bitcoin.org*, Oct. 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf.

[4] S. B. Pandya, H. A. Sanghvi, R. H. Patel, and A. S. Pandya, "GPU and FPGA based deployment of blockchain for cryptocurrency–a systematic review," in *2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, Greater Noida, India, 2022.

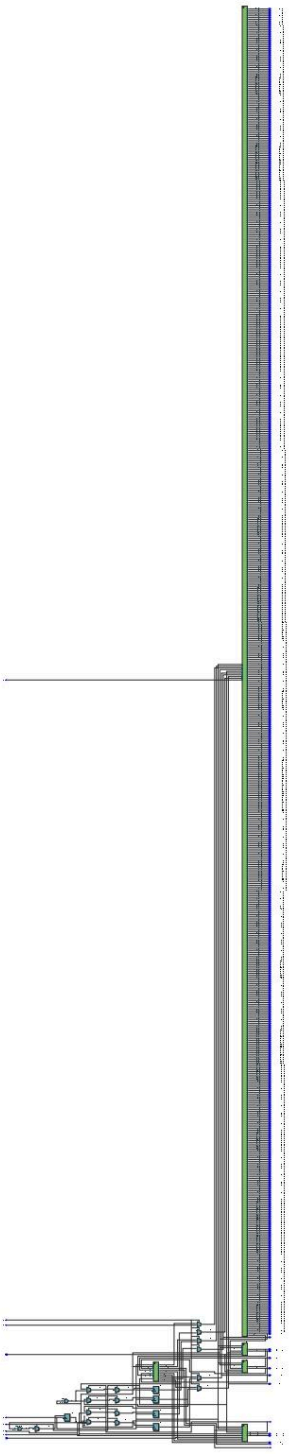
[5] I. Algredo-Badillo, C. Feregrino-Uribe, R. Cumplido, and M. Morales-Sandoval, "FPGA-based implementation alternatives for the inner loop of the Secure Hash Algorithm SHA-256," *Microprocessors and Microsystems*, 2013.

[6] S. Yan, "Analysis on blockchain consensus mechanism based on proof of work and proof of stake," in *2022 International Conference on Data Analytics, Computing and Artificial Intelligence (ICDACAI)*, Zakopane, Poland, 2022

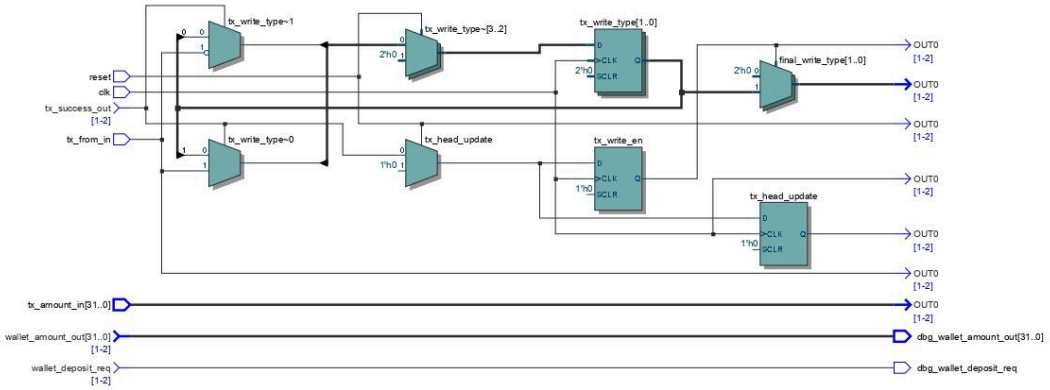
[7] P. Müller, S. Bergsträßer, A. Rizk, and R. Steinmetz, "The bitcoin universe: An architectural overview of the bitcoin blockchain," in *11. DFN-Forum Kommunikationstechnologien*, Gesellschaft für Informatik eV, 2018.

LAMPIRAN

Lampiran A: Skema Proyek

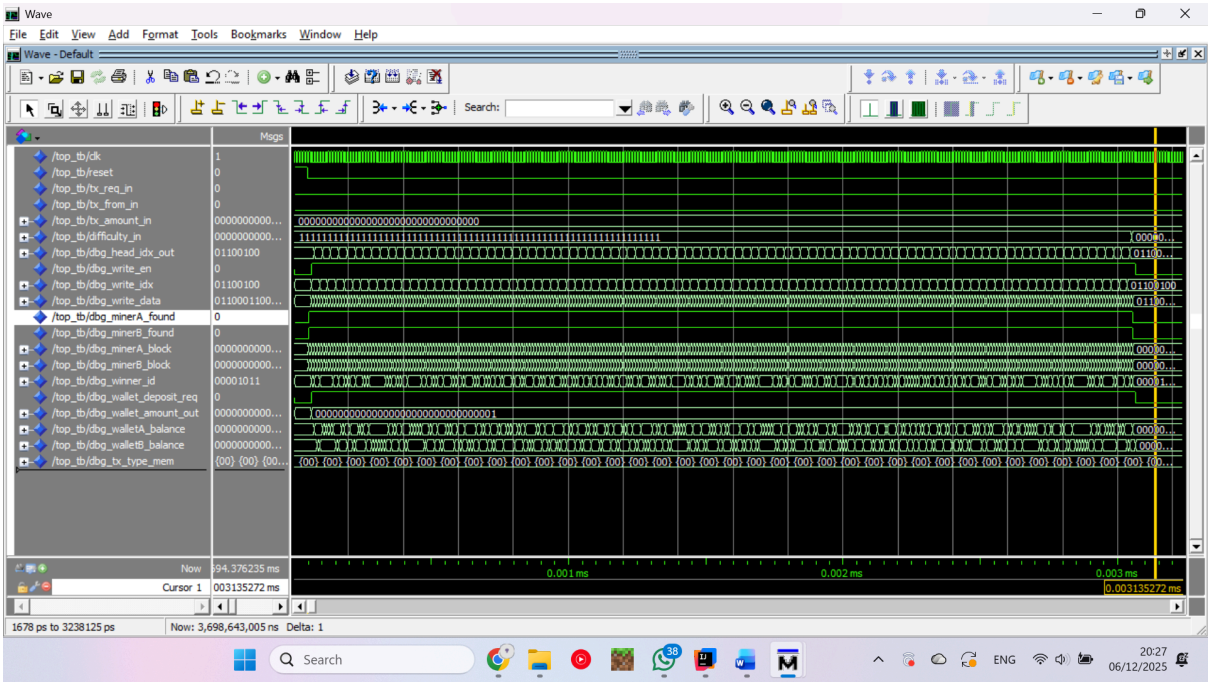


Gambar A.1 Skematik Top-Level Entity dan Interkoneksi Pin



Gambar A.2 Detail Logika RTL untuk Kontrol Transaksi dan Penulisan Blok

Lampiran B: Dokumentasi



Gambar B.1 Tampilan Gelombang Simulasi Penuh di ModelSim