

**Rancang Bangun Pencacah Frekuensi Multi Kanal Dengan Metode
Digital Interpolasi *Reciprocal* Menggunakan *High Resolution* TDC
(*Time to Digital Converter*) pada *Low-Cost* FPGA**

LAPORAN TESIS



Oleh:

Imron Rosadi

166090300111011

**PROGRAM STUDI S2 ILMU FISIKA
MINAT FISIKA INSTRUMENTASI**

PROGRAM PASCASARJANA FAKULTAS MIPA

UNIVERSITAS BRAWIJAYA

MALANG

2018

LEMBAR PENGESAHAN

Rancang Bangun Pencacah Frekuensi Multi Kanal Dengan Metode Digital Interpolasi Reciprocal Menggunakan High Resolution TDC (*Time to Digital Converter*) pada Low-Cost FPGA

LAPORAN TESIS



Nama Mahasiswa : Imron Rosadi
 NIM : 166090300111011
 Program Studi : S2 Fisika
 Minat : Instrumentasi

Menyetujui,
 KOMISI PEMBIMBING

Ketua

Anggota

Prof.Dr.-Ing. Setyawan P Sakti, M.Eng
 196508251990021001

Hari Arief D, M.Eng, Ph.D
 196909201994121001

Mengetahui
 Ketua Program Studi S2 Fisika
 Mauludi Ariesto P, S.Si., M.Si., Ph.D
 197304122000031013

IDENTITAS TIM PENGUJI JUDUL TESIS

Rancang Bangun Pencacah Frekuensi Multi Kanal Dengan Metode Digital Interpolasi Reciprocal Menggunakan High Resolution TDC (*Time to Digital Converter*) pada *Low-Cost* FPGA

Nama : Imron Rosadi

NIM : 166090300111011

Program Studi : S2 Fisika

Bidang Minat : Fisika Instrumentasi

KOMISI PEMBIMBING

Ketua : Prof.Dr.-Ing. Setyawan P Sakti, M.Eng

Anggota : Hari Arief D, M.Eng, Ph.D

TIM DOSEN PENGUJI

Dosen penguji 1 : DR. Eng. Didik Rahadi Santoso, M.Si.

Dosen penguji 2 : Mauludi Ariesto P, S.Si., M.Si.,Ph.D

Tanggal Ujian :

Sk. Penguji :

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, didalam naskah penelitian Tesis ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi. Apabila ternyata didalam naskah penelitian Tesis ini dapat dibuktikan terdapat unsur-unsur penjiplakan (plagiat) tesis, saya bersedia Tesis (megister) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku.

Malang, Desember 2018

Imron Rosadi



RIWAYAT HIDUP PENULIS

Imron Rosadi lahir di kota Sampang salah satu kota di pulau Madura jawa timur pada 15 Maret 1992.

- Pada tahun 1998 masuk SDN Ragung 1
- Pada tahun 2004 Masuk SMPN 1 Torjun
- Pada tahun 2007 Masuk SMAN 1 Torjun
- Pada tahun 2011 masuk kuliah S1 Jurusan Fisika universitas Brawijaya
- Pada tahun 2016 masuk kuliah S2 Jurusan Fisika Universitas Brawijaya.

Selama kuliah S1, penulis aktif didalam kegiatan Lembaga Dakwah Islam Indonesia (LDII), serta memperdalam ilmu Al Quran dan Hadist.



UCAPAN TERIMA KASIH

Puji syukur kehadiran Allah SWT atas segala nikmat dan karunia-Nya sehingga penulis dapat melaksanakan dan menyelesaikan Tesis yang berjudul **“Rancang Bangun Pencacah Frekuensi Multi Kanal Dengan Metode Digital Interpolasi *Reciprocal* Menggunakan *High Resolution* TDC (*Time to Digital Converter*) pada *Low-Cost* FPGA”** dengan sebaik-baiknya. Sholawat dan salam tetap terucap tercurahkan kepada junjungan Nabi Muhammad SAW.

Dalam pelaksanaan maupun penyusunan laporan Tesis ini penulis banyak mendapatkan bantuan baik moril maupun materil dari berbagai pihak, oleh karena itu penulis ingin menyampaikan ucapan terima kasih yang tidak terhingga kepada:

1. Prof.Dr.-Ing. Setyawan P Sakti, M.Eng dan Hari Arief D, M.Eng, Ph.D selaku pembimbing yang telah memberikan bimbingan, nasehat dan arahan kepada penulis.
 2. DR. Eng. Didik Rahadi Santoso, M.Si. dan Mauludi Ariesto P, S.Si., M.Si., Ph.D selaku dosen penguji yang telah banyak memberikan masukan dan saran kepada penulis sehingga Tesis ini menjadi lebih baik.
 3. Seluruh anggota kelompok peneliti ASMAT: Akhmad Sabarudin M.Sc., Dr.Sc., Djoko Herry Santjojo, Ph.D., Dr.Eng. Masruroh, M.Si., mbak nike, ridha, sukma, rara dan anggota ASMAT yang lain terima kasih atas segala bantuan dan kerjasamanya.
 4. Teman-teman LAB. Teknologi sensor: warits, ragil, esa, intan, Daniel dan masih banyak lagi yang belum bisa saya sebutkan. Terima kasih atas bantuan dan kepeduliannya.
 5. Teman-teman djokam diklaseman tuhaiwin marja, sayyid, mas sakti, azis, bukhori dan masih banyak lagi yang belum disebutkan Alhamdulillah. Jaza kumullohu khoiro.
- Secara khusus penulis ingin mengucapkan terima kasih kepada

1. Abah dan ummi tercinta, kakak tajul ula, adela, Tika yang telah banyak memberikan dukungan dan pengorbanan baik secara moril maupun materil sehingga penulis dapat menyelesaikan studi.
2. Bapak Prof.Dr.-Ing. Setyawan P Sakti, M.Eng yang telah bersedia memberikan ide penelitian tesis ini, terima kasih atas segala ilmu yang telah diberikan kesempatan kepada penulis untuk melanjutkan Pendidikan kejenjang Magister. Maaf sebesar-

besarnya atas segala kekeliruan yang telah penulis lakukan selama ini. Saya berharap segala nasehat yang bapak berikan selama ini dapat membuat saya jauh lebih baik kedepannya.

Penulis menyadari bahwa tesis ini masih jauh dari kesempurnaan, maka saran dan kritik yang mendukung dari semua pihak sangat diharapkan. Akhirnya hanya kepada Allah SWT kita kembalikan semua urusan dan semoga tesis ini dapat bermanfaat bagi semua pihak. Semoga Allah SWT meridhoi dan dicatat sebagai amal ibadah disisiNya, Amin.

Malang, 25 Desember 2018

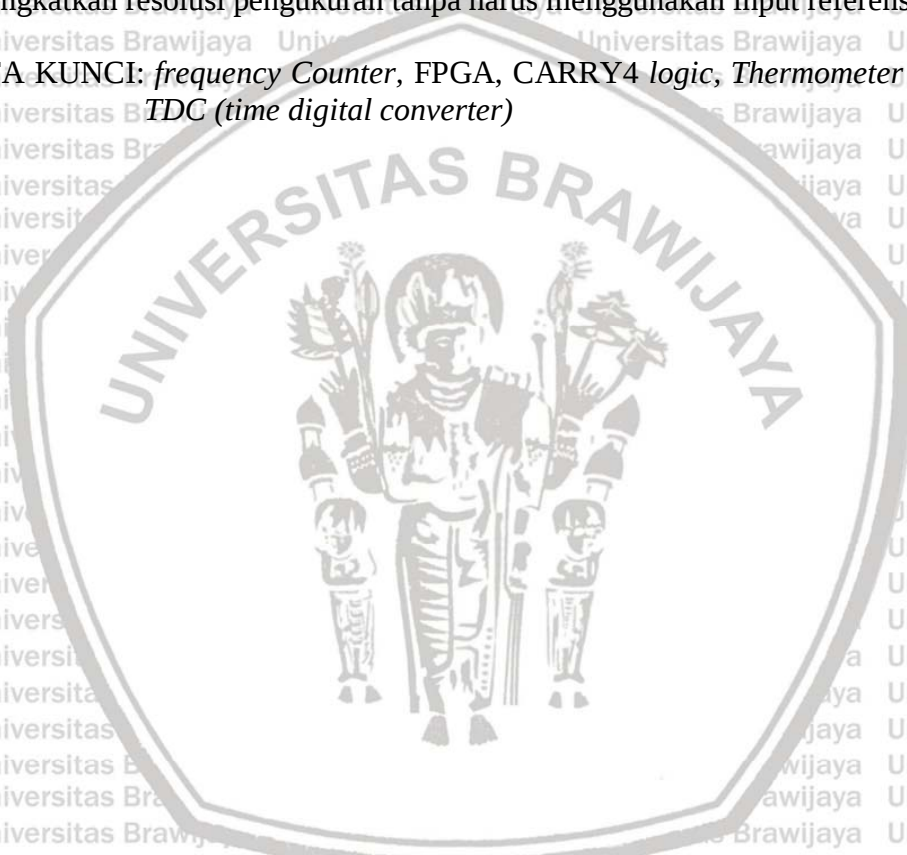
Penulis



RINGKASAN

Frequency counter merupakan instrument yang mengukur frekuensi sinyal input dan frekuensi referensi (input referensi). Frekuensi dan waktu interval merupakan besaran fisika yang harus diukur dengan tingkat presisi yang sangat tinggi, dan pada saat ini *counter* yang sangat canggih dibangun hanya dalam sebuah chip kecil saja, mengubah kuantitatif fisik menjadi frekuensi merupakan metode pendekatan yang paling banyak digunakan dalam mendesain instrumen elektronik. Pengembangan sistem embedded berbasisi FPGA dan mikrokontroler menghasilkan sistem pengukuran frekuensi dengan resolusi pengukuran tinggi. Metode yang digunakan dalam membangun pencacah frekuensi adalah penggabungan antara metode *reciprocal* dan TDC (*time digital converter*). *Thermometer-code* menggunakan CARRY4 logic FPGA digunakan sebagai *delay line*. Sistem embedden dikembangkan ini dapat meningkatkan resolusi pengukuran tanpa harus menggunakan input referensi yang tinggi.

KATA KUNCI: *frequency Counter*, FPGA, CARRY4 logic, *Thermometer code*, *Reciprocal*, TDC (*time digital converter*)



SUMMARY

The frequency counter is an instrument that measures the input signal frequency and reference frequency (reference input). Frequency and interval times are physical quantities that must be measured with very high levels of precision, and at the moment very sophisticated counters are built only on a small chip, changing quantitative physical to frequency is the most widely used approach to designing electronic instruments. The development of an embedded FPGA and microcontroller system produces a frequency measurement system with high measurement resolution. The method used in building frequency counters is a combination of reciprocal and TDC methods (time digital converter). The Thermometer-code CARRY4 logic FPGA is used as delay line. This embedded system developed can increase measurement resolution without having to use high reference inputs.

KEYWORDS: Frequency counter, FPGA, CARRY4 logic, Thermometer code, Reciprocal, TDC (time digital converter)



KATA PENGANTAR

Segala puji hanya milik Allah SWT atas segala rahmatNya. Sehingga penulis dapat menyelesaikan tesis yang berjudul **“Rancang Bangun Pencacah Frekuensi Multi Kanal Dengan Metode Digital Interpolasi Reciprocal Menggunakan TDC High Resolution (Time Digital to Converter) pada Low-Cost FPGA”**

Penulis menyampaikan rasa terima kasih untuk semua pihak yang telah ikut berperan dalam proses penyelesaian proposal tesis ini. Penulis juga sangat mengharapkan saran dan kritik dari pembaca untuk perbaikan proposal tesis ini.

Malang, Desember 2018

Imron Rosadi

166090300111011



DAFTAR ISI

KATA PENGANTAR.....10

DAFTAR ISI.....11

DAFTAR GAMBAR.....13

BAB I..... ERROR! BOOKMARK NOT DEFINED.

PENDAHULUAN ERROR! BOOKMARK NOT DEFINED.

1.1. LATAR BELAKANG.....**ERROR! BOOKMARK NOT DEFINED.**

1.2. RUMUSAN MASALAH.....**ERROR! BOOKMARK NOT DEFINED.**

1.3. TUJUAN PENELITIAN.....**ERROR! BOOKMARK NOT DEFINED.**

1.4. MANFAAT PENELITIAN**ERROR! BOOKMARK NOT DEFINED.**

BAB II..... ERROR! BOOKMARK NOT DEFINED.

TINJUAN PUSTAKA..... ERROR! BOOKMARK NOT DEFINED.

2.1 FPGA (FIELD PROGRAMMABLE GATE ARRAY)**ERROR! BOOKMARK NOT DEFINED.**

2.2 FREQUENCY COUNTER.....**ERROR! BOOKMARK NOT DEFINED.**

2.3.1 Pencacah Konvensional (conventional counting)**Error! Bookmark not defined.**

2.3.2 Pencacah Metode Reciprocal.....**Error! Bookmark not defined.**

2.3.3 Metode Interpolasi Reciprocal.....**Error! Bookmark not defined.**

2.3 TDC (TIME DIGITAL CONVERTER)**ERROR! BOOKMARK NOT DEFINED.**

BAB III..... ERROR! BOOKMARK NOT DEFINED.

KERANGKA KONSEP PENELITIANERROR! BOOKMARK NOT DEFINED.

BAB IV..... ERROR! BOOKMARK NOT DEFINED.

METODE PENELITIAN ERROR! BOOKMARK NOT DEFINED.

4.1 TEMPAT DAN WAKTU PENELITIAN.....**ERROR! BOOKMARK NOT DEFINED.**

4.2 ALAT DAN BAHAN**ERROR! BOOKMARK NOT DEFINED.**

4.3 TAHAP PENELITIAN.....**ERROR! BOOKMARK NOT DEFINED.**

4.4 DESAIN SISTEM.....**ERROR! BOOKMARK NOT DEFINED.**

4.5 PERANCANGAN DAN UJI SISTEM**ERROR! BOOKMARK NOT DEFINED.**

4.5.1 Perancangan Elektronik.....**Error! Bookmark not defined.**

4.5.2 Perancangan Program.....**Error! Bookmark not defined.**

4.5.3 Komunikasi FPGA ke Mikrokontroller**Error! Bookmark not defined.**

4.5.4 Uji Coba Sistem Pencacah.....**Error! Bookmark not defined.**

4.5.5 Pengujian Pencacahan Kristal Rubidium**Error! Bookmark not defined.**

4.5.6 Pengujian Akurasi Pencacah Frekuensi**Error! Bookmark not defined.**

4.5.7 Pengujian Pencacah Multi Kanal,.....**Error! Bookmark not defined.**

4.5.8 Analisa Data dan Pembahasan.....**Error! Bookmark not defined.**

BAB V ERROR! BOOKMARK NOT DEFINED.

HASIL DAN PEMBAHASAN ERROR! BOOKMARK NOT DEFINED.

5.1 HASIL IMPLEMENTASI DESAIN**ERROR! BOOKMARK NOT DEFINED.**

- 5.2 SISTEM PENCACAH FREKUENSI.....**ERROR! BOOKMARK NOT DEFINED.**
- 5.3 PENGOLAHAN DATA PADA MIKROKONTROLER**ERROR! BOOKMARK NOT DEFINED.**
- 5.4 PENGUJIAN PENCACAH FREKUENSI...**ERROR! BOOKMARK NOT DEFINED.**
- 5.4.1. PENGUJIAN KESTABILAN FREKUENSI**ERROR! BOOKMARK NOT DEFINED.**
- 5.4.2. PENGUJIAN AKURASI PENCACAHAN .**ERROR! BOOKMARK NOT DEFINED.**

BAB VIERROR! BOOKMARK NOT DEFINED.

PENTUP.....ERROR! BOOKMARK NOT DEFINED.

- 6.1. KESIMPULAN.....**ERROR! BOOKMARK NOT DEFINED.**

DAFTAR PUSTAKA.....ERROR! BOOKMARK NOT DEFINED.



DAFTAR GAMBAR

Gambar 2.1 Struktur internal FPGA	Error! Bookmark not defined.
Gambar 2.2 Blok Diagram Modul DCM (<i>digital clock manager</i>).Error! Bookmark not defined.	
Gambar 2.3 <i>Asynchronous Counter</i>	Error! Bookmark not defined.
Gambar 2.4 Dasar blok diagram Pencacah digital (<i>digital counter</i>)Error! Bookmark not defined.	
Gambar 2.5 Blok diagram pencacah frekuensi metode konvensional.Error! Bookmark not defined.	
Gambar 2.6 Logiak pencacahan metode <i>reciprocal counters</i> Error! Bookmark not defined.	
Gambar 2.7 error relatif ± 1 menggunakan metode <i>reciprocal</i> dan metode konvensional untuk frekuensi input kurang dari frekuensi referensi (<i>time base</i>).Error! Bookmark not defined.	
Gambar 2.8 Interpolasi analog	Error! Bookmark not defined.
Gambar 2.9 Interpolasi Digital	Error! Bookmark not defined.
Gambar 2. 10 Prinsip pengukuran waktu.....	Error! Bookmark not defined.
Gambar 2. 11 Diagram waktu dari <i>time measurement</i> Error! Bookmark not defined.	
Gambar 2. 12 blok diagram <i>single channel</i> TDC. .Error! Bookmark not defined.	
Gambar 2. 13 <i>delay tap</i> dari <i>carry chain</i> sebagai implementasi <i>CARRY4 logic</i> . CO0 dan CO3 merupakan <i>fine time data output</i> . CIN dan COUT digunakan sbagai <i>Port</i> penyambung.	Error! Bookmark not defined.
Gambar 3.1 Diagram Konsep Penelitian	Error! Bookmark not defined.
Gambar 4.1 Diagram alir langkah penelitian.....	Error! Bookmark not defined.
Gambar 4. 2 Blok pencacah frekuensi metode Digital Interpolasi <i>Reciprocal</i> TDC.Error! Bookmark not defined.	
Gambar 4. 3 Struktur <i>multi-tap delay line</i> menggunakan <i>CARR4 logic</i> pada SLICE FPGA Spartan-6.....	Error! Bookmark not defined.
Gambar 4. 4 Implementasi <i>code thermometer interpolator</i> TDC pada FPGA dengan <i>carrychain</i> sebagai <i>delay line</i>	Error! Bookmark not defined.
Gambar 4. 5 <i>pipelined thermometer to binary decoder</i> dengan <i>counter bit</i> pada tahap akhir	Error! Bookmark not defined.
Gambar 4. 6 <i>Main idea clock interpolator</i>	Error! Bookmark not defined.
Gambar 4. 7. <i>Timing</i> diagram implementasi TDC pada metode interpolasi <i>reciprocal</i>	Error! Bookmark not defined.
Gambar 4. 8 <i>Continuous Timestamping Counter</i> ..Error! Bookmark not defined.	
Gambar 4. 9 Blok diagram sistem pencacah frekuensi metode Digital Interpolasi <i>Reciprocal</i> TDC.....	Error! Bookmark not defined.
Gambar 4.10 Flowchart program pencacah frekuensi.Error! Bookmark not defined.	
Gambar 5. 1 <i>Board</i> sistem pencacah frekuensi	Error! Bookmark not defined.
Gambar 5. 2 <i>Top Level</i> arsitektur sistem pencacah multi kanal.Error! Bookmark not defined.	
Gambar 5. 3. <i>Listing</i> program <i>time_base 1s</i> dan <i>time_sys</i> Error! Bookmark not defined.	
Gambar 5. 4 <i>Listing</i> program kontrol stop dan start <i>counting</i> .Error! Bookmark not defined.	
Gambar 5. 5 <i>Listing</i> program <i>counter</i> sinyal input dan sinyal referensiError! Bookmark not defined.	
Gambar 5. 6 <i>top level</i> arsitektur TDC.....	Error! Bookmark not defined.
Gambar 5. 7 mekaniseme kerja TDC.....	Error! Bookmark not defined.

Gambar 5. 8 *listing* program Interpolator pada sistem pencacahan frekuensi. **Error! Bookmark not defined.**

Bookmark not defined.

Gambar 5. 9. *Top level Pipeline Thermometer Encoder*. **Error! Bookmark not defined.**

Gambar 5. 10 register *buffer* penyimpanan cacahan dengan metode *reciprocal*. **Error! Bookmark not defined.**

Bookmark not defined.

Gambar 5. 11 Mekanisme Multiplekser pada Modul UART. **Error! Bookmark not defined.**

Gambar 5. 12 *flowchart* proses pengolahan data pada Mikrokontroler. **Error! Bookmark not defined.**

Gambar 5. 13 Grafik Hasil pencacaha Kristal Rubidium 10MHz. **Error! Bookmark not defined.**

Gambar 5. 14 Grafik hasil pencacahan oscillator 16MHz. **Error! Bookmark not defined.**

Gambar 5. 15 Grafik hasil pencacahan oscillator 10MHz. **Error! Bookmark not defined.**

Gambar 5. 16 Grafik Hasil cacahan oscillator 50MHz. **Error! Bookmark not defined.**

Gambar 5. 17 Grafik Hasil cacahan oscillator 20MHz. **Error! Bookmark not defined.**



BAB I

PENDAHULUAN

1.1. Latar Belakang

Frequency counter merupakan instrument penghitung frekuensi sinyal input. Frekuensi dan waktu interval merupakan besaran fisika yang harus diukur dengan tingkat presisi yang sangat tinggi, dan pada saat ini *counter* yang sangat canggih dibangun hanya dalam sebuah chip kecil saja, mengubah kuantitatif fisik menjadi frekuensi merupakan metode pendekatan yang paling banyak digunakan dalam mendesain instrumen elektronik. Dengan demikian *counter* merupakan instrumen yang bisa digunakan dalam berbagai aplikasi dengan tingkat resolusi dan presisi pengukuran yang sangat tinggi. *Frequency Counter* telah banyak dikembangkan, salah satu contohnya adalah *Frequency Counter* menggunakan Mikrokontroler, CPLD dan FPGA dengan resolusi pengukuran 1 Hz sampai 0.1 Hz (Rubiola *et al.*, 2016).

Salah satu contoh aplikasi frekuensi counter adalah pengembangan sensor QCM sebagai biosensor. *Quartz Crystal Microbalance* (QCM) merupakan sebuah resonator Kristal kuarsa yang digunakan untuk mendeteksi perubahan massa yang sangat kecil dari orde mikrogram sampai nanogram secara *real time* dengan mendeteksi perubahan frekuensi. Selain itu, kelebihan utama dari QCM adalah kemampuan sensitivitas dan selektifitas yang tinggi untuk mendeteksi target *analyte* dengan penerapan proses *coating* pada permukaan QCM (Ayad and Minisy, 2016). Oleh karena itu, sensor ini dapat digunakan untuk memantau *thin film deposit rate*, berat molekul terhadap afinitas kinetik permukaan pada sistem kimia atau biologi yang praktis dan ekonomis dengan ambang batas pengukuran tertentu (Lal and Tiwari, 2018).

Penghitungan frekuensi secara langsung (metode konvensional) telah lama digantikan oleh metode perhitungan *reciprocal* klasik dengan mengukur periode rata-rata sinyal input dibandingkan dengan input referensi (clock referensi), keuntungan yang diperoleh dari metode tersebut adalah ketidakpastian pengukuran $1/n$ (n adalah sinyal input), dengan demikian dapat disimpulkan bahwa metode diatas dibatasi oleh referensi input (*clock* referensi).

Resolusi yang lebih tinggi diperoleh dengan mengukur fraksi sinyal input dengan menggunakan interpolator. Interpolasi sederhana bekerja pada frekuensi yang tetap, dengan demikian pengukuran tidak dipengaruhi oleh besar dan kecil input referensi. Teknik-Teknik tersebut antara lain *Nutt Interpolator*, *Frequency Vernier*, dan *Thermometer Code*. Semua metode tersebut memiliki resolusi pengukuran pada orde *pico second*.

Oleh karena itu, dalam penelitian ini permasalahan yang akan dibahas adalah bagaimana mengembangkan instrumen frekuensi *counter* dengan resolusi cacahan mencapai orde *pico second* dengan menggunakan metode digital interpolasi menggunakan TDC (*time digital converter*) untuk mengurangi ketergantungan *input* referensi yang sangat tinggi.

Berdasarkan permasalahan dan kebutuhan pada proses pengukuran menggunakan metode digital interpolasi yang telah dikemukakan di atas, maka pada penelitian ini dibangun *Frequency Counter* dengan resolusi orde *pico second* dengan menggunakan metode Digital interpolasi *Reciprocal* dengan memanfaatkan *high resolution* TDC (*time digital converter*) yang diterapkan pada *low-cost* FPGA. Selain dapat meningkatkan resolusi pengukuran, dengan menggunakan metode Digital Interpolasi juga dapat meningkatkan ketelitian hasil pengukuran atau eror yang sangat kecil. Sehingga dari penelitian ini diharapkan menghasilkan suatu sistem instrumentasi pengukuran frekuensi dengan berprinsip *low-cost* dan resolusi yang sangat tinggi.

1.2. Rumusan Masalah

Rumusan masalah yang dibahas pada penelitian ini adalah:

a) Bagaimana mendapatkan frekuensi *counter* multi kanal dengan resolusi yang sangat tinggi.

b) Bagaimana mengembangkan Digital Interpolasi *Reciprocal* dengan menggunakan TDC pada *low-cost* FPGA.

c) Bagaimana mengembangkan pencacah frekuensi dengan metode Digital Interpolasi *Reciprocal* menggunakan TDC dengan memanfaatkan *carry-chain* pada FPGA.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini secara umum adalah mengembangkan suatu perangkat pengukuran frekuensi yang memiliki resolusi dan kesetabilan yang sangat tinggi. Sedangkan tujuan khusus dari penelitian ini adalah:

a) Mengembangkan pencacah frekuensi berbasis Xilinx Spartan-6 FPGA dengan menggunakan Digital Interpolasi.

b) Mengembangkan perangkat pencacah frekuensi resolusi tinggi dengan menggunakan metode Digital Interpolasi *Reciprocal* dengan memanfaatkan *carry chain* pada FPGA untuk membangun TDC.

c) Mengembangkan perangkat pencacah frekuensi dengan mengutamakan *low-Cost* dan resolusi tinggi.

1.4. Manfaat Penelitian

Penelitian ini diharapkan menghasilkan sistem instrumentasi dan pengukuran frekuensi dengan resolusi yang sangat tinggi (orde *picosecond*). Sehingga dapat menunjang penelitian-penelitian yang memerlukan suatu alat ukur penghitung frekuensi dengan presisi dan resolusi sangat tinggi.



BAB II

TINJUAN PUSTAKA

2.1 FPGA (*Field Programmable Gate Array*)

FPGA merupakan perangkat yang tersusun atas matrik *gate array logic circuit* yang dapat dikonfigurasi ulang. Ketika FPGA dikonfigurasi, sirkuit internal pada FPGA akan terhubung dengan cara menciptakan implementasi perangkat keras dari *software* aplikasi. Tidak seperti *processor*, FPGA menggunakan perangkat khusus untuk memproses logika dan tidak memiliki sistem operasi. Satu FPGA dapat menggantikan ribuan komponen diskrit dengan menggabungkan jutaan gerbang logika dalam satu chip sirkuit terpadu (IC). *Internal resources* dari chip FPGA terdiri dari matrik *configurable logic block* (CLB) yang dikelilingi oleh blok *input/output* yang ditunjukkan pada Gambar 2.3.

Blok logika pada FPGA diimplementasikan menggunakan *multiple level low fan-in gates*, sehingga menciptakan desain yang lebih ringkas dibandingkan dua level AND-OR logic.

Terdapat dua acara untuk mengkonfigurasi FPGA antara lain:

1. Persimpangan antar blok logika dan
2. Fungsi dari setiap blok logika.

Logic Block dari FPGA dapat dikonfigurasi sedemikian rupa sehingga dapat memberikan fungsionalitas sesederhana transistor atau serumit seperti mikroprosesor. Hal tersebut dapat digunakan untuk mengimplementasikan fungsi kombinasional dan sekuensial yang berbeda. *Logic block* pada FPGA dapat diimplementasikan beberapa fungsi dibawah ini:

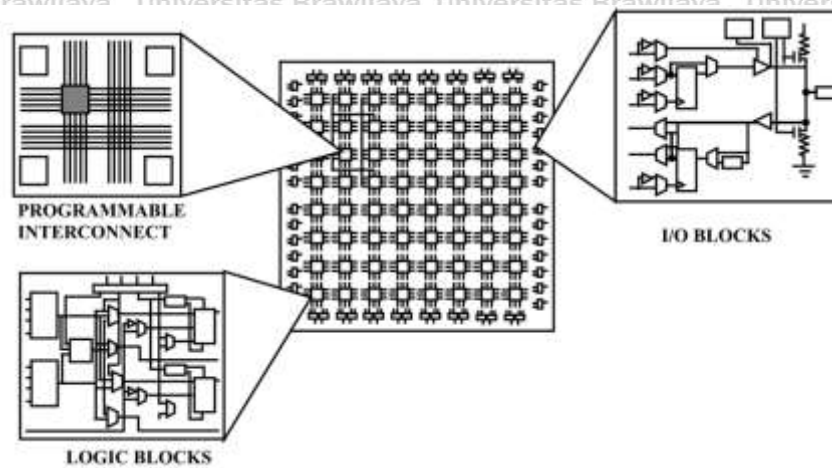
2.1 *Transistor Pairs*

2.2 *Combinational gates* seperti gerbang NAND atau gerbang XOR

2.3 *N-input Lookup table*

2.4 Multiplexer

2.5 Structure Wide fan-in AND-OR

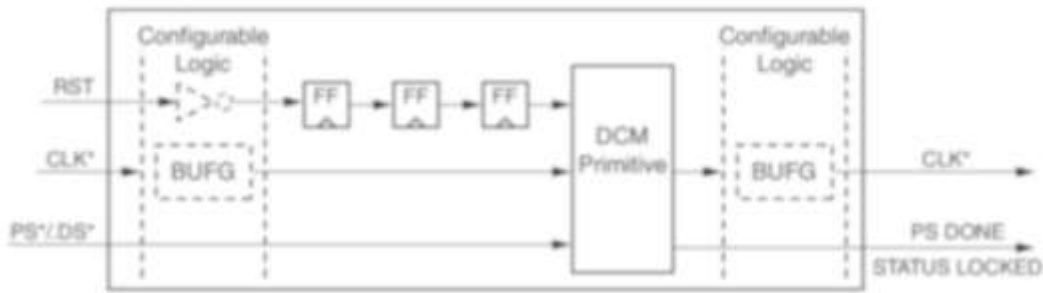


Gambar 2.1 Struktur internal FPGA

Routing pada FPGA terdiri dari *wire segment* dengan panjang bervariasi yang saling terhubung melalui *switch* yang dapat diprogram. Densitas blok logika yang digunakan tergantung pada panjang dan jumlah *wire segment* pada routing. Jumlah segment yang digunakan untuk *interconnection* memiliki ciri *tradeoff* antara densitas blok logika dan jumlah area yang digunakan untuk routing (CHTOUROU dkk. 2017).

Selain *logic blok* didalam FPGA juga terdapat DCM (*digital clock manager*) khususnya pada FPGA Xilinx Spartan. DCM merupakan kemampuan *clocking* tingkat lanjut pada FPGA Xilinx Spartan. DCM pada FPGA digunakan untuk mengimplementasikan *delay locked loop*, *digital phase shifter*, dan *digital spread spectrum* (Xilinx, 2009).

DCM merupakan solusi dari berbagai masalah yang menyangkut *clocking*, khususnya pada aplikasi *high-performance* dan frekuensi tinggi seperti, menghilangkan *clock skew* didalam perangkat FPGA maupun komponen eksternal untuk meningkatkan kinerja sistem secara keseluruhan serta menghilangkan *clock distribution delay*. Selain itu, DCM dapat bekerja untuk *phase shift clock* sinyal baik friksi tetap dari priode *clock* atau peningkatan nilai periode *clock* dan filter *clock input jitter*.



Gambar 2.2 Blok Diagram Modul DCM (*digital clock manager*).

2.2 Frequency Counter

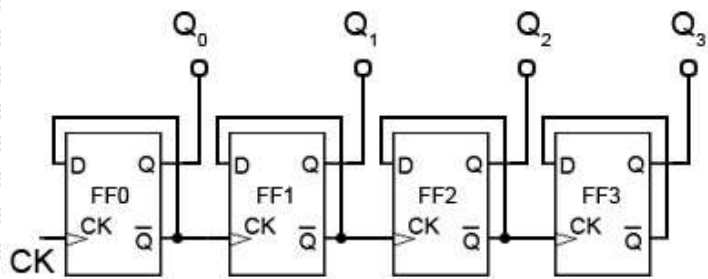
Pencacah frekuensi merupakan perangkat utama dalam sistem pengukuran QCM.

Resolusi pencacah akan sangat menentukan performa sistem pengukuran tersebut. Tingkat presisi dan stabilitas adalah faktor utama dari pencacah frekuensi, oleh sebab itu, pencacah frekuensi harus memiliki kemampuan untuk mendeteksi perubahan frekuensi QCM dibawah 1 Hz setiap satu detik (Sakti, 2014).

Dalam bidang elektronika digital, *counter* merupakan rangkaian logika sekuensial yang tersusun dari beberapa rangkain *flip-flop*. secara umum pencacah digital “*counter*” dibagi menjadi dua jenis yaitu, *synchronous* dan *asynchronous counter*.

Synchronous counter umumnya terdiri dari elemen memori yang diimplementasikan menggunakan *flip-flop*, dan elemen kombinasi berupa gerbang logika. Pada *Synchronous counter*, pulsa input dimasukkan ke pulsa input pada semua *flip-flop* secara simultan (langsung). *Synchronous counter* dikenal juga sebagai paralel sekuensial, contoh dari *synchronous counter* adalah *ring counter* dan *Johnson counter*. Berbeda dengan *Asynchronous counter* transisi *output flip-flop* berfungsi sebagai sumber *trigger* untuk *flip-flop* selanjutnya.

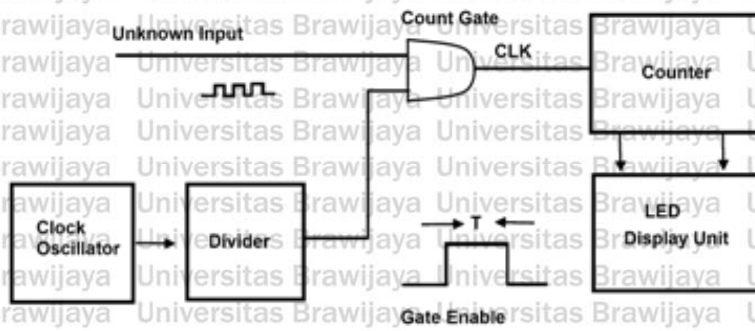
Dengan kata lain, *input pulsa clock* untuk semua *flip-flop*, kecuali yang *flip-flop* pertama yang di *trigger* bukan dari *clock* pulsa yang datang, selanjutnya melalui transisi keluaran *flip-flop* sebelumnya. *Asynchronous counter* disebut juga *serial sequential circuit*, contoh dari *asynchronous counter* adalah *binary Ripple Counter* dan *Up Down Counter*.



Gambar 2.3 Asynchronous Counter

Untuk mengukur frekuensi dari sinyal beresilasi dapat digunakan pencacah digital (*digital counter*). Pengukuran dapat dilakukang dengan cara langsung yaitu mendeteksi *rising edge* sinyal input (frekuensi yang diukur), dihitung melalui *time gate* (dikontrol oleh *clock* referensi) dan nilai frekuensi diperoleh dengan membagi jumlah pulsa dengan interval waktu selama *gate* terbuka. Gambar 2.5 menunjukkan dasar blok diagram *digital counter*, terdapat beberapa komponen utama pada pencacah digital antara lain *Edge Detector* dan *Edge Counters* (Valdes *et al.*, 2008).

Dalam perkembangannya, terdapat beberapa metode dalam membangun sebuah instrument pencacah frekuensi digital (*digital counter*), metode tersebut antara lain *conventional counting*, *reciprocal counting*, *Interpolating Reciprocal Counting*, dan *Multiple Time Stamp Average Continuous Counting* (Johansson, 2005). Dari beberapa metode diatas, *error* pengukuran frekuensi yang terjadi didominasi oleh dua faktor, yaitu faktor akurasi dan stabilitas dari sinyal *clock* referensi yang digunakan (*time base*) dan *error* relatif penentuan waktu *gate time* untuk interval waktu pencacah frekuensi (SiTime, 2013). Metode yang umum digunakan untuk pengukuran frekuensi adalah metode konvensional dan metode *reciprocal* (mencacah sinyal input dan sinyal referensi dalam interval waktu yang sama) (Boven, 2007).

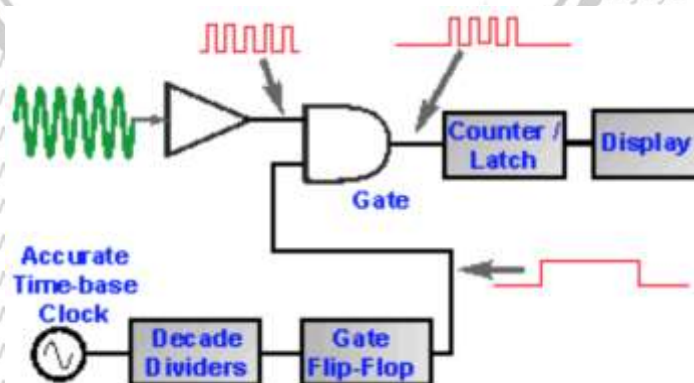


Gambar 2.4 Dasar blok diagram Pencacah digital (*digital counter*)

Tingkat sensitivitas dari pencacah frekuensi dapat didefinisikan sebagai kemampuan untuk mencacah *amplitude* sinyal terendah, dengan kata lain sensitivitas adalah ukuran dari level *hysteresis* pada input komparator (Poole, 2015).

2.3.1 Pencacah Konvensional (*conventional counting*)

Metode konvensional merupakan metode yang pertama dan paling sederhana dalam proses pengukuran frekuensi. Metode konvensional memiliki kinerja dan fleksibilitas pengukuran yang sangat rendah. Pengukuran frekuensi secara konvensional hanya menggunakan register sederhana untuk menghitung setiap siklus dari sinyal input selama interval waktu 1s. Frekuensi dihitung dengan cara mencacah jumlah transisi naik (*falling edge*) dan transisi sinyal turun (*falling edge*) dari sinyal input (*measured clock*) seperti Gambar 2.7 (Johansson, 2005).



Gambar 2.5 Blok diagram pencacah frekuensi metode konvensional.

Pencacah frekuensi konvensional terdapat dua sinyal masukan yaitu sinyal referensi (*accurate time-base clock*) dan sinyal input yang akan diukur. Sinyal input dirubah menjadi sinyal pulsa dengan menggunakan rangkain *Schmitt trigger* dengan level *trigger* tertentu. *Gate* merupakan waktu interval sinyal input yang dicacah, *Gate* atau *Time Base* didapatkan dari *clock* yang telah diketahui nilainya dengan tingkat akurasi tinggi. Pada umumnya *clock (Time Base)* yang digunakan adalah osilator Kristal eksternal.

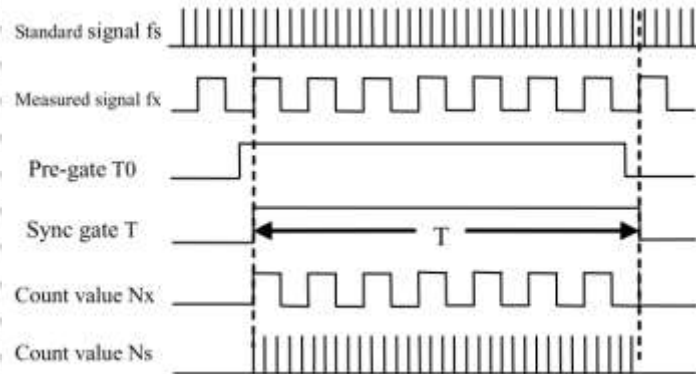
2.3.2 Pencacah Metode *Reciprocal*

Pada awal tahun 1980-an, pencacah frekuensi berbasis mikrokomputer mulai menggunakan metode pengukuran timbal balik / *reciprocal* (Johansson, 2005). *Reciprocal Counters* terdiri dua penghitung (*counters*) untuk mencacah sinyal masuk dan mencacah sinyal referensi (*time base*). Interval waktu cacah diperoleh dari besar periode yang dibutuhkan untuk satu siklus sinyal referensi (*time base*).

Pada masing-masing *caunter*, proses memulai dan berhenti pencacahan dikontrol oleh sinyal yang disebut *gate*. Jika *gate* disinkronkan dengan *clock*, maka *gate time* dikontrol dengan tepat dan jumlah *event* dihitung dengan beberapa ketidakpastian ± 1 *event*. Pada waktu bersamaan, *gate* disinkronkan dengan sinyal *input*, jumlah *event* sinyal input dihitung dengan ketidakpastian ± 1 periode sinyal *input* (Packard, 1997).

Kelebihan penggunaan metode *reciprocal* dengan metode konvensional adalah dapat mencapai presisi yang sama pada seluruh rentang pengukuran frekuensi dan tidak terpengaruh dengan ukuran frekuensi sinyal input (Fang and Chen, 2012). Metode *reciprocal* juga dikenal dengan metode *multi-cycle synchronous*, ciri yang paling menonjol dari metode ini adalah *time gate* bukan nilai tetap tetapi nilai yang terkait dengan besar sinyal yang diukur. Perhitungan dilakukan secara bersama antara sinyal referensi (*time base*) dan sinyal *input*. Kemudian untuk memperoleh frekuensi sinyal input yang diukur melalui rumus matematika. Karena *gate time* merupakan kelipatan bilangan dari periode sinyal terukur, maka eror pengukuran mencapai ± 1

periode (Chen, Liu and Liu, 2011) Gambar 2.8 ditunjukkan logika pencacahan metode *reciprocal counters*.



Gambar 2.6 Logik pencacahan metode *reciprocal counters*

Dari schematic diatas diperoleh persamaan:

$$T = \frac{N_x}{f_x} = \frac{N_s}{f_s} \quad 2.1$$

Setelah melakukan deformasi, diperoleh:

$$f_x = \frac{N_x}{N_s} f_s \quad 2.2$$

Deferensial persamaan diatas, diperoleh:

$$df_s = \frac{f_s}{N_s} dN_x - \frac{N_x}{N_s^2} f_s dN_s + \frac{N_x}{N_s} df_s \quad 2.3$$

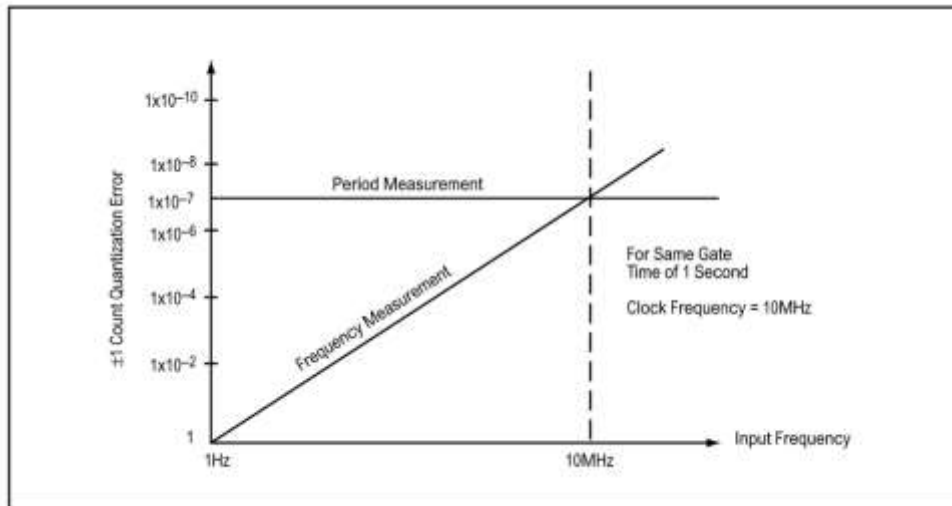
Untuk $dN_x = 0$, $dN_s = \pm 1$, dan $\frac{N_x}{f_x} = \frac{N_s}{f_s}$

Dengan demikian:

$$\frac{df_x}{f_x} = \pm \frac{1}{N_s} + \frac{df_s}{f_s} \quad 2.4$$

Dari persamaan 2.8 dapat disimpulkan metode *reciprocal counters* tidak ada hubungan dengan frekuensi sinyal sinyal terukur, hanya terkait dengan *gate time* dan frekuensi sinyal referensi (*time base*). Semakin lama *time gate* T dan semakin tinggi frekuensi referensi (*time base*) f_s , semakin kecil kesalahan relatif dari frekuensi yang terukur.

Pencacah frekuensi dengan metode *reciprocal* bersifat konstan dan sangat bergantung pada frekuensi *clock* referensi (*time base*). Berbeda dengan metode konvensional yang menghasilkan error yang berbanding lurus dengan frekuensi input yang terukur (Packard, 1997).



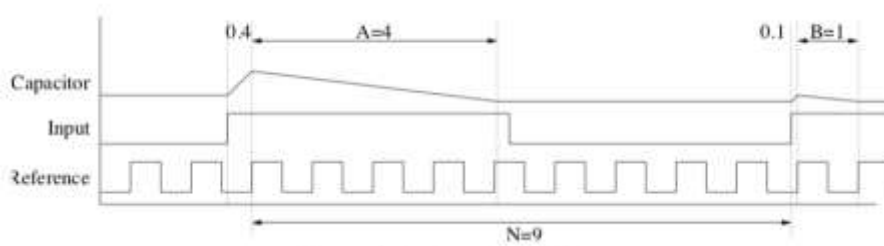
Gambar 2.7 error relatif ± 1 menggunakan metode *reciprocal* dan metode konvensional untuk frekuensi input kurang dari frekuensi referensi (*time base*).

2.3.3 Metode Interpolasi *Reciprocal*

Meningkatkan frekuensi referensi (*time base*) pada metode *reciprocal counter* akan meningkatkan resolusi pengukuran. Tetapi menyebabkan desain *counter* akan lebih rumit dan mahal. Oleh karena itu, digunakan Interpolasi fase sinyal referensi (*time base*) untuk mengurangi ketergantungan frekuensi yang sangat besar dan error pada saat kontrol *gate time*.

Pencacah frekuensi menggunakan interpolasi banyak digunakan pada tahun 1960an – 1970an. Pencacah menggunakan interpolasi menggunakan proses *charging* pada kapasitor yang sangat singkat selama interval antara transisi naik pada sinyal input dan transisi naik pada sinyal referensi (*time base*) selanjutnya. *Charge* yang tersimpan berbanding lurus dengan panjang siklus sinyal referensi (*time base*), muatan (*charge*) kemudian diukur dengan menghitung beberapa lama waktu yang diperlukan untuk *discharge* kapasitor dengan arus yang

sangat kecil. Peningkatan resolusi secara teoritis sama dengan rasio *charge* dan *discharge* arus (Boven, 2007).



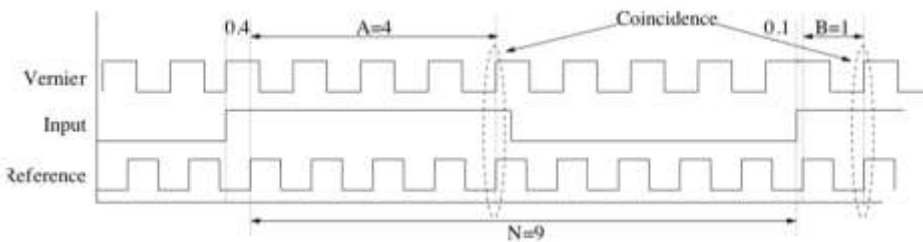
Gambar 2.8 Interpolasi analog

Pada contoh diatas, *counter* utama antara dua *rising edge* sinyal input berjumlah 9 siklus dari *clock* referensi (*time base*). *Start* pengukuran 0.4 siklus lebih cepat dibandingkan dengan sinyal referensi (*time base*). Pada contoh *discharge* kapasitor 10 kali lebih lambat ketika *charge*, sehingga *gain* dari rangkaian $Q = 10$. *Discharge counter* membaca 4 siklus diawal pengukuran dan satu siklus pada akhir pengukuran, sehingga periode sinyal input didapatkan dengan persamaan $t_x = t_{ref} * \left(N + \frac{A-B}{Q}\right)$.

Pada awal 1980an, Hewlett-Packard memperkenalkan metode penghitungan waktu pada HP5370. Instrument tersebut menggunakan *clock* referensi internal 200MHz, tetapi resolusi ditingkatkan 256 kali menggunakan interpolasi digital. Hal tersebut, setara dengan *clock* referensi 50sGHz dan resolusi 20ps (Packard, 1997).

HP5370 menggunakan interpolasi dengan menggunakan osilator yang bekerja pada 255/256 sebagai referensi (*time base*). Frekuensi tetap dijaga oleh rangkaian PLL, tetapi permulaan (*start*) pengukuran segera dimulai kembali setelah *rising edge* sinyal input. PSPLO (*Phase Startable Phase Lockable Oscillator*) terus berjalan dengan frekuensi lama, tetapi fase yang berbeda. Karena frekuensi $(1-1/256) f_{ref}$, hal tersebut bertepatan dengan referensi osilator setiap 255 *clock cycle*. Jumlah *cycle* antara memulai kembali dengan *coincidence* selanjutnya merupakan pengukuran langsung dari fraksi *clock cycle* referensi pada saat memulai osilator.

Hal ini dapat dibandingkan dengan cara *Vernier* untuk meningkatkan resolusi pembacaan frekuensi.

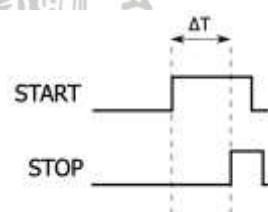


Gambar 2.9 Interpolasi Digital

Pada Gambar 2.11, periode sinyal *Vernier* 1.1 kali sinyal referensi (*time base*) yang dapat meningkatkan 10 kali lipat resolusi pengukuran. *Timing* sinyal input sama seperti pada saat interpolasi analog yang telah dijelaskan sebelumnya. Perhatikan dimana osilator *Vernier* di *restart* ketika *coincidence* pada sinyal referensi.

2.3 TDC (*time digital converter*)

TDC adalah sistem elektronik untuk mengukur interval antara dua kejadian (*rising edge*) dari dua sinyal yang diberikan. Tujuan utama TDC adalah mengubah informasi secara temporal kedalam urutan biner yang dapat dimengerti (diproses selanjutnya) (Dadouche *et al.*, 2015).

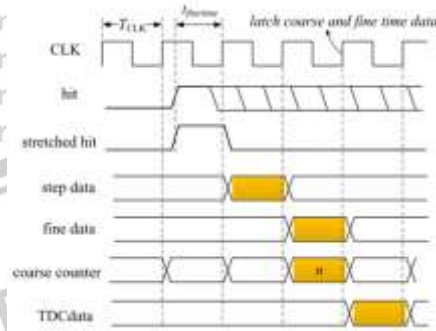


Gambar 2. 10 Prinsip pengukuran waktu

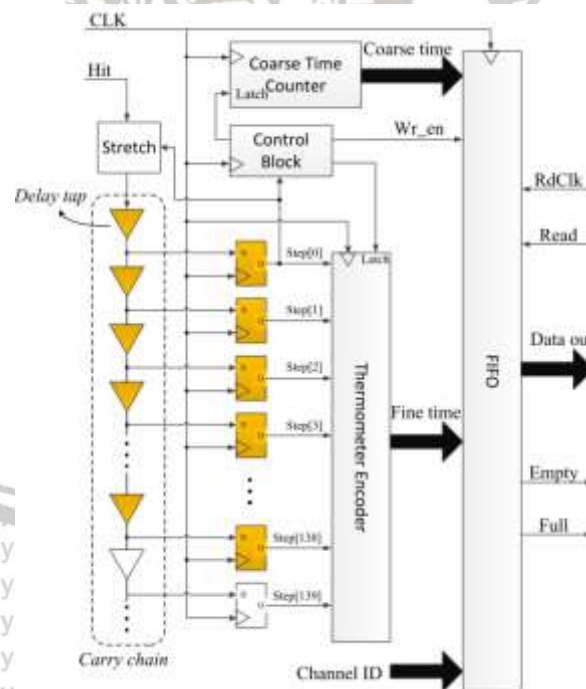
Prinsip proses pengukuran waktu pada metode *single channel* di ilustrasikan pada gambar 2.11 dan gambar 2.12. terdapat dua bagian yang sangat krusial yaitu *carry chain* dan *coarse time counter* dan beberapa *auxiliary* lainnya (Zheng *et al.*, 2017). *Encoder* digunakan mengubah kode *thermometer* menjadi kode biner (*fine data*) pada gambar 2.11 dalam satu siklus *clock*. Blok kontrol berfungsi untuk mengunci (*latches*) *coarse* dan *fine time*. Karena

terdapat *logic delay* untuk mendapatkan *fine time data*, penguncian waktu dilakukan pada saat *rising edge* setelah sinyal *hit* muncul (lihat gambar 2.11). sebagai hasilnya, *time stamp t* dari sinyal *hit* diperoleh sesuai persamaan dibawah ini :

$$t = (n - 2) \cdot T_{CLK} - t_{finedata} \text{ (Wang et al., 2009)}$$



Gambar 2. 11 Diagram waktu dari *time measurement*



Gambar 2. 12 blok diagram *single channel TDC*.

Dalam prakteknya, *dicated fast lookahead carry logic* atau disebut juga CARRY4 digunakan sabagai interpolator waktu. Lokasi CARRY4 terdapat pada *slice* Xilinx Spartan-6 FPGA *configurable logic block* (CLB) dan dapat dibentuk *carry chain* yang

Gambar 2. 13 *delay tap* dari *carry chain* sebagai implementasi CARRY4 logic. CO0 dan CO3 merupakan *fine time data output*. CIN dan COUT digunakan sebagai *Port* penyambung.

BAB III

KERANGKA KONSEP PENELITIAN

Pengukuran frekuensi adalah salah satu metode paling dasar dan paling penting dalam elektronik *measurement*. Sinyal frekuensi memiliki anti interfensi yang kuat, mudah ditransmisikan, dan dapat diukur dengan tingkat presisi yang sangat tinggi. Oleh karena itu, studi tentang metode pengukuran frekuensi penting khususnya pada *engineering application*. Metode pengukuran frekuensi yang umum digunakan antara lain *direct frequency measurement*, *direct period measurement*, *reciprocal frequency counter* dan lain-lain.

Direct frequency counting adalah metode menghitung jumlah pulsa N dalam waktu t, kemudian menghitung jumlah pulsa per satuan waktu, yaitu frekuensi sinyal yang diukur. *Direct period measurement* merupakan metode dimana proses yang pertama dilakukan mengukur periode T dari sinyal yang diukur, kemudian menghitung frekuensi sinyal dengan $f = 1/T$. namun kedua metode tersebut menghasilkan eror pengukuran ± 1 dari sinyal yang diukur. Pada perkembangan selanjutnya terdapat metode selain metode yang telah disebutkan diatas yaitu *reciprocal counter* (pengukuran timbal-balik), metode ini melakukan penghitungan secara bersamaan antara sinyal yang diukur dengan sinyal referensi yang telah diketahui nilainya, biasanya *clock* referensi memiliki nilai yang lebih besar dari sinyal input. metode *reciprocal counter* memiliki eror kurang dari 1, tetapi kelemahan dari metode ini memerlukan input sinyal referensi yang besar, yang berarti pengukuran dibatasi oleh besarnya nilai sinyal referensi.

Setelah memaparkan beberapa metode pengukuran frekuensi dan kelebihan dan kekurangan masing-masing metode, terungkap fakta bahwa dalam proses pengukuran frekuensi terdapat beberapa bagian yang tidak ter-*counting* akibat adanya perbedaan Periode sinyal input dan sinyal referensi. Meski sangat kecil, tetapi hal tersebut sangatlah penting jika sistem pengukuran frekuensi mengharuskan memiliki tingkat presisi tinggi, salah satu contoh aplikasi pengukuran frekuensi yang membutuhkan tingkat presisi tinggi adalah frekuensi *counter* untuk mendeteksi massa pada sensor QCM sebagai biosensor. Oleh sebab itu, perlu ada metode baru untuk meningkatkan resolusi dan presisi pengukuran, salah satunya dengan menggunakan digital interpolator menggunakan TDC (*time digital converter*), metode ini merupakan penggabungan antara metode *reciprocal* dan *reciprocal*. Diharapkan dengan menggunakan metode interpolasi *reciprocal* TDC dapat meningkatkan resolusi pengukuran sampai orde *pico second*.

Pada era modern, teknologi *nano* berkembang sangat pesat, termasuk penggunaan teknologi *nano* dalam sistem pengukuran. Oleh Karena itu, FPGA merupakan pilihan yang tepat untuk menerapkan metode Digital Interpolasi *reciprocal* TDC karena memiliki dimensi kecil, murah, membutuhkan daya rendah, dan dilengkapi dengan fasilitas *CARRY4 logic* untuk membangun TDC (*time digital converter*).

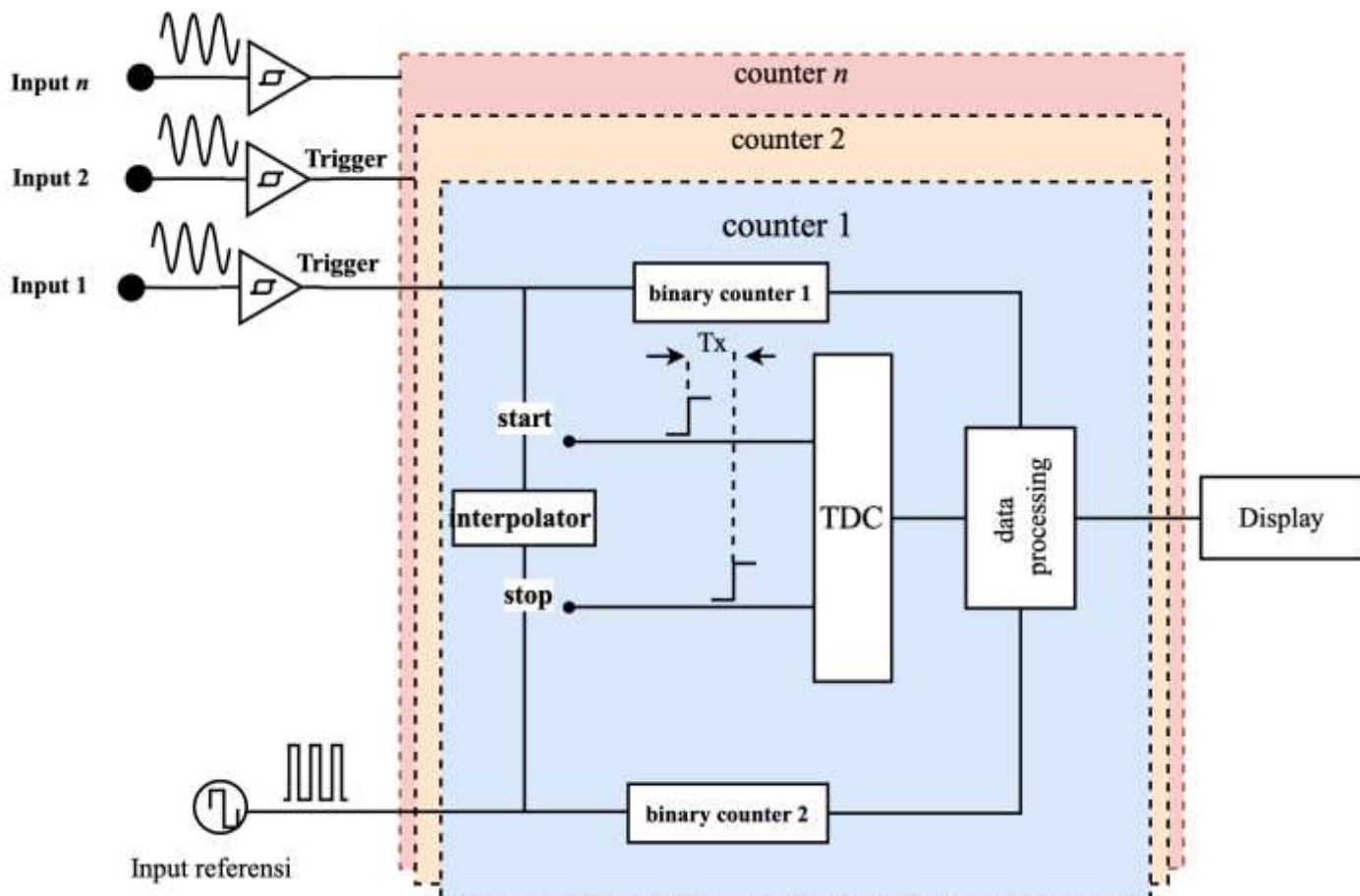
Pokok masalah yang akan dibahas dalam penelitian ini diantaranya adalah bagaimana mendesain frekuensi *counter* menggunakan metode Digital Interpolasi *Reciprocal* dan TDC untuk meningkatkan resolusi dan ketelitian hasil pengukuran menggunakan Xilinx Spartan-6 FPGA sebagai IC utama. Selain itu, masalah yang akan dibahas bagaimana mengurangi ketergantungan penggunaan *clock* referensi

yang sangat besar untuk mendapatkan resolusi tinggi dengan interpolasi digital menggunakan CARRY4 *logic* untuk membangun *multi-tap delay line* pada TDC.

Metode Digital Interpolasi *Reciprocal* merupakan pengembangan dari metode *Reciprocal* dengan penambahan kontrol interpolator dan TDC sehingga dapat mereduksi *error* pada *gate time* dan dapat meningkatkan resolusi pengukuran.

Peta konsep penelitian dan kaitan antara beberapa variabel masalah yang telah dijelaskan tersebut ditunjukkan pada Gambar 3.1.





Gambar 3.1 Diagram Konsep Penelitian

BAB IV

METODE PENELITIAN

4.1 Tempat dan waktu penelitian

Penelitian akan dilakukan di Laboratorium Sensor Sistem Jurusan Fisika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Brawijaya Malang mulai 1 September 2018.

4.2 Alat dan Bahan

Alat yang digunakan dalam penelitian antara lain adalah Textronix TDS 1001B *two channel digital oscilloscope*, Laptop-PC, *Software ISE Design Suite 14.7*, *oscillator RB Specrative 10 MHz*, rangkian osilator untuk QCM, multimeter digital, dan *JTAG HS2 Programming Cable*. Sedangkan bahan yang digunakan antara lain Xilinx Spartan-6 XCS500E FPGA, Sensor QCM 10 MHz, Kristal OCXO AOCJY dan beberapa komponen elektronika.

4.3 Tahap Penelitian

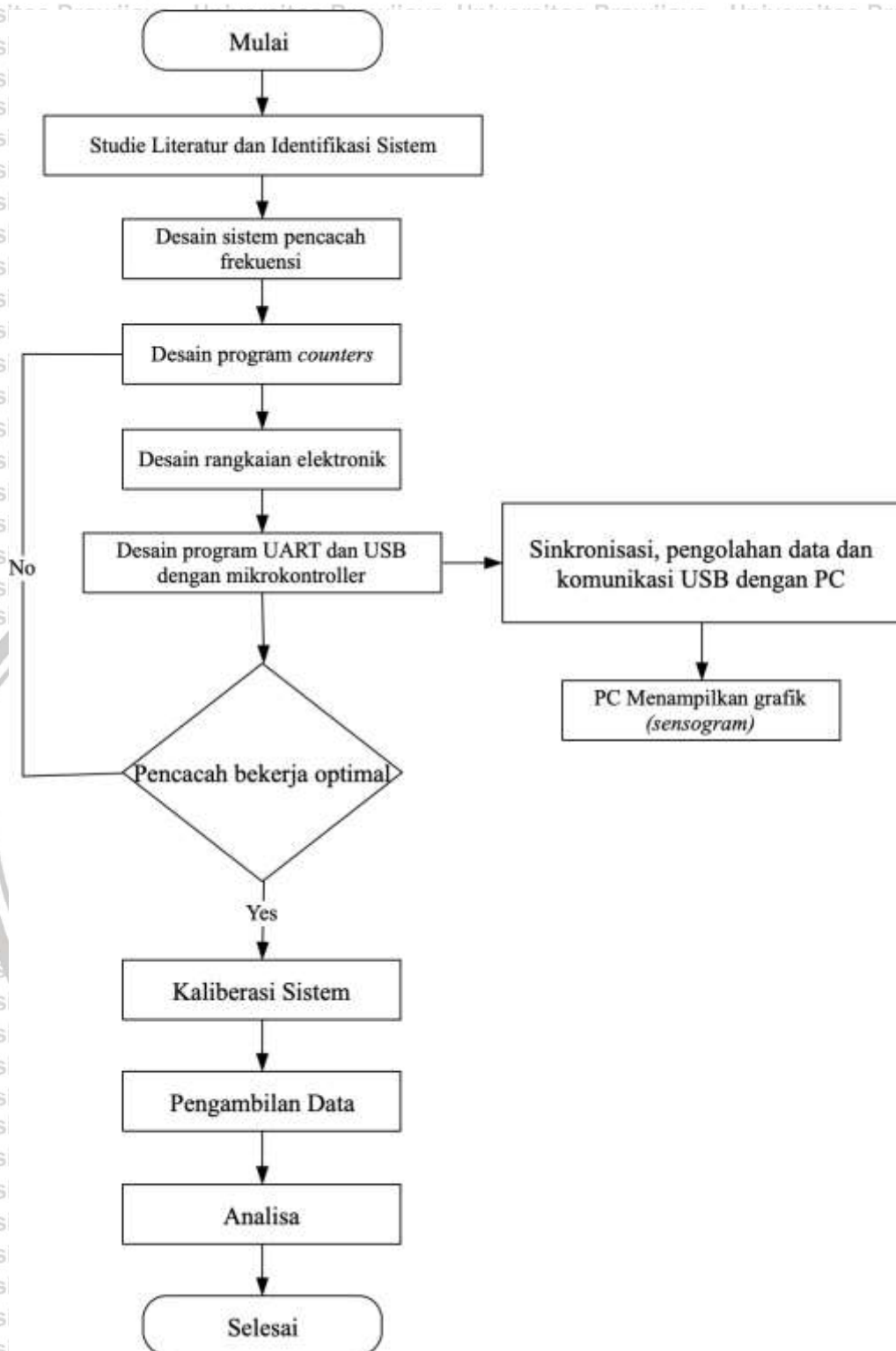
Langkah awal dalam penelitian ini adalah studi literatur dan identifikasi sistem. Tahap selanjutnya perancangan sistem dan pembuatan *listing* program frekuensi *counter* di *ISE Design Suite 14.7* dan mensimulasikan. Pembuatan program terfokus pada frekuensi *counter* menggunakan metode Digital Interpolasi *reciprocal* TDC yang memiliki resolusi sampai orde *pico* dan sistem komunikasi UART antar FPGA dengan mikrokontroler dan komunikasi USB antara mikrokontroler dengan PC (komputer).

Tahap selanjutnya adalah *development* perangkat elektronik dan pengujian sistem. Pengujian sistem dilakukan dengan cara mencoba menyalakan LED.

Setelah pengujian sistem berhasil, maka tahap selanjutnya menjalankan program frekuensi *counter* pada *software ISE Design suite 14.7*.

Setelah program frekuensi *counter* di *download* ke dalam sistem, sistem diuji untuk mencacah sinyal masukan. Jika sistem tidak dapat mencacah atau hasil cacahan tidak sesuai dengan sinyal masukan maka langkah selanjutnya adalah pengecekan untuk dilakukan pembaharuan. Jika sistem telah mencacah sesuai dengan sinyal masukan maka tahap selanjutnya adalah kalibrasi menggunakan osilator rubidium 10 MHz.

Sistem yang telah dikalibrasi selanjutnya dicoba untuk mencacah frekuensi dari sinyal generator dan osilator TCXO sebagai proses pengambilan data. Langkah terakhir data hasil cacahan dianalisis dan dibahas pada laporan penelitian.



Gambar 4.1 Diagram alir langkah penelitian

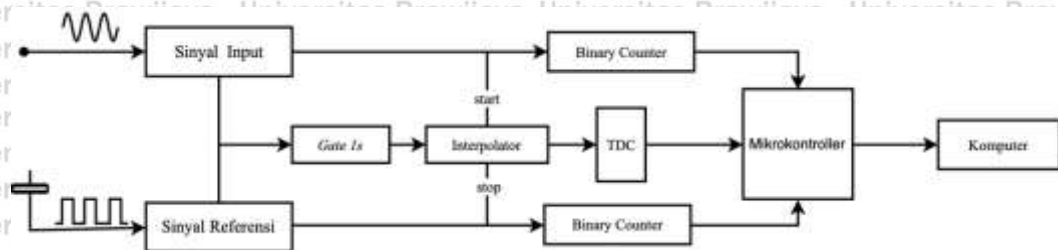
4.4 Desain Sistem

Desain sistem pencacah frekuensi secara garis besar ditunjukkan seperti Gambar 4.2. Sistem pencacah frekuensi dibagi dalam tiga sistem utama, yaitu pencacah frekuensi berbasis FPGA, Modul Mikrontroler 32 bit berbasis PIC32 dan Komputer (PC). Sistem-sistem akan diintegrasikan untuk menghasilkan sebuah perangkat frekuensi multi kanal yang bekerja secara simultan (*real time*).

Mikrokontroller PIC32 menjadi sistem yang akan menangani beberapa fungsi seperti menerima dan mengolah data hasil cacahan yang terukur oleh pencacah frekuensi berbasis FPGA, serta menangani komunikasi dengan PC yang berfungsi menampilkan data hasil cacahan.

Pencacah frekuensi didesain agar mampu mengukur frekuensi dari 4 sinyal input secara *real time* dan simultan. Oleh karena itu, pada penelitian ini akan dikembangkan empat buah pencacah frekuensi dengan sistem kerja *array sensor*. Pencacah frekuensi pada penelitian dibangun dengan IC utama yaitu Xilinx Spartan-6 FPGA. Metode yang digunakan untuk membangun pencacah frekuensi adalah metode Digital interpolasi *reciprocal TDC*, oleh karena itu, terdapat beberapa desain rangkaian digital yang dibangun didalam FPGA. Antara lain, yaitu *multi-tap delay line (code interpolator)*, pencacah digital (*binary counter*), dan modul UART.

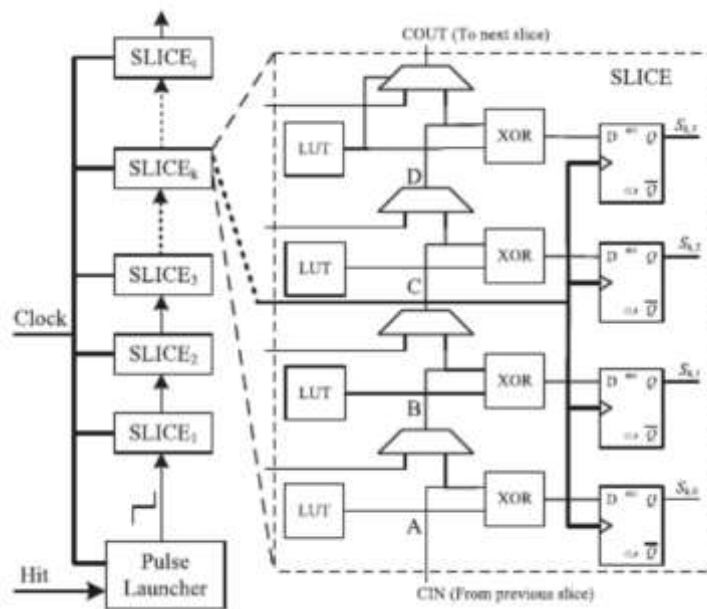
Modul pembagi frekuensi dihasilkan dengan cara mencacah sinyal yang bersumber dari frekuensi osilator eksternal OCXO 100MHz dan dijadikan sinyal *time gate* satu detik. Selain itu, *time-gate* digunakan sebagai *interrupt* dan kontrol komunikasi antara FPGA dan mikrokontroller.



Gambar 4. 2 Blok pencacah frekuensi metode Digital Interpolasi
Reciprocal TDC.

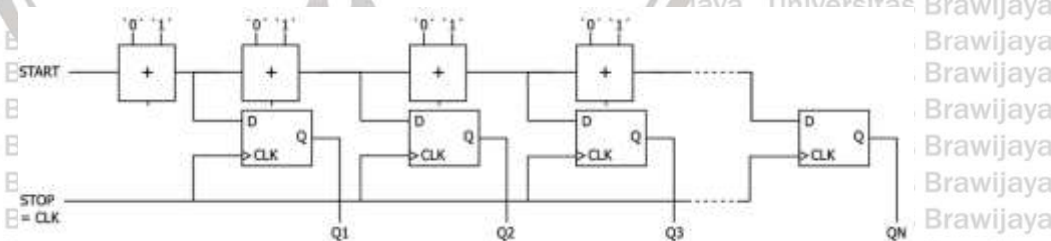
Digital Interpolasi *reciprocal counter*, mulai mencacah dengan *reciprocal counter* dengan ketidakpastian waktu pengukuran 1 *clock cycle*, Digital Interpolasi *reciprocal* digunakan karena adanya fakta bahwa tidak tahu dimana *clock cycle pulse* mulai dan berhenti pada metode *reciprocal* biasa. Dengan metode interpolasi khusus yaitu penambahan proses TDC (*time digital converter*), dapat ditentukan sudut fase *clock pulse* saat memulai dan berhenti pengukuran.

Terdapat CARRY4 *logic* pada FPGA yang dapat dibentuk suatu rangkaian untuk mendeteksi suatu kejadian yang sangat singkat (orde *pico second*). TDC pada penelitian ini mengaplikasikan struktur *multi-tap delay line* dari CARRY4 *logic* dan beberapa rangkaian D flip-flop. *Multi-tap delay line* adalah *carry chain* yang dibentuk dengan cara mengalirkan *carry logic* pada SLICE FPGA. Rangkaian *carry chain* memiliki waktu tunda lebih kecil dari pada 1 *clock* sistem. Setiap *rising edge* dari clock akan men-trigger *coarse counter* dan D register sehingga memberikan nilai waktu tunda sinyal. Gambar 4.3 merupakan struktur *multi-tap delay line* pada FPGA Spartan-6.



Gambar 4. 3 Struktur *multi-tap delay line* menggunakan CARRY4 logic pada SLICE FPGA Spartan-6.

Carry di propagasi (menyebar) melalui *carry chain*, untuk mendapatkan *carry* yang menyebar, jumlah *carry* harus 1 pada saat sinyal input masuk (*carry input* = 1). Oleh karena itu, masing *carry* diberikan input 0 dan 1. *Carry* mengalir sesuai dengan kecepatan elemen *carry*, pada FPGA kecepatan *carry* berada pada rentang 15-25ps, tergantung *platform* FPGA yang digunakan.



Gambar 4. 4 Implementasi *code thermometer interpolator* TDC pada FPGA dengan *carrychain* sebagai *delay line*.

delay line harus mencakup (tergantung pada aplikasi) periode waktu. Panjang *delay line* N dapat dihitung dengan:

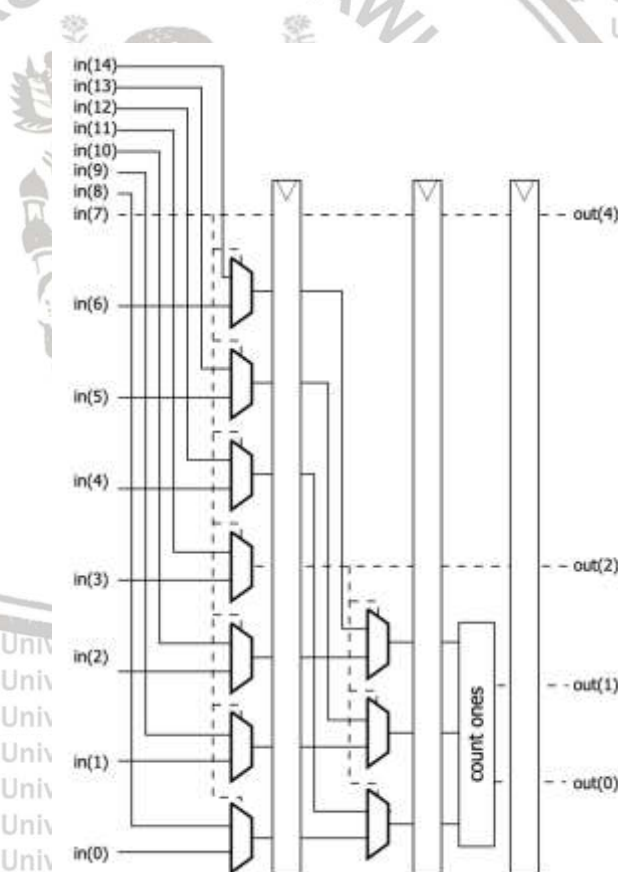
$$N_{\text{carry blok}} = \frac{1}{f_{\text{CLK}} \cdot \text{resolusi}}$$

Dimana resolusi merupakan *carry propagation delay*.

Pada penelitian ini menggunakan 100 MHz input referensi sebagai STOP sinyal, dan panjang *line* 256. *Delay line* dapat diimplementasikan kedalam 64 *SLICE* yang berisi *primitive CARRY4*.

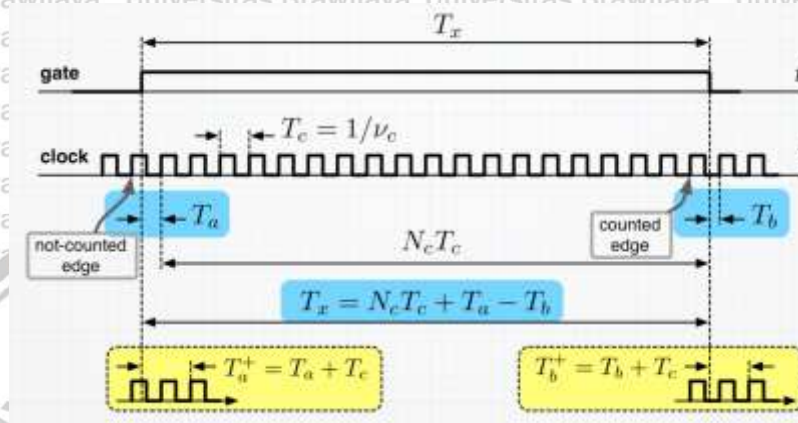
Langkah selanjutnya yang tidak kalah penting adalah merubah *code thermometer interpolator* ke biner agar dapat digunakan secara praktis.

Thermometer encoder sebanding dengan jumlah 2^N dimana N merupakan Panjang dari *delay line*. *Counter* menghitung *bit* tinggi yang tersisa. Gambar 4.5 merupakan struktur *encoder*.



Gambar 4. 5 *pipelined thermometer to binary encoder* dengan *counter bit* pada tahap akhir

Multi-tap delay line dapat diimplementasikan kedalam gerbang *array*, dengan mengeksplotasi fakta bahwa internal *delay* dari gerbang tunggal sangat kecil dan dapat diprediksi. Oleh karena itu *multiple phase* dari sinyal *clock* dapat dihasilkan dengan menggunakan beberapa rangkain *CARRY4 logic* pada *FPGA*. Resolusi waktu ditentukan oleh *gate delay*, bukan waktu *switching*.

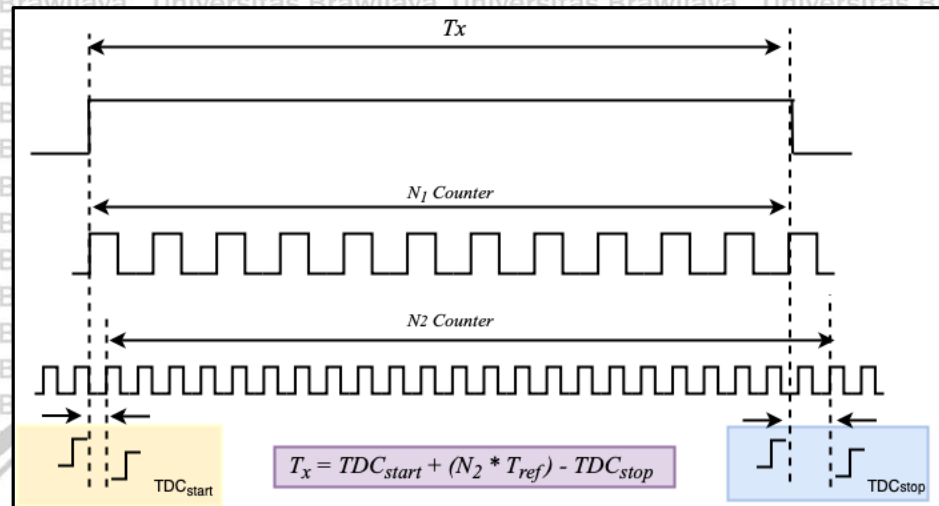


Gambar 4. 6 Main idea clock interpolator

Pada gambar 4.6 merupakan ide dari diperlukannya suatu *clock interpolator* pada metode *reciprocal*. Terdapat dua titik yang tidak ter-counted yaitu pada titik T_a dan T_b yang sangat pendek dan sulit untuk diukur, hal tersebut menyebabkan ada perlu penambahan satu titik yaitu T_c untuk mempermudah perhitungan. Interpolasi sangat dimungkinkan diterapkan dengan adanya fakta bahwa frekuensi *clock* konstan dan akurasi yang telah diketahui.

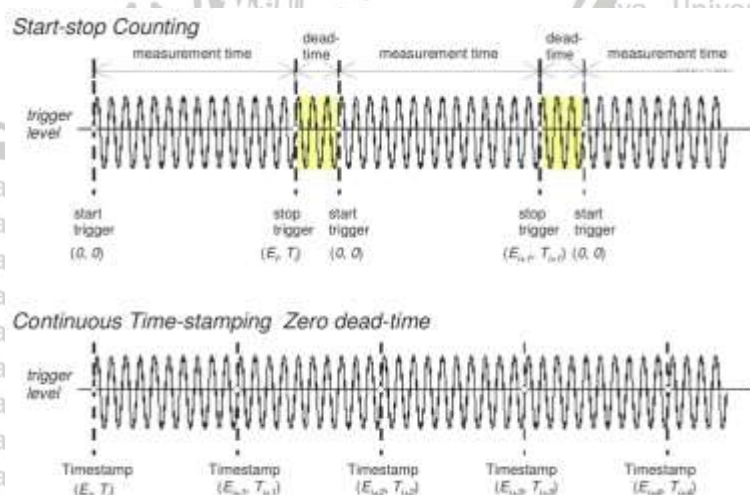
Untuk mendeteksi titik yang tidak ter-counting, maka diperlukan adanya proses lanjutan untuk mendeteksi titik yang tidak ter-counting. TDC digunakan untuk mendeteksi titik tidak ter-counting tersebut. Gambar 4.7 merupakan proses kerja TDC untuk menghitung titik-titik yang tidak ter-counting. Dengan memperhatikan gambar 4.7 frekuensi sinyal input dapat diperoleh dengan persamaan di bawah ini:

$$F_x = \frac{N_x}{T_x}$$



Gambar 4. 7. *Timing diagram implementasi TDC pada metode interpolasi reciprocal.*

Dengan mengimplementasikan *code* interpolator dan TDC pada Gambar 4.7, maka sistem pengukuran frekuensi yang dibangun seperti *Timestamping counter*, dimana titik *dead time* (tidak terukur) akan terukur dan dapat meningkatkan resolusi hasil pengukuran.



Gambar 4.8 *Continuous Timestamping Counter*

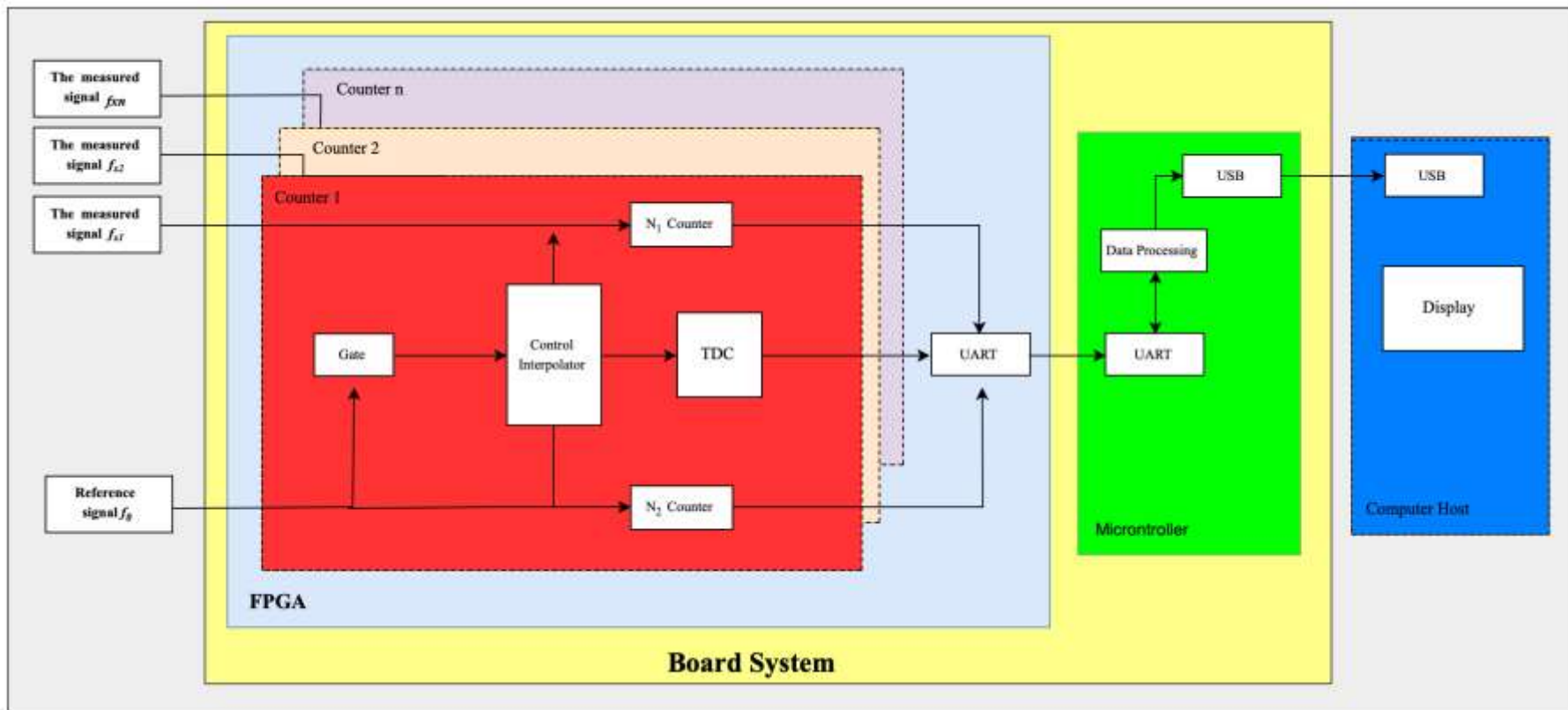
4.5 Perancangan dan Uji Sistem

4.5.1 Perancangan Elektronik

Pada penelitian ini secara garis besar dibangun atas dua komponen utama yang akan berfungsi sesuai dengan program yang ditanam didalamnya. Kedua komponen tersebut adalah IC FPGA Spartan-6 tipe XCS2500 dan mikrokontroler PIC32. Gambar 4.8 merupakan blok diagram sistem pencacah frekuensi metode Digital Interpolasi *Reciprocal* menggunakan TDC.

IC FPGA merupakan komponen utama dalam proses pencacah frekuensi, sedangkan mikrokontroler untuk mengolah data hasil pencacahan. Selain itu mikrokontroler dihubungkan dengan IC FPGA melalui komunikasi UART, dan komunikasi USB HID digunakan untuk menghubungkan antara mikrokontroler dengan komputer (PC).





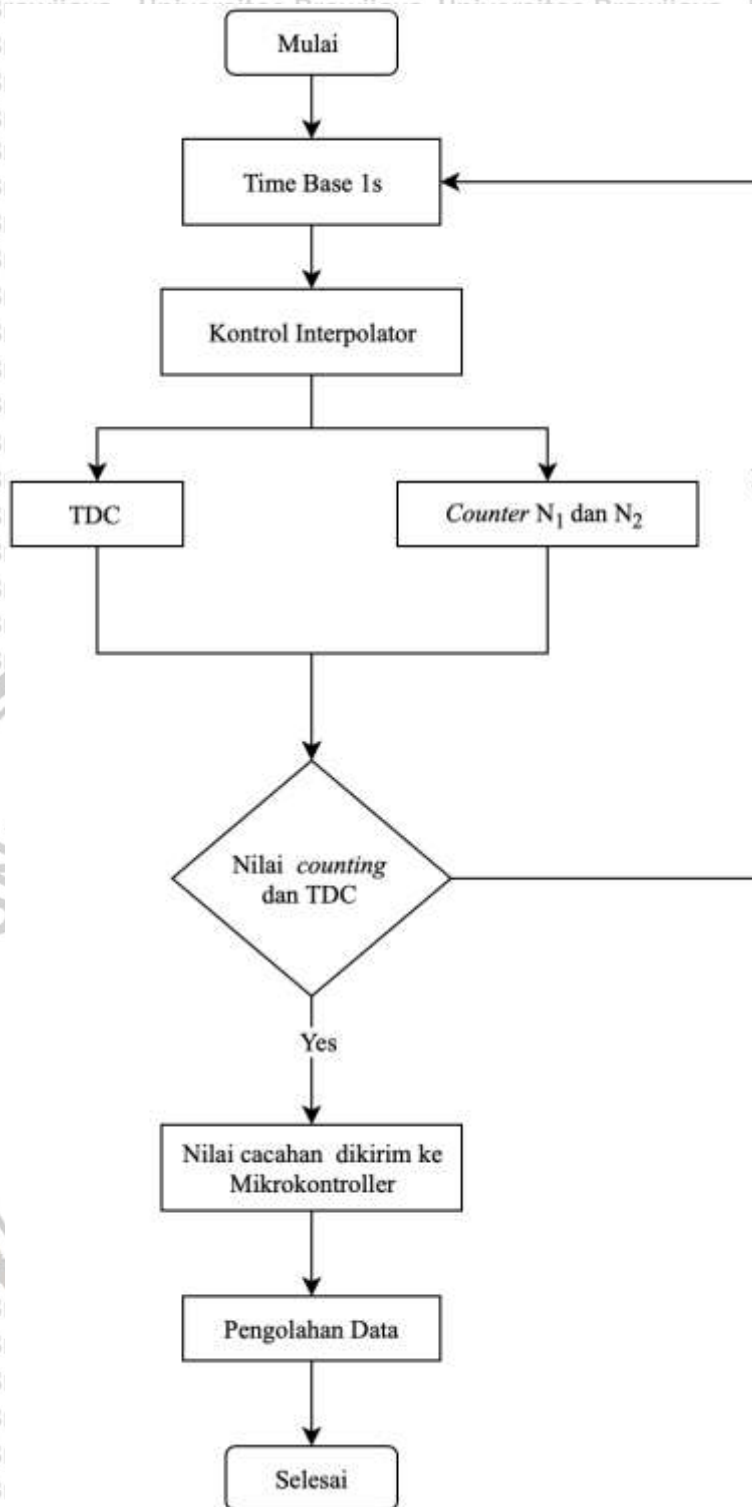
Gambar 4. 9 Blok diagram sistem pencacah frekuensi metode Digital Interpolasi *Reciprocal* TDC.

Pengiriman data ke komputer (PC) dibangun dengan memanfaatkan komunikasi USB.

Disisi lain, komunikasi USB tidak hanya digunakan untuk komunikasi USB, melainkan digunakan untuk sumber daya (*supply*) komponen-komponen pada *board* sistem pencacah frekuensi, oleh sebab itu pada board sistem pencacah tidak dibutuhkan sumber tegangan tambahan dari luar.

4.5.2 Perancangan Program

Perancangan program FPGA menggunakan *software xilinx ISE Design Suite 14.2* dengan menggunakan Bahasa VHDL. *Flowchart* program ditunjukkan pada Gambar 4.9. inti program yang akan di-*download* ke dalam FPGA adalah mencacah sinyal masukan dengan mendeteksi *rising edge* menggunakan metode *reciprocal TDC* dengan memanfaatkan CARRY4 logic pada FPGA. Selain itu terdapat program pada mikrokontroler menggunakan *software mikroC* dengan menggunakan bahasa C, pada mikrokontroler inti program yang dimasukkan adalah komunikasi UART dan USB serta beberapa fungsi aritmatika untuk mengolah data hasil cacahan yang dihasilkan oleh FPGA dan selanjutnya dikirim kekomputer (PC) untuk ditampilkan hasil cacahan pada tiap-tiap kanal.



Gambar 4.10 Flowchart program pencacah frekuensi.

4.5.3 Komunikasi FPGA ke Mikrokontroler

Dalam penelitian ini, pencacah frekuensi dibangun dengan memanfaatkan dua IC utama yang berbeda yaitu IC FPGA dan mikrokontroler. Masing-masing IC memiliki fungsi yang

berbeda, FPGA berfungsi sebagai IC utama untuk membangun digital counter untuk sistem pencacah frekuensi dan IC mikrokontroler digunakan sistem komunikasi UART dengan FPGA, pengolahan data dan pengiriman data hasil cacahan ke komputer (PC). Dari beberapa fungsi tersebut, maka diperlukan antarmuka antara FPGA dengan mikrokontroler.

4.5.4 Uji Coba Sistem Pencacah

Pengujian sistem secara keseluruhan bertujuan untuk mengetahui fungsionalitas disain program yang telah dibuat dalam Bahasa VHDL secara keseluruhan. Proses pengamatan tertuju pada perancangan, seperti bagaimana program *digital counter* bekerja dalam memecah sinyal, sistem pewaktu (*time gate*), sistem penyimpanan data dalam register *buffer*, sistem komunikasi UART, TDC (*time digital converter*) dan pengiriman data ke komputer (PC).

4.5.5 Pengujian Pencacahan Kristal Rubidium

Untuk mengetahui kestabilan sistem pencacah frekuensi dalam memecah sinyal pulsa, perlu dilakukan pengujian dengan menggunakan Kristal dengan tingkat kesetabilan yang sangat tinggi. Maka dalam pengujian ini digunakan Kristal rubidium dengan tingkat kesetabilan tinggi dan frekuensi osilasi 10 MHz.

4.5.6 Pengujian Akurasi Pencacah Frekuensi

Pengujian ini dilakukan untuk melihat keakuratan sistem pencacah frekuensi. Proses pengujian dilakukan dengan cara memecah pulsa sinyal yang dihasilkan oleh generator BK PRECISION. Dari hasil pencacahan sinyal generator, selanjutnya dilakukan pengamatan terhadap perubahan nilai cacahan terhadap waktu.

4.5.7 Pengujian Pencacah Multi Kanal

Pengujian ini dilakukan bertujuan untuk mengetahui keempat kanal pada pencacah frekuensi dapat memecah secara simultan. Proses pengujian dilakukan dengan cara memberikan sinyal dari *signal generator* pada masing-masing kanal.

4.5.8 Analisa Data dan Pembahasan

Untuk mengetahui karakteristik dari sistem pencacah yang telah dibangun perlakuan analisa data dari beberapa hasil pengujian pencacah frekuensi. Data yang dianalisa antara lain: implementasi program pada IC FPGA dan mikrokontroler. Antarmuka komunikasi FPGA dengan mikrokontroler, DCM untuk pergeseran fase sinyal, sistem komunikasi USB antara mikrokontroler, komputer (PC) dan sistem pencacahan.

Data hasil implementasi pada FPGA merupakan hasil *synthesis* dari program VHDL yang dibuat dan hasil *resource* yang digunakan. Untuk program DCM sebagai pergeseran fase dianalisa ketepatan pada masing-masing *gate* untuk masing pencacah. Sedangkan untuk komunikasi UART dan USB telah berjalan baik atau tidak.

Analisa pencacahan dilakukan dengan menggunakan beberapa data pengujian yang telah dilakukan antara lain: pencacah pulsa sinyal 10MHz TCXO, signal generator, dan kristal rubidium dalam bentuk grafik. Grafik dianalisa tingkat akurasi dan presisi dalam interval waktu pengukuran dan analisa kinerja dari perangkat pencacah frekuensi pada masing-masing kanal (*channel*).

BAB V

HASIL DAN PEMBAHASAN

5.1 Hasil Implementasi Desain

Desain sistem pengukuran frekuensi yang telah dibahas pada subbab 4.4 diimplementasikan dalam suatu sistem *embedded* berbasis FPGA Spartan-6 dan Mikrokontroler PIC32. Desain sistem yang dibangun didesain secara *portable* dan *low power consumption*. Oleh karena itu, sistem *embedded* ini akan bekerja untuk berbagai fungsi pencacah frekuensi berbasis FPGA dan mikrokontroler sebagai pengolah data dan sistem komunikasi USB HID dengan komputer (PC).

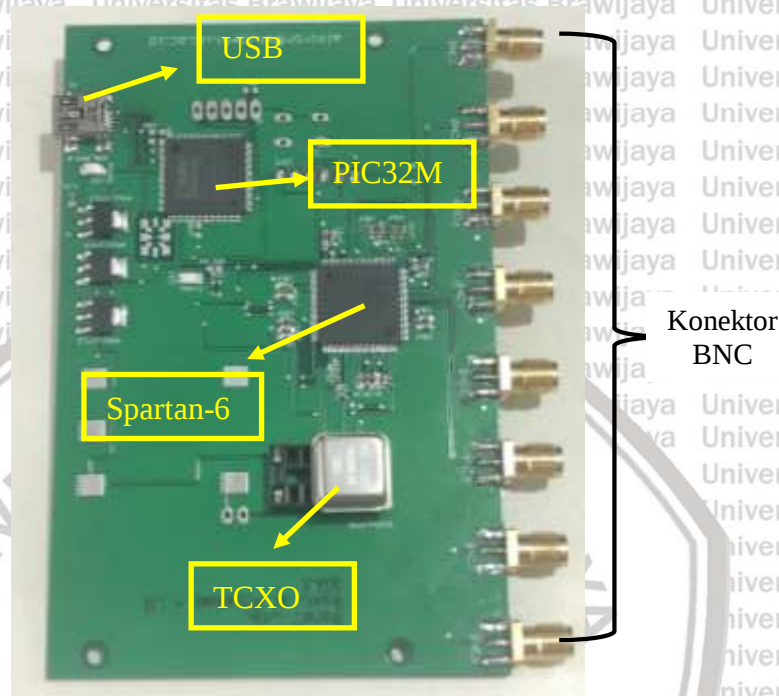
Rangkaian elektronik utama pada penelitian ini terdiri dari IC Spratan-6 FPGA, kristal TCXO 100MHz, regulator 3.3Volt dan 2.5 Volt dan beberapa konektor BNC. Melalui konektor BNC inilah rangkain pencacah frekuensi menerima sinyal input. Mikrokontroler dan USB mini merupakan rangkain elektronik kedua yang bertugas untuk pengolahan data dan sistem komunikasi baik antara FPGA dengan mikrokontroler maupun mikrokontroler dengan Komputer.

Sistem komunikasi antara FPGA dan mikrokontroler menggunakan *protocol* komunikasi *serial* yaitu UART (*universal Asynchronous Receiver/Transmitter*). Sehingga tidak membutuhkan sistem *clock* pengiriman data yang sama antara FPGA dan mikrokontroler.

Selanjutnya komunikasi antara mikrokontroler dengan komputer (PC) menggunakan *protocol* komunikasi USB HID (*human interface device*), sistem komunikasi tersebut dipilih selain dikarenakan lebar pengiriman data yang banyak yaitu 64 bit dan juga merupakan sistem komunikasi dengan kecepatan pengiriman tinggi, cocok untuk mengirimkan data hasil cacahan 4 kanal dengan setiap kanal membutuhkan lebar data 32 bit tiap detik secara bersamaan.

repository.ub.ac.id

Pada penelitian ini, alat yang dibangun tidak membutuhkan *power supply* tambahan, cukup menggunakan *power supply* 5V yang bersumber dari USB yang terhubung dengan komputer (PC). Gambar 5.1 merupakan keseluruhan rangkaian *embedded* sistem untuk pencacah frekuensi multi kanal.



Gambar 5. 1 Board sistem pencacah frekuensi

Desain sistem menggunakan IC FPGA sebagai IC utama untuk mencacah sinyal input multi kanal secara simultan (*real time*) dengan kapasitas pengukuran 32 bit. Pencacah frekuensi yang dibangun memiliki kemampuan untuk mendeteksi perbedaan *rising edge* (sinyal naik) antara sinyal input dengan sinyal referensi. Resolusi pengukuran waktu pada sistem pencacah frekuensi mencapai 40ps.

5.2 Sistem pencacah frekuensi

Perangkat yang dirancang secara *portable* dan mampu mencacah secara simultan (*real time*). Oleh karena itu, pada penelitian ini dikembangkan beberapa modul pencacah frekuensi yang bekerja secara bersamaan untuk mengukur frekuensi sinyal input. Pencacah pada penelitian ini menggunakan Spartan-6 FPGA metode yang digunakan adalah metode

UNIVERSITAS BRAWIJAYA

interpolasi *reciprocal* menggunakan TDC seperti yang telah dibahas pada subbab sebelumnya. Dengan demikian, dibangun beberapa rangkaian digital yang diimplementasikan kedalam FPGA, yaitu rangkaian *time base 1s*, *binary counter*, *code thermometer interpolator* (TDC), *pipeline thermometer (encoder)*, multiplexer, dan UART. Masing-masing kanal terdapat rangkaian tersebut kecuali modul UART. Gambar 5.2 menunjukkan *top level* arsitektur sistem pencacah multi kanal dalam bahasa VHDL.

```
entity top is
  generic (
    STAGES          : integer := 256;
    FINE_BITS       : integer := 6;
    Xoff             : integer := 8;
    Yoff            : integer := 2);
  port (
    ref_clk          : in std_logic;
    reset            : in std_logic;
    sample_clk       : in std_logic;
    sample_clk_1     : in std_logic;
    sample_clk_2     : in std_logic;
    sample_clk_3     : in std_logic;
    txd              : out std_logic);
end top;
```

Gambar 5.2 *Top Level* arsitektur sistem pencacah multi kanal.

Rangkaian pembagi frekuensi yang dikembangkan pada penelitian ini menggunakan sinyal referensi TCXO 100MHz, kemudian dicacah dan dibentuk menjadi sinyal *time base 1s*. Pada penelitian ini terdapat dua sinyal hasil pencacahan sinyal referensi yaitu *time base 1s* dan *time_sys*. *Time_base 1s* hanya digunakan untuk mengontrol proses pencacahan, dan tidak digunakan sebagai referensi waktu cacahan, karena waktu cacahan diperoleh berdasarkan hasil perhitungan antara interpolasi *reciprocal* menggunakan TDC. Sedangkan untuk *time_sys* digunakan untuk sistem *interrupt* komunikasi UART antara FPGA dan mikrokontroler. Gambar 5.3 merupakan implementasi program sistem pembagi frekuensi pada sistem pencacah frekuensi multi kanal.

```

time_base_1s :process(ref_clk)
begin
    if ref_clk'event and ref_clk='1' then
        counter <= counter + 1;
        if counter < 100000000 then -- 100000000Hz
            gate_1s <= '1';
            gate_sys <= '1';
        elsif counter = 100000000 then
            gate_1s <= '0';
            gate_sys <= '0';
        elsif counter > 100000000 and counter < 100000000+5 then -- 100000005Hz
            gate_sys <= '0';
            counter <= 0;
        end if;
    end if;
end Process;

```

Gambar 5. 3. Listing program *time_base_1s* dan *time_sys*

Pengukuran frekuensi pada penelitian ini menggunakan metode interpolasi *reciprocal*, metode ini mengkombinasikan antara metode *reciprocal* dan TDC (*time digital convert*) dengan sinyal *time_1s* sebagai sinyal kontrol. Secara garis besar, mekanisme pengukuran sinyal input diimplementasikan dalam bentuk program dalam bahas VHDL seperti pada gambar dibawah ini.

```

process(sample_clk, gate_1s)
begin
    if rising_edge(sample_clk) then
        start_i <= gate_1s;
    end if;
end process;

```

Gambar 5. 4 Listing program kontrol stop dan start counting.

Pada Gambar 5.4 merupakan program untuk memulai dan memberhentikan *counting* berdasarkan *rising edge* (sinyal naik) dari input sinyal yang akan dihitung. Dari *listing* program tersebut juga menegaskan bahwa sinyal *gate_1s* hanya dijadikan sebagai kontrol memulai dan memberhentikan *counter*, dan tidak berpengaruh terhadap hasil *counting*.

Setelah berhasil membentuk sinyal *gate* berdasarkan mekanisme *rising edge* sinyal input, langkah selanjutnya melakukan proses *counting* secara bersamaan sinyal input dan sinyal

referensi. Hal ini dilakukan untuk mengetahui waktu proses *countin* sinyal input dan tidak terlepas adanya fakta bahwa input referensi memiliki priode tetap (pada penelitian menggunakan sinyal referensi periode 10ns). Gambar 5.5 merupakan *listing* program untuk proses *counting* sinyal input dan sinyal referensi.

```
counter_sample_1 : process(sample_clk_1, start_i_1)
begin
    if start_i_1 = '1' then
        if rising_edge(sample_clk_1) then
            sample_count_1 <= sample_count_1 + 1;
        end if;
    elsif start_i_1 = '0' then
        sample_count_1 <= 0;
    end if;
end process;

counter_ref_1 : process(ref_clk, start_i_1)
begin
    if start_i_1 = '1' then
        if rising_edge(ref_clk) then
            ref_count_1 <= ref_count+1;
        end if;
    elsif start_i_1 = '0' then
        ref_count_1 <= 0;
    end if;
end process;
```

Gambar 5. 5 *Listing* program *counter* sinyal input dan sinyal referensi

Pada tahap ini metode yang digunakan dalam proses *counting* adalah metode *reciprocal* dengan resolusi pengukuran 0.1 Hz.

Meskipun telah mendapatkan hasil pengukuran frekuensi dengan resolusi kurang dari 1 *clock system*, masih perlu adanya proses lanjutan untuk meningkatkan resolusi pengukuran.

Selain itu, pada subbab sebelumnya telah dijelaskan bahwa terdapat bagian yang tidak ter-*counting* jika hanya menggunakan metode *reciprocal* saja. Maka pada penelitian ini ditambah metode baru yaitu TDC (*time digital converter*) agar titik yang tidak ter-*counting* dapat dihitung. Dengan penambahan proses TDC maka hasil akhir dari proses ini adalah waktu yang dibutuhkan selama *start* sampai *stop*. Pada penelitian ini TDC yang digunakan menggunakan

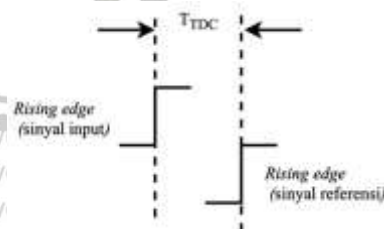
algoritma *code thermometer interpolator* dengan Panjang *line* sebanyak 256. Gambar 5.6 merupakan *top level* arsitektur yang digunakan pada penelitian ini. Secara teori resolusi pengukuran TDC dengan menggunakan Panjang *line* 256 dan *clock* referensi 100 MHz sebesar 40 ps.

```

Ins_TDC : fine_tdc_with_encoder
    GENERIC MAP (
        STAGES          => STAGES,
        FINE_BITS        => FINE_BITS,
        Xoff             => Xoff,
        Yoff             => Yoff)
    PORT MAP (
        clock            => ref_clk,
        reset            => reset,
        hit              => sample_clk,
        value_fine       => value_vine);
  
```

Gambar 5. 6 *Top level* arsitektur TDC.

Hal terpenting dalam TDC adalah sinyal *start* dan sinyal *stop*, pada *listing* program diatas disesuaikan dengan metode *reciprocal* sebelumnya maka yang menjadi sinyal *start* adalah sinyal input dan sinyal *stop* (*clock*) adalah sinyal referensi. Gambar 5.7 merupakan ilustrasi TDC yang digunakan.



Gambar 5. 7 mekanisme kerja TDC.

Untuk mengetahui titik *start* dan *stop* yang tidak *ter-counting* perlu ditambahkan suatu proses interpolator. Pada bagian ini nilai TDC yang akan dikirim ke mikrokontroler hanya pada titik memulai dan titik akhir *counter*. Sama halnya seperti metode *reciprocal* sebelumnya, siny

gate 1s menjadi kontrol pada proses interpolator ini. Gambar 5.8 merupakan *listing* program interpolator, data TDC yang dikirim pada proses telah memiliki panjang data 8 bit.

```
process(start_i, value_vine)
begin
    if start_i = '1' then
        hasil_tdc_start <= unsigned(value_vine);
    elsif start_i = '0' then
        hasil_tdc_stop <= unsigned(value_vine);
    end if;
end process;
```

Gambar 5. 8 *listing* program Interpolator pada sistem pencacah frekuensi.

Proses selanjutnya adalah pengiriman data ke modul UART kemudian dikirim ke mikrokontroler. Akan tetapi, perlu diperhatikan juga adalah proses *encoder* pada TDC. Pada penelitian ini, algoritma yang digunakan pada proses *encoder* yaitu *pipeline thermometer code*. Lebar *bit* pada hasil *encoder* disesuaikan dengan panjang *line* pada TDC. Pada penelitian ini panjang *line* yang digunakan adalah 256, maka lebar data hasil *encoder* adalah $2^N = 256$, jadi lebar data hasil *encoder* adalah 8bit. Gambar 5.9 adalah *top level* arsitektur dari *pipeline thermometer code*.

```
t2b : therm2bin_pipeline_count
GENERIC MAP (
    b      => FINE_BITS)
PORT MAP (
    clock   => clock,
    reset   => reset,
    valid   => valid,
    thermo  => fine_value_reg,
    bin     => fine_value_bin);
```

Gambar 5. 9. *Top level Pipeline Thermometer Encoder*

Tahap terakhir proses *counting* pada FPGA adalah mengirimkan data ke modul UART dan selanjutnya ditransfer ke mikrokontroler untuk proses pengolahan data. Sebelum dilakukan pengiriman ke modul UART perlu dilakukan penyimpanan sementara (*buffer*) selama proses

counting berlangsung. Register *buffer* dikontrol oleh sinyal yang sama dengan sinyal kontrol modul UART. Dibawah ini *listing* program untuk register *buffer* sebelum dikirimkan ke modul UART.

```
process(gate_sys)
begin
  if gate_sys'event and gate_sys = '0' then
    sample_r   <= to_unsigned(sample_count,32);
    ref_r       <= to_unsigned(ref_count,32);

    sample_r_1 <= to_unsigned(sample_count_1,32);
    ref_r_1     <= to_unsigned(ref_count_1,32);

    sample_r_2 <= to_unsigned(sample_count_2,32);
    ref_r_2     <= to_unsigned(ref_count_2,32);

    sample_r_3 <= to_unsigned(sample_count_3,32);
    ref_r_3     <= to_unsigned(ref_count_3,32);
  end if;
end process;
```

Gambar 5. 10 Register *buffer* penyimpanan cacahan dengan metode *reciprocal*.

Pada modul UART data yang dikirim dilakukan secara bergantian (multiplekser) hal ini disesuaikan dengan standart protokol UART yaitu 8bit data. Pada penelitian *baudrate* yang digunakan sebesar 9600. *Clock* pada modul UART dihasilkan oleh sinyal referensi. Gambar 5.11 merupakan mekanisme pengiriman data pada modul UART. Masing-masing data dikirim secara berurutan berdasarkan urutan *byte* yang telah ditentukan.

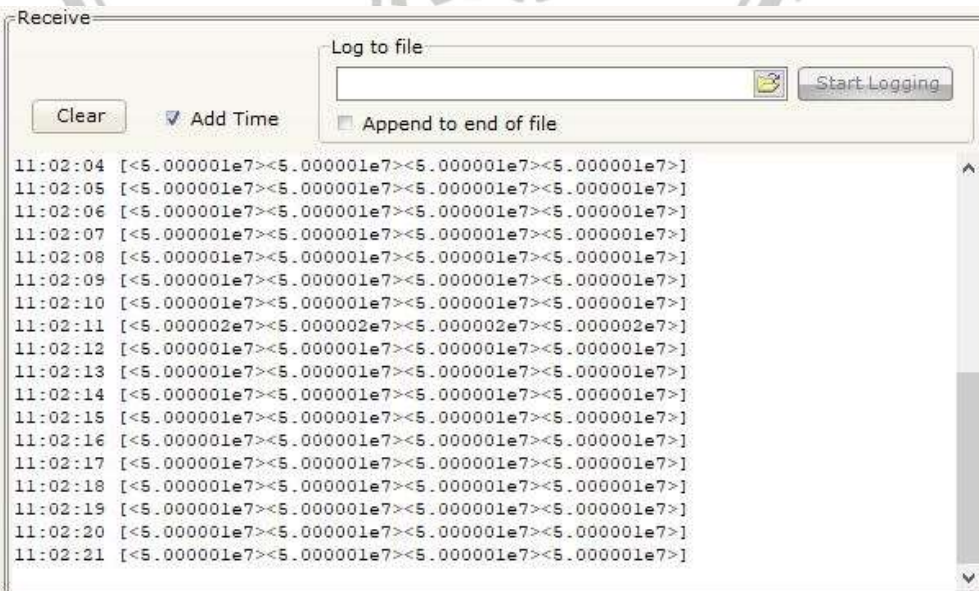
```

if done_counting= '1' then
  if baudrate_clock'event and baudrate_clock = '1' and char_index < 42 then
    txd <= '1';
    if bit_counter = 9 then
      bit_counter<=x"0";
      char_index  <= char_index+1;
      urut_data   <= urut_data+1;
      case urut_data is
        when 0 =>
          shift_register <= "1" & data10 & "0";
        when 1 =>
          shift_register <= "1" & data9 & "0";
        when 2 =>
          shift_register <= "1" & data8 & "0";
        when 3 =>
          shift_register <= "1" & data7 & "0";
        when 4 =>
          shift_register <= "1" & data6 & "0";
        when 5 =>
          shift_register <= "1" & data5 & "0";
        when 6 =>
          shift_register <= "1" & data4 & "0";
        when 7 =>
          shift_register <= "1" & data3 & "0";
        when 8 =>
          shift_register <= "1" & data2 & "0";
        when 9 =>

```

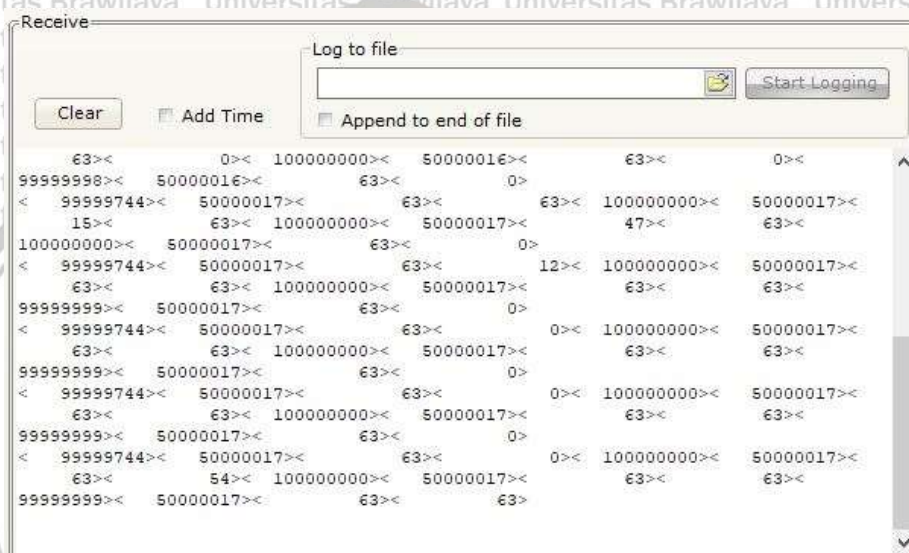
Gambar 5. 11 Mekanisme Multiplexer pada Modul UART.

Data dikirim secara per-byte, urutan *byte* diatur sesuai dengan jumlah kanal pada sistem *counter*. Antara FPGA dan komputer diatur batas jumlah *byte* pengiriman, sehingga data terkirim setiap 1 detik. Dibawah ini adalah gambar monitor hasil pengolahan pada mikrokontroler, data dikirim ke komputer melalui komunikasi USB HID.



Gambar 5. 12. Hasil pengolahan data pada mikrokontroler.

Untuk memastikan data yang dihitung merupakan hasil pencacahan pada setiap kanal, maka perlu ada pengecekan terlebih dahulu untuk memastikan tidak ada urutan *byte* yang tertukar. Pengecekan dilakukan dengan cara mengirimkan data hasil konversi dari data HEX kedalam kode ASCII pada masing-masing kanal dan tidak dilakukan proses perhitungan (konversi ke nilai frekuensi hasil cacahan). Gambar dibawah ini merupakan hasil pengiriman data konversi HEX ke ASCII.



Gambar 5. 13 konversi data HEX to ASCII.

5.3 Pengolahan Data Pada Mikrokontroler

Pada penelitian ini mikrokontroler yang digunakan adalah PIC32. Perlu diperhatikan bahwa mikrokontroler hanya bertugas untuk pengolahan data dan pengiriman data kekomputer.

Bahasa yang digunakan untuk memprogram mikrokontroler adalah bahasa C dan menggunakan *compiler* mikroC. Pada mikrokontroler terdapat beberapa proses konversi dan perhitungan aritmatika untuk memperoleh nilai frekuensi sinyal input dari masing-masing kanal pada FPGA. Untuk lebih jelasnya, perhatikan gambar 5.12 *flowchart* proses dari pengolahan data sampai data dikirimkan ke komputer.

Terdapat beberapa persamaan yang digunakan untuk proses aritmatika di dalam mikrokontroler antara lain sebagai berikut :

- Menghitung nilai T_x :

$$T_x = TDC_{start} + N_{ref} * T_{ref} - TDC_{stop}$$

Dimana :

$TDC = time$ • resolusi TDC (pada penelitian ini 40 ps).

$T_{ref} = 10 \text{ ns}$ (pada penelitian ini menggunakan 100 MHz).

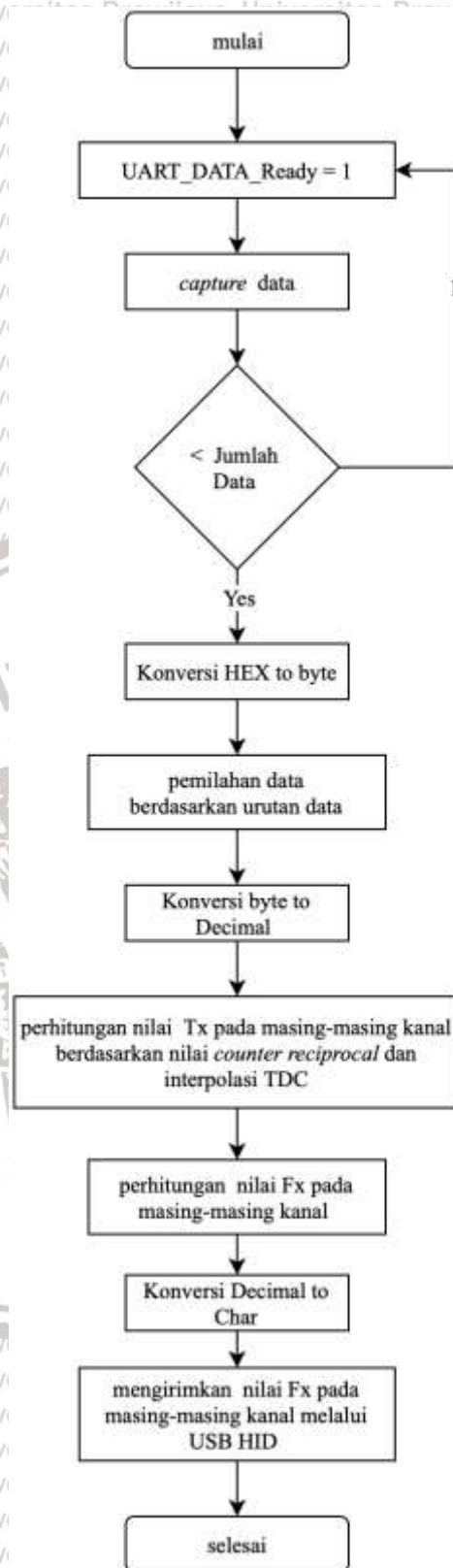
T_x yang dihasilkan masih dalam bentuk satuan ns (*nanosecond*), maka dirubah kedalam satuan s (sekon) dengan mengalikan $10e-9$.

- Menghitung nilai F_x (Frekuensi input) :

$$f_x = \frac{N_x}{T_x}$$

Dimana:

N_x = merupakan hasil *counting* sinyal input.

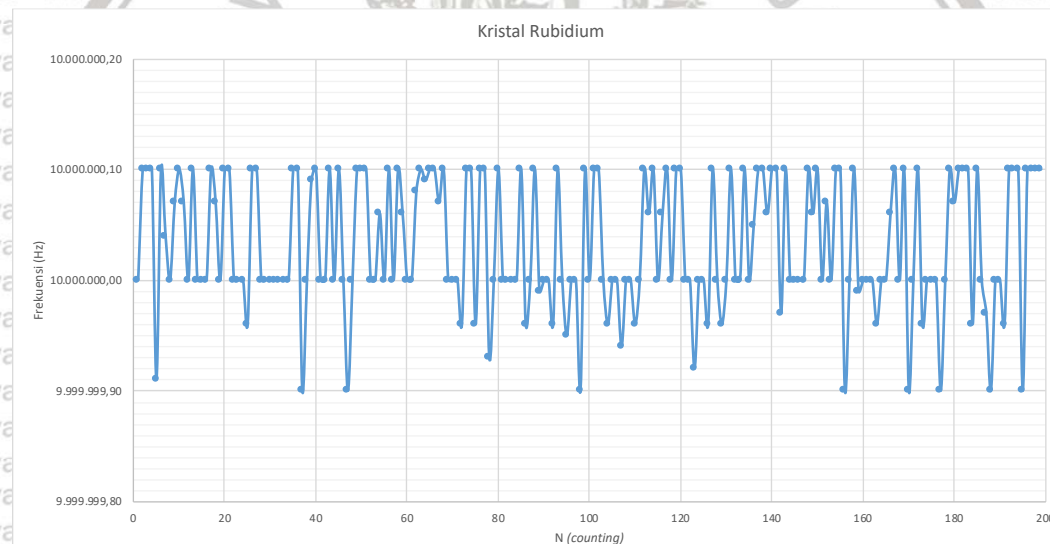


Gambar 5. 14 Flowchart proses pengolahan data pada Mikrokontroler.

5.4 Pengujian Pencacah Frekuensi

5.4.1. Pengujian Kestabilan Frekuensi

Data hasil cacahan sinyal input dari kristal rubidium dengan tingkat kestabilan yang sangat tinggi dapat dilihat pada gambar 5.13. jika dianalisis secara mendalam hasil pencacahan frekuensi dapat mendeteksi perubahan frekuensi yang dihasilkan oleh kristal rubidium mencapai 0.01 Hz. Dengan demikian, dapat disimpulkan dengan penambahan interpolator TDC dapat meningkatkan resolusi hasil pengukuran. Secara teori pengukuran frekuensi dengan metode *reciprocal* dengan menggunakan input referensi 100MHz dan input sinyal 10 MHz memiliki resolusi pengukuran yang dihasilkan sebesar 0.1 Hz, Namun TDC berhasil mendeteksi perbedaan *rising edge* dari sinyal referensi dan sinyal input sehingga dapat meningkat resolusi hasil. pengukuran.

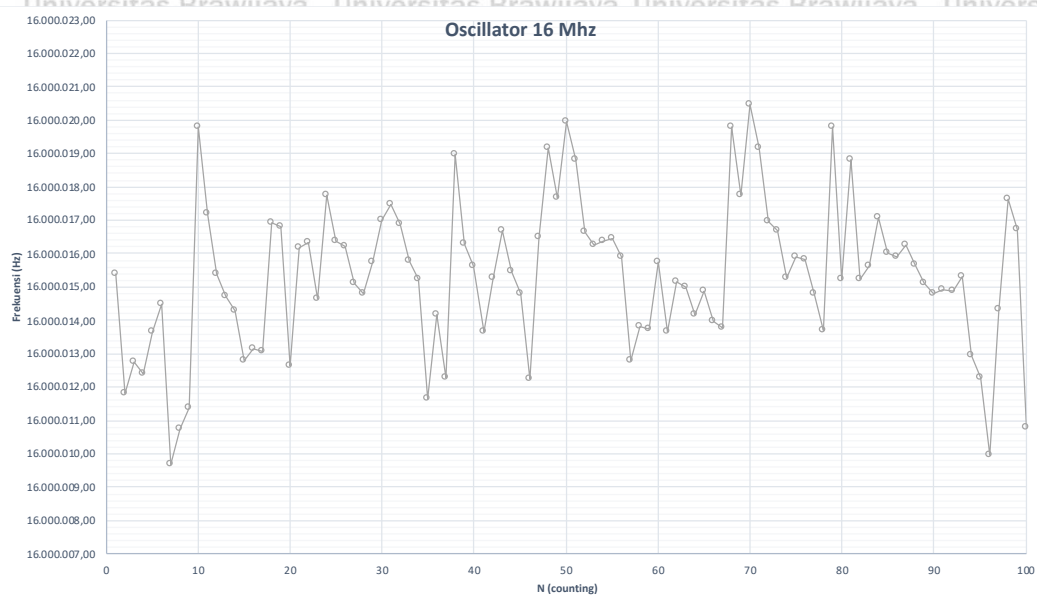


Gambar 5. 15 Grafik Hasil pencacahan Kristal Rubidium 10MHz.

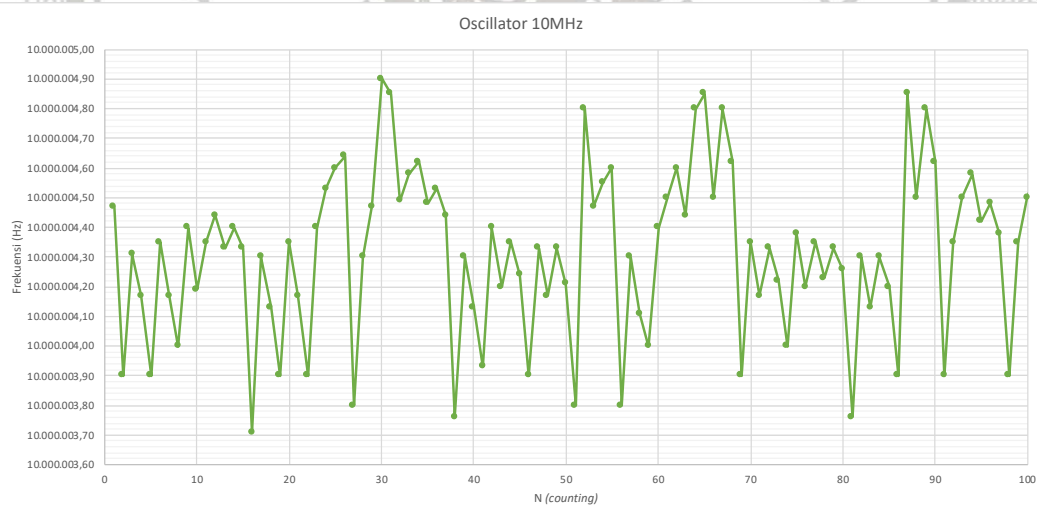
5.4.2. Pengujian Akurasi Pencacahan

Pada pengujian akurasi sinyal input yang digunakan dari beberapa *oscillator*. Pada penelitian ini menggunakan 4 pengujian, disesuaikan dengan jumlah kanal yang diprogram pada FPGA. *Oscillator* yang digunakan antara lain 10MHz, 16MHz, 20MHz dan 50MHz.

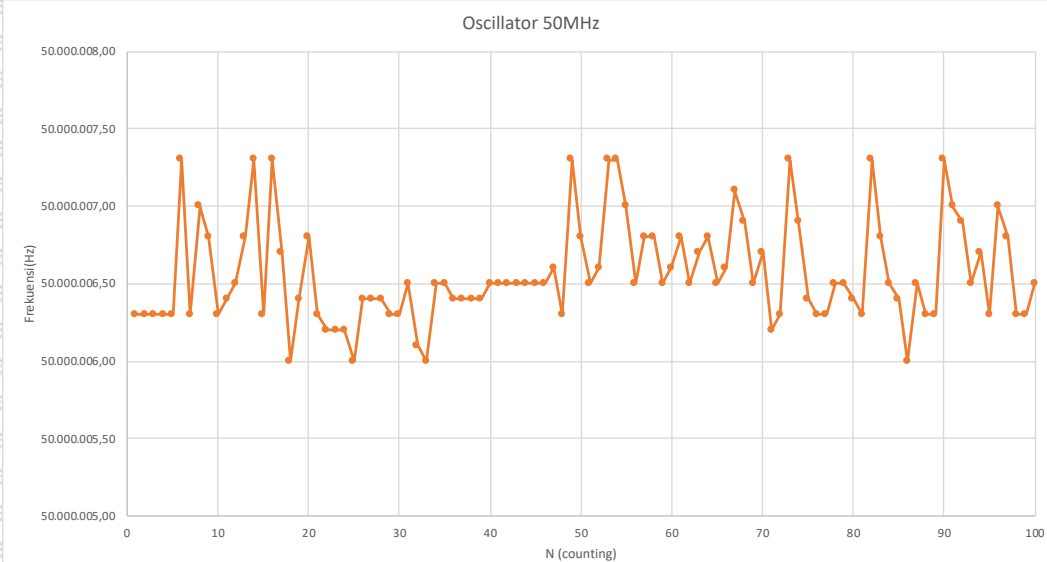
masing-masing memiliki eror yang bervariasi. Gambar dibawah ini merupakan hasil pencacahan oscillator 16MHz, 10MHz, 20MHz, 50MHz.



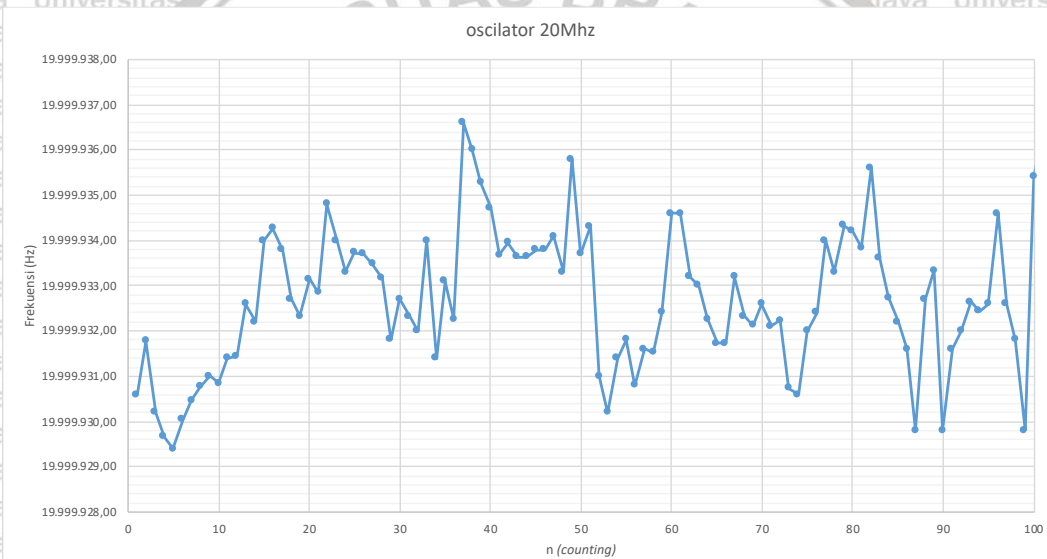
Gambar 5. 16 Grafik hasil pencacahan oscillator 16MHz.



Gambar 5. 17 Grafik hasil pencacahan oscillator 10MHz.



Gambar 5. 18 Grafik Hasil cacahan oscillator 50MHz.



Gambar 5. 19 Grafik Hasil cacahan oscillator 20MHz.

Masing-masing input sinyal memiliki kestabilan osilasi lebih dari 1 Hz, akan tetapi jika diperhatikan secara seksama dari gambar grafik diatas, terlihat bahwa sistem pencacah yang dibuat dalam penelitian ini tetap mampu mendeteksi perubahan frekuensi dibawah 0.1 Hz.

Untuk lebih memperkuat hasil analisis dilakukan metode statistik. Untuk mengetahui error pada setiap pencacahan. Hasil dari Analisa statistik ditampilkan pada table 5.1 dibawah ini.

oscillator	Rata-rata (Hz)	Standar Deviasi
Rb 10MHz	10000000,05	0,050600001
10MHz	10000004,53	0,055
16MHz	16000014,065	1,325
20MHz	19999933,5	2,899999999
50MHz	50000006,4	0,100000001

Tabel 5. 1 Tabel Hasil analisis pencacahan *oscillator*.

Dari hasil Analisa metode statistik juga dapat disimpulkan bahwa pada proses pencacahan kristal rubidium resolusi pengukuran yang dihasilkan mencapai lebih kecil dari 0,1 Hz yaitu 0,05 Hz tanpa menambah atau meningkatkan nilai input referensi (*time base*).



BAB VI

PENUTUP

6.1. Kesimpulan

Pada penelitian ini dibangun sistem pencacah frekuensi dengan metode penggabungan antara metode *reciprocal* dan TDC (*time digital converter*). Metode ini telah diterapkan ke dalam IC Spartan-6 FPGA untuk menghasilkan pencacah frekuensi 32-bit dengan resolusi pengukuran 0.01Hz dengan resolusi TDC 40ps. Hasil cacaha pada FPGA dikirim ke mikrokontroler melalui komunikasi UART dengan mekanisme pengiriman secara multiplekser yang dibangun didalam FPGA. Data hasil pencacahan selanjutnya diolah didalam mikrokontroler, setelah proses pengolahan data selesai, maka data akan dikirim kekomputer (PC) melalui komunikasi USB HID. Komputer akan menampilkan hasil pencacahan dalam bentuk grafik.

Interpolator TDC pada sistem pencacah ini memiliki peran penting untuk meningkatkan resolusi pengukuran. Dengan menggunakan TDC dapat menghitung titik-titik yang tidak dapat *ter-counting*. TDC dibangun menggunakan algoritma *thermometer code* dengan CARRY4 *logic* sebagai *delay line*. Dengan menggunakan panjang *line* 256 (256 CARRY4 *logic*) dan *clock system* 100MHz resolusi pengukuran maksimum yang dapat diperoleh sebesar 40ps.

DAFTAR PUSTAKA

Ayad, M. M. and Minisy, I. M. (2016) 'Detection and kinetics of methylamine on chitosan film coated quartz crystal microbalance electrode', *Progress in Organic Coatings*. Elsevier B.V., 100, pp. 76–80. doi: 10.1016/j.porgcoat.2016.01.012.

Boven, P. (2007) 'Increasing the resolution of reciprocal frequency counters', *UKW-Tagung Weinheim*.

Chen, Y., Liu, L. and Liu, H. (2011) 'A New High-precision Frequency and Duty Cycle Measurement Method with FPGA Implementation', pp. 0–3.

CHTOUROU, S. *et al.* (2017) 'Performance analysis and optimization of cluster-based mesh FPGA architectures: design methodology and CAD tool support', *Turkish Journal of Electrical Engineering & Computer Sciences*, 25, pp. 2044–2054. doi: 10.3906/elk-1506-51.

Dadouche, F. *et al.* (2015) 'New Design-methodology of High-performance TDC on a Low Cost FPGA Targets', *Sensors & Transducers*, 193(10), pp. 123–134. Available at: http://www.sensorsportal.com/HTML/DIGEST/october_2015/Vol_193/P_2745.pdf.

Fang, Y. and Chen, X. (2012) 'Design of Equal Precision Frequency Meter Based on FPGA *', 2012(October), pp. 696–700.

Johansson, S. (2005) 'New frequency counting principle improves resolution', in *Proceedings of the 2005 IEEE International Frequency Control Symposium and Exposition*, 2005. IEEE, pp. 628–635. doi: 10.1109/FREQ.2005.1574007.

Lal, G. and Tiwari, D. C. (2018) 'Investigation of nanoclay doped polymeric composites on piezoelectric Quartz Crystal Microbalance (QCM) sensor', *Sensors and Actuators, B: Chemical*. Elsevier B.V., 262, pp. 64–69. doi: 10.1016/j.snb.2018.01.200.

Packard, H. (1997) 'Fundamentals of the Electronic Counters', *USA, Hewlett-Packard Company*, 4844, pp. 1–44. doi: 10.1016/j.lcats.2008.08.011.

Poole, I. (2015) *RF Digital Frequency Counter*. Available at: http://www.radio-electronics.com/info/t_and_m/frequency_counter/counter_basics.php (Accessed: 11 April 2018).

Rubiola, E. *et al.* (2016) 'The Ω Counter, a frequency counter based on the linear regression', *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 63(7), pp. 961–969. doi: 10.1109/TUFFC.2016.2570604.

Sakti, S. (2014) 'Dual Channel High Precision 26 Bit Frequency Counter Using CPLD XC95108XL for QCM Sensor System', *International Journal of Information and Electronics Engineering*, 4(3). doi: 10.7763/IJIEE.2014.V4.441.

SiTime (2013) 'Frequency Measurement Guidelines for Oscillators 1 Introduction 2 Common Frequency Measurement Issues 2 - 1 Measurements made with different frequency counters do not match', (December), pp. 1–13.

Valdes, M. D. *et al.* (2008) 'IMPLEMENTATION OF A FREQUENCY MEASUREMENT CIRCUIT FOR HIGH-ACCURACY QCM SENSORS', *Reading and Writing*, 6(1), pp. 233–236.

Wang, J. *et al.* (2009) 'A Fully Fledged TDC Implemented in', *Time*, pp. 290–294.

Xilinx (2009) *Digital Clock Manager (DCM) Module*.

Zhao, L. *et al.* (2013) 'The design of a 16-channel 15 ps TDC implemented in a 65 nm FPGA', *IEEE Transactions on Nuclear Science*, 60(5), pp. 3532–3536. doi: 10.1109/TNS.2013.2280909.

Zheng, J. *et al.* (2017) 'Low-Cost FPGA TDC With High Resolution and Density', *IEEE Transactions on Nuclear Science*, 64(6), pp. 1401–1408. doi: 10.1109/TNS.2017.2705802.

