



**RANCANG BANGUN DAN IMPLEMENTASI ALAT UKUR DIGITAL
UNTUK MENGUKUR RELASI GIGI ANTERIOR**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Program Studi Teknik Elektro (S1) dan mencapai gelar Sarjana Teknik

Oleh
MUHAMAD SETYO WIBOWO
NIM 101910201050

**PROGRAM STUDI STRATA 1 TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS JEMBER
2015**

PERSEMBAHAN

Alhamdulillah, segala puji bagi Allah SWT yang telah memberikan limpahan nikmat yang sangat luar biasa kepada penulis dan tidak lupa juga sholawat kepada Nabi besar Muhammad SAW yang telah membawa kita semua menuju peradaban manusia yang lebih baik. Skripsi bagi saya adalah salah satu fase kehidupan dimana saya berada di titik terendah dengan berbagai cobaan hidup sehingga butuh banyak dukungan agar saya mampu bangkit dan akhirnya menyelesaikannya. Oleh karena itu, dengan kerendahan hati penulis mempersembahkan karya tulis ini untuk :

1. Kedua orang tuaku, Ibunda Erly Nurwahyu Ningsih dan Ayahanda Muljadi yang selalu mendo'akan dan selalu mendukung baik moral maupun materil.
2. Kakakku Dian Amelia dan Isnani Hayuningtias yang selalu menjadi motivasi penulis selalu mendo'akan agar sukses selalu.
3. Semua Dosen Jurusan Teknik Elektro Fakultas Teknnik Universitas Jember yang telah membimbing dan memberikan ilmu. Terutama Bapak Mohamad Agung Prawira Negara, S.T., M.T selaku DPU, drg. Leliana Sandra Devi A.P Sp.Ort selaku DPA.
4. Nurul Dwi Kurniati yang selalu memberikan semangat dan mendo'akan agar sukses selalu.
5. Seluruh Guru-guruku dari TK, SD, MTsN, SMAMDA dan Guru Mengaji yang telah membimbing dengan sabar dan memberikan ilmu.
6. Teman-teman teknik, khususnya Teknik Elektro angkatan 2010.
7. Teman-teman para penghuni Laboratorium Jaringan Komputer dan Laboratorium Sistem Kendali.
8. Teman-teman UKM ROBOTIKA dan Seluruh Pengurus HME Periode 2011-2015.
9. Teman-teman kontraan Anam, Fahmi, Galang, Saviq dan Riska Selalu menemani dan memberi semangat.

MOTTO

“Dan mengharapkanlah kalian pada rahmatNya Allah (Surga)
dan takutlah pada siksaNya (Neraka)”
(terjemahan Q.S Al Isra’ ayat 57)

Man jadda wa jada (siapa yang bersungguh – sungguh akan berhasil)
Man shobaro zafiro (siapa yang bersabar akan beruntung)
Man saaro“ala darbi washola (siapa yang berjalan dijalurNya akan sampai)
(A.Fuadi)

“Perjuangan, kegagalan dan kesuksesan adalah satu paket yang tidak bisa di beli
terpisah jalani perjuangannya bangkit dari kegagalanya dan miliki kesuksesannya”
(Nurul Dwi Kurniati)

“Perjuangan kerasnya hidup sendiri dan tak ada yang menemani selain Allah SWT”
(Riki Setian)

“Sing Penting Wani Disek”
(Elektro 2010)

PERNYATAAN

Saya yang bertanda tangan dibawah ini

Nama : Muhamad Setyo Wibowo

NIM : 101910201050

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “RANCANG BANGUN DAN IMPLEMENTASI ALAT UKUR DIGITAL UNTUK MENGUKUR GIGI *ANTERIOR*” adalah benar-benar karya sendiri, kecuali kutipan yang sudah ada disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak mana pun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, Januari 2015

Yang menyatakan,

Muhamad Setyo Wibowo

NIM. 101910201050

SKRIPSI

**RANCANG BANGUN DAN IMPLEMENTASI ALAT UKUR DIGITAL
UNTUK MENGUKUR RELASI GIGI ANTERIOR**

Oleh

MUHAMAD SETYO WIBOWO

NIM 101910201050

Pembimbing

Dosen Pembimbing Utama : Mohamad Agung Prawira Negara, S.T., M.T.

Dosen Pembimbing Anggota : drg. Leliana Sandra Devi Ade Putri, Sp.Ort

PENGESAHAN

Skripsi yang berjudul “RANCANG BANGUN DAN IMPLEMENTASI ALAT UKUR DIGITAL UNTUK MENGUKUR GIGI *ANTERIOR*” telah diuji dan disahkan pada:

hari, tanggal :

tempat : Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

Tim Penguji

Pembimbing Utama,

Pembimbing Anggota,

Mohamad Agung P.N , S.T., M.T

NIP. 19871217 201212 1 003

drg. Leliana Sandra Devi A.P Sp.Ort

NIP. 19720824 200112 2 001

Penguji I,

Penguji II,

Bambang Supeno, S.T., M.T.

NIP. 19690630 199512 1 001

Satryo Budi Utomo, S.T., M.T.

NIP. 19850126 20081 1 002

Mengesahkan,
Dekan Fakultas Teknik,

Ir. Widyono Hadi, M.T.

NIP. 19610414 198902 1 001

**RANCANG BANGUN DAN IMPLEMENTASI ALAT UKUR DIGITAL
UNTUK MENGUKUR GIGI ANTERIOR**

Muhamad Setyo Wibowo

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

ABSTRAK

Sampai saat ini alat untuk mengukur maloklusi gigi *anterior* masih dilakukan secara manual, yaitu dengan cara membuat model gigi pasien yang di ukur menggunakan jangka berujung runcing, penggaris, dan *digital caliper*. Oleh karena itu, terciptalah sebuah ide membuat alat ukur digital untuk mengukur relasi gigi anterior secara vertikal dengan sensor flex agar mempermudah dokter gigi mengetahui jarak *overbite* dan *overjet* pada gigi anterior. Kalibrasi alat ukur digital ini yang menggunakan sensor flex dapat dilakukan dengan dua cara, yaitu menggunakan perhitungan rumus persamaan garis dan menggunakan bantuan tool *trandline* di Excel. Pada perbandingan data *overjet* dan *overbite* antara pengukuran manual dengan digital *error* persen *overjet* terkecil 0% dan yang terbesar 490%, untuk *overbite error* persen 0% dan yang terbesar 19%.

Kata Kunci: *Alat ukur digital, Overjet, Overbite*

DESIGN AND IMPLEMENTATION MEASURE FOR MEASURE DENTAL DIGITAL ANTERIOR

Muhamad Setyo Wibowo

Electrical Engineering Department, Engineering Faculty, University Jember

ABSTRACT

Until now measuring malocclusion instrument for anterior tooth is still using manual methods, namely to create a model of the patient's teeth which then measured using a compass needle-nose pliers, a ruler, and a digital caliper. Therefore, we had an idea of making a digital measuring instrument for measuring the vertical anterior tooth relationship using flex sensors in order to facilitate the dentist to know the distance overbite and overjet on anterior teeth. The digital calibration of measuring instruments that use flex sensors can be done in two ways, which are using the calculation equations and using the help line trandline tool in Excel. The comparison of data between the overjet and overbite for digital manual measurement with the smallest error percent for overjet is 0% and 490% for the largest, as for overbite the smallest error 0% and 19% for the largest.

Keywords: *Digital measuring instrument, overjet, overbite*

RINGKASAN

Sampai saat ini alat untuk mengukur maloklusi gigi *anterior* masih dilakukan secara manual dengan model gigi pasien dan di ukur dengan jangka berujung runcing, penggaris, dan *digital caliper* (jangka sorong). Hal ini tentu membutuhkan ketelitian di setiap pengukurannya. Sistem penyimpanan data pun pada umumnya masih dilakukan secara manual. Dari sinilah tercipta sebuah ide membuat alat ukur digital untuk mengukur relasi gigi anterior secara vertikal sensor flex agar mempermudah dokter mengetahui jarak *overbite* dan *overjet* pada gigi anterior.

Besar *overjet* ditentukan oleh posisi gigi anterior maksila dan mandibula. *Overjet* tergantung pada inklinasi dari gigi-gigi insisivus dan hubungan antara antero-posterior dari lengkung gigi. *Overbite* adalah jarak vertikal antara gigi-gigi insisal insisivus atas dengan insisal insisive bawah. Dipengaruhi oleh derajat perkembangan vertikal dari segmen *dento-alveolar anterior*.

Proses kalibrasi ini ada beberapa tahap untuk mendapatkan hasil nilai yang sesungguhnya dari nilai ADC. Pengolahan data yang pertama yaitu pengambilan 10 sampel yang terdiri dari beberapa ukuran gigi yang berbeda. Untuk linier nilai keluaran dari sensor ini peneliti menggunakan excel untuk mempermudah pembuatan grafik Linier dan menggunakan rumus persamaan garis.

Pada pengujian *overjet* dan *overbite* dengan sampel yang telah di tentukan yaitu mahasiswa Diploma III teknik elektro angkatan 2012 Universitas Jember. Pada perbandingan data *overjet* dan *overbite* antara pengukuran manual dengan digital *error* persen *overjet* terkecil 0% dan yang terbesar 490%, untuk *overbite error* persen 0% dan yang terbesar 19%.

PRAKATA

Bismillahirrohmanirrohim,

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul " RANCANG BANGUN DAN IMPLEMENTASI ALAT UKUR DIGITAL UNTUK MENGUKUR GIGI *ANTERIOR*". Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Keberadaan skripsi ini tidak terlepas dari bantuan serta dorongan dari berbagai pihak. Untuk itu penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT, yang telah memberikan karunia kehidupan sehingga dapat terselesaikan tulisan ini.
2. Bapak Ir. Widyono Hadi, MT selaku Dekan Fakultas Teknik Universitas Jember.
3. Bapak Dr. Triwahju Hardianto, S.T., M.T., Selaku ketua Jurusan Teknik Elektro, Fakultas Teknik Universitas Jember.
4. Bapak Bambang Supeno, S.T., M.T. Selaku komisi bimbingan yang senantiasa mencurahkan segenap waktunya dalam penyusunan skripsi ini.
5. Agung Prawira Negara, S.T., M.T., selaku Dosen Pembimbing Utama (DPU) yang senantiasa mencurahkan segenap waktunya dalam penyusunan skripsi ini.
6. drg. Leliana Sandra Devi A.P Sp.Ort selaku Dosen Pembimbing Anggota (DPA) yang senantiasa mencurahkan segenap waktunya dalam penyusunan skripsi ini.
7. Dr. Azmi Saleh S.T., M.T. selaku Dosen Pembimbing Akademik.
8. Seluruh Dosen Teknik Elektro Universitas Jember yang tidak dapat saya sebutkan satu-persatu, terima kasih atas bimbingan yang telah diberikan.

9. Kedua orang tuaku, Ibunda Erly Nurwahyu Ningsih dan Ayahanda Muljadi yang selalu mendo'akan dan selalu mendukung baik moral maupun materil.
10. Kakakku Dian Amelia dan Isnani Hayuningtias yang selalu menjadi motivasi penulis selalu mendo'akan agar sukses selalu
11. Nurul Dwi Kurniati yang selalu memberikan semangat dan mendo'akan agar sukses selalu.
12. Teman-teman penghuni Laboratorium Jaringan Komputer dan Laboratorium Sistem Kendali yang telah membantu dalam proses penyelesaian tugas akhir ini.
13. Keluarga Besar kontaraan Sumber Alam yang telah memberikan berbagai pengalaman yang tak ternilai harganya.
14. Semua teman-teman teknik elektro 2010 dengan mottonya “ Seng Penting Wani Disik” atas semangat, dukungan, dan motivasi, serta memberikan warna di sini.
15. Semua pihak teman-teman elektro 2012 yang terutama D3 yang telah meluangkan waktu untuk pengambilan data.
16. Buat temen temen fakultas kedokteran gigi terutama fais yang telah membantu pembuatan model gigi.

Semoga laporan tugas akhir ini dapat bermanfaat dalam mengembangkan ilmu pengetahuan khususnya untuk disiplin ilmu teknik elektro, kritik dan saran diharapkan terus mengalir untuk lebih menyempurnakan proyek akhir ini dan diharapkan dapat dikembangkan untuk penelitian-penelitian selanjutnya.

Jember, Januari 2015

Penulis

DAFTAR ISI

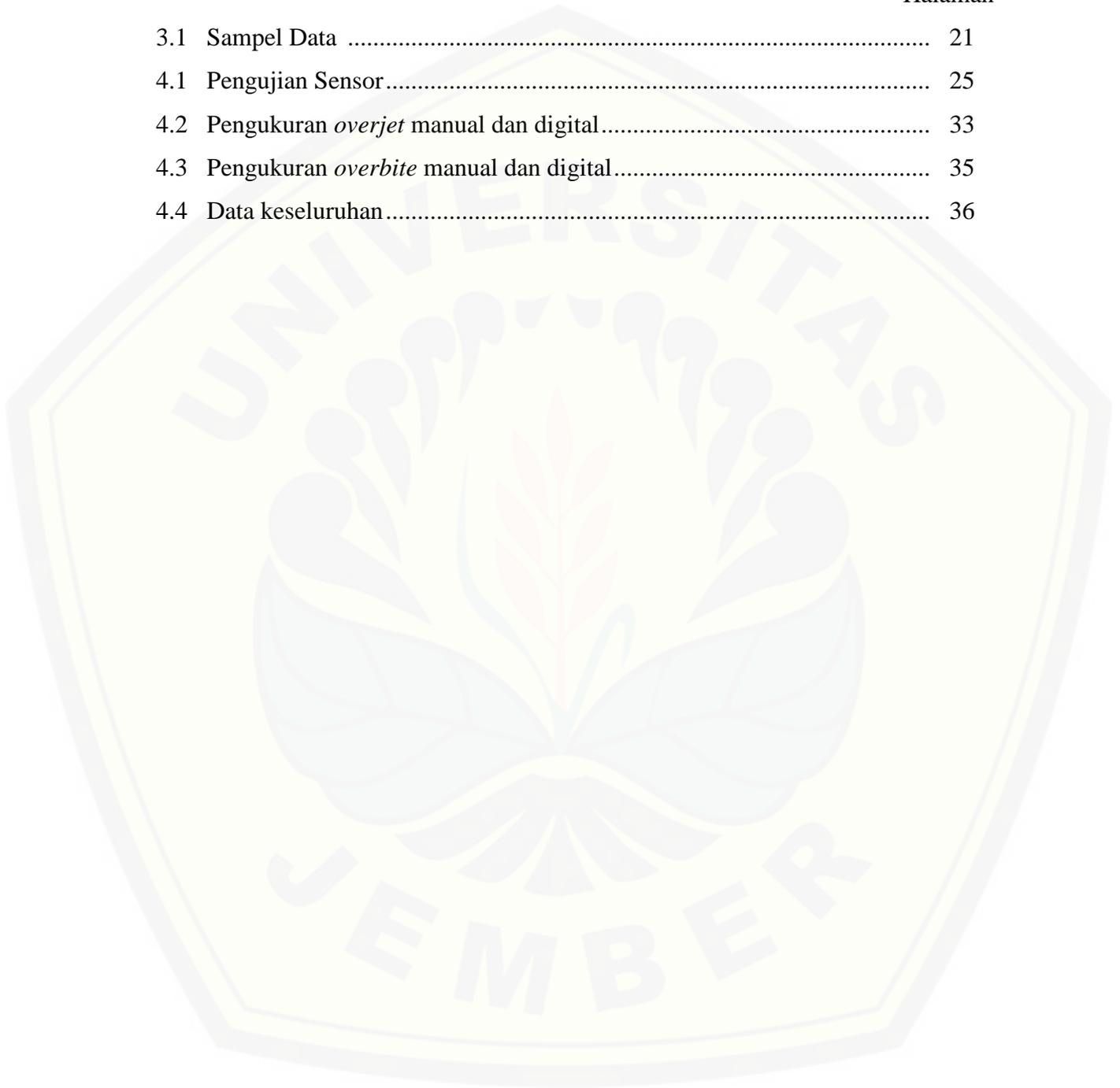
| | Halaman |
|-----------------------------------|---------|
| HALAMAN JUDUL | i |
| HALAMAN PERSEMBAHAN | ii |
| HALAMAN MOTO | iii |
| HALAMAN PERNYATAAN | iv |
| HALAMAN PEMBIMBING | v |
| HALAMAN PENGESAHAN | vi |
| ABSTRAK | vii |
| ABSTRACT | vii |
| RINGKASAN | ix |
| PRAKATA | xi |
| DAFTAR ISI | xii |
| DAFTAR TABEL | xv |
| DAFTAR GAMBAR | xvi |
| BAB 1. PENDAHULUAN | |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 1 |
| 1.3 Tujuan | 2 |
| 1.4 Manfaat | 2 |
| BAB 2. TINJAUAN PUSTAKA | |
| 2.1 Ortodonsi | 3 |
| 2.2 Nomenkuatur gigi | 3 |
| 2.3 Maloklusi | 4 |
| 2.4 Overjet | 4 |
| 2.5 Overbite | 5 |
| 2.6 Arduino uno | 5 |
| 2.5.1 Daya..... | 6 |
| 2.5.2 Memori | 7 |

| | |
|---|-----------|
| 2.5.3 Input dan Output..... | 7 |
| 2.5.4 Komunikasi..... | 8 |
| 2.7 Flex Sensor | 9 |
| 2.8 Analog to Digital Converter..... | 9 |
| 2.9 Rangkaian Pembagi Tegangan | 11 |
| 2.10 Software Visual Studio 2010..... | 11 |
| BAB 3. METODOLOGI PENELITIAN | |
| 3.1 Jenis Penelitian | 13 |
| 3.2 Rancangan Penelitian | 13 |
| 3.3 Waktu dan Tempat Penelitian | 13 |
| 3.3.1 Waktu Penelitian | 13 |
| 3.3.2 Tempat Penelitian | 13 |
| 3.4 Alat dan Bahan | 13 |
| 3.4.1 Alat..... | 13 |
| 3.4.2 Bahan | 13 |
| 3.5 Populasi dan Sampel Penelitian | 14 |
| 3.5.1 Populasi | 14 |
| 3.5.2 Besar Sampel | 14 |
| 3.6 Cara pengukuran manual <i>Overjet</i> dan <i>Overbite</i>..... | 14 |
| 3.6.1 Pengukuran <i>Overjet</i> | 14 |
| 3.6.2 Pengukuran <i>Overbite</i> | 14 |
| 3.7 Pengukuran secara digital..... | 15 |
| 3.8 Blok Diagram | 15 |
| 3.9 Flowchart Program..... | 15 |
| 3.10 Hasil perancangan alat..... | 17 |
| 3.11 Aplikasi GUI | 20 |
| 3.12 Prose Kalibrasi | 21 |
| BAB 4. HASIL DAN PEMBAHASAN | |
| 4.1 Pengujian Sensor | 25 |
| 4.2 Pengujian Perangkat Lunak | 25 |
| 4.3 Pengujian Perangkat Keras | 30 |

| | |
|---|-----------|
| 4.3.1 Pengujian pengiriman data..... | 30 |
| 4.3.2 Pengujian tombol..... | 31 |
| 4.3.3 Pengujian masukan sensor | 31 |
| 4.4 Pengujian <i>Overjet</i> | 32 |
| 4.5 Pengujian <i>Overbite</i>..... | 34 |
| 4.6 Analisi sitem keseluruhan..... | 36 |
| | |
| BAB 5. PENUTUP | |
| 5.1 Kesimpulan | 38 |
| 5.2 Saran | 38 |
| DAFTAR PUSTAKA | 39 |
| LAMPIRAN | 40 |

DAFTAR TABEL

| | Halaman |
|--|---------|
| 3.1 Sampel Data | 21 |
| 4.1 Pengujian Sensor..... | 25 |
| 4.2 Pengukuran <i>overjet</i> manual dan digital..... | 33 |
| 4.3 Pengukuran <i>overbite</i> manual dan digital..... | 35 |
| 4.4 Data keseluruhan..... | 36 |



DAFTAR GAMBAR

| | Halaman |
|---|---------|
| 2.1 Gigi Permanen | 3 |
| 2.2 <i>Overjet</i> | 4 |
| 2.3 <i>Overbite</i> insisal | 5 |
| 2.4 Skematik arduino uno | 6 |
| 2.5 Bentuk fisik <i>flex sensor</i> | 9 |
| 2.6 Rangkaian Pembagi Tegangan | 10 |
| 2.7 Tampilan dinding kerja Microsoft Visual Studio C# 2010 Ultimate | 11 |
| 3.1 Blog Diagram | 15 |
| 3.2 <i>Flowchart Overjet</i> | 16 |
| 3.3 <i>Flowchart Overbite</i> | 17 |
| 3.4 Perancangan alat | 18 |
| 3.5 Hasil pembuatan alat | 18 |
| 3.6 Hasil pembuatan alat dan <i>interface</i> laptop | 19 |
| 3.7 Tampilan GUI | 20 |
| 3.8 Grafik hubungan antara pengukuran <i>overjet</i> terhadap nilai ADC | 22 |
| 3.9 Grafik hubungan antara pengukuran <i>overjet</i> terhadap nilai ADC | 22 |
| 4.1 Tampilan Aplikasi yang sudah jalan | 26 |
| 4.2 Tampilan ketika tombol <i>Start</i> di tekan | 27 |
| 4.3 Tampilan <i>textbox</i> di isi | 28 |
| 4.4 Tampilan excel setelah di tekan <i>save</i> | 29 |
| 4.5 Tampilan kiriman dari arduino di Laptop | 30 |
| 4.6 Tampilan ketika di tekan tombol | 31 |
| 4.7 Tampilan masukan sensor | 32 |

DAFTAR LAMPIRAN

| | Halaman |
|--|---------|
| 1. Alat dan bahan untuk cetak gigi..... | 41 |
| 2. Proses pembuatan adonan dengan alginad..... | 41 |
| 3. Proses pencetakan gigi..... | 42 |
| 4. Hasil cetakan atas dan bawah | 42 |
| 5. Proses pengecoran..... | 43 |
| 6. Hasil cetak model gigi | 43 |
| 7. Pengukuran <i>overbite</i> dengan penggaris | 44 |
| 8. Pengukuran <i>overjet</i> dengan penggaris | 44 |
| 9. Pengukuran <i>overjet</i> dan <i>overbite</i> digital | 45 |
| 10. Tampilan GUI <i>minus overjet</i> | 46 |
| 11. Tampilan GUI gigitan terbuka | 46 |
| 12. <i>Listing</i> program arduino..... | 47 |
| 13. <i>Listing</i> program C# | 49 |

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Dunia elektronika khususnya bidang biomedika adalah suatu bidang yang saat ini sedang mengalami perkembangan dengan pesat hal ini dikarenakan memang tuntutan jaman yang menuntut suatu kinerja yang otomatis sehingga bisa memudahkan manusia.

Dunia kedokteran gigi saat ini tidak kalah pentingnya dalam perkembangan ilmu maupun perkembangan alat-alat yang dipergunakan di dalam praktek. Maloklusi adalah penyimpangan letak gigi atau malrelasi lengkung gigi (rahang) di luar rentang kewajaran yang dapat diterima. Maloklusi dapat timbul karena faktor keturunan dimana ada ketidak sesuaian besar rahang dengan besar gigi-gigi di dalam mulut. Misalnya, ukuran rahang mengikuti garis keturunan Ibu, dimana rahang berukuran kecil, sedangkan ukuran gigi mengikuti garis keturunan bapak yang giginya lebar-lebar. *Overjet* adalah jarak horizontal antara gigi gigi insisivus atas dan bawah pada keadaan oklusi. *Overbite* adalah jarak vertikal antara gigi-gigi insisal insisivus atas dengan insisal insisive bawah. Beberapa orang berusaha memperbaiki abnormalitas giginya dengan melakukan beberapa perawatan untuk memperbaiki masalah pada giginya.

Sampai saat ini alat untuk mengukur maloklusi gigi *anterior* masih dilakukan secara manual, yaitu dengan cara membuat model gigi pasien yang di ukur menggunakan jangka berujung runcing, penggaris, dan *digital caliper*. Hal ini tentu membutuhkan ketelitian di setiap pengukurannya. Sistem penyimpanan data pun pada umumnya masih dilakukan secara manual. Oleh karena itu, terciptalah sebuah ide membuat alat ukur digital untuk mengukur relasi gigi anterior secara vertikal dengan sensor flex agar mempermudah dokter gigi mengetahui jarak *overbite* dan *overjet* pada gigi anterior.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan permasalahan yaitu:

1. Bagaimana mendesain alat ukur digital untuk mengukur relasi gigi *anterior*?
2. Bagaimana cara kalibrasi alat ukur digital untuk mengukur relasi gigi *anterior*?

1.3 Tujuan

Penelitian ini memiliki beberapa tujuan diantaranya:

1. Membuat alat untuk alat ukur digital untuk mengukur relasi gigi *anterior*.
2. Mengetahui cara kalibrasi alat ukur digital untuk mengukur relasi gigi *anterior*.

1.4 Manfaat

Penelitian ini memiliki beberapa manfaat diantaranya:

1. Memudahkan pengukuran gigi *anterior*.
2. Mengembangkan alat untuk teknologi kedokteran gigi.

BAB 2. TINJAUAN PUSTAKA

2.1 Ortodonsi

Ilmu ortodonsi adalah gabungan ilmu dan seni yang berhubungan dengan perkembangan dan menegakkan atau merawat anomali dari geligi, rahang, dan muka serta pengaruhnya terhadap kesehatan fisik, estetik dan mental. Di dalam ilmu ortodonsi ini juga diulas tentang *oklusi* dan maloklusi. Oklusi adalah berkontaknya gigi geligi rahang atas dengan permukaan gigi geligi rahang bawah pada saat kedua rahang tersebut menutup sesuai fungsinya. Maloklusi merupakan setiap yang menyimpang dari *oklusi* itu sendiri.(Foster, 1997)

2.2 Nomenkuatur gigi

| | | | | | | | | | | | | | | | |
|------------------------|-----------|-----------|-----------|-----------|----------|-----------|-----------|--------------------------|-----------|----------|-----------|-----------|-----------|-----------|-----------|
| 1 Rahang atas kanan | | | | | | | | 2 Rahang atas kiri | | | | | | | |
| <i>Dextra Superior</i> | | | | | | | | <i>Sinistra Superior</i> | | | | | | | |
| M3 | M2 | M1 | P2 | P1 | C | I2 | I1 | I1 | I2 | C | P1 | P2 | M1 | M2 | M3 |
| 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |
| | | | | | | | | | | | | | | | |
| M3 | M2 | M1 | P2 | P1 | C | I2 | I1 | I1 | I2 | C | P1 | P2 | M1 | M2 | M3 |
| 48 | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
| 4 Rahang bawah kanan | | | | | | | | 3 Rahang bawah kiri | | | | | | | |
| <i>Dextra Inferior</i> | | | | | | | | <i>Sinistra Inferior</i> | | | | | | | |

Gambar 2.1 Gigi Permanen (Sumber: Minasari, 2008)

Sistem 2 angka dari Federation Dental International (FDI) :

1. Angka pertama menunjukkan kuadran gigi
2. Angka kedua menunjukkan elemen gigi

2.3 Maloklusi

Maloklusi adalah setiap keadaan yang menyimpang dari oklusi normal, *maloklusi* juga diartikan sebagai suatu kelainan susunan gigi geligi atas dan bawah yang berhubungan dengan bentuk rongga mulut serta fungsi.

Maloklusi dapat timbul karena faktor keturunan dimana ada ketidaksesuaian besar rahang dengan besar gigi-gigi di dalam mulut. Misalnya, ukuran rahang mengikuti garis keturunan Ibu, dimana rahang berukuran kecil, sedangkan ukuran gigi mengikuti garis keturunan bapak yang giginya lebar-lebar. Gigi-gigi tersebut tidak cukup letaknya di dalam lengkung gigi. (Pambudi raharjo, 2009).

2.4 Overjet

Besar *overjet* ditentukan oleh posisi gigi anterior maksila dan mandibula. *Overjet* tergantung pada inklinasi dari gigi-gigi insisivus dan hubungan antara antero-posterior dari lengkung gigi. *Overjet* adalah jarak horizontal antara gigi insisivus atas dan bawah pada keadaan oklusi (gambar 2.1).



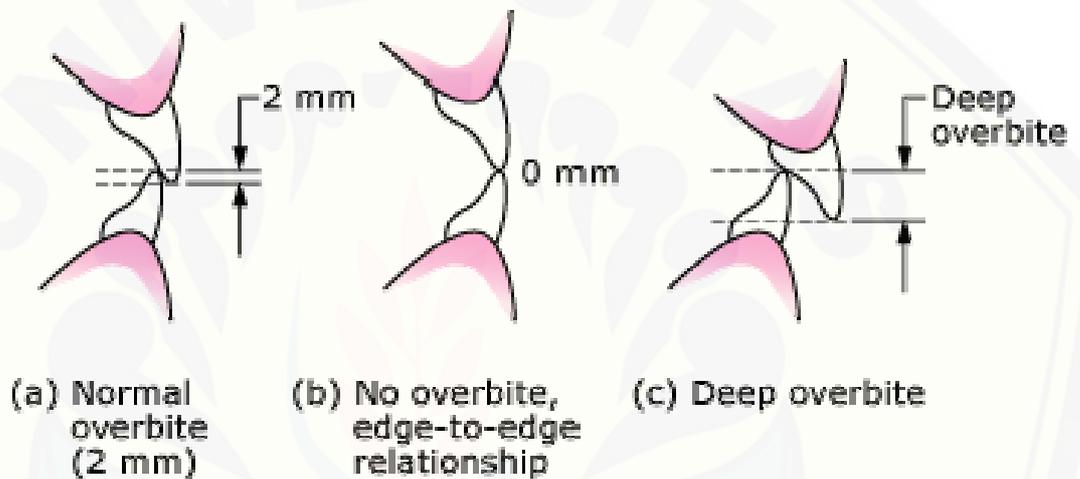
Gambar 2.2 *overjet* (Sumber: Foster,1997)

relasi gigi dalam arah anteroposterior dan lateral (fasiolingual) :

1. *Overjet* besar / berlebihan (> 4 mm)
2. *Overjet* normal (2 – 4 mm)
3. *Overjet* kecil (< 2 mm)
4. *Edge to edge bite* (0 mm)
5. *Crossbite* (bernilai negatif)

2.5 Overbite

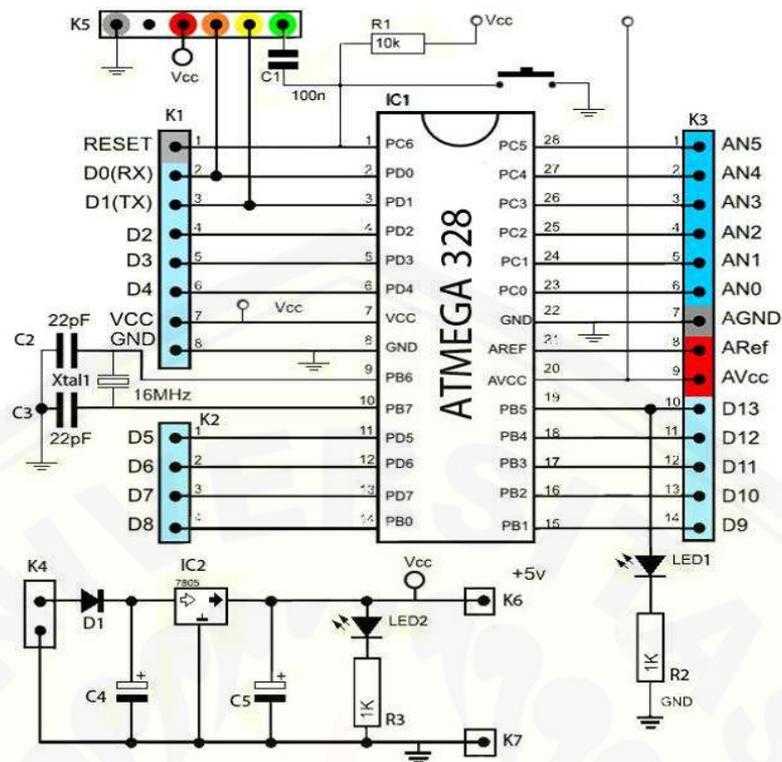
Overbite adalah jarak vertikal antara gigi-gigi insisal insisivus atas dengan insisal insisive bawah. Dipengaruhi oleh derajat perkembangan vertikal dari segmen *dento-alveolar anterior*. Idealnya, gigi-gigi Insisivus bawah harus berkontak dengan sepertiga permukaan palatal dari insisivus atas, pada keadaan oklusi, namun bisa juga terjadi *overbite* yang berlebih atau tidak ada kontak insisal. Pada keadaan ini *overbite* disebut tidak sempurna jika insisivus bawah di atas ketinggian edge insisal atas, atau gigitan terbuka, jika insisivus bawah lebih pendek dari *edge* insisal atas pada oklusi. (Foster,1997)



Gambar 2.3 *Overbite* insisal (Sumber: Lia,2014)

2.6 Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328 (*datasheet*). Memiliki 14 *pin input* dari *output* digital dimana 6 *pin input* tersebut dapat digunakan sebagai *output* PWM dan 6 *pin input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP header, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan *Board* Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC yang ke adaptor DC atau baterai untuk menjalankannya.



Gambar 2.4 Skematik Arduino Uno

(Agung Tarecha, 2012)

Uno berbeda dengan semua *board* sebelumnya dalam hal koneksi *USB to serial* yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter *USB to serial* berbeda dengan *board* sebelumnya yang menggunakan chip FTDI *driver USB to serial*.

Nama “Uno” berarti *satu* dalam bahasa Italia, untuk menandai peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian *board* USB Arduino, dan sebagai model referensi untuk platform Arduino, untuk perbandingan dengan versi sebelumnya, lihat indeks *board* Arduino.

2.6.1 Daya

Uno Arduino dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal (otomatis). Eksternal (non USB) daya dapat berasal baik dari AC ke adaptor DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan

plug jack pusat positif ukuran 2.1mm konektor *power*. Ujung kepala dari baterai dapat dimasukkan kedalam Gnd dan Vin *pin header* dari konektor *power*.

Kisaran kebutuhan daya yang disarankan untuk *board* Uno adalah 7 sampai dengan 12 volt, jika diberi daya kurang dari 7 volt kemungkinan pin 5v Uno dapat beroperasi tetapi tidak stabil kemudian jika diberi daya lebih dari 12V, regulator tegangan bisa panas dan dapat merusak board Uno.

Pin listrik adalah sebagai berikut:

1. V_{IN} . Tegangan masukan kepada *board* Arduino ketika itu menggunakan sumber daya eksternal (sebagai pengganti dari 5 volt koneksi USB atau sumber daya lainnya).
2. 5V. Catu daya digunakan untuk daya mikrokontroler dan komponen lainnya.
3. 3v3. Sebuah pasokan 3,3 volt dihasilkan oleh regulator on board.
4. GND. *Ground pin*.

2.6.2 Memori

ATmega328 memiliki 32 KB (dengan 0,5 KB digunakan untuk *bootloader*), 2 KB dari SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis dengan EEPROM *library*).

2.6.3 Input dan Output

Masing-masing dari 14 pin digital di Uno dapat digunakan sebagai masukan atau keluaran, dengan menggunakan fungsi *pinMode* (), *digitalWrite* (), dan *digitalRead* (), beroperasi dengan daya 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki internal *pull up* resistor (secara *default* terputus) dari 20-50 kOhms. Selain itu, beberapa *pin* memiliki fungsi khusus:

1. *Serial*: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. *Pin* ini dihubungkan ke *pin* yang berkaitan dengan chip Serial ATmega8U2 USB to TTL.
2. *Interrupt Eksternal*: 2 dan 3. *Pin* ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai. Lihat (*attach Interrupt*) fungsi untuk rincian lebih lanjut.

3. *PWM*: 3, 5, 6, 9, 10, dan 11. Menyediakan *output* PWM 8-bit dengan fungsi *analogWrite ()*.
4. *SPI*: 10 (SS), 11 (Mosi), 12 (MISO), 13 (SCK). *Pin* ini mendukung komunikasi SPI menggunakan *SPI library*.
5. *LED*: 13. Ada *built in* LED terhubung ke *pin* digital 13. Ketika *pin* bernilai nilai *HIGH*, LED on, ketika *pin* bernilai *LOW*, LED *off*.
6. Uno memiliki 6 masukan analog, berlabel A0 sampai dengan A5, yang masing-masing menyediakan 10 bit dengan resolusi (yaitu 1024 nilai yang berbeda). Selain itu, beberapa *pin* memiliki fungsi khusus.
7. *I2C*: A4 (*SDA*) dan A5 (*SCL*). Dukungan I2C (TWI) komunikasi menggunakan perpustakaan *Wire*.
8. *Aref*. Tegangan referensi (0 sampai 5V saja) untuk *input* analog. Digunakan dengan fungsi *analogReference ()*.
9. *Reset*. Bawa baris ini *LOW* untuk *reset* mikrokontroler.
10. Lihat juga *mapping pin* Arduino dan *port* ATmega328.

2.6.4 Komunikasi

Uno Arduino memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di *pin* digital 0 (RX) dan 1 (TX). Sebuah ATmega8U2 sebagai saluran komunikasi serial melalui USB dan sebagai *port* virtual com untuk perangkat lunak pada komputer. Firmware '8 U2 menggunakan *driver* USB standar COM, dan tidak ada *driver* eksternal yang diperlukan. Namun, pada Windows diperlukan, sebuah *file* inf. Perangkat lunak Arduino terdapat monitor serial yang memungkinkan digunakan memonitor data tekstual sederhana yang akan dikirim ke atau dari *board* Arduino. LED RX dan TX di papan tulis akan berkedip ketika data sedang dikirim melalui *chip* USB to serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada *pin* 0 dan 1).

Sebuah *Software Serial library* memungkinkan untuk berkomunikasi secara serial pada salah satu *pin* digital pada *board* Uno's. ATmega 328 juga mendukung I2C (TWI) dan komunikasi SPI. Perangkat lunak Arduino termasuk

perpustakaan Kawat untuk menyederhanakan penggunaan bus I2C, lihat dokumentasi untuk rincian. Untuk komunikasi SPI, menggunakan perpustakaan SPI. (Suhendri, 2013)

2.7 *Flex sensor*

Sensor *Flex* adalah sensor yang memiliki perubahan resistansi akibat adanya perubahan lekukan pada bagian sensor. Sensor ini memiliki *output* berupa resistansi. Sensor ini membutuhkan tegangan sebesar +5V agar bisa bekerja. *Output* resistansi ini akan diberikan tegangan yang nantinya akan dibaca oleh mikrokontroler. Sensor ini digunakan untuk mendeteksi pergerakan lari tangan pada manusia / bagian lekukan lainnya. Mikrokontroler mengubah data menggunakan ADC (*analog to digital converter*), dimana data masukannya didapat dari tegangan yang sudah terkena resistansi.



Gambar 2.5 Bentuk Fisik *Flex Sensor*

(Sumber: Danny,2009)

Dari bentuk Fisik sensor *Flex*, sensor ini memiliki 2 jalur *output*. Sensor ini prinsip kerjanya mirip dengan *variable resistor*. Untuk dapat menggunakan sensor ini dibutuhkan sebuah rangkaian pembagi tegangan yang nantinya dihubungkan di mikrokontroler. (Danny, 2009)

2.8 *Analog to Digital Converter*

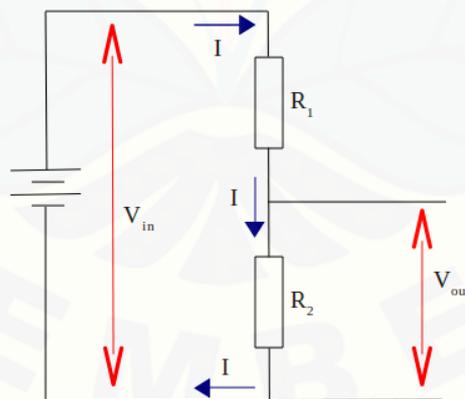
ADC (*Analog To Digital Converter*) adalah perangkat elektronika yang berfungsi untuk mengubah sinyal analog (sinyal kontinyu) menjadi sinyal digital. Perangkat ADC (*Analog To Digital Conversion*) dapat berbentuk suatu modul atau rangkaian elektronika maupun suatu chip IC. ADC (*Analog To Digital Converter*) berfungsi untuk menjembatani pemrosesan sinyal analog oleh sistem digital.

Prinsip kerja ADC adalah mengkonversi sinyal analog ke dalam bentuk besaran yang merupakan rasio perbandingan sinyal input dan tegangan referensi. Sebagai contoh, bila tegangan referensi 5 volt, tegangan input 3 volt, rasio input terhadap referensi adalah 60%. Jadi, jika menggunakan ADC 8 bit dengan skala maksimum 255, akan didapatkan sinyal digital sebesar $60\% \times 255 = 153$ (bentuk decimal) atau 10011001 (bentuk biner). (Danny, 2009)

$$\begin{aligned} \text{signal} &= \frac{\text{sample}}{\text{Max_value}} \times \text{reference_voltage} & (2.1) \\ &= \frac{153}{255} \times 5 \\ &= 3 \text{ Volts} \end{aligned}$$

2.9 Rangkaian Pembagi Tegangan

Menghubungkan resistor seri seperti ini pada tegangan DC memiliki satu keuntungan, tegangan yang berbeda muncul di setiap resistor menghasilkan sebuah rangkaian yang disebut Rangkaian Pembagi Tegangan. Rangkaian yang ditunjukkan di atas adalah pembagi tegangan sederhana di mana tiga 1V, 2V dan 6V dihasilkan dari satu *supply* tegangan *battery* 9V. Hukum tegangan Kirchoff menyatakan bahwa "tegangan dalam rangkaian tertutup sama dengan jumlah semua tegangan (IR) di seluruh rangkaian". Rangkaian dasar Resistor Seri sebagai Pembagi Tegangan dapat dilihat pada Gambar rangkaian dibawah ini:



Gambar 2.6 Rangkaian Pembagi Tegangan

(Sumber: Danny,2009)

Dari rangkaian dijelaskan bahwa terdapat dua buah resistor dimana fungsi dari kedua resistor tersebut adalah membagi tegangan. Dari gambar di atas dapat dilihat bahwa V_{out} memiliki rumus sebagai berikut:

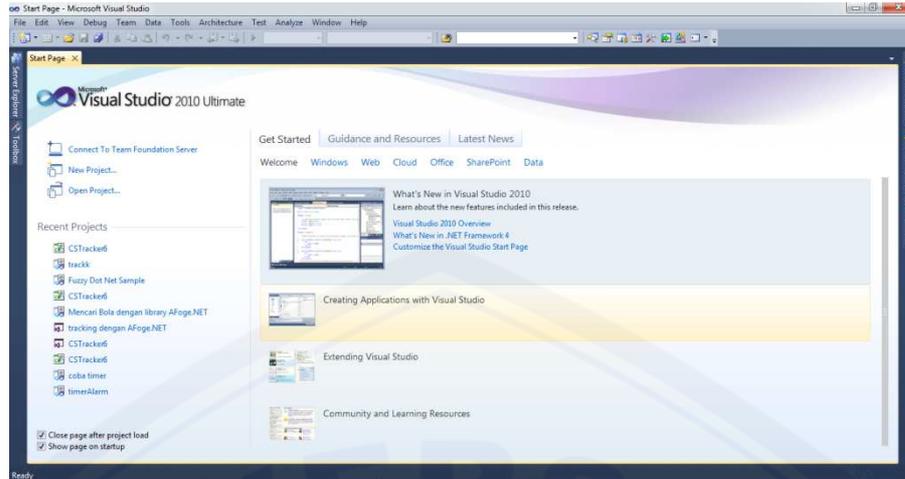
$$V_{out} = V_{In} \frac{R_2}{R_1 + R_2} \quad (2.2)$$

Rangkaian diatas dapat memberikan tegangan yang berubah sesuai dengan besar resistansi pada R1. Dalam penerapan sehari-hari rangkaian ini biasanya terdapat pada potensiometer. Sebuah resistor variabel, potensiometer atau sering disebut juga Pot, adalah contoh yang baik dari multi-resistor pembagi tegangan. Pengaturan dengan memutar/menggeser tombol pada potensiometer akan menghasilkan Resistansi yang berbeda pada kaki resistor sehingga akan menghasilkan tegangan keluaran yang kita harapkan dengan lebih akurat. (Danny, 2009)

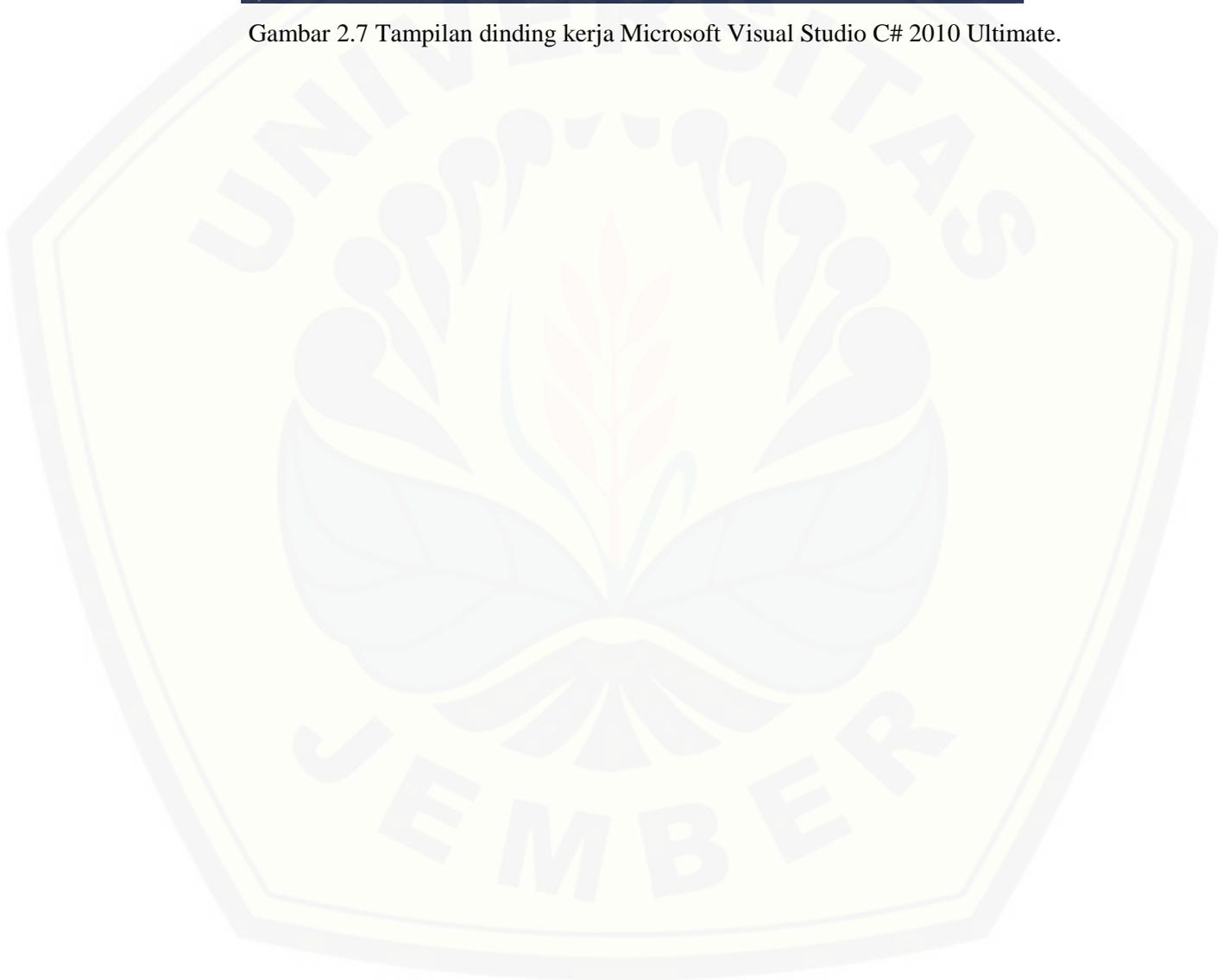
2.10 Software Visual Studio 2010

Microsoft Visual Studio 2010 merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi *console*, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, SDK, *Integrated Development Environment (IDE)*, dan dokumentasi (umumnya berupa *MSDN Library*). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi *Windows Mobile* (yang berjalan di atas .NET Compact Framework). Visual studio yang digunakan tugas akhir ini adalah visual studio 2010 yang mendukung *library up to date computer vision* dalam proses pengolahan citra. Microsoft Visual Studio C# 2010 Ultimate dipilih karena dapat digunakan dan diintegrasikan dengan *library* OpenCV terbaru. (John Sharp, 2010)



Gambar 2.7 Tampilan dinding kerja Microsoft Visual Studio C# 2010 Ultimate.



BAB 3. METODOLOGI PENELITIAN

3.1 Jenis Penelitian

Jenis penelitian yang digunakan adalah *Applied research*.

3.2 Rancangan Penelitian

Rancangan penelitian yang digunakan adalah Pengukuran Variabel Penelitian.

3.3 Waktu dan Tempat Penelitian

3.3.1 Waktu Penelitian : September - Desember 2014

3.3.2 Tempat Penelitian : Laboratorium Sistem kendali dan Laboratorium jaringan komputer Universitas Jember

3.4 Alat Dan Bahan

3.4.1 Alat

- | | |
|----------------------|---------------------|
| 1. Multimeter | 7. Penyedot timah |
| 2. Solder | 8. Gunting |
| 3. Spatula | 9. Obeng |
| 4. Sendok cetak gigi | 10. Penggaris logam |
| 5. Tang kombinasi | 11. Pensil |
| 6. Bor | 12. Jangka sorong |

3.4.2 Bahan

- | | |
|------------------------|---------------------------------------|
| 1. Flex Sensor | 5. Arduino uno |
| 2. <i>Push button</i> | 6. <i>Connector</i> dan <i>jumper</i> |
| 3. LCD | 7. Model Studi |
| 4. Komponen eletronika | 8. Alginad |

3.5 Populasi dan sampel Penelitian

3.5.1 Populasi

Populasi penelitian ini adalah mahasiswa Universitas Jember fakultas teknik elektro Diploma III angkatan 2012 dengan jumlah 20 mahasiswa aktif .

3.5.2 Besar Sampel

Besar sampel di ambil dari 20 mahasiswa yang masih aktif.

3.5.3 Sampel diambil secara acak dari Mahasiswa dengan kriteria :

1. Gigi pada model studi tidak mengalami kerusakan ataupun porus.
2. Morfologi gigi normal.

3.6 Cara pengukuran manual *overjet* dan *overbite*

3.6.1 Pengukuran *overjet*

Pengukuran dilakukan pada model studi dengan mengukur jarak dari tepi labio-insisal gigi insisivus sentral atas ke permukaan labial gigi insisivus sentral bawah dalam milimeter. Dengan penggaris yang diletakkan di tengah-tengah kedua gigi insisivus sentral atas.

3.6.2 Pengukuran *overbite*

1. Menarik garis khayal dari dataran insisal gigi insisif pertama rahang bawah sejajar dengan dataran oklusal.
2. Tarik garis khayal kedua dari ujung insisal gigi insisif pertama rahang atas sejajar dengan dataran oklusal. Jarak antara dua garis tersebut diukur. Jarak tersebut menunjukkan besarnya *overbite* pada model.

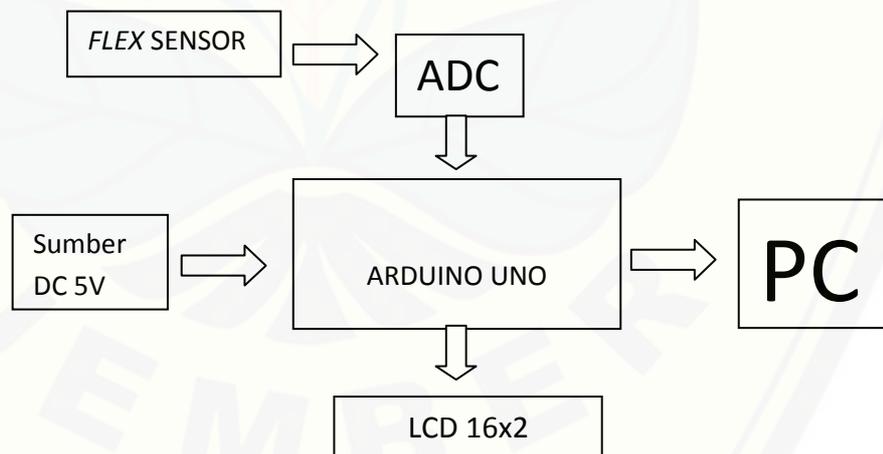
3.7 Pengukuran secara digital

1. Pasien menggigit sensor flex yang telah di sediakan selama 3 detik.
2. Sensor tegak lurus dengan dagu.
3. Lalu tekan tombol *Start* untuk pengambilan data.
4. Setelah itu lakukan sebanyak tiga kali untuk pengambilan data.
5. Hasil dari pengambilan data tiga kali itu di rata-rata untuk menentukan hasil yang sebenarnya.

3.7.1 Cara pengukuran digital untuk *minus overjet* dan *open bite*

1. Pasien menggigit sensor flex yang telah di sediakan selama 3 detik.
2. Sensor tegak lurus dengan dagu.
3. Lalu tekan tombol *minus overjet* untuk pasien *minus overjet* dan tekan *open bite* untuk pasien *open bite*.
4. Setelah itu lakukan sebanyak tiga kali untuk pengambilan data.
5. Hasil dari pengambilan data tiga kali itu di rata-rata untuk menentukan hasil yang sebenarnya.

3.8 Blok Diagram



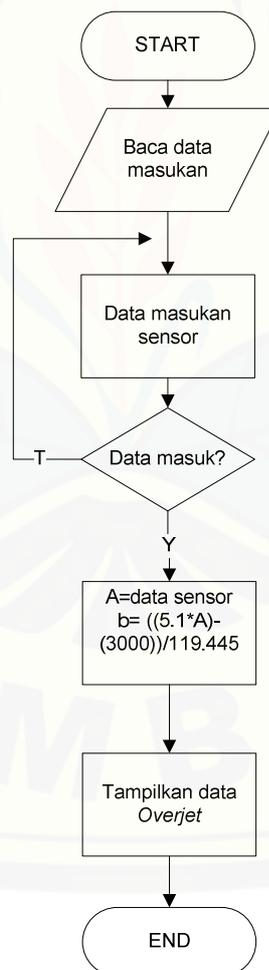
Gambar 3.1 Blok Diagram

Prinsip kerja alat ini data masukan berupa nilai ADC dari data sensor yang lalu data dikirim ke mikrokontroler untuk di proses data tersebut. data tersebut dalam bentuk sinyal digital di kirim ke *serial* untuk di proses ke PC.

Sensor flex disini digunakan untuk mengukur *overjet dan overbite* pada gigi *insisivus*. Di karena sifat sensor flex cocok untuk di gunakan mengukur sudut kemiringan pada gigi dan sifatnya yang fleksibel.

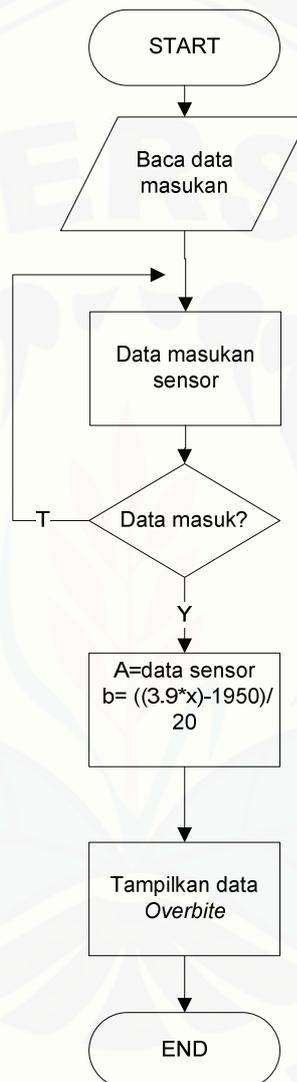
3.9 Flowchart Program

Pada *flowchart* gambar 3.2 adalah *flowchart overjet* Proses pengambilan data dari Flex sensor, variabel untuk data ADC sensor menggunakan huruf A. Setelah penginisialan data sensor lalu proses kalibrasi menggunakan perhitungan garis lurus untuk linier dari hasil pembacaan sensor dan di ubah menjadi angka yang kita inginkan.



Gambar 3.2 Flowchart Overjet

Pada *flowchart* gambar 3.3 adalah *flowchart overbite* Proses pengambilan data dari Flex sensor, variabel untuk data ADC sensor menggunakan huruf A. Setelah penginisialan data sensor lalu proses kalibrasi menggunakan perhitungan garis lurus untuk linierkan dari hasil pembacaan sensor dan di konvert menjadi angka yang kita inginkan.

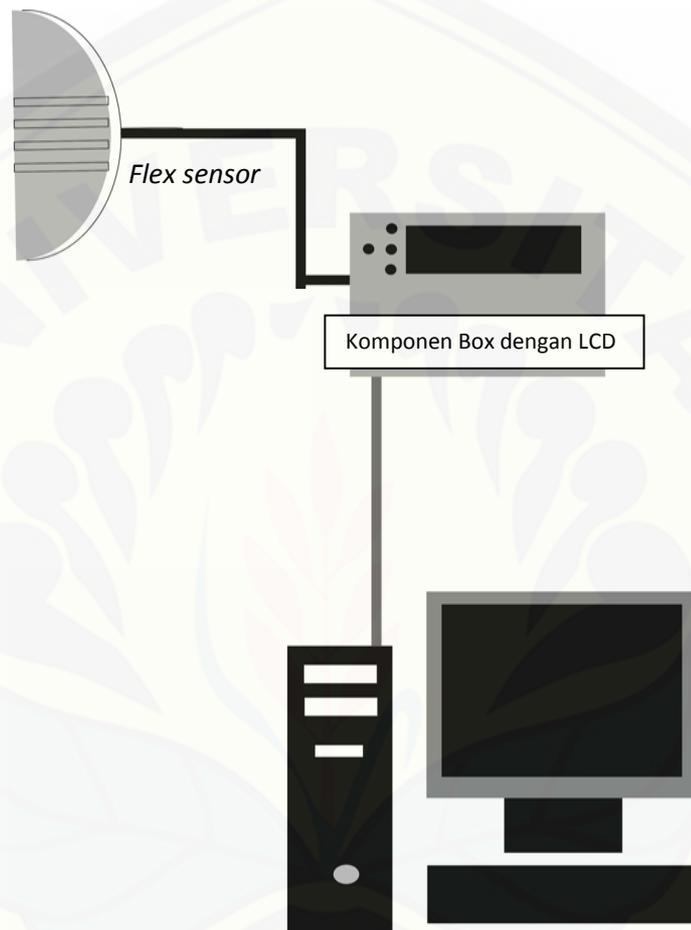


Gambar 3.3 *Flowchart overbite*

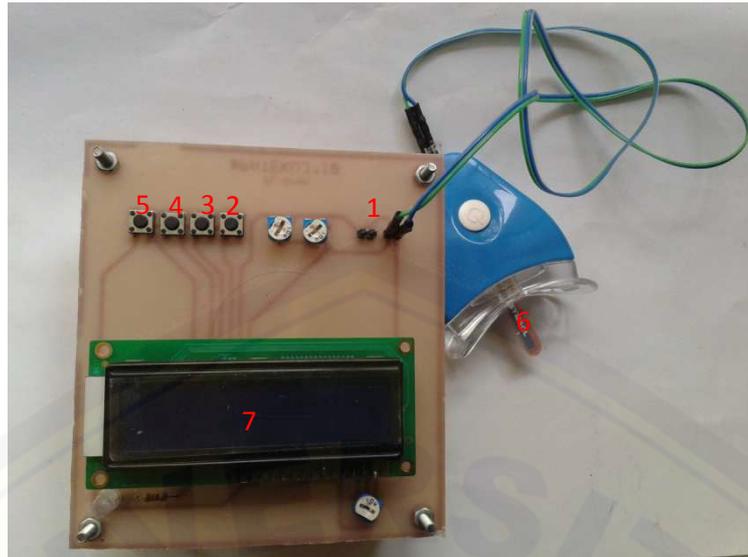
3.10 Hasil perancangan alat

Pada gambar 3.4 adalah gambar rancangan awal dimana flex sensor untuk di gigit setelah data di olah di dalam komponen *box*. Di dalam komponen *box* terdapat arduino uno dan LCD untuk menampilkan hasil pembacaan sensor dan

push button sebagai tombol untuk *Start* dan *stop*. Setelah diproses dari arduino lalu di kirim ke PC/laptop untuk ditampilkan di layar dan juga bisa untuk menyimpan data yang di hasilkan.



Gambar 3.4 perancangan Alat

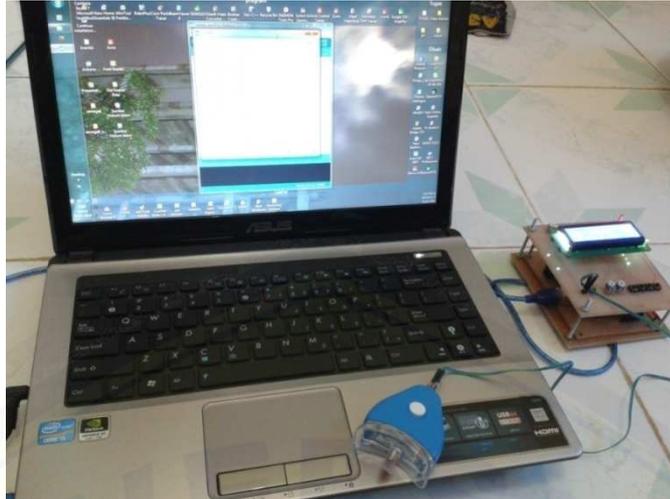


Gambar 3.5 Hasil pembuatan alat

Pada gambar 3.5 adalah bentuk hasil pembuatan alat. Sebagai berikut ini adalah fungsi dari masing masing kegunaan pada perangkat keras:

1. Variabel resistor sebagai kalibrasi sensor ke titik Nol.
2. Tombol untuk mulai pengambilan data.
3. Tombol untuk berhenti pengambilan data.
4. Tombol untuk *overjet* minus.
5. Tombol untuk *openbite*.
6. Sensor Flex untuk di gigit.
7. LCD untuk tampilan data.

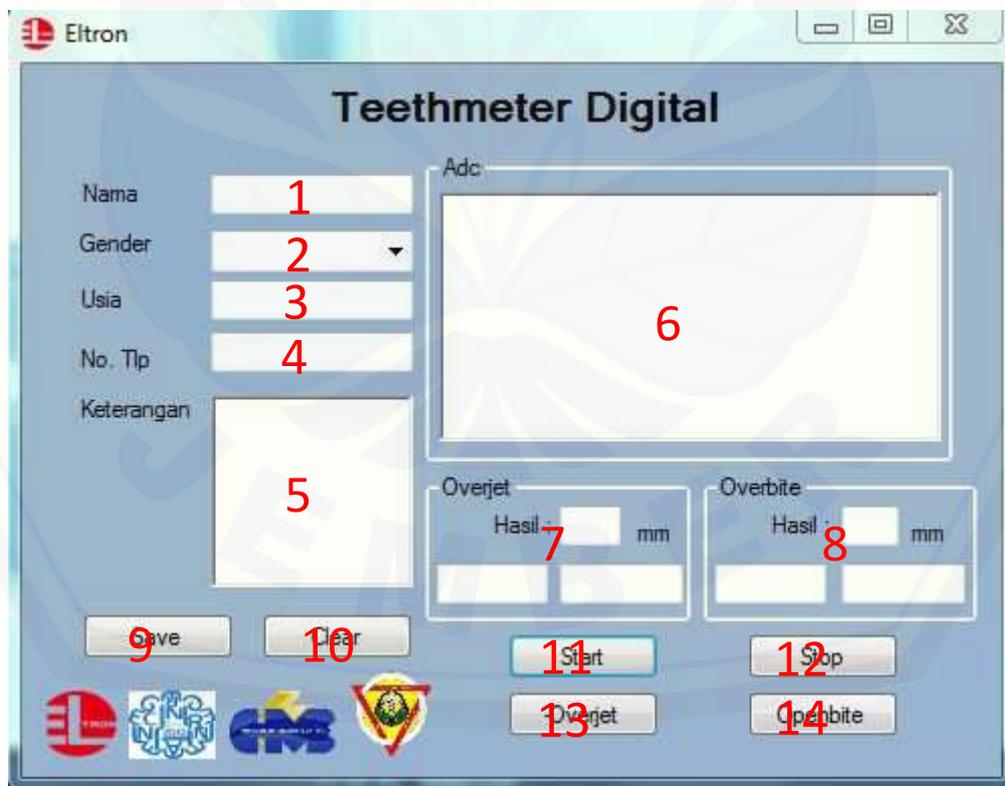
Pada gambar 3.5 adalah hasil dari pembuatan alat nampak atas kelihatan LCD, VR, tombol dan sensor flex. Masing-masing komponen berbeda fungsinya, LCD disini sebagai penampil nilai dari sensor. Variabel resistor berfungsi untuk mengatur kalibrasi ulang setelah pemakaian untuk mengembalikan ke nilai nol lagi. Tombol disini berfungsi untuk memulai pengambilan data dan memberhentikan pengambilan data.



Gambar 3.6 Hasil pembuatan alat dan *Interface* laptop

3.11 Aplikasi GUI

Aplikasi GUI yang telah dibuat menggunakan Microsoft visual studio 2010 ultimate untuk memantau nilai-nilai setiap sensor. Menggunakan komunikasi serial sebagai komunikasi antara Laptop dengan arduino.



Gambar 3.7 Tampilan GUI (*Grafik User Interface*)

Graphic user Interface (GUI) ini dilengkapi dengan pengaktifan penyimpanan data. Penjelasan bagian GUI pada gambar 3.7 sebagai berikut :

1. Tampilan untuk memasukan nama.
2. Tampilan untuk jenis kelamin.
3. Tampilan untuk memasukan usia.
4. Tampilan untuk memasukan nomer telepon.
5. Tampilan untuk keterangan.
6. Tampilan data ADC dari *flex* sensor.
7. Tampilan hasil nilai kalibrasi *Overjet*.
8. Tampilan hasil nilai kalibrasi *Overjet*.
9. Tombol penyimpan ke Microsoft Excel.
10. Tombol untuk membersihkan data yang telah di ketik.
11. Tombol untuk memulai pengambilan data.
12. Tombol berhenti.
13. Tombol untuk pengaturan minus *overjet*.
14. Tombol untuk pengaturan *openbite*.

Berdasarkan hasil pengujian dapat dikatakan aplikasi GUI itu berjalan dengan baik dalam menerima data dengan menggunakan komunikasi serial.

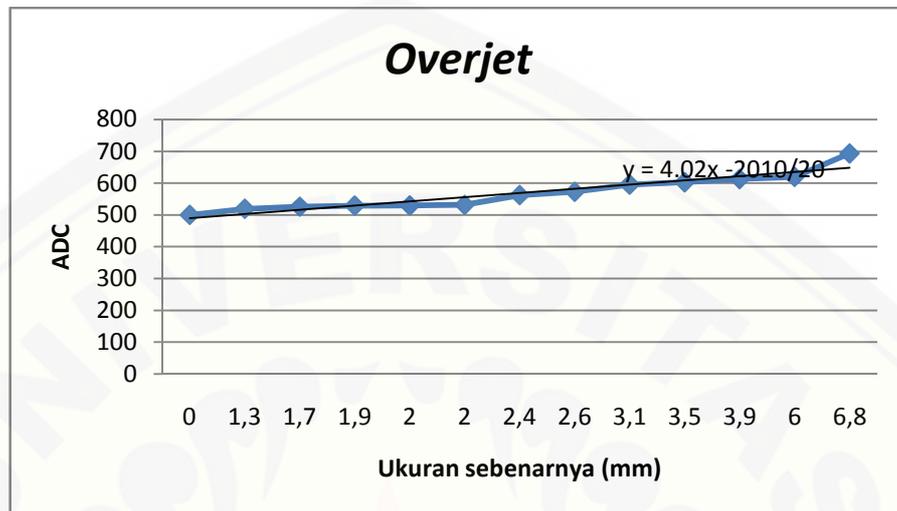
3.12 Proses Kalibrasi

Proses kalibrasi ini ada beberapa tahap untuk mendapatkan hasil nilai yang sesungguhnya dari nilai ADC. Pengolahan data yang pertama yaitu pengambilan 10 sampel yang terdiri dari beberapa ukuran gigi yang berbeda. Data awal kalibrasi sebagai berikut ini:

Tabel 3.1 Sampel data

| No | Nama | <i>Overjet</i> | <i>overbite</i> | ADC |
|----|---------|----------------|-----------------|---------|
| 1 | Ferdika | 3,9 | 3 | 612,697 |
| 2 | Kepler | 1,9 | 2 | 531,588 |
| 3 | Angga | 1,3 | 1.6 | 519,385 |
| 4 | Putri | 2,4 | 2,2 | 595,403 |
| 5 | Chechen | 3,1 | 2,26 | 572,666 |
| 6 | Amiin | 6,8 | 5,1 | 692,991 |
| 7 | Cemet | 6 | 4,2 | 619,445 |
| 8 | Victor | 2 | 2,5 | 529,048 |
| 9 | Robin | 1,7 | 0,84 | 525,756 |
| 10 | Ulfa | 2,6 | 2,2 | 562,24 |

Setelah mendapatkan nilai ADC dari setiap sampel gigi, lalu mencetak gigi dari sampel tersebut untuk di lakukan pengukuran dengan penggaris agar dapat mengetahui ukuran *overjet* dan *overbite* sebenarnya. Untuk linier nilai keluaran dari sensor ini peneliti menggunakan Excel untuk mempermudah pembuatan grafik linier sebagai berikut ini adalah gambar grafik yang di buat:



Gambar 3.8 Grafik hubungan antara pengukuran *overjet* terhadap nilai ADC

Gambar 3.8 adalah gambar garfik hubungan antara nilai *overjet* dengan nilai ADC, lalu klik kanan garis grafik dan memilih *trandline* setelah itu pilih linier dan akan keluar garis lurus dan keluar rumus garis lurus untuk kalibrasi. Dan dapatlah rumus seperti berikut:

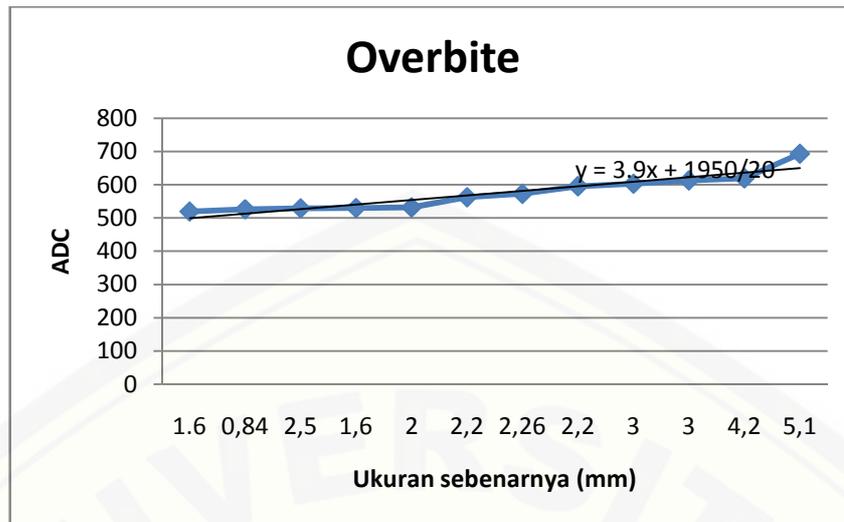
$$y = \frac{(4.02 * x) - 2010}{20} \quad (3.1)$$

dimana x adalah nilai ADC dan y adalah nilai *overjet* yang dicari. Maka dari rumus tersebut di masukan ke dalam *listing* program.

Sama halnya seperti *overjet* Gambar 3.8 adalah gambar garfik hubungan antara nilai *overjet* dengan nilai ADC, lalu klik kanan garis grafik dan memilih *trandline* setelah itu pilih linier dan akan keluar garis lurus untuk kalibrasi. Dan dapatlah rumus seperti berikut:

$$y = \frac{(3.9 * x) - 1950}{20} \quad (3.2)$$

dimana x adalah nilai ADC dan y adalah nilai *overjet* yang di cari. Maka dari rumus tersebut di masukan ke dalam *listing* program.



Gambar 3.9 Grafik hubungan antara pengukuran *overjet* terhadap nilai ADC

Untuk proses kalibrasi minus *overjet* peneliti menggunakan rumus garis lurus untuk nilai sebenarnya. Pada sampel yang di gunakan ini berbeda dari populasi yang di rencanakan di karenakan pada populasi tersebut tidak ada yang minus *overjet*. Agar kalibrasi alat sempurna maka peneliti mencari sampel data dari luar populasi. Sampel yang di dapat dari pengukuran manual *overjet* adalah -5,1mm dan nilai *overbite* adalah 5mm dan nilai untuk nilai ADC 486,7. Dapat dilihat nilai dari pengukuran manual dengan penggaris nilai *overjet* -5,1mm, apabila menggunakan rumus kalibrasi seperti (3.1) maka hasilnya berbeda oleh karena peneliti menggunakan rumus persamaan garis sebagai berikut ini:

$$\frac{y-y_1}{y_2-y_1} = \frac{x-x_1}{x_2-x_1} \quad (3.3)$$

y = nilai yang di cari

y_1 = nilai dalam kondisi awal

y_2 = nilai dalam kondisi akhir

x = masukan dari pembacaan sensor

x_1 = nilai ADC dalam kondisi awal

x_2 = nilai ADC dalam kondisi akhir

Dapat di lihat dari penyerderhanaan rumus 3.3 menjadi sederhana ketika dimasukan nilai *overjet* yang di dapat rumus menjadi sederhana seperti rumus 3.4. Dari rumus yang di dapat maka akan disederhanakan menjadi seperti ini:

$$\begin{aligned}\frac{y-y_1}{y_2-y_1} &= \frac{x-x_1}{x_2-x_1} \\ \frac{y-0}{-5,1-0} &= \frac{x-500}{486,7-500} \\ y &= \frac{5,2x-2550}{-13,3}\end{aligned}\quad (3.4)$$

Pada kalibrasi *overbite* dengan nilai minus maka menggunakan rumus garis lurus tapi dalam kondisi awal tidak nol. Untuk sampel *openbite* peneliti mencari sampel dari luar populasi dikarenakan dalam populasi tidak ada sampel dalam katagori tersebut. Nilai yang di dapat dari pengukuran manual dengan penggaris nilai *overjet* 3,9mm dan nilai *overbite* 2,3mm untuk nilai ADC yang di dapat 562,2. Maka penyederhanaan dari rumus 3.3 mencari rumus untuk *overbite* sebagai berikut:

$$\begin{aligned}\frac{y-(-4)}{-2,3-(-4)} &= \frac{x-500}{562,2-500} \\ y &= \frac{1,7x+850}{62,2} -4\end{aligned}\quad (3.5)$$

Untuk mendapatkan rumus kalibrasi *overjet* sebagai berikut:

$$\begin{aligned}\frac{y-4}{3,9-4} &= \frac{x-500}{562,2-500} \\ y &= \frac{-0,1x+50}{62,2} + 4\end{aligned}\quad (3.6)$$

Dari rumus tersebut akan di masukan dalam pemrograman kalibrasi sensor untuk mendapatkan nilai yang di inginkan. Dari rumus diatas peneliti menggunakan kondisi awal posisi sensor sudah mengukur 4 mm di karenakan gigi *anterior* sudah terbuka. Apabila saya menggunakan titik awal 0 mm maka hasil tidak sesuai apa yang di inginkan.

BAB 4. HASIL DAN PEMBAHASAN

4.1 Pengujian Sensor

Sensor yang diuji adalah sensor flex, sensor flex diuji dengan penggaris busur dan *multimeter* untuk menentukan perubahan resistansi ketika ditekuk sesuai derajat penggaris busur. Maka hasil yang didapatkan sebagai berikut.

Tabel 4.1 sudut dan resistansi sensor

| No | Sudut(°) | Resistansi (KΩ) |
|----|----------|-----------------|
| 1 | 0 | 33 |
| 2 | 20 | 35 |
| 3 | 40 | 37 |
| 4 | 60 | 42 |
| 5 | 80 | 44 |
| 6 | 90 | 46 |

Dapat dilihat dari hasil pengukuran resistansi dengan *multimeter* posisi awal sensor flex resistansinya 33 KΩ. Perubahan ketika ditekuk 20 derajat maka resistansinya menjadi 35 KΩ selisih perubahan sudut dari 0 ke 20 derajat hanya 2 saja. Ketika ditukuk lagi menjadi 40 derajat besar resistansi sensor menjadi 37 KΩ. Lalu sensor ditekuk sampai 60, 80 dan 90 derajat perubahan resistansinya semakin kecil hanya selisih 2 KΩ. Dari data yang diambil bisa dikatakan bahwa sensor ini masih dalam kondisi baik dan dapat digunakan.

4.2 Pengujian Prengkat Lunak

Pengujian aplikasi dilakukan untuk mengetahui apa aplikasi yang dibuat dengan Visual Studio C# sudah berjalan sesuai yang diharapkan atau belum. Pengujian pertama dilakukan adalah dengan cara melakukan *start debugging* untuk mengetahui apakah *listing* program yang ditulis sudah benar atau tidak. Lalu digunakan untuk memantau apakah terjadi *error* jika program dijalankan. Jika terdapat kesalahan pada program maka program tidak akan dapat dijalankan dan aplikasi yang ditulis sudah betul maka secara otomatis aplikasi akan langsung

terbuka dengan sendirinya dan dapat digunakan sesuai dengan fungsi-fungsi perintah yang ditulis dalam program.



Gambar 4.1 Tampilan Aplikasi yang sudah jalan

Untuk menampilkan data yang ada diperangkat keras dikirim melalui serial. Aplikasi akan menerima data dengan listing program sebagai berikut.

```

button1.Enabled = false;
button2.Enabled = true;
arduino.PortName = "COM87";
arduino.BaudRate = 9600;
arduino.Parity = Parity.None;
arduino.DataBits = 8;
arduino.StopBits = StopBits.One;
arduino.Open();

```

Listing program diatas untuk koneksikan data yang ada di perangkat lunak dan perangkat keras. Perangkat ini menggunakan nama port COM87 dengan *baudrate* 9600, perangkat ini menggunakan *baudrate* 9600 karena pada pengiriman arduino menggunakan *baudrate* yang sama. *Databits* arduinonya ada 8. Variabel arduino tersebut adalah nama *tool* serial yang sudah diganti nama.

Setelah itu coba menekan fungsi tombol yang ada di perangkat keras tersebut apakah berfungsi atau tidak. Pertama tekan tombol *start* untuk mengaktifkan penerima data yang dikirim dari perangkat lunak.



Gambar 4.2 Tampilan ketika tombol *Start* ditekan.

Dapat dilihat pada gambar 4.2 tampilan diatas setelah di tekan tombol *start* maka data yang dikirim dari perangkat keras ditampilkan. Berikut ini adalah listing program yang ada pada tombol *Start*.

```

for (int g = 1; g <= 3; g++)
{
    kode = arduino.ReadLine();
    float b = float.Parse(kode);
    float bb = b / 100;
    double j = ((4.02 * bb) - 2010);
    double data = j / 20;
    float adc = b / 100;
    double bagi = ((3.9 * adc) - 1950);
    double hasil = bagi / 20;
    c = Convert.ToString(data);
    decimal aa = Convert.ToDecimal(c);
    aa = Math.Round(aa, 1);
    richTextBox1.AppendText(kode);
    textBox1.Text = Convert.ToString(aa);
}

```

```

double aaa = Convert.ToDouble(aa);
segmentGauge1.Value = aaa;
d = Convert.ToString(hasil);
decimal dec = Convert.ToDecimal(d);
dec = Math.Round(dec, 1);
textBox5.Text = Convert.ToString(dec);
double segmen = Convert.ToDouble(dec);
segmentGauge2.Value = segmen;

```

Dapat dilihat pada *listing* program diatas menggunakan *for* untuk *looping* sebanyak 3 kali. Di dalam *listing* program diatas ada perhitungan matematis berfungsi untuk mengubah data ADC menjadi data yang diinginkan. Data yang di kirim dari perangkat keras adalah data dalam bentuk *String*, oleh karena itu sebelum di eksekusi program berikutnya harus di *convert* dari *string* ke *float* agar bisa menggunakan operator matematik. Setelah itu untuk menampilkan ke *textbox* data yang berbentuk *float* di *convert* ke *string* agar dapat muncul di *textbox*.

Langkah berikutnya yaitu dengan menekan tombol *-overjet* tombol tersebut memiliki fungsi yang sama seperti tombol *start* bedanya hanya pada perhitungan matematis yang ada pada *listing* program. Berikut ini adalah *listing* program dari *-overjet*.

```

kode = arduino.ReadLine();
float b = float.Parse(kode);
float bb = b / 100;
double j = ((-5.1 * bb) + 2550);
double data = j / -13.2;
//bite
float adc = b / 100;
double bagi = ((5 * adc) - 2500);
double hasil = bagi / -13.3;

```

Listing program tersebut hampir sama seperti tombol *start*, bedanya terletak pada perhitungannya agar mendapatkan nilai *minus* maka yang perhitungan *overjet* dikalikan dengan nilai *minus* juga. Pada gambar 4.3 tampilan setelah di isi sesuai perintah. Dari gambar diatas *textbox* dapat di isi. Setelah itu coba klik tombol *save* untuk mengetahui fungsi tombol tersebut.

Nama: tyok
 Gender: Laki Laki
 Usia: 23
 No. Tlp: 085648880700
 Keterangan: Coba save

Gambar 4.3 Tampilan *textbox* di isiGambar 4.4 Tampilan excel setelah ditekan *save*

Pada gambar 4.4 adalah tampilan setelah ditekan tombol *save* dan muncul di *window* Microsoft Office Excel. Semua yang diisi di *textbox* tampil di kolom dan baris Microsoft Office Excel. Berikut ini adalah *listing* program untuk *save* data di Microsoft Office Excel.

```

Microsoft.Office.Interop.Excel.Application xls = new
Microsoft.Office.Interop.Excel.Application();
Workbook wb = xls.Workbooks.Add(XlSheetType.xlWorksheet);
Worksheet wa = (Worksheet)xls.ActiveSheet;

xls.Visible = true;

wa.Cells[1, 1] = "Nama";
wa.Cells[1, 2] = "Gender";
wa.Cells[1, 3] = "Usia";
wa.Cells[1, 4] = "No Telpn";
wa.Cells[1, 5] = "overjet";
wa.Cells[1, 6] = "overbite";
wa.Cells[1, 9] = "keterangan";
wa.Cells[1, 7] = "Kondisi overjet";
wa.Cells[1, 8] = "Kondisi overbite";

wa.Cells[2, 1] = nama.Text;
wa.Cells[2, 2] = comboBox1.Text;
wa.Cells[2, 3] = usia.Text;
wa.Cells[2, 4] = tlp.Text;
wa.Cells[2, 5] = textBox5.Text;
wa.Cells[2, 6] = textBox1.Text;
wa.Cells[2, 9] = keterangan.Text;
wa.Cells[2, 7] = textBox2.Text;
wa.Cells[2, 8] = textBox3.Text;

```

Dapat dilihat pada *listing* program diatas ada pemanggilan data *library* ke Microsoft. Microsoft.Office.Interop.Excel.Application xls = new fungsinya

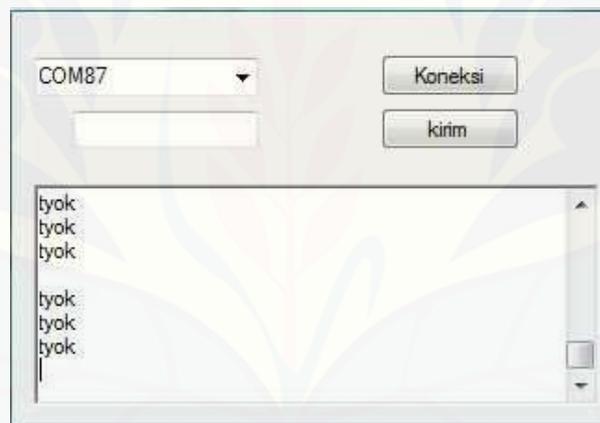
adalah untuk memanggil Microsoft Office Excel, setelah itu membuat lembaran baru program `wa.Cells[1, 1] = "Nama"`; berfungsi untuk mengatur baris dan kolom pada Microsoft Office Excel koordinatnya pada program seperti ini [baris, kolom].

Dalam pengujian perangkat lunak bisa dikatakan bahwa perangkat lunak berjalan sesuai apa yang peneliti inginkan.

4.3 Pengujian Perangkat Keras

4.3.1 Pengujian pengiriman data

Pengujian pengiriman data dari arduino ke laptop dengan komunikasi serial. Untuk mengirim data dari arduino menggunakan *listing program* `Serial.println("tyok")`; yang berada di dalam kurung adalah data yang dikirimkan dari arduino. Setelah tampilan dari data yang dikirimkan seperti gambar berikut ini:

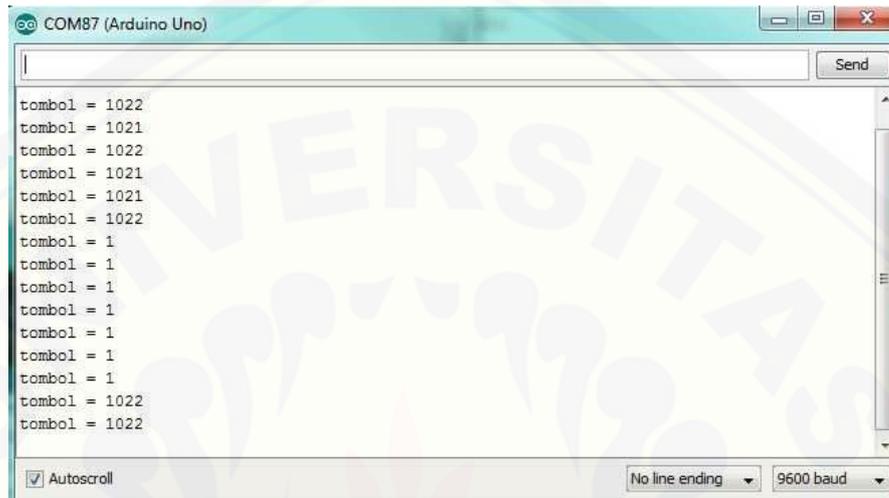


Gambar 4.5 tampilan kiriman dari arduino di Laptop

Dapat dilihat pada gambar 4.5 pengiriman data yang dikirim berupa karakter kata "tyok". Dalam pengiriman data terdapat kehilangan data yang dikirim setiap 3 kali pengiriman data pasti data yang sesudahnya hilang. Oleh karena diambil hanya 3 data ADC untuk diproses dalam alat ini.

4.3.2 Pengujian tombol

Dalam pengujian tombol ini *pin* yang digunakan adalah *pin* analog. rangkaian tombol menggunakan aktif *LOW*, Jadi ketika tombol ditekan mikrokontroler bekerja ketika *pin* itu mendapat logika 0. untuk membuktikanya maka dibuat pembacaan tombol dengan nilai ADC yang ditampilkan di serial. Gambar tampilan sebagai berikut:

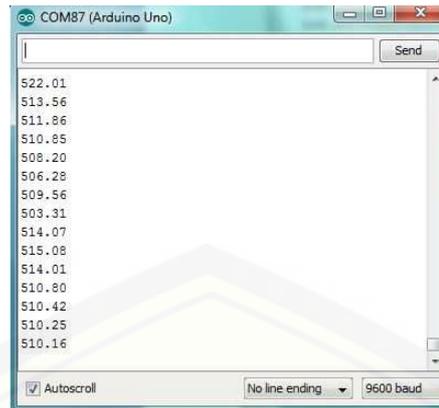


Gambar 4.6 Tampilan ketika di tekan tombol

Dari gambar 4.6 pembacaan ADC dengan 10 bit data jadi jumlah maksimal pembacaan ADC 1022 . Ketika tombol tidak ditekan pembacaan nilai ADC yang 1022 lalu ketika ditekan maka akan berubah nilainya menjadi 1. Dari pengujian tersebut maka tombol bisa digunakan.

4.3.3 Pengujian masukan sensor

Pada pengujian ini menggunakan masukan *pin* A0 dan menggunakan sensor flex dengan rangkaian pembagi tegangan untuk mengeluarkan tegangan analog yang diubah menjadi digital. Data dari sensor ditampilkan di serial monitor arduino. Maka akan muncul sebagai berikut ini.



Gambar 4.7 tampilan masukan sensor

Dari gambar 4.7 dapat dilihat bahwa masukan sensor sesuai dengan apa yang diinginkan peneliti. Nilai dari sensor sudah dalam bentuk ADC yang sudah di kelolah oleh arduino.

4.4 Pengujian *Overjet*

Pada pengujian *overjet* dengan sampel yang telah ditentukan yaitu mahasiswa Diploma III teknik elektro angkatan 2012 Universitas Jember. Untuk mendapatkan hasil perbandingan yang akurat maka dilakukan pembuatan model gigi pada sampel untuk memudahkan pengukuran dengan penggaris. Setelah itu sampel diukur digital dengan cara mengigit sensor flex dengan tiga kali gigitan hasil dari gigitan tersebut diambil rata-rata. Dapat dilihat pada Tabel 4.2 ini adalah hasil dari pengambilan data sampel secara manual dan digital.

Pada Tabel 4.2 tersebut terlihat perbandingan antara pengukuran manual dengan penggaris dan pengukuran digital menggunakan sensor flex. Disini pengukuran digital mengacu pada pengukuran manual. Pengukuran digital ini hanya mengukur relasi antara gigi 11 dengan 41 dan 21 dengan 31. Pada data sampel yang diambil hanya satu relasi gigi yaitu gigi 11 dengan 41 dan ada juga yang gigi 21 dengan 31 dikarenakan gigi 11 mengalami patah pada giginya. Sampel pertama bernama Munir pengukuran manual diperoleh 4.02 mm dengan pengukuran digital di peroleh 4.2 mm. *Error* persen yang 4.4% diperoleh dari perhitungan dengan rumus sebagai berikut:

$$Error = \frac{hasil\ manual - hasil\ digital}{hasil\ manual} \times 100\% \quad (4.1)$$

Tabel 4.2 Pengukuran *overjet* manual dan digital

| No | Nama | Pengukuran <i>Overjet</i> (mm) | | <i>Error</i> (%) | Kriteria |
|----|---------|-----------------------------------|---------|------------------|-----------|
| | | Manual | Digital | | |
| 1 | Munir | 4.02 | 4.2 | 4.4 | Bertambah |
| 2 | Ferdika | 3.9 | 3.72 | 4.6 | Bertambah |
| 3 | Kepler | 1.9 | 2.12 | 2.2 | Berkurang |
| 4 | Angga | 1.3 | 1.82 | 8 | Berkurang |
| 5 | Putri | 2.4 | 2.5 | 4.1 | Bertambah |
| 6 | Chechen | 3.1 | 3.3 | 6.4 | Normal |
| 7 | Amiin | 6.8 | 6.5 | 4.4 | Bertambah |
| 8 | Cemet | 6 | 6.3 | 5 | Bertambah |
| 9 | Victor | 2 | 2.12 | 6 | Normal |
| 10 | Robin | 1.7 | 1.8 | 5 | Berkurang |
| 11 | Ulfa | 2.6 | 2.6 | 0 | Normal |
| 12 | Jhon | 2 | 2.21 | 5 | Normal |
| 13 | Ival | -5.1 | -5,5 | 7 | Minus |
| 14 | Anam | 2 | 2.0 | 0 | Normal |
| 15 | Kusuma | 1.3 | 1.4 | 3.3 | Berkurang |
| 16 | Zainul | 3.6 | 3.9 | 8.3 | Normal |
| 17 | Roin | 1 | 6.02 | 490 | Berkurang |
| 18 | Fandi | 3.9 | 4 | 2.5 | Normal |
| 19 | Rizky | 3.9 | 3.3 | 15.3 | Normal |

Sampel berikutnya yang diambil adalah Ferdika dengan pengukuran manual 3.9 mm lalu di ukur dengan pengukuran digital 3.72 mm didapat *error*(%) 4.6%. Dari pengambilan data pada sampel yang bernama Roin terdapat *erro*(r%) hingga melebihi 100% dikarenakan kelemahan alat ini ketika nilai jarak *overbite* dan *overjet* selisih terlalu jauh, akibatnya hasil pengukuran nilai terkecil menjadi besar *error*(%). Karena pada kalibrasi mengambil nilai garis lurus yang mendekati data pengukuran manual dengan penggaris. Nilai masukan ADC dari sensor sama antara *overbite* dan *overjet*, jadi nilai yang dihasilkan selisih jauh antara *overbite* dan *overjet*. Sebagai contohnya nilai pengukuran manual *overbite* 5,8 mm dan *overjet* 1 mm dengan nilai ADC dari alat 553. Maka dimasukkan rumus dari kalibrasinya.

Untuk rumus *overjet* :

$$y = \frac{(4.02 * x) - 2010}{20}$$

$$y = \frac{(4.02 * 530) - 2010}{20}$$

$$y = 6.02$$

untuk rumus *Overbite*:

$$y = \frac{(3.9 * x) - 1950}{20}$$

$$y = \frac{(3.9 * 530) - 1950}{20}$$

$$y = 5.85$$

Dapat dilihat dari hasil yang didapat dari penghitungan dengan rumus kalibrasi dengan ADC yang sama selisih dari kedua perhitungan dekat. Dari salah satu perhitungan ada yang mendekati nilai *overbite* mendekati nilai pengukuran manual dan nilai *overjet* sangat jauh.

4.5 Pengujian *Overbite*

Pada pengujian *overjet* dengan sampel yang telah ditentukan yaitu mahasiswa Diploma III Teknik Elektro angkatan 2012 Universitas Jember. Untuk mendapatkan hasil perbandingan yang akurat maka dilakukan pembuatan model gigi pada sampel untuk memudahkan pengukuran dengan penggaris. Setelah itu sampel diukur digital dengan cara mengigit sensor flex dengan tiga kali gigitan hasil dari gigitan tersebut diambil rata-rata. Tabel 4.3 adalah hasil dari pengambilan data sampel secara manual dan digital.

Dapat dilihat dari tabel 4.3 perbandingan antara pengukuran *overbite* secara manual dan digital. Hasil dari pengukuran digital mengacu pada pengukuran manual dengan penggaris. Pengukuran digital ini hanya mengukur relasi antara gigi 11 dengan 41 dan 21 dengan 31. Pada data sampel yang diambil hanya satu relasi gigi yaitu gigi 11 dengan 41 dan ada juga yang diambil sampel relasi gigi 21 dengan 31 dikarenakan gigi 11 mengalami patah pada giginya.

Dari data sampel yang bernama Munir, pengukuran manual dengan penggaris 3.9 mm lalu diukur dengan menggunakan sensor flex 4 mm jadi *error*

2.5%. pengukuran selanjutnya adalah Ferdika dengan pengukuran manual 3 mm dan pengukuran digital dengan hasil 2,8 mm dengan *error* 6%. Dari data yang diambil ada *error* yang paling tinggi yaitu 19%. Dalam kalibrasi diambil nilai garis lurus yang mendekati data pengukuran manual dengan penggaris. Dan nilai ADC sama antara *overbite* dan nilai *overjet*, dari kondisi ini nilai selisih jauh antara *overbite* dan *overjet*.

Tabel 4.3 Pengukuran *overbite* manual dan digital

| No | Nama | Pengukuran Overbite (mm) | | <i>Error</i> % | Kriteria |
|----|---------|-----------------------------|---------|----------------|-----------|
| | | Manual | Digital | | |
| 1 | Munir | 3.9 | 4 | 2.5 | Normal |
| 2 | Ferdika | 3 | 2.8 | 6 | Normal |
| 3 | Kepler | 2 | 2.1 | 5 | Normal |
| 4 | Angga | 1.6 | 1.7 | 5 | Berkurang |
| 5 | Putri | 2.2 | 2.32 | 5.1 | Normal |
| 6 | Chechen | 2.26 | 2.38 | 5.3 | Normal |
| 7 | Amiin | 5.1 | 4.98 | 4.4 | Bertambah |
| 8 | Cemet | 4.2 | 5 | 19 | Bertambah |
| 9 | Victor | 2.5 | 2.62 | 4 | Normal |
| 10 | Robin | 0.84 | 1 | 1.9 | Berkurang |
| 11 | Ulfa | 2.2 | 2.2 | 0 | Normal |
| 12 | Jhon | 1.6 | 1.72 | 5 | Berkurang |
| 13 | Ival | 5 | 5.7 | 14 | Bertambah |
| 14 | Anam | 1.6 | 1.9 | 18 | Berkurang |
| 15 | Kusuma | 1.3 | 1.4 | 3.3 | Berkurang |
| 16 | Zainul | 3.8 | 3.9 | 2.9 | Normal |
| 17 | Roin | 6 | 5.8 | 3.3 | Bertambah |
| 18 | Fandi | 3.5 | 3.8 | 8 | Normal |
| 19 | Rizky | -2.3 | -1.8 | 19 | Minus |

Pada pengambilan data ini pengaruh dari tekanan gigit dan sudut sensor sangat berpengaruh. Ketika pengambilan data sensor goyang sedikit maka hasil yang didapat akan berbeda dari pengambilan awal oleh karena itu pengambilan data harus dilakukan 3x lalu di rata-rata.

4.6 Analisis Sistem Keseluruhan

Pada subbab terakhir ini menganalisis sistem keseluruhan dari pengujian yang dilakukan pada alat yang telah dibuat dan data pengukuran yang telah dilakukan. Pengukuran *overjet* dan *overbite* saling terkait dua pengukuran digital tersebut, dalam kedua pengukuran tersebut dikarenakan menggunakan satu sensor untuk mendapatkan dua nilai pengukuran. Dapat dilihat pada tabel 4.4 ini:

Tabel 4.4 Data keseluruhan

| No | Nama | Pengukuran Overjet | | Pengukuran Overbite | |
|----|---------|--------------------|---------|---------------------|---------|
| | | Manual | Digital | Manual | Digital |
| 1 | Ferdika | 4.02 | 4.2 | 3.9 | 4 |
| 2 | Munir | 3.9 | 3.72 | 3 | 2.8 |
| 3 | Kepler | 1.9 | 2.12 | 2 | 2.1 |
| 4 | Angga | 1.3 | 1.82 | 1.6 | 1.7 |
| 5 | Putri | 2.4 | 2.5 | 2.2 | 2.32 |
| 6 | Chechen | 3.1 | 3.3 | 2.26 | 2.38 |
| 7 | Amiin | 6.8 | 6.5 | 5.1 | 4.98 |
| 8 | Cemet | 6 | 6.3 | 4.2 | 5 |
| 9 | Victor | 2 | 2.12 | 2.5 | 2.62 |
| 10 | Robin | 1.7 | 1.8 | 0.84 | 1 |
| 11 | Ulfa | 2.6 | 2.6 | 2.2 | 2.2 |
| 12 | Jhon | 2 | 2.21 | 1.6 | 1.72 |
| 13 | Ival | -5.1 | -5,5 | 5 | 5.7 |
| 14 | Anam | 2 | 2.0 | 1.6 | 1.9 |
| 15 | Kusuma | 1.3 | 1.4 | 1.3 | 1.4 |
| 16 | Zainul | 3.6 | 3.9 | 3.8 | 3.9 |
| 17 | Roin | 1 | 5.9 | 6 | 5.8 |
| 18 | Fandi | 3.9 | 4 | 3.5 | 3.8 |

Dapat dilihat pada tabel 4.4 perbandingan dari kedua data pengukuran digital selisi pengukurannya berdekatan antara *overbite* dan *overjet*. Ada beberapa jaraknya yang sama disebabkan karena pengiriman data dari arduino menuju PC mengalami keterlambatan. Pembacaan program terlambat sebagian sudah membaca nilai ADC yang baru dan program setelahnya membaca nilai ADC yang lama. Dengan menggunakan persamaan garis harusnya hanya beberapa titik saja yang mempunyai nilai yang sama.

Dari pengujian sensor apabila sensor sering ditekuk atau digigit mempengaruhi resistansinya. Sensor sering digigit atau ditekuk sensor mengalami penambahan jumlah resistansi pada kondisi awal. Pengaruh perbedaan

pengukuran dikarenakan letak gigitan berbeda dan bisa disebabkan karena pada saat pengukuran sensor goyang tidak tegak lurus dengan mulut.

Kelebihan alat ukur digital ini hanya memerlukan waktu 3 detik untuk pengambilan data. Sedangkan pengukuran data manual peneliti memerlukan waktu kurang lebih 10 detik untuk setiap pengambilan.



BAB 5. PENUTUP

5.1 Kesimpulan

Pada pengujian yang telah dilakukan dari penelitian dengan judul “Rancang Bangun dan Impementasi Alat Ukur Digital untuk Mengukur Relasi Gigi Anterior” maka dapat disimpulkan sebagai berikut.

1. Pada pengujian sensor flex perubahan sudut berpengaruh dengan resistansi ketika sudut semakin besar resistasi semakin besar dan ketika sudut semakin kecil maka resistansi semakin berkurang.(Tabel 4.1 hal 25)
2. Pada perbandingan data *overjet* dan *overbite* antara pengukuran manual dengan digital *error* persen *overjet* terkecil 0% dan yang terbesar 490%, untuk *overbite error* persen 0% dan yang terbesar 19%.(Tabel 4.4 hal 36)
3. Dari waktu pengukuran digital didapatkan hasil bahwa pengukuran digital dengan waktu 3 detik dikarenakan menggunakan perulangan selama 3 detik.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan tentang “Rancang Bangun dan Impementasi Alat Ukur Digital untuk Mengukur Relasi Gigi Anterior”. Penulisan memberikan saran berikut dengan harapan untuk menyempurnakan manfaat yang lebih baik di masa mendatang :

1. Untuk penelitian selanjutnya pengukuran lebih cepat dapat menggunakan sensor yang lebih lebar sehingga dalam satu kali pengukuran dapat memunculkan semua nilai pengukuran gigi *anterior*.
2. Flex sensor dapat di ganti dengan sensor lain yang lebih kuat masa pemakaiannya.
3. Untuk penelitian selanjutnya jumlah sampel *overjet* dan *overbite* bernilai negatif di tambah agar semakin akurat pengambilan datanya.

DAFTAR PUSTAKA

- Abdulhamid. 2006. Jenis-jenis Penelitian.
http://abdulhamid.files.wordpress.com/.../materi_kuliah_3_19_feb_06.doc.
Diakses 21 Desember 2014
- Badan Penerbit Universitas Jember. 2012. *Pedoman Penulisan Karya Ilmiah*, Edisi Ketiga. Jember: Badan penerbit Universitas Jember.
- Etiawan, Iwan .2009. *Buku Ajar Sensor dan Transduser*. In: Sensor dan Transduser. Faculty of Engineering, Diponegoro University
- Foster, T.D . 1997. “Buku Ajar Ortodonsi”. Edisi III. Jakarta: EGC
- Sharp, john. 2010. “Microsoft Visual C# 2010 Step by Step”. Edisi I. Redmond: Microsoft prees
- Mico, Pardosi. 2003. “Buku Panduan Microsoft Visual Basic 6.0”Surabaya: PT Dua Selaras.
- Nasution, Minasari Imran. 2008.” Morfologi Gigi Desidui dan Gigi Permanen”. Medan: USU press
- Lia ,Martina. 2013. *Index Maloklusi*.
<http://fairytoot.blogspot.com/2013/10/indeks-maloklusi.html>. Diakases 19 Maret 2014
- Pazz, T. 2010. Prinsip kerja strain gauge. *elektronika-dasar.web.id/tag/prinsip-kerja-strain-gauge/* . Diakses 19 Maret 2014
- Raharjo,Pambudi. 2009. Ortodonti dasar. Surabaya: Airlangga University press.
- PUGUH ,PUGUH,2010 “ Aplikasi Flex Sensor dan Accelerometer Berbasis Arduino Utuk Pembelajaran Bahasa Isyarat Tangan” , Semarang: Fakultas Teknik Universitas Dian Nuswantoro Semarang
- Suhendri. 2013. Arduino uno.
<http://belajar-dasar-pemrograman.blogspot.com/2013/03/arduino-uno.html>.
Diakses 19 Maret 2014
- Wardana, kusuma ,dkk. 2010. Pengembangan Perangkat *Finger Motion Capture* Berbasis *Flex Sensor*. Surabaya: ITS

LAMPIRAN

1. Alat dan bahan untuk cetak gigi



2. Proses pembuatan adonan alginat



3. Proses pencetakan gigi



4. Hasil cetakan atas dan bawah



5. Proses pengecoran



6. Hasil pencetakan model gigi



7. Pengukuran overbite dengan penggaris



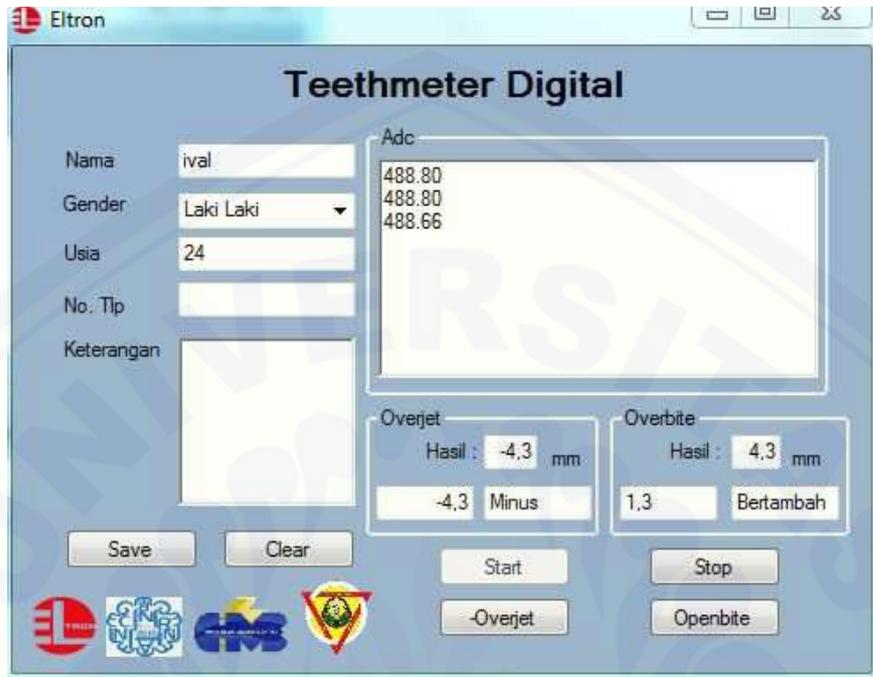
8. Pengukuran overjet dengan penggaris



9. Pengukuran *overjet* dan *overbite* digital



10. Tampilan GUI saat *minus overjet*.



11. Tampilan GUI gigitan terbuka



12. Listing Program arduino

```
void loop()
{
  butt1=analogRead(A2);
  butt2=analogRead(A3);
  butt3=analogRead(A4);
  butt4=analogRead(A5);

  if (butt1<50)
  { c=1; }
  else if (butt2<50)
  {c=0;}
  else if (butt3<50)
  {c=2;}
  else if (butt4<50)
  {c=3;}

  else if (c==1)
  {
    for(i=0;i<15000;i++)
    {
      sensorValue = analogRead(A0);
      total = total + sensorValue;
    }
    total=total/15000;
    b= ((4.02*total)-(2010))/20;
    j=((3.9*total)-(1950))/20;
  }
}
```

```
else if (c==2)
{
  for(i=0;i<15000;i++)
  {
    sensorValue = analogRead(A0);
    total = total + sensorValue;
  }
  total=total/15000;
  b= ((-5.1*total)-(2550))/-13.3;
  j=((5*total)-(2500))/-13.3;
}
```

```
else if (c==3)
{
  for(i=0;i<15000;i++)
  {
    sensorValue = analogRead(A0);
    total = total + sensorValue;
  }
  total=total/15000;
  b= (((1.7*total)-(850))/ 9)-4;
  j= (((-0.1*total)+(50))/1)+4;
}
else
{total=0;
}
```

```
lcd.setCursor(0,0);
lcd.print(">>>>");
lcd.setCursor(13,0);
lcd.print("<<<<");
lcd.setCursor(3,0);
lcd.print("ADC:");
lcd.setCursor(7, 0);
lcd.print(total,1);
lcd.setCursor(0,1);
lcd.print("OB:");
lcd.setCursor(3,1);
lcd.print(b,1);
lcd.setCursor(9,1);
lcd.print("OJ:");
lcd.setCursor(12,1);
lcd.print(j,1);
Serial.println(total);
```

13. Listing program C#

```
private void button1_Click(object sender, EventArgs e)
{
    button1.Enabled = false;
    button2.Enabled = true;
    arduino.PortName = "COM87";
    arduino.BaudRate = 9600;
    arduino.Parity = Parity.None;
    arduino.DataBits = 8;
    arduino.StopBits = StopBits.One;
    arduino.Open();
    //Detak.Start();
    for (int g = 1; g <= 3; g++)
    {
        kode = arduino.ReadLine();
        float b = float.Parse(kode);
        float bb = b / 100;
        double j = ((4.02 * bb) - 2010);
        double data = j / 20;
        //bite
        float adc = b / 100;
```

```
double bagi = ((3.9 * adc) - 1950);
double hasil = bagi / 20;

c = Convert.ToString(data);
decimal aa = Convert.ToDecimal(c);
aa = Math.Round(aa, 1);
richTextBox1.AppendText(kode);
textBox4.Text = Convert.ToString(aa);

d = Convert.ToString(hasil);
decimal dec = Convert.ToDecimal(d);
dec = Math.Round(dec, 1);
textBox6.Text = Convert.ToString(dec);

if (aa > 4)
{
    decimal aku = aa - 4;
    textBox1.Text = Convert.ToString(aku);
    textBox2.Text = "Bertambah";
}
else if (aa < 0)
{
    decimal aku = aa - 0;
    textBox1.Text = Convert.ToString(aku);
    textBox2.Text = "Minus";
}
else if (aa < 2)
{
    decimal aku = aa - 2;
    textBox1.Text = Convert.ToString(aku);
    textBox2.Text = "Berkurang";
}
else
{
    textBox1.Text = "0";
    textBox2.Text = "Normal";
}

if (dec > 3)
{
    decimal saya = dec - 3;
    textBox5.Text = Convert.ToString(saya);
    textBox3.Text = "Bertambah";
}
else if (dec < 0)
{
    decimal saya = dec - 0;
    textBox5.Text = Convert.ToString(saya);
    textBox3.Text = "Minus";
}
else
{
    if (dec < 1)
    {
```

```
        decimal saya = dec - 1;
        textBox5.Text = Convert.ToString(saya);
        textBox3.Text = "Minus";
        textBox3.Text = "Berkurang";
    }
    else
    {
        textBox5.Text = "0";
        textBox3.Text = "Normal";
    }
}
}

private void button2_Click(object sender, EventArgs e)
{
    button1.Enabled = true;
    button2.Enabled = false;

    arduino.Close();
    if (arduino.IsOpen)
    {
        arduino.Close();
    }
}
string kode;

private void serialPort1_DataReceived(object sender,
SerialDataReceivedEventArgs e)
{
    this.Invoke(new EventHandler(tampil));
}

string c, d;

private void tampil(object sender, EventArgs e)
{
}

private void comboBox1_SelectedIndexChanged(object sender,
EventArgs e)
{
    //comboBox1.Items.Add("perempuan");
    //kode.comboBox1.Items.Add("Laki laki");
}

private void Form1_Load(object sender, EventArgs e)
{
}
}
```

```
private void button5_Click(object sender, EventArgs e)
{
    Microsoft.Office.Interop.Excel.Application xls = new
Microsoft.Office.Interop.Excel.Application();
    Workbook wb = xls.Workbooks.Add(XlSheetType.xlWorksheet);
    Worksheet wa = (Worksheet)xls.ActiveSheet;

    xls.Visible = true;

    wa.Cells[1, 1] = "Nama";
    wa.Cells[1, 2] = "Gender";
    wa.Cells[1, 3] = "Usia";
    wa.Cells[1, 4] = "No Telpon";
    wa.Cells[1, 5] = "overjet";
    wa.Cells[1, 6] = "overbite";
    wa.Cells[1, 9] = "keterangan";
    wa.Cells[1, 7] = "Kondisi overjet";
    wa.Cells[1, 8] = "Kondisi overbite";

    wa.Cells[2, 1] = nama.Text;
    wa.Cells[2, 2] = comboBox1.Text;
    wa.Cells[2, 3] = usia.Text;
    wa.Cells[2, 4] = tlp.Text;
    wa.Cells[2, 5] = textBox5.Text;
    wa.Cells[2, 6] = textBox1.Text;
    wa.Cells[2, 9] = keterangan.Text;
    wa.Cells[2, 7] = textBox2.Text;
    wa.Cells[2, 8] = textBox3.Text;
}

private void button6_Click(object sender, EventArgs e)
{
    nama.Clear();
    usia.Clear();
    tlp.Clear();
    keterangan.Clear();
    textBox1.Clear();
    textBox5.Clear();
    richTextBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
}
float sec = 0;

private void button3_Click(object sender, EventArgs e)
{
    button1.Enabled = false;
    button2.Enabled = true;
    arduino.PortName = "COM87";
```

```
arduino.BaudRate = 9600;
arduino.Parity = Parity.None;
arduino.DataBits = 8;
arduino.StopBits = StopBits.One;
arduino.Open();

for (int g = 1; g <= 3; g++)
{
    kode = arduino.ReadLine();
    float b = float.Parse(kode);
    float bb = b / 100;
    double j = ((-5.1 * bb) + 2550);
    double data = j / -13.3;
    //bite
    float adc = b / 100;
    double bagi = ((5 * adc) - 2500);
    double hasil = bagi / -13.3;

    c = Convert.ToString(data);
    decimal aa = Convert.ToDecimal(c);
    aa = Math.Round(aa, 1);
    richTextBox1.AppendText(kode);
    textBox4.Text = Convert.ToString(aa);

    //textBox1.Clear();
    //a = Convert.ToDouble(b);
    d = Convert.ToString(hasil);
    decimal dec = Convert.ToDecimal(d);
    dec = Math.Round(dec, 1);
    textBox6.Text = Convert.ToString(dec);

    if (aa > 4)
    {
        decimal aku = aa - 4;
        textBox1.Text = Convert.ToString(aku);
        textBox2.Text = "Bertambah";
    }
    else if (aa < 0)
    {
        decimal aku = aa - 0;
        textBox1.Text = Convert.ToString(aku);
        textBox2.Text = "Minus";
    }
    else if (aa < 2)
    {
        decimal aku = aa - 2;
        textBox1.Text = Convert.ToString(aku);
        textBox2.Text = "Berkurang";
    }
    else
    {
        textBox1.Text = "0";
        textBox2.Text = "Normal";
    }
}
```

```
if (dec > 3)
{
    decimal saya = dec - 3;
    textBox5.Text = Convert.ToString(saya);
    textBox3.Text = "Bertambah";
}
else if (dec < 0)
{
    decimal saya = dec - 0;
    textBox5.Text = Convert.ToString(saya);
    textBox3.Text = "Minus";
}
else
{
    if (dec < 1)
    {
        decimal saya = dec - 1;
        textBox5.Text = Convert.ToString(saya);
        textBox3.Text = "Berkurang";
    }
    else
    {
        textBox5.Text = "0";
        textBox3.Text = "Normal";
    }
}
```