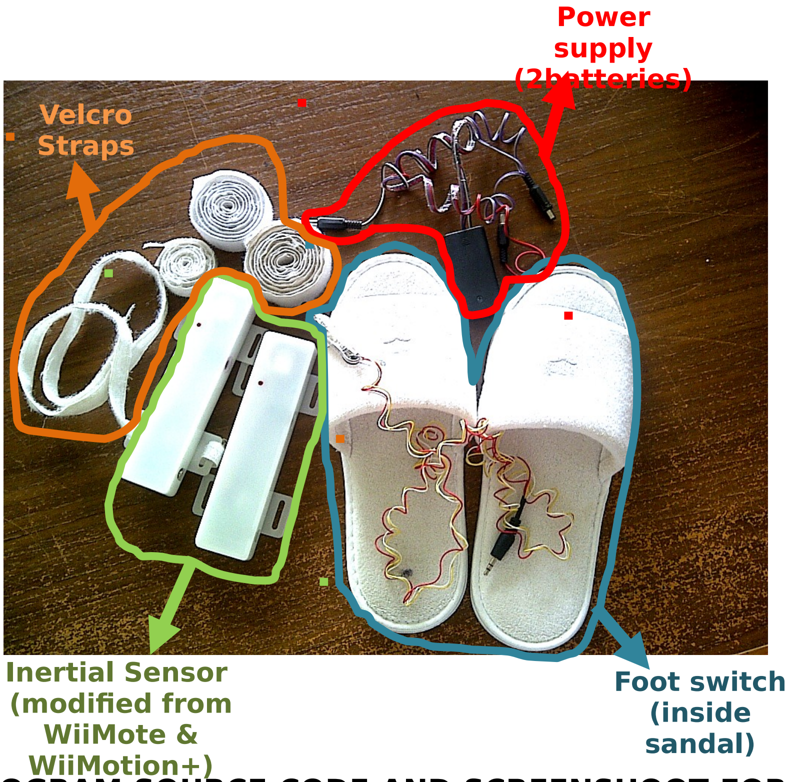
# LAMPIRAN *OUTPUT* PENELITIAN

## 1. HARDWARE PROTOTYPE


Power supply (2batteries)
Velcro Straps
Inertial Sensor (modified from WiiMote & WiiMotion+)
Foot switch (inside sandal)

## 2. PROGRAM SOURCE CODE AND SCREENSHOOT FOR DATA ACQUISITION (USING C#)

### 2.1 Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;
using WiimoteLib;

namespace Parameter_Gait
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new AnalisisGait());
        }
    }
}
```

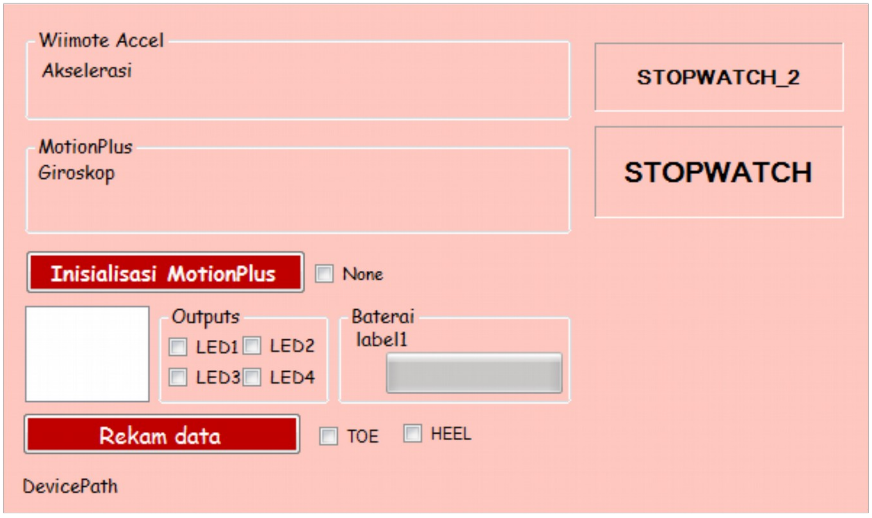
### 2.2 AnalisisGait.cs

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using WiimoteLib;
using System.Drawing.Drawing2D;
using System.Runtime.InteropServices;
/*using AForge.Video;
using AForge.Video.VFW;
using AForge.Video.DirectShow;*/

namespace Parameter_Gait
{
    public partial class AnalisisGait : Form
    {
        // map a wiimote to a specific state user control dealie
        Dictionary<Guid, parameter> mWiimoteMap = new Dictionary<Guid, parameter>();
        WiimoteCollection mWC;
        /*public FilterInfoCollection VideoCaptureDevices;
        public VideoCaptureDevice FinalVideo;*/

        public AnalisisGait()
        {   InitializeComponent();}

        private void AnalisisGait_Load(object sender, EventArgs e)
        {
            // find all wiimotes connected to the system
            mWC = new WiimoteCollection();
            int index = 1;

            try
            {  mWC.FindAllWiimotes();}
            catch (WiimoteNotFoundException ex)
            {    MessageBox.Show(ex.Message, "Wiimote not found error",
                 MessageBoxButtons.OK, MessageBoxIcon.Error); }
            catch (WiimoteException ex)
            {
                MessageBox.Show(ex.Message, "Wiimote error", MessageBoxButtons.OK,
                 MessageBoxIcon.Error);
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message, "Unknown error", MessageBoxButtons.OK,
                 MessageBoxIcon.Error);
            }

            foreach (Wiimote wm in mWC)
            {
                // create a new tab
                TabPage tp = new TabPage("Wiimote " + index);
                tabWiimotes.TabPages.Add(tp);

                // create a new user control
                parameter wi = new parameter(wm);
                tp.Controls.Add(wi);

                // setup the map from this wiimote's ID to that control
                mWiimoteMap[wm.ID] = wi;

                // connect it and set it up as always
                wm.WiimoteChanged += wm_WiimoteChanged;
                wm.WiimoteExtensionChanged += wm_WiimoteExtensionChanged;

                wm.Connect();
                if (wm.WiimoteState.ExtensionType != ExtensionType.BalanceBoard)
                    wm.SetReportType(InputReport.IRExtensionAccel,
                     IRSensitivity.Maximum, true);

                wm.SetLEDs(index++);
            }
        }

        void wm_WiimoteChanged(object sender, WiimoteChangedEventArgs e)
        {
            parameter wi = mWiimoteMap[((Wiimote)sender).ID];
            wi.UpdateState(e);
        }

     void wm_WiimoteExtensionChanged(object sender, WiimoteExtensionChangedEventArgs e)
        {
            // find the control for this Wiimote
            parameter wi = mWiimoteMap[((Wiimote)sender).ID];
            wi.UpdateExtension(e);

            if (e.Inserted)
                ((Wiimote)sender).SetReportType(InputReport.IRExtensionAccel, true);
            else
                ((Wiimote)sender).SetReportType(InputReport.IRAccel, true);
        }

        private void AnalisisGait_FormClosing(object sender, FormClosingEventArgs e)
        {
            foreach (Wiimote wm in mWC)
                wm.Disconnect();
        }
    }
}
```

### 2.3 parameter.cs

```
using System;
using System.IO;
using System.Collections;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using System.Drawing.Imaging;
using System.Data;
using System.Diagnostics;
using System.Linq;
using System.Runtime.InteropServices;
using System.Text;
using System.Windows.Forms;
using WiimoteLib;
using Microsoft.Office.Core;
using Excel = Microsoft.Office.Interop.Excel;


namespace Parameter_Gait
{
    public partial class parameter : UserControl
    {
         //inisiasi delegate untuk meng-update wiimote state & perubahan extension
        private delegate void UpdateWiimoteStateDelegate(WiimoteChangedEventArgs args);
        private delegate void
UpdateExtensionChangedDelegate(WiimoteExtensionChangedEventArgs args);

        private Wiimote mWiimote;
        private int rollfast = 0; private int pitchfast = 0; private int yawfast = 0;
        private int toecek = 0; private int heelcek = 0;

        double[] state = new double[14];

        public parameter()
        {
            InitializeComponent();
            //g = Graphics.FromImage(b);
        }
        /* WIIMOTE: fungsi-fungsi untuk Wiimote
         */
        public parameter(Wiimote wm):this ()
        {
             mWiimote = wm;
        }

        //BEGININVOKE prosedur untuk memanggil delegate update wiimote state
        public void UpdateState(WiimoteChangedEventArgs args)
        {
            BeginInvoke(new UpdateWiimoteStateDelegate(UpdateWiimoteChanged), args);
        }

        //BEGININVOKE prosedur untuk memanggil delegate update perubahan extension
        public void UpdateExtension(WiimoteExtensionChangedEventArgs args)
        {
            BeginInvoke(new UpdateExtensionChangedDelegate(UpdateExtensionChanged),
args);
        }

        private void chkLED_CheckedChanged(object sender, EventArgs e)
        {
            mWiimote.SetLEDs(chkLED1.Checked, chkLED2.Checked, chkLED3.Checked,
chkLED4.Checked);
        }


        //prosedur meng-update perubahan wiimote  state
        private void UpdateWiimoteChanged(WiimoteChangedEventArgs args)
        {
            WiimoteState ws = args.WiimoteState;

            toe.Checked = ws.ButtonState.Left;
            if (ws.ButtonState.Left)
            { toecek = 1; }
            else
            { toecek = 0; }

            heel.Checked = ws.ButtonState.Up;
            if (ws.ButtonState.Up)
            { heelcek = 1; }
            else
            { heelcek = 0; }

            lblAccel.Text = ws.AccelState.Values.ToString();

            switch (ws.ExtensionType)
            {   //extension yang digunakan hanya MotionPlus; nunchuck, balance board,
classic controllers, dll. tidak digunakan
                case ExtensionType.MotionPlus:
                    lblMotionPlus.Text = ws.MotionPlusState.RawValues.ToString();
                    clbSpeed.SetItemChecked(0, ws.MotionPlusState.YawFast);
                    clbSpeed.SetItemChecked(1, ws.MotionPlusState.PitchFast);
                    clbSpeed.SetItemChecked(2, ws.MotionPlusState.RollFast);
                    if (ws.MotionPlusState.YawFast)
                    { yawfast = 1; }
                    else
                    { yawfast = 0; }

                    if (ws.MotionPlusState.PitchFast)
                    { pitchfast = 1; }
                    else
                    { pitchfast = 0; }

                    if (ws.MotionPlusState.RollFast)
                    { rollfast = 1; }
                    else
                    { rollfast = 0; }
                    break;
            }

            //g.Clear(Color.Black);

            pbBattery.Value = (ws.Battery > 0xc8 ? 0xc8 : (int)ws.Battery);
            lblBattery.Text = ws.Battery.ToString();
            lblDevicePath.Text = "Device Path: " + mWiimote.HIDDevicePath;

            state[1] = rollfast;
            state[2] = pitchfast;
            state[3] = yawfast;
            state[4] = (double)(ws.AccelState.Values.X * 9.8);
            state[5] = (double)(ws.AccelState.Values.Y * 9.8);
            state[6] = (double)(ws.AccelState.Values.Z * 9.8);
            state[7] = (double)(ws.MotionPlusState.RawValues.X);
            state[8] = (double)(ws.MotionPlusState.RawValues.Y);
            state[9] = (double)(ws.MotionPlusState.RawValues.Z);
            state[10] = toecek;
            state[11] = heelcek;

            chkLED1.Checked = ws.LEDState.LED1;
            chkLED2.Checked = ws.LEDState.LED2;
            chkLED3.Checked = ws.LEDState.LED3;
            chkLED4.Checked = ws.LEDState.LED4;
        }

        //prosedur meng-update perubahan extension
        private void UpdateExtensionChanged(WiimoteExtensionChangedEventArgs args)
        {
            chkExtension.Text = args.ExtensionType.ToString();
            chkExtension.Checked = args.Inserted;
        }

        public Wiimote Wiimote
        {
            set { mWiimote = value; }
        }

        /* fungsi-fungsi user control untuk form
         */
        private void initMPlus_Click(object sender, EventArgs e)
        {
            mWiimote.InitializeMotionPlus();
        }

        System.Globalization.CultureInfo oldCI;
        //get the old CurrenCulture and set the new, en-US
        void SetNewCurrentCulture()
        {
            oldCI = System.Threading.Thread.CurrentThread.CurrentCulture;
            System.Threading.Thread.CurrentThread.CurrentCulture = new
System.Globalization.CultureInfo("en-US");
        }
        //reset Current Culture back to the originale
        void ResetCurrentCulture()
        {
            System.Threading.Thread.CurrentThread.CurrentCulture = oldCI;
        }

        public void data()
        {
            Stopwatch sw = new Stopwatch();
            //buat kalarray
            int col = 0;
            double[,] dataarray = new double[2100, 12];
            sw.Start();
            while (sw.ElapsedMilliseconds<21000)
            {
                swcounter.Text = sw.ElapsedMilliseconds.ToString();
                swsecond.Text = sw.Elapsed.Seconds.ToString();
                //isi kolom dataarray 0 - 11
                dataarray[col, 0] = sw.ElapsedMilliseconds;
                for (int j = 1; j < 12; j++)
                {
                    dataarray[col, j] = state[j];
                }
                PauseForMilliSeconds(10);
                col++;
            }
            sw.Stop();

            SetNewCurrentCulture();
            Excel.Application myExcelApp = new Excel.ApplicationClass();
            myExcelApp.Visible = true;
            object misValue = System.Reflection.Missing.Value;
            Excel.Workbooks myExcelWorkbooks = myExcelApp.Workbooks;
            Excel.Workbook myExcelWorkbook = myExcelWorkbooks.Add(misValue);

            Excel.Worksheet kal = (Excel.Worksheet)myExcelWorkbook.ActiveSheet;

            kal.Cells[1, 1] = "Waktu";
            kal.Cells[1, 2] = "FastBit Roll";
            kal.Cells[1, 3] = "FastBit Pitch";
            kal.Cells[1, 4] = "FastBit Yaw";
            kal.Cells[1, 5] = "AX";
            kal.Cells[1, 6] = "AY";
            kal.Cells[1, 7] = "AZ";
            kal.Cells[1, 8] = "GX";
            kal.Cells[1, 9] = "GY";
            kal.Cells[1, 10] = "GZ";
            kal.Cells[1, 11] = "toe";
            kal.Cells[1, 12] = "heel";

            kal.get_Range("A2", "L2101").Value2 = dataarray;
            ResetCurrentCulture();
        }
        public static DateTime PauseForMilliSeconds(int MilliSecondsToPauseFor)
                {
                    System.DateTime ThisMoment = System.DateTime.Now;
                    System.TimeSpan duration = new System.TimeSpan(0, 0, 0, 0, 
MilliSecondsToPauseFor);
                    System.DateTime AfterWards = ThisMoment.Add(duration);

                    while (AfterWards >= ThisMoment)
                    {
                        System.Windows.Forms.Application.DoEvents();
                        ThisMoment = System.DateTime.Now;
                    }
                    return System.DateTime.Now;
                }
        private void rekam_Click(object sender, EventArgs e)
        {  data();}
   }
}
```

## 2.4 PROGRAM SCREENSHOOT FOR DATA ACQUISITION (USING C#)



## 3. PROGRAM SOURCE CODE AND SCREENSHOOT FOR INERTIAL SENSOR DATA PROCESSING (USING MATLAB)

### 3.1 data_gait.m

```
% nama file: data_gait.m
clear; clc; close all;
% ---------------------------------------------------------------------
% JANGAN LUPA CEK "NAMA FILE", "KOLOM", "RANGE", & "INDEX" (untuk i)
% 1. lihat ekstensi file excel "*.XLS" (excel 2003) atau "*.XLSX" (excel 2007)
% 2. kalibrasi giroskop XYZ
    % "calt" = file saat giroskop diam/kalibrasi
% 3. "s" = shank/betis; "t" = thigh/paha
% --------------------------------------------------------------------------
% REFERENSI UNTUK KALIBRASI
    % uref = 8192; unit referensi
    % vref = 1350; mV referensi
    % vds = 2.27; mV/deg/s
    % dref = vref/vds = 594.7136564 deg/s
    % dalam (unit/deg/s)--> referensi unit: udref = uref/dref
     udref = 13.7746963;
% --------------------------------------------------------------------------
% THIGH SEGMENT
% BACA DATA WIIMOTE
    kalt='kalt_1610.xlsx';
    data1='t3_1610.xlsx';
% nilai kalibrasi -> nilai rata-rata unit G0X, G0Y, G0Z
    calt = xlsread(kalt,'E2:J1318');
    Acalt = [mean(calt(:,1)) mean(calt(:,2)) mean(calt(:,3))];
    Gcalt = [mean(calt(:,4)) mean(calt(:,5)) mean(calt(:,6))];

% "FB" = Fast Bit; bit indikator untuk mode cepat (pitchfast, rollfast, yawfast)
% Waktu dalam "detik"
    data_t = xlsread(data1,'A2:L1346');

    Tt = data_t(:,1)/1000;
    FBt = [data_t(:,2) data_t(:,3) data_t(:,4)];
    Att = [data_t(:,5) data_t(:,6) data_t(:,7)];
    Gtt = [data_t(:,8) data_t(:,9) data_t(:,10)];

% FOOT SWITCH: cek sajbungannya (ke Tt atau S?)
    TH = [data_t(:,11) data_t(:,12)];
% --------------------------------------------------------------------------
% NILAI AKSELERASI LINIER At & NILAI KECEPATAN SUDUT Gt
% "Go" = nilai giroskop sebelum disamakan orientasinya dengan akselerometer
% nilai kec.angular giroskop (dlm deg/s -> sensor frame) setelah dikalibrasi
    GoT=zeros(length(data_t),3);
    for i=1:length(data_t)
        for j=1:3
           if (FBt(i,j) == 0)
           GoT(i,j) = (Gtt(i,j)- Gcalt(1,j))/udref;
           else
           GoT(i,j) = ((Gtt(i,j)- Gcalt(1,j))/udref)*(2000/440);
           end
        end
    end
% koordinat giroskop wiimotionplus berbeda dari wiimote
    Gt    = [GoT(:,2), -1*GoT(:,3), -1*GoT(:,1)];
    At=zeros(length(data_t),3);
    for i=1:length(data_t)
        for j=1:3
            At(i,j) = (Att(i,j)- Acalt(1,j));
        end
    end
% --------------------------------------------------------------------------
% SHANK SEGMENT
% BACA DATA WIIMOTE
    kals='kals_1610.xlsx';
    data2='s3_1610.xlsx';
% nilai kalibrasi -> nilai rata-rata unit G0X, G0Y, G0Z
    cals = xlsread(kals,'E2:J2083');
    Acals = [mean(cals(:,1)) mean(cals(:,2)) mean(cals(:,3))];
    Gcals = [mean(cals(:,4)) mean(cals(:,5)) mean(cals(:,6))];

% "FB" = Fast Bit; bit indikator untuk mode cepat (pitchfast, rollfast, yawfast)
% waktu dalam "detik"
    data_s = xlsread(data2,'A2:L2096');

    Ts = data_s(:,1)/1000;
    FBs = [ data_s(:,2) data_s(:,3) data_s(:,4)];
    Ass = [ data_s(:,5) data_s(:,6) data_s(:,7)];
    Gss = [ data_s(:,8) data_s(:,9) data_s(:,10)];
% -------------------------------------------------------------------------
% NILAI AKSELERASI LINIER As & NILAI KECEPATAN SUDUT Gs
% "Go" = nilai giroskop sebelum disamakan orientasinya dengan akselerometer
% nilai kec.angular giroskop (dlm deg/s -> sensor frame) setelah dikalibrasi
    GoS=zeros(length(data_s),3);
    for k=1:length(data_s)
        for m=1:3
           if (FBs(k,m) == 0)
           GoS(k,m) = (Gss(k,m)- Gcals(1,m))/udref;
           else
           GoS(k,m) = ((Gss(k,m)- Gcals(1,m))/udref)*(2000/440);
           end
        end
    end
% koordinat giroskop wiimotionplus berbeda dari wiimote
    Gs    = [GoS(:,2), -1*GoS(:,3), -1*GoS(:,1)];
    As=zeros(length(data_s),3);
    for k=1:length(data_s)
        for m=1:3
            As(k,m) = (Ass(k,m)- Acals(1,m));
        end
    end
% -------------------------------------------------------------------------
% PLOT/FIGURE
% -------------------------------------------------------------------------
% THIGH
figure ('Name','3_1610_At','NumberTitle','off')
    subplot(3,1,1);line(Tt(1:(i-1)),At(1:(i-1),1),'marker','.','color', 'black');
    title('Grafik Akselerasi Linier - 1');
    subplot(3,1,2);line(Tt(1:(i-1)),At(1:(i-1),2),'marker','.','color','red');
    ylabel('Akselerasi (m/s2)');
    subplot(3,1,3);line(Tt(1:(i-1)),At(1:(i-1),3),'marker','.','color', 'blue');
    xlabel('Waktu (s)');
    saveas(gcf,'3_1610_At','fig');
    saveas(gcf,'3_1610_At','jpg');

figure ('Name','3_1610_Gt','NumberTitle','off')
    subplot(3,1,1);line(Tt(1:(i-1)),Gt(1:(i-1),1),'marker','.','color', 'black');
    title('Grafik Kecepatan Sudut - 1');
    subplot(3,1,2);line(Tt(1:(i-1)),Gt(1:(i-1),2),'marker','.','color','red');
    ylabel('Kecepatan Sudut (sudut/s)');
    subplot(3,1,3);line(Tt(1:(i-1)),Gt(1:(i-1),3),'marker','.','color', 'blue');
    xlabel('Waktu (s)');
    saveas(gcf,'3_1610_Gt','fig');
    saveas(gcf,'3_1610_Gt','jpg');
% -------------------------------------------------------------------------
% SHANK
figure ('Name','3_1610_As','NumberTitle','off')
    subplot(3,1,1);line(Ts(1:(k-1)),As(1:(k-1),1),'marker','.','color', 'black');
    title('Grafik Akselerasi Linier - 2');
    subplot(3,1,2);line(Ts(1:(k-1)),As(1:(k-1),2),'marker','.','color','red');
    ylabel('Akselerasi (m/s2)');
    subplot(3,1,3);line(Ts(1:(k-1)),As(1:(k-1),3),'marker','.','color', 'blue');
    xlabel('Waktu (s)');
    saveas(gcf,'3_1610_As','fig');
    saveas(gcf,'3_1610_As','jpg');

figure ('Name','3_1610_Gs','NumberTitle','off')
    subplot(3,1,1);line(Ts(1:(k-1)),Gs(1:(k-1),1),'marker','.','color', 'black');
    title('Grafik Kecepatan Sudut  - 2');
    subplot(3,1,2);line(Ts(1:(k-1)),Gs(1:(k-1),2),'marker','.','color','red');
    ylabel('Kecepatan Sudut (sudut/s)');
    subplot(3,1,3);line(Ts(1:(k-1)),Gs(1:(k-1),3),'marker','.','color', 'blue');
    xlabel('Waktu (s)');
    saveas(gcf,'3_1610_Gs','fig');
    saveas(gcf,'3_1610_Gs','jpg');
% -------------------------------------------------------------------------
% TOE & HEEL
```

```
figure ('Name','3_1610_TH','NumberTitle','off')
    subplot(2,1,1);line(Tt(1:(i-1)),TH(1:(i-1),1),'marker','.','color', 'black');
    title('Grafik TOE-HEEL');
    ylabel('TOE');
    subplot(2,1,2);line(Tt(1:(i-1)),TH(1:(i-1),2),'marker','.','color','red');
    ylabel('HEEL');
    saveas(gcf,'3_1610_TH','fig');
    saveas(gcf,'3_1610_TH','jpg');
% -------------------------------------------------------------------------
% KECEPATAN SUDUT-Z & TOE-HEEL
figure, line(Tt(1:(i-1)),Gt(1:(i-1),3),'marker','.','color','blue')
    hold on
    line(Ts(1:(k-1)),Gs(1:(k-1),3),'marker','.','color','red')
    hold on
    line(Tt(1:(i-1)),300*TH(1:(i-1),1),'marker','.','color','black'), line(Tt(1:(i-
    1)),300*TH(1:(i-1),2),'marker','.','color','green');
    legend ('Kec.Sudut THIGH','Kec.Sudut SHANK','TOE','HEEL')
    saveas(gcf,'3_1610_GzTH','jpg');
    saveas(gcf,'3_1610_GzTH','fig');
hold off
% -------------------------------------------------------------------------
% SHIFTING & CUTTING PLOT
% -------------------------------------------------------------------------
% Menyamakan waktu pengambilan data segmen paha (thigh) & betis (shank)
% Potong grafik pada bagian berjalan saat direkam saja
% ALERT: TIAP DATA BERBEDA!!
% t_sf = Tt(x)-Ts(x);dimana x adalah INDEX ketika nilai At(x)/Gt(x)= +-0
t_sf = 0.21;
Ts_s = Ts-t_sf;

idx_t = 194:771 ;
idx_s = 300:1198 ;
% -------------------------------------------------------------------------
% SAVE FILE .mat
% -------------------------------------------------------------------------
savefile = '3_1610.mat';
    Tt  = Tt(idx_t); TH = TH(idx_t,:);
    Axt = At(idx_t,1);Ayt = At(idx_t,2);Azt = At(idx_t,3);
    Gxt = Gt(idx_t,1);Gyt = Gt(idx_t,2);Gzt = Gt(idx_t,3);
    Ts  = Ts_s(idx_s);
    Axs = As(idx_s,1);Ays = As(idx_s,2);Azs = As(idx_s,3);
    Gxs = Gs(idx_s,1);Gys = Gs(idx_s,2);Gzs = Gs(idx_s,3);
save(savefile, 'Tt','Axt','Ayt','Azt','Gxt','Gyt','Gzt',...
    'TH','Ts','Axs','Ays','Azs','Gxs','Gys','Gzs');
```

### 3.2 data_gait.m

```
% nama file: gait_ictc.m
clear all; clc; close all;
% -------------------------------------------------------------------------
load '3_1610.mat'
figure, line(Tt,Gzt,'marker','.','color','blue')
    hold on
    line(Ts,Gzs,'marker','.','color','red')
    hold on

line(Tt,300*TH(:,1),'marker','.','color','black'),line(Tt,300*TH(:,2),'marker','.','col
or','green');
    legend ('Kec.Sudut THIGH','Kec.Sudut SHANK','TOE','HEEL')
    hold off
% -------------------------------------------------------------------------
% FILTERING
% frekuensi sampling (fs) dihitung dalam "lp5orbt.m"
% lp5orbt: LowPass filter Orde "X"(biasanya 5), Butterworth
% lp5orbt(Tt,a,N):
    % Tt=waktu
    % a=untuk frekuensi cut-off (fs/a)
    % N=orde filter

Gzt=filter(lp5orbt(Tt(:),5,5),Gzt(:));
Gzs=filter(lp5orbt(Ts(:),5,5),Gzs(:));
Axt=filter(lp5orbt(Tt(:),5,5),Axt(:));
```

```
Axs=filter(lp5orbt(Ts(:),5,5),Axs(:));
% -------------------------------------------------------------------------
% PROCESSING
% FIND PEAKS -> midswing || VALLEYS -> IC/HS,TC/TO
% ALERT: Pastikan INDEX T&DATA yang dicari peak-valley sama ukurannya
%         (sudah di CUTTING))
ic_t = ic(Tt(:), Gzt);tc_t = tc(Tt(:), Gzt);
ic_s = ic(Ts(:), Gzs);tc_s = tc(Ts(:), Gzs);
ic_TH = icTH(Tt,TH);
tc_TH = tcTH(Tt,TH);

savefile = 'ictc.mat';
```

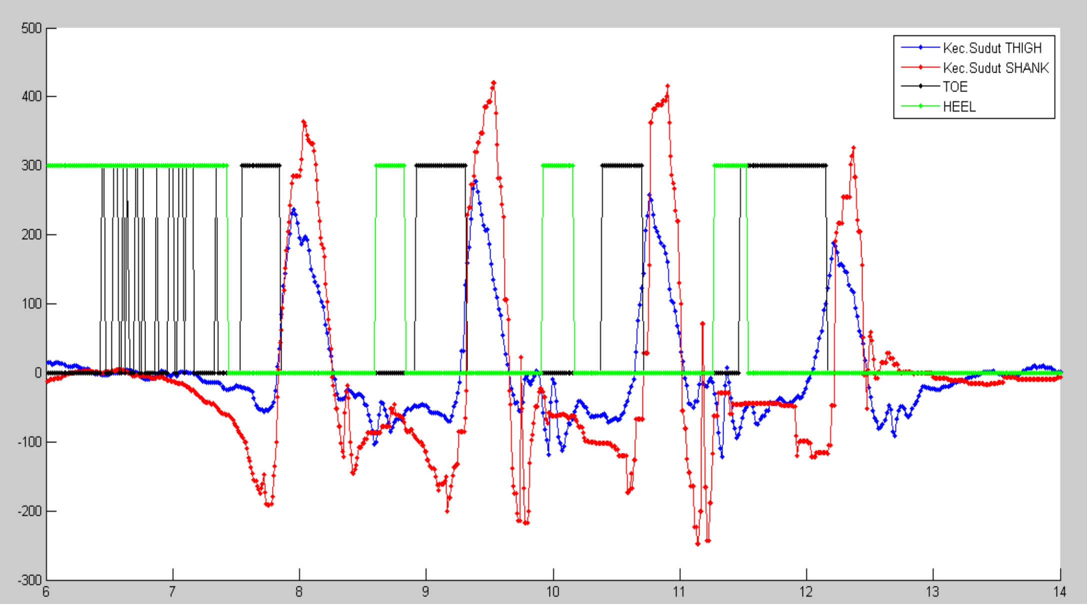
### 3.3 par_gait.m

```
% ------------------------------------------------------------------
% nama file: par_gait.m
clear all; clc; close all;
% ------------------------------------------------------------------
t1_ic=T(T>=T(T==ic(1,1))&T<=T(T==ic(2,1)));
t2_ic=T(T>=T(T==ic(2,1))&T<=T(T==ic(3,1)));

gz1_ic=Gz(T>=T(T==ic(1,1))&T<=T(T==ic(2,1)));
gz2_ic=Gz(T>=T(T==ic(2,1))&T<=T(T==ic(3,1)));
gzf1_ic=Gz1(T>=T(T==ic(1,1))&T<=T(T==ic(2,1)));
gzf2_ic=Gz1(T>=T(T==ic(2,1))&T<=T(T==ic(3,1)));

ax1_ic=Ax(T>=T(T==ic(1,1))&T<=T(T==ic(2,1)));
ax2_ic=Ax(T>=T(T==ic(2,1))&T<=T(T==ic(3,1)));
axf1_ic=Ax1(T>=T(T==ic(1,1))&T<=T(T==ic(2,1)));
axf2_ic=Ax1(T>=T(T==ic(2,1))&T<=T(T==ic(3,1)));
% ------------------------------------------------------------------
% SUDUT V.02 T==ic(1,1) ~ T==ic(length(ic),1)
incl1(1)=0;
for iag=1:length(t1_ic)-1
       incl1(iag+1)=trapz(t1_ic(1:iag+1),gzf1_ic(1:iag+1));
end
incl2(1)=0;
for iag=1:length(t2_ic)-1
       incl2(iag+1)=trapz(t2_ic(1:iag+1),gzf2_ic(1:iag+1));
end

% PARAMETER: KNEE ANGLES
incl1=incl1';
incl2=incl2';
incl1g = 90-incl1;
incl2g = 90-incl2;

figure
subplot(311),plot(t1_ic,gzf1_ic,t2_ic,gzf2_ic),legend('kec.sudut ic1-ic2','kec.sudut
ic2-ic3')
subplot(312),plot(t1_ic,incl1,t2_ic,incl2),legend('sudut ic1-ic2','sudut ic2-ic3')
subplot(313),plot(t1_ic,incl1g,t2_ic,incl2g),legend('90-sudut ic1-ic2','90-sudut ic2-
ic3')
saveas(gcf,'data1_sudut','fig');
% ------------------------------------------------------------------
% PARAMETER: TIMES
%stride times or CYCLE TIMES
idsr=1;
for ts=1:length(ic)-1
    tsr(idsr) = ic(ts+1,1)-ic(ts,1);
    idsr=idsr+1;
end
mean_tsr = mean(tsr);
% ------------------------------------------------------------------
% swing times
idsw=1;
for ts=1:length(ic)
    tsw(idsw) = ic(ts,1)-tc(ts,1);
    idsw=idsw+1;
end
mean_tsw = mean(tsw);
```

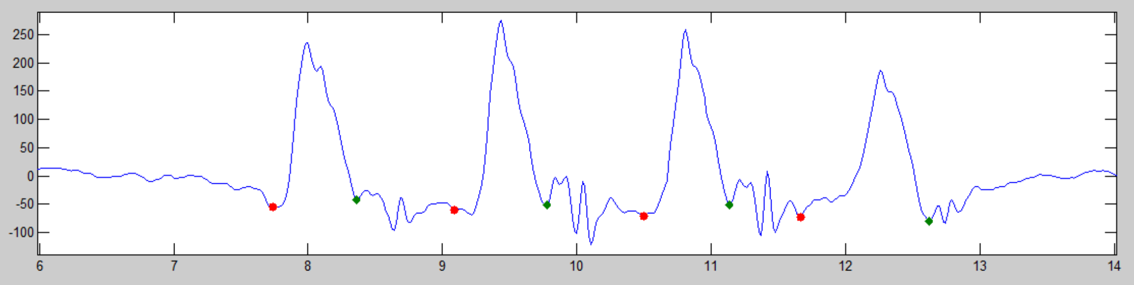
```
% -------------------------------------------------------------------
% stance times
idst=1;
for ts=1:length(tsr)
    tst(idst) = tsr(ts)-tsw(ts+1);
    idst=idst+1;
end
mean_tst = mean(tst);

temps = [max, tc, ic];
par_temps = [mean_tsr mean_tsw mean_tst];
% par_temp = {'t stride','t swing','t stance'; mean_tsr mean_tsw mean_tst};
% -------------------------------------------------------------------
% PARAMETER: CADENCE (steps/minute)
% stride time = 120/cadence
cadence = 120/mean_tsr;
cadence_r=round(cadence);
% -------------------------------------------------------------------
% PARAMETER: LENGTHS
% double integral Ax filtered T
vx1(1)=0;
disX1(1)=0;
for iag=1:length(T)-1
       vx1(iag+1)=trapz(T(1:iag+1),Ax1(1:iag+1));
       disX1(iag+1)=trapz(T(1:iag+1),vx1(1:iag+1));
end
% stride length T filtered (sum)
for n=1:length(ic)-1
    posl1=find(T==ic(n,1));
    posl2=find(T==ic(n+1,1));
    sl1(n)=disX1(posl2)-disX1(posl1);
end

figure
subplot(211),plot(T,vx1),legend('kec.linier filtered')
subplot(212),plot(T,disX1),legend('perpindahan filtered')
saveas(gcf,'data1_VxDxFilter','fig');
SL = mean(sl1(:));
V = SL*cadence/120;
% -------------------------------------------------------------------
% double integral Ax filtered
% T==ic(1,1) ~ T==ic(length(ic),1)
vl1(1)=0;
for iag=1:length(t1_ic)-1
       vl1(iag+1)=trapz(t1_ic(1:iag+1),axf1_ic(1:iag+1));
       dlx1(iag+1)=trapz(t1_ic(1:iag+1),vl1(1:iag+1));
end
for iag=1:length(t2_ic)-1
       vl2(iag+1)=trapz(t2_ic(1:iag+1),axf2_ic(1:iag+1));
       dlx2(iag+1)=trapz(t2_ic(1:iag+1),vl2(1:iag+1));
end

DLX1=(dlx1(length(dlx1))-dlx1(1));
DLX2=(dlx2(length(dlx2))-dlx2(1));
VL1 = DLX1*cadence/120;
VL2 = DLX2*cadence/120;

% stride length ic1~ic3 filtered
DLX= mean([DLX1 DLX2]);
VL = DLX*cadence/120;
% -------------------------------------------------------------------
% double integral Ax unfiltered
% T==ic(1,1) ~ T==ic(length(ic),1)
vu1(1)=0;
for iag=1:length(t1_ic)-1
       vu1(iag+1)=trapz(t1_ic(1:iag+1),ax1_ic(1:iag+1));
       dx1(iag+1)=trapz(t1_ic(1:iag+1),vu1(1:iag+1));
end
vu2(1)=0;
for iag=1:length(t2_ic)-1
       vu2(iag+1)=trapz(t2_ic(1:iag+1),ax2_ic(1:iag+1));
       dx2(iag+1)=trapz(t2_ic(1:iag+1),vu2(1:iag+1));
end

DX1 =(dx1(length(dx1))-dx1(1));
DX2 =(dx2(length(dx2))-dx2(1));
VU1 = DX1*cadence/120;
VU2 = DX2*cadence/120;

% stride length ic1~ic3 unfiltered
DX = mean([DX1 DX2]);
VU = DX*cadence/120;

figure
subplot(221),plot(t1_ic,axf1_ic),legend('aksel.linier ic1-ic2 filtered')
subplot(222),plot(t2_ic,axf2_ic),legend('aksel.linier ic2-ic3 filtered')
subplot(223),plot(t1_ic,ax1_ic),legend('aksel.linier ic1-ic2 unfiltered')
subplot(224),plot(t2_ic,ax2_ic),legend('aksel.linier ic2-ic3 unfiltered')
saveas(gcf,'data1_Aksel','fig');

figure
subplot(221),plot(t1_ic,vl1,t2_ic,vl2),legend('kec.linier ic1-ic2 filtered','kec.linier
ic2-ic3 filtered')
subplot(222),plot(t1_ic,dlx1,t2_ic,dlx2),legend('jarak ic1-ic2 filtered','jarak ic2-ic3
filtered')
subplot(223),plot(t1_ic,vu1,t2_ic,vu2),legend('kec.linier ic1-ic2
unfiltered','kec.linier ic2-ic3 unfiltered')
subplot(224),plot(t1_ic,dx1,t2_ic,dx2),legend('jarak ic1-ic2 unfiltered','jarak ic2-ic3
unfiltered')
saveas(gcf,'data1_KecJarak','fig');
```

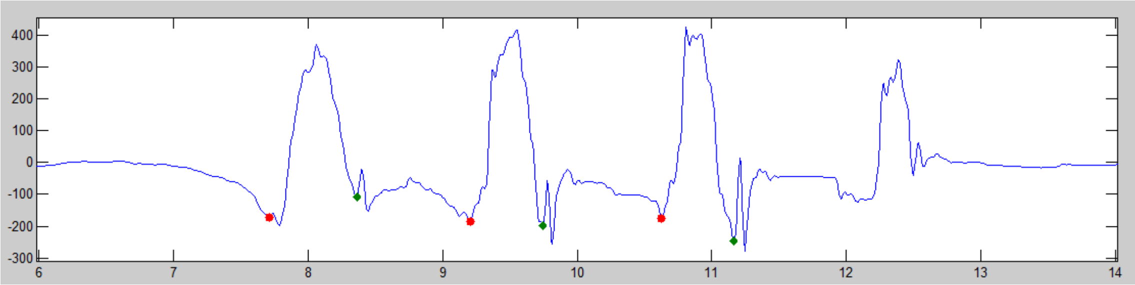
## 3.4 PROGRAM SCREENSHOOT FOR DATA PROCESSING (USING MATLAB)

### Angular Velocity of thigh and shank segments and foot switch result



### Initial Contact and Terminal Contact points at thigh (top) dan shank segment (below)




400
300
200
100
0
-100
-200
-300
6
7
8
9
10
11
12
13
14