Rancang Bangun Pendeteksi Suhu, Tekanan Darah dan Detak Jantung untuk Medical Check Up

Akuwan Saleh, Anang Budikarso

Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh November (ITS) Surabaya
Kampus PENS-ITS, Keputih, Sukolilo, Surabaya, Telp: +62+031+5947280
Email: anang_bk@eepis-its.edu

ABSTRAK

Penelitian ini membahas tentang tahap awal pemeriksaan medis dimana biasanya dilakukan pemeriksaan kesehatan sebelum penyakit seseorang didiagnosa. Biasanya kondisi yang diperiksa yaitu suhu tubuh, tekanan darah dan detak jantung.

Maka dari itu dirancang alat untuk mengukur parameter-parameter tersebut dengan cara mengukur suhu tubuh dengan sensor suhu, mengukur tekanan darah menggunakan sensor Tekanan darah dan mengukur detak jantung menggunakan sensor detak jantung.

Data hasil pengukuran akan diolah dan dikirim secara serial oleh Mikrokontroler AVR dan selanjutnya ditampilkan di PC dengan Microsoft Visual Basic.

Hasil pengukuran suhu di PC dibandingkan dengan termometer terdapat nilai error sebesar 0,44%. Tekanan dibandingkan dengan tensimeter digital terdapat nilai error sebesar 3,01% dan detak jantung di nilai error sebesar 3,55%.

Kata kunci : suhu tubuh, detak jantung, tekanan darah, mikrokontroler AVR, Microsoft Visual Basic.

1. Latar Belakang

Dengan berkembangnya teknologi dalam kehidupan sehari-hari sudah sangat dirasakan oleh setiap orang. Perkembangan dalam bidang elektronika terjadi suatu waktu, mulai dari hal yang sangat bersifat sederhana. Bahkan perkembangan teknologi elektronika sudah dapat dikembangkan dalam bidang medis terutama dalam melakukan diagnosa. Antara lain bisa sebagai alat kontrol kesehatan, alat bantu penyembuhan dan lain-lain.

Untuk tahap awal pemeriksaan medis, biasanya dilakukan *medical check up* sebelum penyakit seseorang didiagnosa. Dari hasil *medical check up* akan diketahui apakah seseorang dalam kondisi sehat atau tidak. Biasanya pada *medical check up* yang dilakukan di rumah sakit, beberapa kondisi yang akan diperiksa adalah suhu tubuh, detak jantung, tes darah, THT, dan lain sebagainya.

Dari kondisi di atas timbul gagasan untuk mengatasi kendala yang ada. Perancangan dan pembuatan suatu alat yang dapat digunakan untuk membantu mengukur atau mengetahui suhu tubuh, detak jantung dan tekanan darah dengan bantuan sensor. Hasil dari pengukuran sensor akan ditampilkan di personal computer berupa grafik dan karakter angka. Diharapkan alat ini dapat membantu mengetahui kondisi kesehatan dengan parameter diatas setidaknya untuk tahap awal pendeteksian.

1. Dasar teori

Makalah ini ditunjang oleh beberapa teori, antara lain sebagai berikut:

2.1. Sistem Saraf

Sistem saraf ini dibagi dalam 2 bagian yaitu sistem simpatis dan sistem parasimpatis. Fungsi saraf simpatik adalah serabut-serabut saraf simpatis mensarafi otot jantung, otot-otot tak sadar semua pembuluh darah, serta semua alat-alat dalam seperti lambung, pankreas dan usus. Melayani serabut sekretorik pada kelenjar keringat, serabut-serabut motorik pada otot sadar dalam kulit yaitu *arrectores pilorum* serta mempertahankan tonus semua otot, termasuk tonus otot sadar.

2.2. Temperatur Tubuh

Kulit sebagai organ pengatur panas. Suhu tubuh seseorang adalah tetap, meskipun terjadi perubahan suhu lingkungan. Hal itu dipertahankan karena penyesuaian antara panas yang hilang dan panas yang dihasilkan, yang diatur oleh pusat pengatur panas. Pusat ini segera menyadari bila ada perubahan pada panas tubuh, karena suhu darah yang mengalir melalui edula oblongata. Suhu normal (sebelah dalam) tubuh, yaitu suhu visera dan otak ialah 36° sampai 37.5°C. suhu kulit sedikit lebih rendah.

2.3. Denyut Jantung

Daya pompa jantung, pada orang yang sedang istirahat jantungnya berdebar sekitar 70 kali semenit dan memompa 70 ml setiap denyut (volume denyutan adalah 70 ml). jumlah darah yang setiap menit dipompa dengan demikian adalah 70 x 70 ml atau sekitar 5 liter. Sewaktu banyak bergerak

kecepatan jantung dapat menjadi 150 setiap menit dan volume denyut lebih dari 150 ml, yang membuat daya pompa jantung 20 sampai 25 liter setiap menit. Tiap menit sejumlah volume yang tepat sama kembali dari vena ke jantung. Akan tetapi, bila pengmbalian dari vena tidak seimbang dan ventrikel gagal mengimbangnya dengan daya pompa jantung, maka terjadi payah jantung.

2.4. Tekanan Darah

Tekanan sistolik pada masa bayi sekitar 70-90 mmHg dan tekanan diastolik kira-kira 50 mmHg. Tekanan darah berubah menurut umur. Pada dewasa muda tekanan sistolik kira-kira 110-125 mmHg dan tekanan diastolik 60-70 mmHg. Sedangkan pada umur yang lebih tua tekanan sistolik kira-kira 130-150 mmHg dan tekanan diastolik sekitar 80-90 mmHg. Perbedaan tekanan antara sistole dan diastole disebut tekanan nadi dan nomalnya berkisar antara 30-50 mmHg. Batas terendah tekanan sistole pada orang dewasa diperkirakan 105 mmHg dan batas teratas ialah 150. Pada wanita tekanan darahnya ialah 5-10 mmHg lebih rendah daripada pria.

2.5 CodeVision AVR

2.5.1 Cara Membuat Proyek Baru

1. Tanpa CodeWizardAVR

Buka pogram CodeVisionAVR dengan cara klik ganda ikon **CVAVR** pada *desktop* atau dengan cara klik **Start** → **Programs** → **CodeVisionAVR Evaluation** → **CodeVision AVR C Compiler**. Pada menu **File** pilih **New** atau klik ikon , kemudian pada jendela *Create New File* pilih **Project** dan klik **OK**. Untuk membuat proyek baru tanpa bantuan CodeWizard AVR klik **No** pada jendela yang muncul berikutnya.

Setelah itu pilih direktori untuk menyimpan proyek yang akan dibuat, misalnya pada direktori "C:\Coba CVAVR\", kemudian isi nama proyek tersebut. Lalu klik **Save**.

Pada jendela *Configure Project .prj*, tab *C Compiler*, pilih mikrokontroler yang digunakan misalnya ATMEGA 8535 dan pilih frekuensi *clock* (kristal) yang digunakan oleh mikrokontroler, misalnya 4 MHz. Kemudian klik **OK**.

Langkah pembuatan proyek baru selesai tapi belum dapat digunakan untuk membuat program C. Agar dapat digunakan untuk membuat program C diperlukan pembuatan *file c* yaitu klik menu **File** pilih **New** atau klik ikon kemudian pada jendela *Create New File* pilih **Source** dan klik **OK**. Akan terdapat *file c* dengan nama "untitled.c". Pada menu **File** pilih **Save As...**, kemudian isi nama *file* yang akan dibuat.

Untuk penggabungan file yang berekstensi .c dengan file berekstensi .prj, yaitu klik pada menu **Project**, pilih **Configure**. Pada label *Files*, pilih **Add**.

Kemudian pilih file yang berekstensi .c dan klik **Open**. Program C tersebut ditambahkan dalam proyek yang telah dibuat sebelumnya, kemudian klik **OK** pada jendela *Configure Project Test*.

Setelah selesai membuat suatu program C, perlu dilakukan pemeriksaan apakah *syntax* (sintaksis) program yang dibuat benar atau salah yaitu dengan menekan tombol **F9** (*compile*) atau melalui menu **Project** kemudian pilih **Compile**. Jika terdapat kesalahan sintaksis akan terdapat pesan *error* dimana letak kesalahan terjadi.

2. Dengan CodeWizardAVR

Selain membuat proyek baru dengan cara yang telah dibahas sebelum ini, cara lain yang lebih mudah dan cepat yaitu dengan menggunakan CodeWizardAVR. Langkah-langkahnya sebagai berikut:

Pada menu **File** pilih **New** atau klik ikon , kemudian pada jendela *Create New File* pilih **Project** dan klik **OK**. Untuk membuat proyek baru dengan bantuan CodeWizardAVR klik **Yes**.

Akan muncul jendela *CodeWizardAVR – untitled.cwp*, pilih mikrokontroler yang digunakan dan pilih frekuensi *clock* yang digunakan oleh mikrokontroler.

Pada CodeWizardAVR ini telah disediakan beberapa fungsi yang terdapat pada mikrokontroler yang digunakan sehingga kita dapat menggunakan bantuan CodeWizardAVR untuk membuat suatu program C agar dapat menggunakan fungsi-fungsi tersebut. Tetapi program C yang dibuat oleh CodeWizardAVR hanya untuk pengaturan awal saja sehingga masih diperlukan tambahan program untuk menggunakan fungsi-fungsi tersebut.

Setelah selesai memilih fungsi yang digunakan, pada menu **File** pilih **Generate, Save and Exit**. Kemudian tentukan nama file program C yang akan dibuat, klik **Save**. Kemudian tentukan nama *project* yang akan dibuat, klik **Save**. dan terakhir yaitu tentukan nama *wizard* untuk pengaturan fungsi pada CodeWizardAVR, klik **Save**. Setelah selesai, secara otomatis akan muncul program C yang dibuat oleh CodeWizardAVR pada *project*. Jika program C yang dibuat oleh CodeWizardAVR terdapat pengaturan yang tidak diperlukan maka program C tersebut dapat dihapus.

Setelah selesai membuat program C, *compile* program tersebut apakah ada kesalahan sinaksis atau tidak dengan menekan tombol **F9**. Jika sintaksis program sudah benar maka tidak ada pesan kesalahan (*error*).

2.6 Konfigurasi Port Serial

Gambar 1. adalah gambar konektor port serial DB-9 pada bagian belakang CPU. Pada komputer IBM PC kompatibel biasanya dapat menemukan dua konektor port serial DB-9 yang biasa dinamai COM1 dan COM2.

Gambar 1 Konektor serial DB-9 pada bagian belakang CPU

3. Desain Rangkaian

3.1 Block Diagram

Gambar 2. Blok diagram fungsional

3.2 Prinsip Kerja

Pada prinsipnya, komunikasi serial adalah komunikasi dimana pengiriman data dilakukan per bit, sehingga lebih lambat jika dibandingkan komunikasi paralel.

Komunikasi serial membutuhkan port sebagai saluran data. Yang umum digunakan sebagai port serial adalah DB9 yang terdapat 2 jenis konektor yaitu pin 25 dan pin 9.

Pada modul DT-AVR dengan ATmega 8535 sebagai mikro kontrolernya sudah tersedia kabel untuk komunikasi serial dengan jenis konektor 9 pin di satu sisi dan konektor RJ-11 di sisi yang lain.

Pengiriman hasil konversi data analog dari sensor suhu dan sensor tekanan darah pada port A pin 0 dan pin 7 akan dikirim secara terus menerus (real-time) dengan delay 200 ms. Begitu juga sinyal pulse sebagai hasil keluaran dari sensor detak jantung pada port D pin 2 juga akan dikirim dengan delay yang sama. Ketiga jenis data yang dikirim akan ditampilkan secara berurutan pada terminal *Code Vision AVR*. Pada icon *Run the terminal*, dapat diketahui apakah data berhasil diterima oleh PC atau belum. Data yang diterima PC sudah diubah kembali dalam bentuk ASCII.

Dengan *Microsoft Visual Basic*, data hasil pengukuran yang dikirim ke PC akan ditampilkan berupakarakter angka dan grafik.

4. Pengujian dan Analisa

4.1 Pengukuran Suhu di PC

Gambar 3. Diagram pengujian tampilan suhu di PC

Setelah dilakukan perbandingan antara hasil yang diperoleh termometer dengan hasil pengukuran yang ditunjukkan di PC, maka didapatkan hasil seperti dibawah ini:

Tabel 1. Hasil perbandingan pengukuran suhu tubuh dengan termometer dan pengukuran di PC

Waktu (detik)	Termometer (°C)	Pengukuran di PC (°C)	Error (%)
10	35,1	35,375	0,783
20	35,2	35,375	0,498
30	35,8	35,874	0,206
40	35,8	35,874	0,206
50	35,8	35,874	0,206
60	35,9	35,874	0,072

Dalam pengujian ini perlu dilakukan pengkalibrasian dengan menggunakan program pada codevision AVR agar pembacaan sesuai antara keluaran LM 35 dengan tampilan di PC. Adapun perintah dalam penggunaan program sebagai berikut:

```
// Global enable interrupts
#define sei()
while(1)
{
    k=0;
    k=read_adc(0);
    k=(k/1024*5)/0.0098;
    ftoa(k,3,sadc);
    printf("%6s",sadc);
}
```

Analisa program:

'k' adalah deklarasi variabel masukan, kemudian diposisikan di port A ke 0 sebagai ADC. Selanjutnya, 'k' dibagi dengan 10 bit yang ada dalam port A ($2^{10} = 1024$), setelah itu dikali dengan tegangan referensi sebesar 5 V. Hasilnya dibagi dengan 0,0098.

0.0098 adalah angka yang terdapat pada datasheet LM 35 dimana perubahan tegangan keluaran mencapai 10 mV tiap °C. Setelah dilakukan kalibrasi dengan termometer ternyata terdapat perbedaan hasil antara suhu yang ditunjukkan termometer dengan hasil yang ditampilkan di PC.

Gambar 4. Tampilan suhu di Visual Basic

```
Private Sub Timer1_Timer()
    data = MSComm1.Input
    Text4.Text = data
    If aktif = True Then
        suhu = Mid(Text4.Text, 1, 6)
        Text1.Text = Mid(Text4.Text, 1, 6)
    End If

    ....

    If Val(Text1.Text) > 37 Then
        Label1.Enabled = False
        Label2.Enabled = True
    Else

        If Val(Text1.Text) >= 33.1 And Val(Text1.Text) <= 37 Then
            Label1.Enabled = True
            Label2.Enabled = False
        Else
            If Val(Text1.Text) >= 30 And Val(Text1.Text) <= 33 Then
                Label1.Enabled = False
                Label2.Enabled = False
            End If
        End If
    End If

    ....
End Sub
```

Analisa program :

Dengan MSComm1.Input, port serial akan dibuka dan data dari mikrokontroler AVR dikirim. Seluruh data akan ditampilkan pada Text4.Text aktif adalah variabel global yang didefinisikan sebagai logika boolean. Dari data yang ditampilkan di Text4.Text diambil karakter angka mulai dari angka pertama samapi keenam sebagai hasil pengukuran suhu tubuh.

Sedangkan kategori normal atau demam ditentukan melalui seleksi kondisi dengan perintah *if...then*. Jika dari hasil pengukuran diperoleh nilai antara 33.1 sampai dengan 37, maka yang ditampilkan adalah kata 'Normal'. Namun jika dari hasil pengukuran diperoleh nilai lebih dari 37, maka yang akan ditampilkan adalah kata 'Demam'. Sedangkan untuk nilai kurang dari atau sama dengan 33 maka kedua label di atas tidak akan ditampilkan karena dianggap bahwa pengukuran belum dilakukan.

4.2 Pengukuran Tekanan Darah di PC

Gambar 5. Diagram pengujian tekanan darah

Setelah dilakukan pengukuran dengan spygmanometer digital (tensimeter digital) kemudian hasilnya dibandingkan dengan pengukuran yang dilakukan di PC. Hasil yang diperoleh dari kedua pengukuran tersebut adalah sebagai berikut :

Tabel 2. Hasil perbandingan pengukuran tekanan darah dengan tensimeter digital dan pengukuran di PC

Tekanan	Pengukuran dengan Spygmanometer digital	Pengukuran dengan MPX5100 yang ditampilkan di PC	% error
Sistole	112 mm Hg	111,74 mm Hg	0,23
Diastole	77 mm Hg	74,082 mm Hg	3,78

Dalam pengujian ini maka perlu dilakukan pengkalibrasian dengan menggunakan program pada codevision AVR agar pembacaan sesuai antara keluaran sensor dengan tampilan PC .Adapun perintah dalam penggunaan program sebagai berikut:

```
k=0;
k=read_adc(7);
k=((k/1024*5)/0.01)+15;
ftoa(k,3,sadc);
printf("%6s",sadc);
```

Analisa program :

Dimana inisialisasi 'k' adalah deklarasi variabel masukan, kemudian diposisikan di port A ke 7 sebagai ADC. Selanjutnya rumusnya, 'k' dibagi dengan 10 bit yang ada dalam port A ($2^{10} = 1024$),setelah itu dikali dengan tegangan referensi sebesar 5 v. Hasilnya dibagi dengan 0,01 v. Untuk menyamakan hasil agar sama pada tampilan di PC maka setelah dibagi dengan 0,01 maka ditambah 15, jadi hampir mendekati keluaran sensor dengan tampilan di PC.

Gambar 6. Tampilan hasil pengukuran tekanan darah

```
Text5.Text = Mid(Text4.Text, 7, 6)
Text15.Text = Mid(Text4.Text, 13, 1)

If aktif = False Then
    jantung = Text15.Text
    If jantung = 1 Then
        j = j + 1

        If j = 1 Then
            Text16.Text = Mid(Text4.Text, 7, 6)
            End If

        If j = 2 Then
            Text17.Text = Mid(Text4.Text, 7, 6)
            End If
        End If
    End If
End If
....
```

Analisa program :

Tekanan yang diberikan oleh spygmanometer ditampilkan di text5.text dengan pengambilan data pada urutan karakter angka ketujuh sebanyak enam karakter. Hasil pengukuran sistole akan ditampilkan jika terdengar bunyi detak pertama dan diastole akan ditampilkan jika terdengar bunyi detak kedua.

4.3 Pengukuran Detak Jantung

Gambar 7. Diagram pengujian detak jantung

Pada pengukuran ini akan diperoleh hasil berupa jumlah detak jantung dalam satu menit. Pengujian dilakukan saat tubuh dalam keadaan istirahat dan setelah beraktifitas. Terlihat bahwa setelah beraktifitas jumlah detak jantung lebih banyak dibandingkan sebelum beraktifitas.

Tabel 3. Hasil perbandingan pengukuran detak jantung dengan tensimeter digital dan pengukuran di PC

Keadaan detak jantung	Mic kondensor (Kali / Menit)	Tensimeter digital (kali / menit)	Error(%)
Istirahat	75	73	2,7
Aktivitas	86	90	4,4

Adapun pengujian dilakukan dengan cara pengkalibrasian dengan tensimeter digital.

Tensimeter digital yang digunakan : OMRON, Model : IA2 (HEM-7011-C1), Serial no : 5800774L

```
unsigned int frek,pulse;

// External Interrupt 0 service routine
interrupt [EXT_INT0] void ext_int0_isr(void)
{
    pulse++;
    // Place your code here
}

// Standard Input/Output functions
#include <stdio.h>

// Timer 1 output compare A interrupt service routine
interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{
    TCNT1=0;
    frek=pulse;
    pulse=0;
    // Place your code here
}

....
printf("%lu", frek);
```

Analisa program :

Variable *frek* dideklarasikan sebagai integer dan *pulse* didefinisikan sebagai *frek*. Pada interrupt eksternal 0 service routine, nilai pulse akan dijumlahkan. Sedangkan pada timer1, keluaran pulse akan dcompare dengan nilai 0. Jika terdapat masukan (detak) maka akan dikirim sebagai nilai '1'. Namun jika tidak, nilai yang dikirim ke PC adalah karakter '0'.

Pengukuran jumlah detak jantung dilakukan dengan mik kondensor yang dihubungkan dengan stetoskop. Hasil dari pengukuran ini dibandingkan dengan hasil pengukuran yang dilakukan dengan spygmanometer digital (tensimeter digital) dimana pada alat ini jumlah detak jantung juga ditampilkan sebagai hasil pengukuran.

Sedangkan pada PC, pengukuran terhadap detak jantung akan ditampilkan seperti pada gambar di bawah ini.

Gambar 8. Tampilan hasil pengukuran detak jantung

Analisa data:

Pada pengukuran menunjukkan bahwa di saat istirahat atau dalam keadaan tenang maka frekuensi jantung akan berdetak normal. Dan pada saat melakukan aktifitas yang berlebihan atau dalam keadaan yang mencemaskan, jantung sebagai pompa yaitu dipaksa untuk melakukan aktifitas atau frekuensi detak yang berlebihan. Catatan, dalam penggunaan mik kondensor kemudian dimasukkan ke dalam pipa stetoskop untuk mendeteksi bunyi detak sangat sensitif sehingga perlu hati-hati dalam pengukuran. Jadi posisi stetoskop dalam keadaan tenang, tidak boleh geser atau ada guncangan.

```
Private Sub Text15_Change()  
jantung = Text15  
If jantung = 1 Then  
    If Timer2.Enabled = True Then  
        jan = jan + 1  
    End If  
End If  
Text6 = jan  
End Sub
```

Analisa program:

Jika timer 2 aktif maka setiap detak jantung yang diterima dan dikirim ke PC akan dijumlahkan. Penjumlahan terhadap detak jantung ini akan berhenti setelah 60 detik. Hasil pengukuran akan ditampilkan pada text6.text.

V. KESIMPULAN

Setelah melakukan proses pengamatan dan analisa terhadap sistem yang telah dibuat dapat diambil kesimpulan bahwa:

1. Hasil pengukuran suhu di PC dibandingkan dengan termometer terdapat nilai error sebesar 0,44%. Nilai error yang didapat kecil dikarenakan keluaran tegangan sensor LM 35 linier terhadap temperatur dengan kenaikan sebesar 10 mV/°C.

2. Hasil pengukuran tekanan darah di PC dibandingkan dengan tensimeter digital terdapat nilai error sebesar 3,01%. Terjadi nilai error yang cukup besar disebabkan pemakaian mik kondensor (integrasi dengan stetoskop) untuk mengukur sistole dan diastole terlalu sensitif.
3. Hasil pengukuran detak jantung di PC dibandingkan dengan tensimeter digital terdapat nilai error sebesar 3,55%. Nilai error yang didapat cukup besar dikarenakan sensitivitas mik kondensor yang digunakan terlalu peka, geser sedikit atau ada guncangan mempengaruhi hasil data.

VI. DAFTAR PUSTAKA

- [1]. Budiharto, Widodo, Gamayel Rizal, "12 Proyek Mikrokontroler untuk Pemula", PT Elex Media Komputindo, Jakarta, 2007.
- [2]. Evelyn, C.Pearce, "Anatomi dan Fisiologi untuk Paramedis", PT.Gramedia Pustaka Utama, Jakarta, 2006.
- [3]. Rahman, Abdul, "Perencanaan dan Pembuatan Monitoring Kondisi Ketegangan Mental", Surabaya, 2006.
- [4]. Suhata, "VB sebagai Pusat Kendali Peralatan Elektronika", Elex Media Komputindo, Jakarta, 2005
- [5]. Wiyono, Didik, "Panduan Praktis Mikrokontroler Keluarga AVR Menggunakan DT-Combo AVR-51 Starter Kit Dan DT-Combo AVR Exercise Kit", Innovative Electronics, Surabaya, 2007.