

Percobaan 2

Pemrograman IVR Dasar(Inbound)

2.1. Tujuan :

Setelah melaksanakan praktikum ini mahasiswa diharapkan mampu :

- Melakukan pemrograman pengesetan fungsi Dialogic menggunakan Visual C++
- Melakukan pemrograman *Device Input/Output* (open channel, getdigit, wait ring)
- Membuat sebuah aplikasi layanan dengan *Voice Processing Board*

2.2. Peralatan :

- PABX sebagai penyedia jalur telepon
- 2 pesawat Telepon
- 1 PC dilengkapi dengan Dialogic Card (D/41JCT-LS atau D/4PCI-U) sebagai IVR Server

2.3. Teori :

Pemrograman Voice meliputi 3 hal : *Voice Driver*, *Voice Library* dan Fungsi Voice. *Voice Driver*, digunakan untuk berkomunikasi dan mengontrol voice hardware. Voice hardware berisi board penyimpanan dan penghantar voice.

Beberapa fitur Voice yang bisa diaplikasikan pada board-board voice ini meliputi :

- Rekam dan playback data suara
- Pengaturan Kecepatan dan Volume play
- Peng-handel-an panggilan
- Analisa panggilan - Basic dan Lanjutan
- Pembangkitan dan deteksi tone DTMF, MF, and R2MF.
- Pembangkit dan Pendeteksi Tone Global
- Pembangkit Tone berirama

Voice Library menyediakan interface dengan Voice Driver. Voice Library yang digunakan untuk aplikasi *single-threaded* dan *multi-threaded* terdiri dari :

- *libdxxmt.lib* → Voice Library utama
- *libsrlmt.lib* → Standard Run-time Library

Library dari fungsi-fungsi “C” dapat digunakan untuk :

- utilisasi seluruh voice board
- Menulis aplikasi menggunakan model pemrograman *Single-threaded Asynchronous* atau *Multi-threaded Synchronous*
- Mengkonfigurasi peralatan
- Meng-handle kejadian-kejadian yang muncul pada device
- Mengembalikan informasi device.

Library *Standard Run-time* menyajikan sejumlah fungsi-fungsi sistem yang tidak tergantung dari device dan dapat diaplikasikan ke seluruh device Dialogic.

a. Model Pemrograman Single-threaded Asynchronous

Model ini memungkinkan sebuah program tunggal dapat mengontrol berbagai kanal suara dalam satu kendali. Model ini dapat digunakan untuk pengembangan aplikasi kompleks dimana beberapa tugas dapat dikoordinir secara simultan. Model Pemrograman Asynchronous mensupport baik manajemen polled maupun callback.

b. Model Pemrograman Multi-threaded Synchronous

Model ini menggunakan fungsi-fungsi yang mem-blok eksekusi aplikasi sampai seluruh fungsi selesai. Pada model ini, aplikasi mengontrol masing-masing kanal dari kendali yang berbeda. Model ini memungkinkan untuk mengatur aplikasi berbeda pada kanal yang berbeda secara dinamis dan realtime.

Driver Dialogic board sudah dilengkapi dengan ***fungsi-fungsi Voice*** khusus untuk menjalankan aplikasi-aplikasi pada Dialogic board. Fungsi ini tinggal dipanggil dengan menggunakan bahasa pemrograman C++. Beberapa fungsi-fungsi khusus tersebut adalah :

2.3.1. Fungsi Manajemen Device

Fungsi ini digunakan untuk membuka dan menutup device (board dan kanal).

dx_open(), yang berfungsi membuka kanal

dx_close() yang berfungsi menutup kanal.

Pada konfigurasi CT Bus, sebuah board meliputi device interface digital (dtiBxTx) dan device suara (dxxBxCx), yang masing-masing mempunyai kanal yang independent. B diikuti dengan nomor board, C diikuti dengan nomor kanal suara (1 s/d 4). T diikuti nomor time slot interface digital (1 s/d 24 untuk T1, dan 1 s/d 30 untuk E1).

2.3.2. Fungsi I/O

Fungsi I/O digunakan untuk mentransfer data ke dan dari kanal idle yang terbuka. Semua fungsi I/O akan menyebabkan kanal menjadi sibuk ketika proses pentransferan data sedang berlangsung dan kembali idle jika proses transfer data sudah selesai.

Jenis-jenis fungsi I/O adalah :

dx_playiottdata() · play data suara dari berbagai sumber

dx_rec() · merekam data suara ke satu atau lebih tujuan

dx_reciottdata() · merekam data suara ke berbagai tujuan

dx_RxIottData() · menerima data pada kanal tertentu

dx_setdigbuf() · set mode buffering digit

dx_stopch() · stop I/O yang sedang berlangsung

dx_TxIottData() · Transmit data pada kanal tertentu

dx_TxRxIottData() · mulai penerimaan inisiasi transmit data

dx_wink() · wink a channel

2.3.3. Fungsi Play dan Record

Fungsi Play dan Record digunakan untuk memainkan atau merekam data suara, baik dari sebuah kanal atau lebih. Jenis-jenis fungsi Play dan Record adalah sebagai berikut :

dx_play() · play data suara yang sudah direkam

dx_playf() · play data suara yang sudah direkam dari sebuah file tunggal

dx_playiottdata() · play data suara dari berbagai sumber

dx_playvox() · plays file vox tunggal

dx_playwav() · plays file wave tunggal

dx_mreciottdata() · rekam data suara dari dua kanal menjadi sebuah file, device atau memory tunggal..

dx_rec() · rekam data suara ke satu atau lebih tujuan

dx_recf() · rekam data suara ke file tunggal

dx_reciottdata() · rekam data suara ke berbagai tujuan

dx_recvox() · rekam data suara ke file vox tunggal

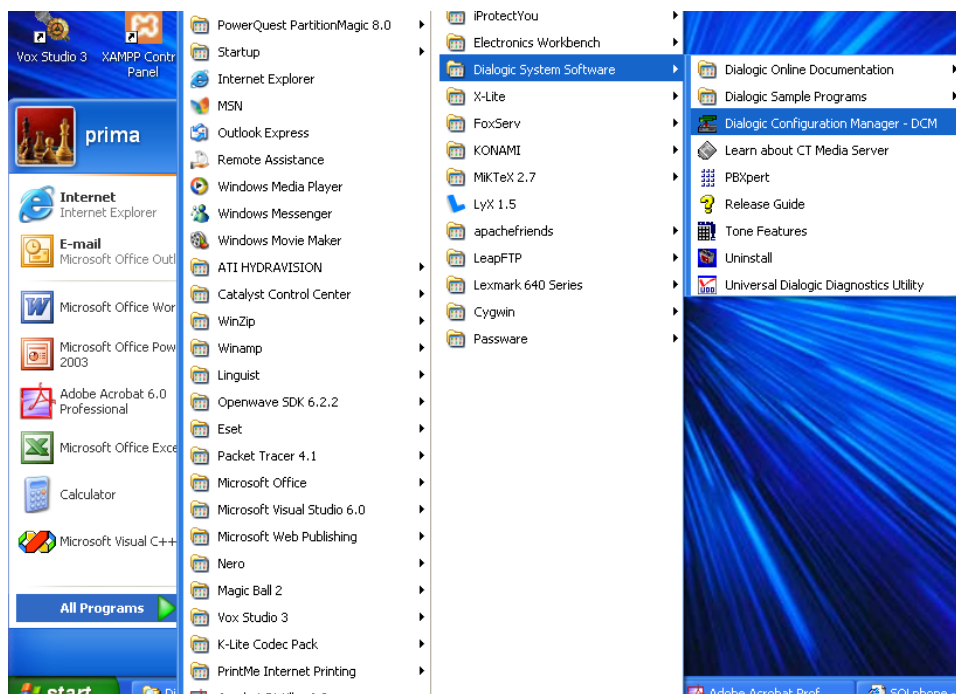
dx_recwav() · rekam data suara ke file wave tunggal

2.4. Bekerja pada Visual C++ dengan menggunakan Dialogic Card

2.4.1 Pengaktifan Dialogic

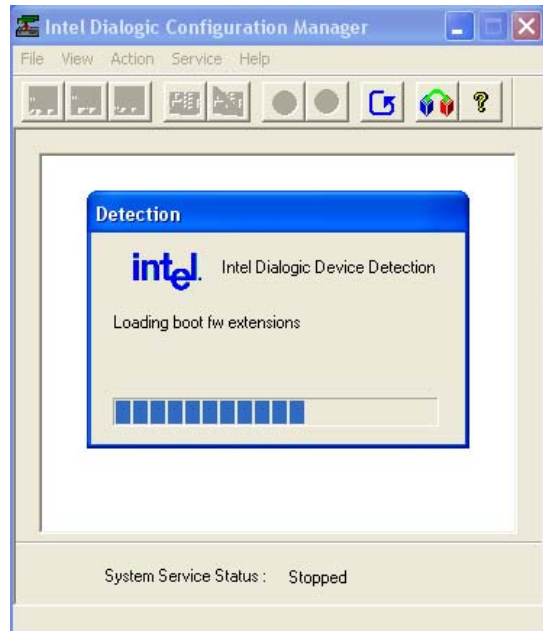
Langkah-langkah pengaktifan dialogic adalah sebagai berikut:

1. Start *Dialogic Configuration manager*. Melalui *Start* menu, pilih *All Program*, kemudian *Dialogic System Software*, selanjutnya pilih *Dialogic Configuration Manager (DCM)*



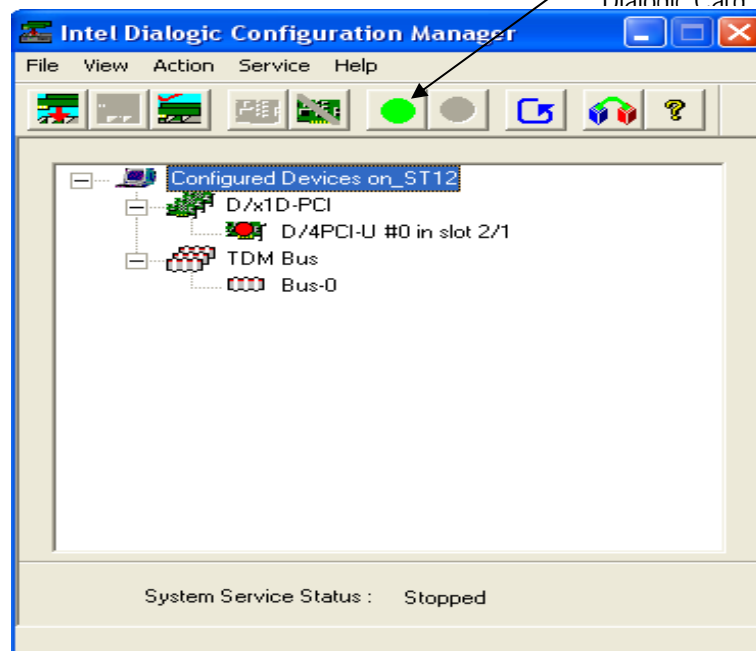
Gambar 2.1. Start Dialogic Configuration Manager

2. Pendeteksian Lokasi Dialogic Card



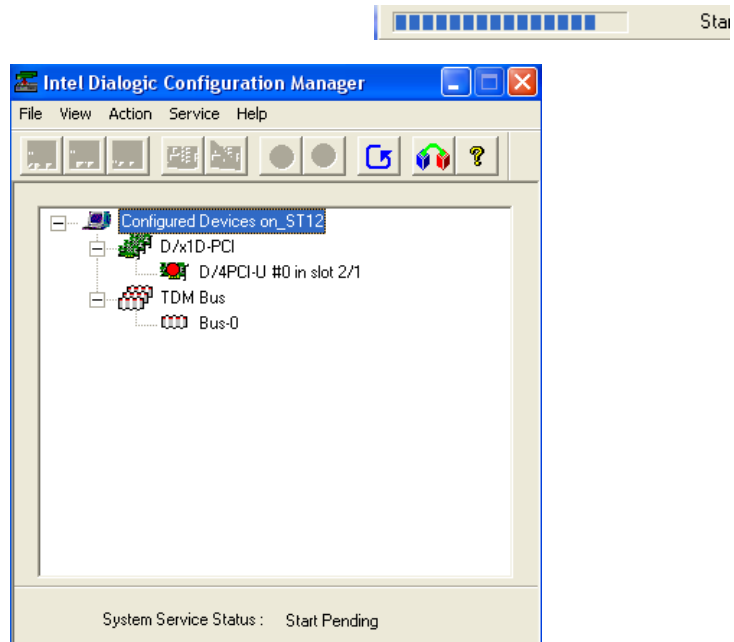
Gambar 2.2. Pendeteksian Lokasi Card

3. Lokasi card sudah terdeteksi. Model dan nomor card yang terpasang di slot tertentu akan ditampilkan (contoh di sini model card D/4PCI-U nomer 0, terpasang pada slot PCI ke-2)



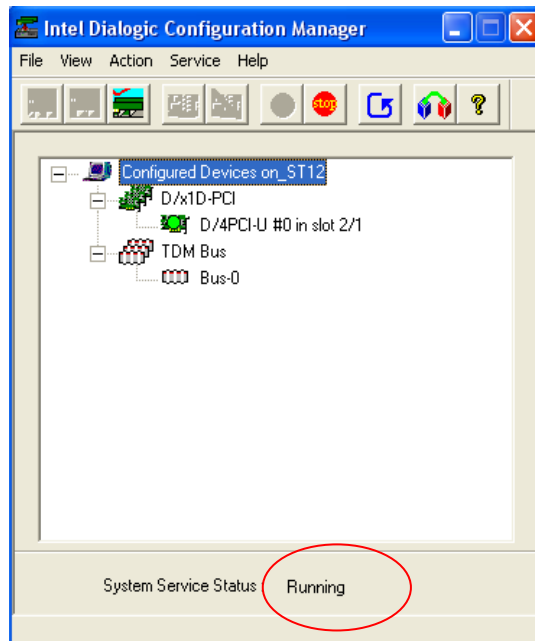
Gambar 2.3. Model dan Nomer Card yang terpasang sudah terdeteksi

4. Untuk mulai menjalankan fungsi Dialogic Card, klik tanda lingkaran hijau pada tools menu bagian atas. Tunggu sebentar. Perhatikan, di sebelah kanan atas dari PC ada display bar yang menyatakan proses aktivasi sedang terjadi.



Gambar 2.4. Mulai menjalankan fungsi Dialogic Card.

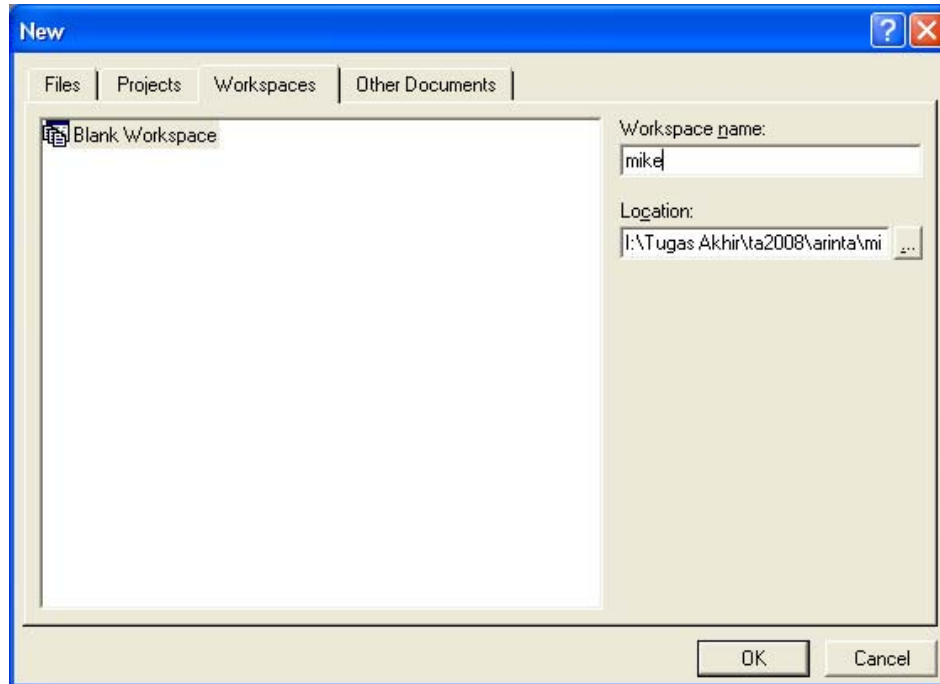
5. Dialogic Card sudah aktif dan siap digunakan, ditunjukkan dengan *system service status = Running*.



Gambar 2.5. Dialogic Card sudah aktif

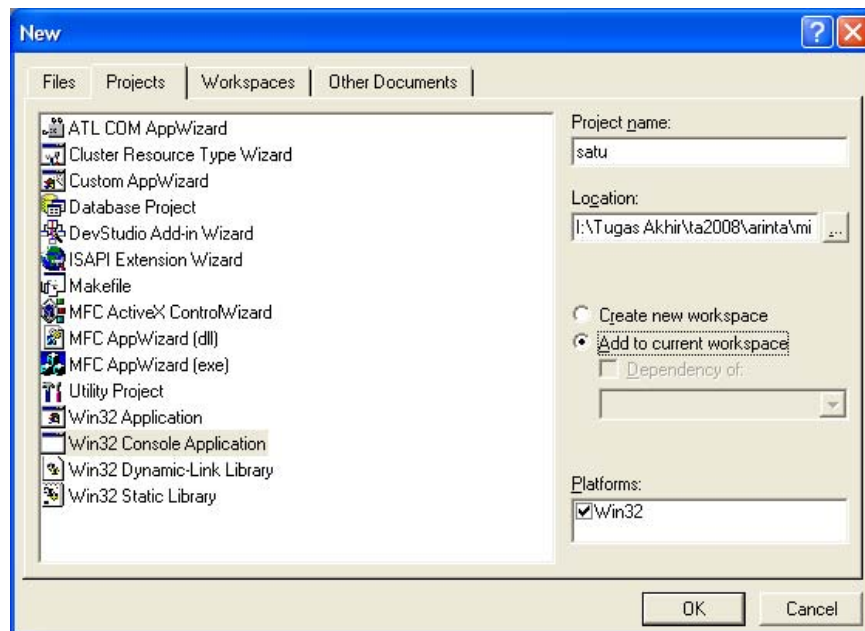
2.4.2 Visual C++

1. Buka Visual C++, File→new→workspace→beri nama kemudian klik OK seperti tampak pada gambar 2.6 dibawah ini



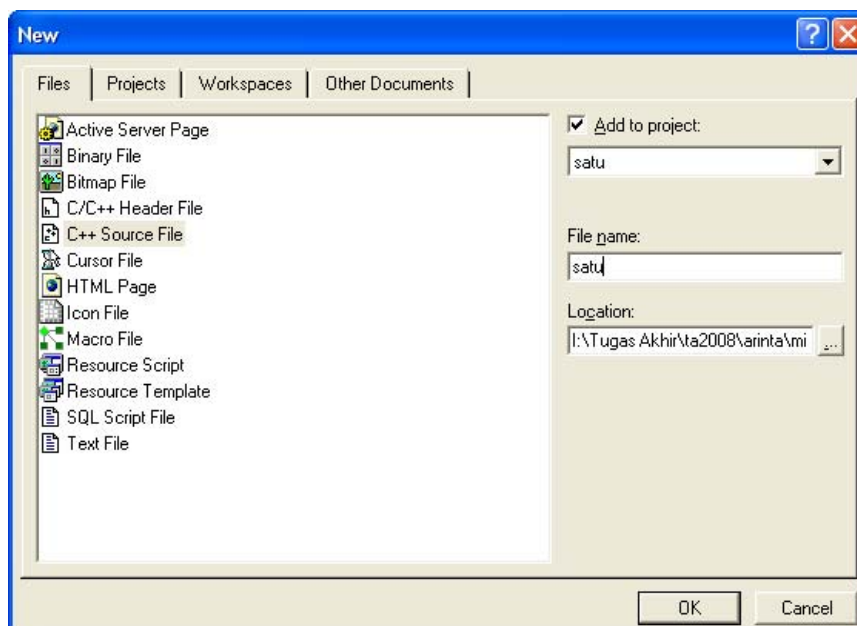
Gambar 2.6. Tampilan Workspace

6. Buka File→new→project→Win32 Console Application→beri nama→add to current workspace kemudian klik OK seperti tampak pada gambar 2.7 dibawah ini



Gambar 2.7. Tampilan Project

7. Buka File→new→Files→C++ sources File→beri nama kemudian klik OK seperti tampak pada gambar 2.8 dibawah ini



Gambar 2.8. Tampilan Files

4. Setelah membuat project dan file *.cpp, pilih *project* → *setting*

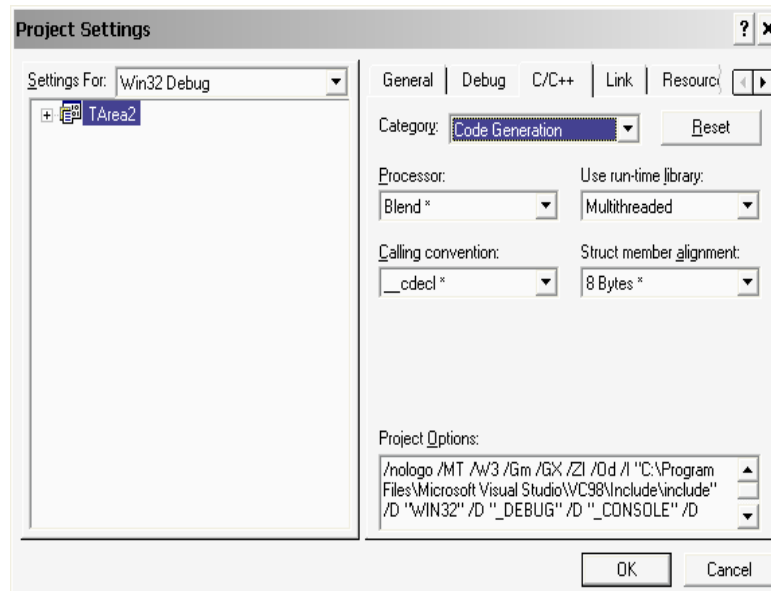
a. Tab: C/C++

- [*Category: Code Generation*]

Use run-time library:

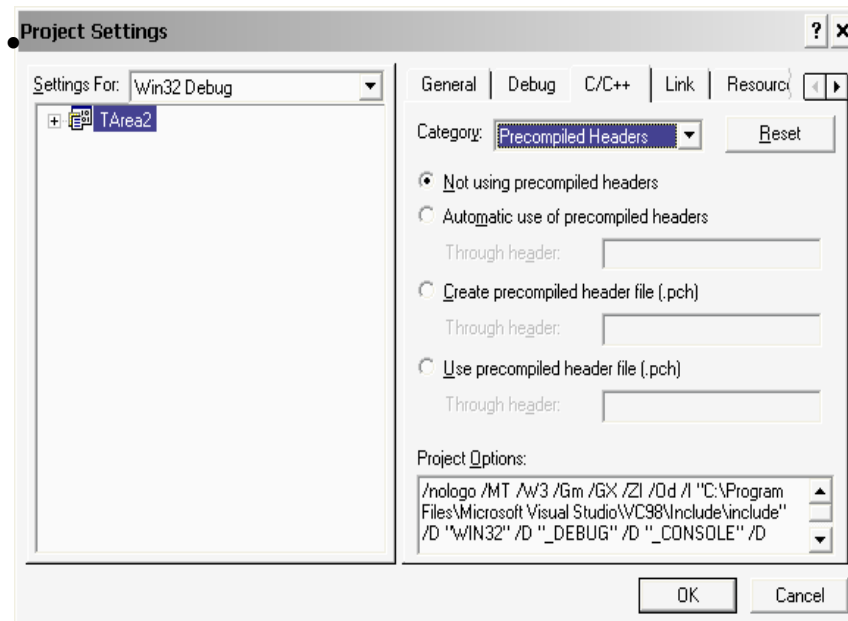
Multithreaded

Hasil pengesetannya seperti pada 2.9



Gambar 2.9 Setting Category Code Generation

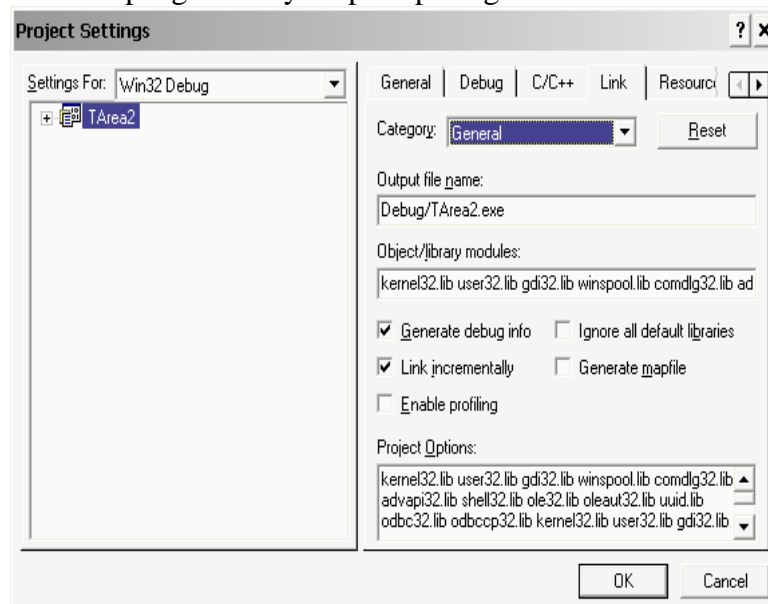
- [Category: Precompiled Headers]
Select 'Not using precompiled headers'
Hasil pengesetannya seperti pada gambar 2.10



Gambar 2.10 Setting Category Precompiled Headers

b. Tab:Link

- [Category: General]
Object/library modules: (tuliskan)
libshlmt.lib dan libdxxmt.lib
Hasil pengesetannya seperti pada gambar 2.11

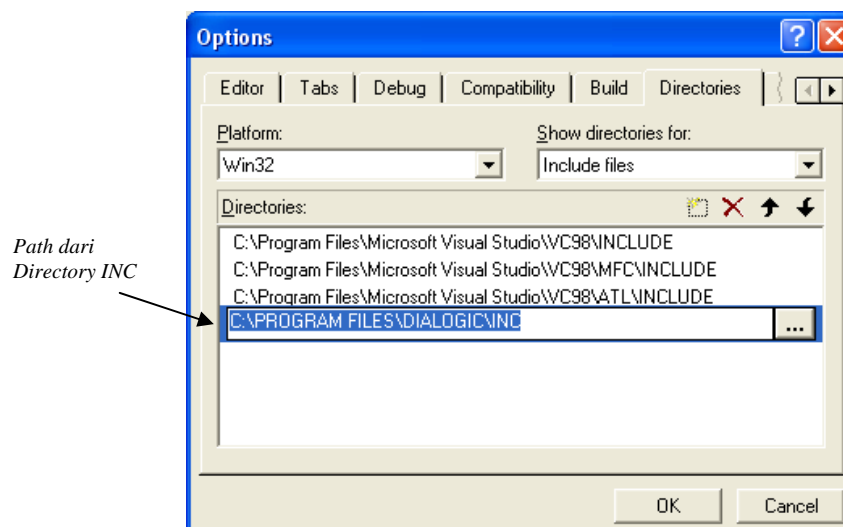


Gambar 2.11 Setting Category General

5. Menambahkan Directory INC dan LIB

Masih pada sheet Proyek → **Tools** → **Options** → **Directories** → **Show Directories** for : pilih Include file → browse folder dimana Directory INC untuk Dialogic berada, sehingga didapatkan : C:\Program Files\Dialogic\INC

Show Directories for : pilih **Library file** → browse folder dimana Directory LIB untuk Dialogic berada, sehingga didapatkan : C:\Program Files\Dialogic\LIB → Ok



Gambar 2.12. Menambahkan Directory INC dari Source Dialogic

6. . Menambah *header-header Dialogic* pada Project yang dibuat.

```
#include <windows.h>
#include <fcntl.h>
#include <srllib.h>
#include <dxxplib.h>
#include <stdio.h>
```

7. Mengisi fungsi baru dengan program dari Dialogic

a. Fungsi Deteksi Digit

Fungsi ini digunakan untuk mendeteksi digit yang dikirimkan oleh pesawat telepon user ke IVR server.

```
char DetectDigit(int chdev, char digit[10], int x)
{
    DV_TPT tpt[3];
    DV_DIGIT digp;
    int numdigs, cnt;
    dx_clrtpt(tpt, 3);
    tpt[0].tp_type=IO_CONT;
    tpt[0].tp_termno=DX_MAXDTMF;
    tpt[0].tp_length=x;
    tpt[0].tp_flags=TF_MAXDTMF;
    tpt[1].tp_type=IO_CONT;
    tpt[1].tp_termno=DX_LCOFF;
    tpt[1].tp_length=10;
    tpt[1].tp_flags=TF_LCOFF|TF_10MS;
    tpt[2].tp_type=IO_EOT;
    tpt[2].tp_termno=DX_MAXTIME;
    tpt[2].tp_length=50;
    tpt[2].tp_flags=TF_MAXTIME;

    //Get digit//

    if((numdigs=dx_getdig(chdev, tpt, &digp, EV_SYNC))== -1)
    {
        printf("Error get digit\n");
        exit(1);
    }
    printf("get digit succes\n");
    for(cnt=0; cnt<numdigs; cnt++)
    {
        digit[cnt]=digp.dg_value[cnt];
    }
    return(digp.dg_value[0]);
}
```

Gambar 3.13. Isi dari Fungsi DetectDigit()

b. Fungsi Play Suara

Fungsi ini akan digunakan untuk memainkan suara yang sudah direkam sebelumnya. Untuk menghemat waktu, proses perekaman suara bisa dilakukan melalui *Multi Thread Mode* yang sudah tersedia pada Sample Program.

```

void PlaySuara(int chdev, char fname[10])
{
    int fd;
    DX_IOTT iott;
    DV_TPT tpt;
    DX_XPB xpb;
    if((fd=dx_fileopen(fname,O_RDONLY|O_BINARY)) ==-1)
    {
    }
    tpt.tp_type =IO_EOT;
    tpt.tp_termno =DX_MAXDTMF;
    tpt.tp_length =1;
    tpt.tp_flags =TF_MAXDTMF;
    iott.io_fhandle =fd;
    iott.io_bufp =0;
    iott.io_offset =0;
    iott.io_length =-1;
    iott.io_type =IO_DEV|IO_EOT;
    xpb.wFileFormat =FILE_FORMAT_WAVE;
    xpb.wDataFormat =DATA_FORMAT_DIALOGIC_ADPCM;
    xpb.nSamplesPerSec =DRT_8KHZ;
    xpb.wBitsPerSample =4;
    if(dx_playiottdata(chdev,&iott,&tpt,&xpb,EV_SYNC)==-1)
    {
        printf("Error play wav file\n");
        exit(1);
    }

    printf("play wav succes\n");
}

```

Gambar 2.14 Isi dari Fungsi Play Suara

8. Buat Prototype fungsi dibawah header-header dialogic yang sudah dibuat

```

#include <windows.h>
#include <fcntl.h>
#include <srllib.h>
#include <dxxplib.h>
#include <stdio.h>

```

Prototype fungsi

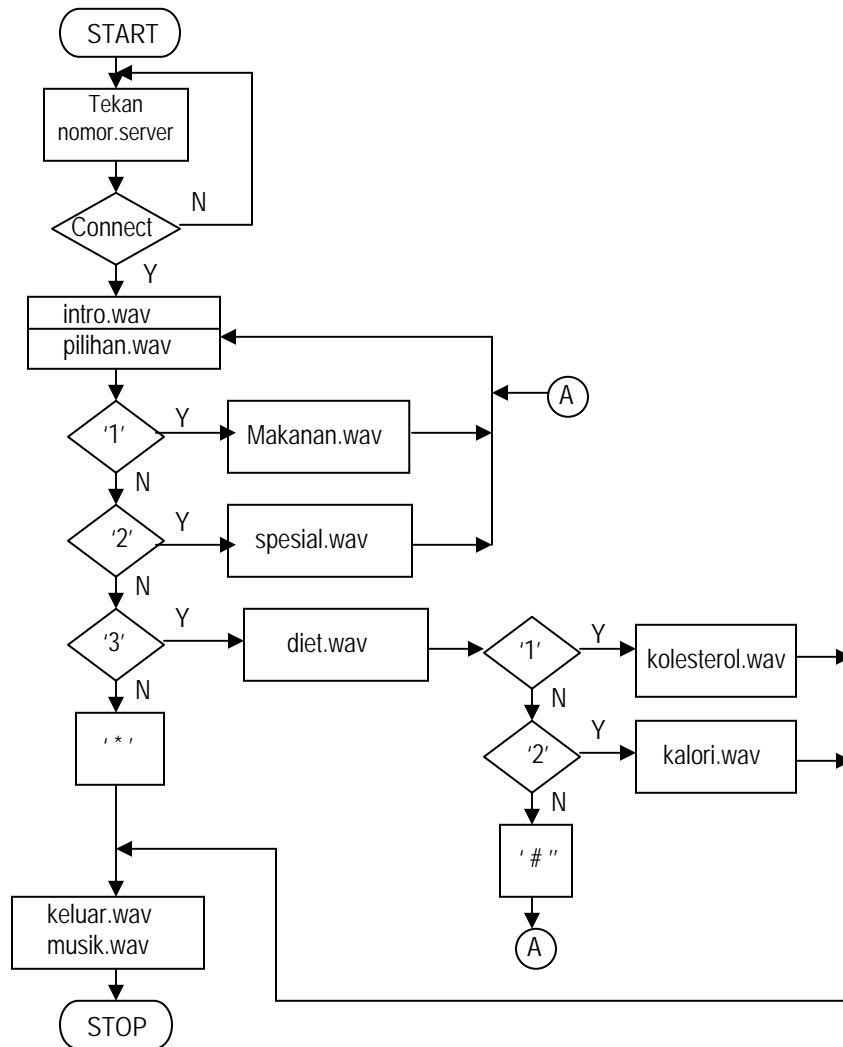
```

char DetectDigit(int chdev, char digit [10], int x);
void PlaySuara(int chdev, char fname[10]);

```

2.4.3. Prosedur Percobaan :

Aplikasi Layanan Informasi yang akan dibuat adalah tentang Menu yang disediakan oleh sebuah Restoran Cepat Saji bernama "Mak Nyuss". Diagram alir dari pelaksanaan Layanan ini ditunjukkan pada gambar 2.15.



Gambar 2.15. Diagram Alir Layanan Informasi Restoran Cepat Saji "Mak Nyuss"

Untuk membuat Aplikasi Layanan Informasi "Restoran Cepat Saji Mak Nyuss", lakukan langkah-langkah sebagai berikut :

1. Rekam file-file di bawah ini dengan menggunakan *Multi Thread Mode* pada Dialogic Sample Program, dengan WAVE Linear PCM 8 kHz. Jadikan dalam sebuah folder tersendiri.

Nama File	Kalimat
intro.wav	Selamat datang dalam Layanan Informasi Restoran Cepat Saji Makan Enak
pilihan.wav	Tekan 1 untuk informasi menu makanan Tekan 2 untuk informasi menu spesial hari ini Tekan 3 untuk informasi menu diet Tekan * untuk keluar
makanan.wav	Ayam saus Inggris Ayam Bakar Mentega Kepiting Saus Asam Manis Udang Cah Jamur Mie goreng spesial
spesial.wav	Menu spesial hari ini adalah Ikan Bakar Madu
diet.wav	Tekan 1 untuk informasi diet rendah kolesterol Tekan 2 untuk informasi diet rendah kalori Tekan # untuk kembali ke menu utama
kolesterol.wav	Menu rendah kolesterol yang disarankan adalah nasi putih tumis kangkung tempe bacem
kalori.wav	Menu rendah kalori yang disarankan adalah kentang stik daging salad buah
keluar.wav	Terima kasih telah mengakses layanan Informasi Restoran Cepat Saji Makan Enak

2. Membuat main program yang merupakan lanjutan dari langkah 8 pada subbab

2.4.2

Beberapa fungsi lain yang diperlukan dalam Main Program ini adalah :

dx_open() → membuka kanal

dx_sethook() → mengeset on hook

dx_wtring() → menunggu ring tone

```
int main(int argc, char* argv[])
{
    int chdev;
    char number[4],dig[10];

    if((chdev = dx_open("dxxxB1C1",NULL))== -1)
    {
        printf("Error open channel");
        exit(1);
    }
    printf("open channel succes\n"); //Set on hook//
    if(dx_sethook(chdev,DX_ONHOOK,EV_SYNC)== -1)
    {
        printf("Error on hook");
        exit(1);
    }
    printf("ONhook succes\n"); //Wait ring tone//
    if(dx_wtring(chdev,2,DX_OFFHOOK,-1)== -1)
    {
        printf("Error off hook");
        exit(1);
    }
}
```

```

}
printf("off hook succes\n");

PlaySuara(chdev, "intro.wav");
awal:

PlaySuara(chdev, "pilihan.wav");
number[0]=DetectDigit(chdev,dig,1);
number[1]='\0';

if(number[0]=='1')
{
    PlaySuara(chdev, "makanan.wav");
    goto awal;
}
else if(number[0]=='2')
{
    PlaySuara(chdev, "spesial.wav");
    goto awal;
}
else if(number[0]=='3')
{
    PlaySuara(chdev, "diet.wav");
    number[0]=DetectDigit(chdev,dig,1);
    number[1]='\0';
    if(number[0]=='1')
    {

        PlaySuara(chdev, "kolesterol.wav");
        goto keluar;
    }
    else if(number[0]=='2')
    {
        PlaySuara(chdev, "kalori.wav");
        goto keluar;
    }
    else
        goto awal;
}
else
{
    keluar:

    PlaySuara(chdev, "keluar.wav");
    PlaySuara(chdev, "musik.wav");
}}

```


2.6 Analisa

1. Apa yang terjadi jika program yang telah dibuat dirubah seperti program dibawah ini:

```
if((chdev = dx_open("dxxxB1C1",NULL))== -1)
{
    printf("Error open channel");
    exit(1);
}
printf("open channel succes\n"); //Set on hook//
if(dx_sethook(chdev,DX_ONHOOK,EV_SYNC)== -1)
{
    printf("Error on hook");
    exit(1);
}
printf("ONhook succes\n");
if(dx_wtring(chdev,2,DX_OFFHOOK,-1)== -1)
{
    printf("Error off hook");
    exit(1);
}
printf("off hook succes\n");
if(dx_sethook(chdev,DX_ONHOOK,EV_SYNC)== -1)
{
    printf("Error on hook");
    exit(1);
}
printf("ONhook succes\n");
```

2. Tuliskan perubahan program jika setelah pemilihan kolesterol.wav dan kalori.wav, maka user akan kembali pada pilihan.wav!

2.7. Pertanyaan & Tugas :

1. Buat sebuah program untuk mendeteksi digit, dengan jumlah dua digit setiap kali memasukkan digit.
2. Dengan jenis-jenis fungsi yang sudah dipelajari, buat aplikasi layanan informasi untuk sebuah agen penjualan ticket pesawat. Tentukan parameter apa saja yang bisa dijadikan layanan informasi. Buat flow chart-nya dan lengkapi dengan breakdown menu pilihan serta Tabel suara.yang diperlukan.