

参考文献：「パソコン用手作り外部インターフェース」、土井滋貴著、CQ 出版社

■ タイマとマルチメディア・タイマ

● タイマ

Visual Studio .NET 2003 を起動。

ファイル > 新規作成 > プロジェクト

Visual C++プロジェクト > Win32 コンソールプロジェクト

プロジェクト名: List204

場所: D:\¥takase¥制御

新しいソリューション名: List204

[OK]

概要 ・ コンソールアプリケーション

[完了]

次のようにソースを入力する。

```
// List204.cpp : コンソール アプリケーションのエントリ ポイントを定義します。
```

```
//
```

```
#include "stdafx.h"
```

```
#include <conio.h>
```

```
#include <windows.h>
```

```
#include <stdio.h>
```

```
VOID CALLBACK TimerProc(HWND, UINT, UINT, DWORD); // Timer プロシージャ
```

```
int main()
```

```
{
```

```
    MSG msg;
```

```
    UINT timerID; // Timer の ID
```

```
    // 1 秒ごとに TimerProc を呼び出す
```

```
    timerID = SetTimer(NULL, 0, 1000, (TIMERPROC)TimerProc);
```

```
    printf("タイマの ID %d¥n", timerID);
```

```
    // Timer には GetMessage() と DispatchMessage() が必要
```

```
    while(GetMessage(&msg, NULL, 0, 0)) {
```

```
        printf("メッセージ発生 番号 = 0x%x 時間 = %d ¥n", msg.message, msg.time);
```

```
        DispatchMessage(&msg);
```

```
        if (kbhit()) break;
```

```
    }
```

```
    printf("メッセージループを終了しました。¥n");
```

```
    // Timer の終了処理、(ウィンドウのハンドル, タイマの ID)
```

```
    KillTimer(NULL, timerID);
```

```
    getch();
```

```
    printf("何かキーを押すと終了します。¥n");
```

```
    getch();
```

```
}
```

```
VOID CALLBACK TimerProc(HWND, UINT, UINT, DWORD time) // Timer プロシージャ
{
    printf("何かキーを押すと終了します 経過時間 %d msec ¥n", time);
    Beep(440, 100);
    return;
}
```