

Fine Kernel ToolKit システム (Visual Studio 2005/2008 版) セットアップマニュアル

FineKernel Project
(2009 年 1/7 版)

1 Visual Studio のバージョンについて

Fine Kernel ToolKit システム (以下 FK) を Visual Studio(以下 VS) 上で利用するにあたっては、VS2005 と VS2008 ではほとんど同様の操作で開発を行うことができる。本マニュアルではこれらのバージョンについてまとめて解説を行っているが、メニューの項目名などが微妙に異なる場合がある。その場合は適宜読み替えること。また、VS2005 と VS2008 にはそれぞれ通常の製品版の他に、誰でも無償で使用できる Express Edition が存在するが、こちらのバージョンも問題なく使用できる。

最新の VS2008 の Express Edition は以下の URL からインストールが可能である。

<http://www.microsoft.com/japan/msdn/vstudio/Express/>

VS2005 の Express Edition を使用している場合は、別途プラットフォーム SDK をダウンロードし、インストールと設定を行う必要がある。その手順は Microsoft 社が公開している以下の記事を参照すること。

<http://www.microsoft.com/japan/msdn/vstudio/express/2005/visualc/usingpsdk/>

2 FK システムのセットアップ

VS2005/2008 のセットアップが完了したら、FK のセットアップを以下の手順で行う。

1. 以下の URL から、「FK ToolKit VisualStudio 用インストーラ」の項目にあるインストーラをダウンロードする。

<http://sourceforge.jp/projects/fktoolkit/releases/>

VS2005 版は「FK_VC05」、VS2008 版は「FK_VC08」から始まるファイル名になっているので、必要な方を選択すること。

2. インストーラを起動し、指示に従ってインストールする。標準の設定では“C:\FK_VC05(08)”という名称のフォルダにインストールされるので、異なる場所に移動した場合は以下の記述を適宜読み替えること。

3 Visual Studio の環境設定

次に、FK を用いた開発を行うために必要な VS の環境設定について述べる。本節の内容は、基本的には一回しておけばその後再設定をする必要はないが、再インストールの際や何らかの原因でこれらの設定が初期化されてしまう場合もあるので、その場合はもう一度再設定する必要がある。

1. VS を起動し、メニュー中の「ツール」から「オプション」を選択する。
2. ダイアログの左側にあるツリーメニューの中の「プロジェクトおよびソリューション」というツリーを展開する。

- 出てくるリストのうち「VC++ ディレクトリ」を選択する。
- ダイアログの右上にある「ディレクトリを表示するプロジェクト」リストボックスで、「インクルードファイル」を選択する。
- 「ディレクトリを表示するプロジェクト」リストボックスの下側にある 5 個のボタンのうち、左側から二番目のボタン (フォルダのアイコンが表示されているボタン) を押す。
- カーソルが下に現れるので、「C:\FK_VC05(08)」と入力して Enter キーを押す。
- 次に、「ディレクトリを表示するプロジェクト」リストボックスから今度は「ライブラリファイル」を選択する。
- 「ディレクトリを表示するプロジェクト」リストボックスの下側にある 5 個のボタンのうち、左側から二番目のボタン (フォルダのアイコンが表示されているボタン) を押す。
- カーソルが下に現れるので、「C:\FK_VC05(08)」と入力して Enter キーを押す。
- 次に、「ディレクトリを表示するプロジェクト」リストボックスから今度は「実行可能ファイル」を選択する。
- 「ディレクトリを表示するプロジェクト」リストボックスの下側にある 5 個のボタンのうち、左側から二番目のボタン (フォルダのアイコンが表示されているボタン) を押す。
- カーソルが下に現れるので、「C:\FK_VC05(08)\bin」と入力して Enter キーを押す。

4 プロジェクトの作成方法

VS では、アプリケーションを作成する際には必ず“プロジェクト”と呼ばれるものを作成しなければならない。開発には単なるプログラムソースだけでなく、様々な開発用の設定ファイルが必要となるが、“プロジェクト”というのはそれらを全て引くくるめたものだと考えれば良い。

通常、プロジェクトには様々な設定を行う必要があるが、FK を前述の手順でセットアップした場合は、「プロジェクトウィザード」を利用することで簡易に設定を済ませることができる。以下にその手順を述べる。

- VS を起動し、メニューから「ファイル」「新規作成」「プロジェクト」を選択する。
- 表示されたダイアログのうち、左側の「プロジェクトの種類」というツリーメニューで「Visual C++」というツリーを選択し、さらにその中の「FK」というツリーを選択する。
- 右側の「テンプレート」から「FK ToolKit Project」を選択する。
- 「プロジェクト名」というテキストボックス中に適当なプロジェクト名を入力する。
- 「場所」に適当な場所を設定しておく。この場所にプロジェクトのフォルダを配置し、プログラムソースを保管することになるので、きちんと場所を把握しておく必要がある。
- 上記の全ての入力を終えたら「OK」を押す。すると「FK ToolKit プロジェクトウィザードによろこそ」というダイアログが開くが、通常は設定を変更する必要はないので、そのまま「完了」ボタンを押す。

この手順で、“プロジェクト”の作成と設定が完了する。実際にはプロジェクト自体に加えて、その上位に位置する“ソリューション”と呼ばれるファイルも同時に作成される。これは eclipse でいうところの“ワークスペース”のようなもので、複数のプロジェクトを 1 つのソリューションで管理するという働きを持っている。

5 コンパイルと実行

コンパイルは、ソリューションエクスプローラー内にあるプロジェクト名を選択したのち、メニュー中の「ビルド」「[プロジェクト名]のビルド」で行える。エラーが発生した場合はメッセージを確認し、ソースファイルにおけるエラーの場合はそれを修正する必要がある。ヘッダファイルのインクルードや、リンクの時点でエラーが発生している場合は、これまでの設定に何らかの不備がある可能性が高いため、もう一度手順を確認すること。

コンパイルが正常に終了した場合、設定ファイル (.sln および .vcproj) 中にある .exe フォルダ中に実行ファイルが生成される。実行するには、メニューの「デバッグ」「デバッグなしで開始」を選択する。この際カレントフォルダは設定ファイルの存在しているフォルダになるため、ファイルを読み込んで処理を行うプログラムを作成した場合は、パスの設定に注意が必要である。

6 プロジェクトの保存とロード

プロジェクト全体の状態を保存するには、メニューの「ファイル」「すべてを保存」を実行する。これにより、各プロジェクトの情報は「.vcproj」及び「.sln」を拡張子に持つファイルに保存される。

保存したプロジェクトをロードするには、「.sln」を拡張子に持つファイルをダブルクリックするか、VS を起動後メニュー中の「ファイル」「ソリューションを開く」を選択し、「.sln」を拡張子に持つファイルを選択すればよい。

7 補足

以下はプロジェクトの設定に関する補足である。前節までで一通りの解説は済んでいるので、本節は必要に応じて参照すること。

7.1 デバッグモードとリリースモードについて

開発を行っていくうち、複雑なバグなどが発生した場合にはデバッグモードでビルドして、問題を検証するのが有効である。プロジェクトウィザードで作成したプロジェクトを使用している場合は、ツールバー中のプルダウンメニューから「Release」と「Debug」を変更することが可能なので、状況に応じて切り替えるとよい。

7.2 Direct3D 版について

FK は OpenGL をベースとして動作しているが、OpenGL が正常に動作しない一部の Windows 環境では、Direct3D を使用して動作させることができる。VS 版のパッケージには Direct3D 版の FK も収録してあるため、プロジェクトの設定を切り替えるだけで Direct3D を使用した動作モードに切り替えることが可能である。具体的には以下の設定を変更する。

- Microsoft 社のサイトから DirectX SDK をダウンロードする。動作確認済みの SDK のバージョンは、「Mar 2008」「Nov 2008」である。
- DirectX SDK をインストールし、ヘッダやライブラリのパスを VS に登録する。
- 「プロジェクトのプロパティ 構成プロパティ C/C++ プリプロセッサ プリプロセッサの定義」に「FK_D3D」を追加する。
- 「プロジェクトのプロパティ 構成プロパティ リンカ 入力 追加の依存ファイル」に入力するファイル名を、「Lib_D3D_Release(Debug).txt」というファイル内に記述されているものに置き換える。

DirectX SDK のインストールや設定に関しては、Web 上の資料や書籍などを参照すること。プロジェクトの設定に関しては、ウィザードで Direct3D に対応した設定に切り替えることも可能なので、適宜利用するとよい。

なお、Direct3D 版はどうしても OpenGL が動作しない場合にのみ使用することを前提としており、動作の再現性や安定性に若干の問題が見られる。このため、極力 OpenGL 版を利用することを推奨する。

7.3 プロジェクトの設定項目について

通常プロジェクトは、ウィザードを利用することで簡易に作成できるが、他のライブラリと組み合わせて使用する場合には、個々の設定項目にも熟知していなければならない。

以下に、FK システム側で変更しているプロジェクトの設定項目を列挙する。自力でプロジェクトを作成する場合や、他のライブラリとの兼ね合いで設定を変更する際の参考にして欲しい。

- 構成プロパティ

- 全般

- * 文字セット : 「マルチバイト文字セットを使用する」
- * 共通言語ランタイム サポート : 「共通言語ランタイム サポートを使用しない」

- C/C++

- プリプロセッサ

- * プリプロセッサの定義 : [Direct3D 版使用時のみ] 「FK_D3D」を追記

- コード生成

- * ランタイムライブラリ : [Release] 「マルチスレッド (/MT)」, [Debug] 「マルチスレッド デバッグ (/MTd)」

- リンカ

- 入力

- * 追加の依存ファイル : 項目をクリックすると右側にボタンが現れるのでそれを押し、表示されるダイアログのテキストエリア中に次の文字列を入力する。

```
TinyFK.lib FK2_base.lib FK2_ftkWin.lib FK2_ftkErr.lib  
ftkgl.lib ftkjpeg.lib ftkpng.lib ftkimages.lib ftkforms.lib ftk.lib ftkz.lib  
imdkcv.lib freetype2MT.lib opengl32.lib glu32.lib imm32.lib winmm.lib wsock32.lib comctl32.lib
```

これはあまりに長すぎるので、同一内容が最初にセットアップした FK_VC05(08) フォルダ内の「Lib_Release.txt」(デバッグモードの場合は「Lib_Debug.txt」) というテキストファイルに記述されているので、これをコピー & ペーストして設定することを推奨する。

- * 特定のライブラリの無視 : [Release] 空欄, [Debug] 「LIBCMT」

- システム

- * サブシステム : コンソールウィンドウを表示させたい場合は「コンソール」を、表示させたくない場合は「Windows」を選択する。