

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
(2010 年 4/23 版)

1 Visual Studio のバージョンについて

Fine Kernel ToolKit システム (以下 FK) は Visual Studio(以下 VS) 2005 と 2008 上において、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、VS には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、VS2008 においてはこちらのバージョンも問題なく使用できる。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/>

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ファイル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、VS2005 版は“C:\FK_VC05”、VS2008 版は“C:\FK_VC08”という名称のフォルダにインストールされる。

セットアップ完了後に、VS への設定情報追加が行われる。環境によってはこの処理がうまく完了しない場合があるため、その場合は手動での設定が必要となる。詳細は本マニュアルの補足の項を参照すること。

3 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”を作成する必要がある。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 のセットアップが正常に完了した場合は、「プロジェクトウィザード」を利用することで簡単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。

1. 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。
2. 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。
3. 右側の「テンプレート」から「FK ToolKit Project」を選択する。

4. 「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。
5. 「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。また「ソリューションのディレクトリを作成」にチェックが入っていると、フォルダ階層が不必要に深くなるため、チェックを外しておくことを推奨する。
6. 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るこそ」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。
7. 作成したプロジェクトが開かれたら、ツールバー内の「Debug」という文字列が選択されているプルダウンメニューを「Release」に変更する。

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。VS で作成したプロジェクトを開く場合は、ソリューションを通じて行う。

4 コンパイル(ビルド)

ソースファイルが作成できたらコンパイルを行い、実行ファイルを生成する。VS においては、コンパイルを初めとした一連の作業のことを“ビルド”と呼ぶ。ビルドを行うには、メニュー中の「ビルド」ソリューションのビルド」を選択する。

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ので、もう一度これまでの手順を確認すること。

5 実行

ビルドが正常に終了した場合、設定ファイル (.vcproj) と同じフォルダ内にある .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プロジェクト設定ファイル (.vcproj) 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。VS を通さずプログラムを実行する場合は、実行ファイルをプロジェクトフォルダにコピーしてから実行するとよい。

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、全てのソースファイルとプロジェクトの情報が保存される。

保存したプロジェクトをロードするには、「.sln」を拡張子を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」を拡張子を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 デバッグビルドとリリースビルドについて

FK を使用したプログラムは、通常リリースビルドで実行することを推奨する。初期状態ではデバッグビルドになっているが、この状態ではプログラムの処理速度が大幅に低下するため、上記の手順においてはリリースビルドへの切り替えを指示している。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。

7.3 Visual Studio の環境設定

本節の内容は、今回のバージョンからはセットアップと同時に行うようになったため、基本的には再設定をする必要はない手順である。何らかの原因で設定の変更がうまく行かなかったり、これらの設定が初期化されてしまった場合には、以下の手順で再設定すること。なお本マニュアルではインストール先が C:\FK_VC08 であるとして記述してあるが、VS のバージョンや自分で設定したインストール先に応じて読み替えること。

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し、出てくるリストのうち「VC++ ディレクトリ」を選択する。
3. 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。
4. 「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。
5. カーソルが下に現れるので、「C:\FK_VC08」と入力して Enter キーを押す。
6. 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。

7. 「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。
8. カーソルが下に現れるので、「C:\FK_VC08」と入力して Enter キーを押す。
9. 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。
10. 「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。
11. カーソルが下に現れるので、「C:\FK_VC08\bin」と入力して Enter キーを押す。

7.4 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、他のシステムと共存させる場合や、ウィザードが利用できない場合は、以下の設定項目を参考にしてプロジェクトの設定を調整すること。

- 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
ftkggl.lib ftkjpeg.lib ftkpng.lib ftkimages.lib ftkforms.lib ftk.lib ftkz.lib
imdkev.lib freetype2MT.lib opengl32.lib glu32.lib imm32.lib winmm.lib wsock32.lib comctl32.lib
```

同一内容が最初にインストール先のフォルダ内の「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してある。

- * 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。