

# StkThreadGui

---

## 1. StkThreadGuiについて

### 1.1 StkThreadGui概要

### 1.2 システム要件

### 1.3 GUIの構成

### 1.4 API詳細

# 1. StkThreadGuiについて

---

## 1.1 StkThreadGui概要

---

StkThreadGuiはスレッド(StkThread)管理用GUIを提供する静的ライブラリで、C言語用APIを提供します。StkThreadGuiは内部的にStkThread.Libの機能呼び出します。また、StkThreadGuiの機能は、場合によりStkThread.Libの機能の事前設定を前提とするため、StkThreadGui利用プログラム側では、StkThreadGuiの機能だけでなく、StkThread.Libの機能呼び出す必要があります。

StkThreadGuiは以下のファイルから構成されます。

stkthreadgui.lib	StkThreadGuiの静的ライブラリです。
stkthreadgui.h	StkThreadGui利用プログラムのためのヘッダファイルです。

## 1.2 システム要件

---

StkThreadGuiは以下の環境で動作します。

オペレーティングシステム:

- Windows Vista Business/Ultimate SP2
- Windows 7 Professional/Ultimate SP1
- Windows 8 Pro
- Windows 8.1 Pro
- Windows Server 2008 Standard SP2
- Windows Server 2008 R2 Standard SP1
- Windows Server 2012 Standard
- Windows Server 2012 R2 Standard

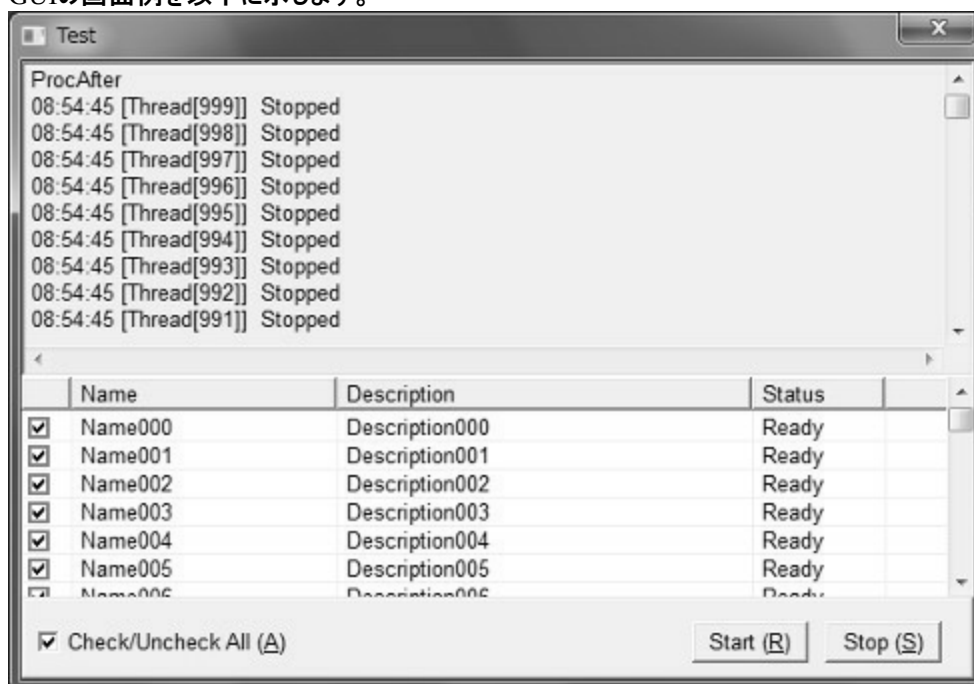
前提となるソフトウェア:

- Microsoft Visual Studio 2008 (Visual C++)

## 1.3 GUIの構成

---

GUIの画面例を以下に示します。



GUIはStkThreadGuiのShowStkThreadControllerメソッドを呼び出すことにより起動します。GUIで表示している情報は以下の2種類となります。

- ログ情報。StkThreadGuiのユーザープログラムで任意のログを出力することができます。

- スレッド一覧。StkThreadGuiのAddStkThreadForGuiメソッドを呼び出して追加したスレッドの一覧が表示されます。

スレッド一覧から操作したいスレッドを選択し、スレッドの開始／停止操作を行うことができます。

## 1.4 API詳細

### 1.4.1 int ShowStkThreadController(HWND hWnd, int ParentIconId)

スレッド管理用GUIを表示します。

hWnd [入力]	親ウィンドウのハンドル
ParentIconId [入力]	親ウィンドウのアイコンID
戻り値	常に0を返す

### 1.4.2 void AddStkThreadForGui(int Id, TCHAR\* Name, TCHAR\* Description, void\* Init, void\* Final, void\* Main, void\* Start, void\* Stop)

StkThreadをStkThreadGuiの管理リソースとして追加します。

スレッド管理用GUIを使用してスレッド(StkThread)を管理する場合、StkThread.LibのAddStkThreadではなく、本メソッドを使用する必要があります。

AddStkThreadForGuiメソッドは内部的にStkThread.LibのAddStkThreadメソッドを呼び出します。詳細はStkThread.LibのAPI仕様を参照してください。

Id [入力]	StkThread ID (StkThreadが内部的に管理するスレッドのIDであり一般的なスレッドIDではない)
Name [入力]	スレッドの名前
Description [入力]	スレッドの詳細説明
Init [入力]	ElemStkThreadInit(コールバックメソッド)へのポインタ。NULLを指定した場合コールバックは無効。
Final [入力]	ElemStkThreadFinal(コールバックメソッド)へのポインタ。NULLを指定した場合コールバックは無効。
Main [入力]	ElemStkThreadMain(コールバックメソッド)へのポインタ。NULLを指定した場合コールバックは無効。
Start [入力]	ElemStkThreadStart(コールバックメソッド)へのポインタ。NULLを指定した場合コールバックは無効。
Stop [入力]	ElemStkThreadStop(コールバックメソッド)へのポインタ。NULLを指定した場合コールバックは無効。

### 1.4.3 void DeleteStkThreadForGui(int Id)

StkThreadをStkThreadGuiの管理リソースから削除します。

スレッド管理用GUIを使用してスレッド(StkThread)を管理する場合、StkThread.LibのDeleteStkThreadではなく、本メソッドを使用する必要があります。

DeleteStkThreadForGuiメソッドは内部的にStkThread.LibのDeleteStkThreadメソッドを呼び出します。詳細はStkThread.LibのAPI仕様を参照してください。

Id [入力]	削除対象のStkThread ID (StkThreadが内部的に管理するスレッドのIDであり一般的なスレッドIDではない)
---------	--

### 1.4.4 void StartAllStkThreads()

AddStkThreadForGuiで登録されたすべてのスレッドを実行します。

StartAllStkThreadsメソッドは内部的にStkThread.LibのStartSpecifiedStkThreadsメソッドを呼び出します。

### 1.4.5 void StopAllStkThreads()

AddStkThreadForGuiで登録されたすべてのスレッドを停止します。  
StopAllStkThreadsメソッドは内部的にStkThread.LibのStopSpecifiedStkThreadsメソッドを呼び出します。

#### 1.4.6 void ClearStkThreadLog()

ログ出力をクリアします。

#### 1.4.7 void AddStkThreadLog(TCHAR\* Msg)

指定された文字列をログとして出力します。

Msg [入力]	ログとして出力する文字列。
----------	---------------

#### 1.4.8 void AddStkThreadLogWithThreadInfo(TCHAR\* Name, TCHAR\* Msg)

指定された文字列をログとして出力します。  
出力される形式は, "時:分:秒 [Name] Msg"となります。

Name [入力]	ログ出力元の名称。通常スレッド名を指定します。
Msg [入力]	ログとして出力する文字列。

#### 1.4.9 void GetStkThreadLog(TCHAR\* Out, int Length)

現在出力されているログを取得します。

Out [出力]	取得した文字列が格納される領域。
Length [入力]	取得した文字列が格納される領域のサイズ(WORD数;TCHARの要素数)。

#### 1.4.10 int GetStkThreadLogUpdateVersion()

ログ出力の更新バージョンを取得します。  
ログ出力の更新バージョンは, AddStkThreadLogメソッドおよびAddStkThreadLogWithThreadInfoメソッドを呼び出す度に増加します。また, ClearStkThreadLogメソッドを呼び出すと0にクリアされます。

戻り値	ログ出力の更新バージョン
-----	--------------

#### 1.4.11 int GetStkThreadLogMaxSize()

ログ出力領域の最大サイズ(WORD数;TCHARの要素数)を取得します。

戻り値	ログ出力領域の最大サイズ(WORD数;TCHARの要素数)
-----	-------------------------------

#### 1.4.12 int GetStkThreadLogSize()

ログ出力領域に現在格納されている文字列のサイズ(WORD数;TCHARの要素数)を取得します。

戻り値	ログ出力領域に現在格納されている文字列のサイズ(WORD数;TCHARの要素数)
-----	--

#### 1.4.13 void ChangeStkThreadLogSize(int Size)

ログ出力領域のサイズ(WORD数;TCHARの要素数)を変更します。  
デフォルトは10,000文字です。

Size [入力]	ログ出力領域のサイズ(WORD数;TCHARの要素数)
-----------	-----------------------------