Subject: CreateSymbolicLink doesn't compile using MinGW
Posted by _dka_ on Tue, 26 Dec 2017 10:53:55 GMT
View Forum Message <> Reply to Message

Hello all,

Sorry, i use google translator.
I used upp-mingw (last nightly builds and stable upp-mingw-11540).
I want to create a symbolic link to a folder.
I tried to create console applications in projects of different types.

My source (Core console project):

```
#include <Core/Core.h>
using namespace Upp;
CONSOLE_APP_MAIN
{
 CreateSymbolicLink("","",1);
}
```

Compiler error:

```
MyApps_MkLinks.cpp (7): error: 'CreateSymbolicLink' was not declared in this scope
 (): CreateSymbolicLink("","",1);
MyApps_MkLinks.cpp (7): note: suggested alternative: 'CreateHardLink'
 (): CreateSymbolicLink("","",1);
 (): CreateHardLink
```

So far, I've solved this problem by adding lines from the winbase.h from mingw:

```
#include <Core/Core.h>
using namespace Upp;

#define SYMBOLIC_LINK_FLAG_DIRECTORY (0x1)
WINBASEAPI BOOLEAN APIENTRY CreateSymbolicLinkA ( LPCSTR lpSymlinkFileName,
LPCSTR lpTargetFileName, DWORD dwFlags );
WINBASEAPI BOOLEAN APIENTRY CreateSymbolicLinkW ( LPCWSTR lpSymlinkFileName,
LPCWSTR lpTargetFileName, DWORD dwFlags );
#define CreateSymbolicLink __MINGW_NAME_AW(CreateSymbolicLink)

CONSOLE_APP_MAIN
{
 CreateSymbolicLink("","",1);
}
```

but now there is a link error:

```
 (): Linking has failed
```

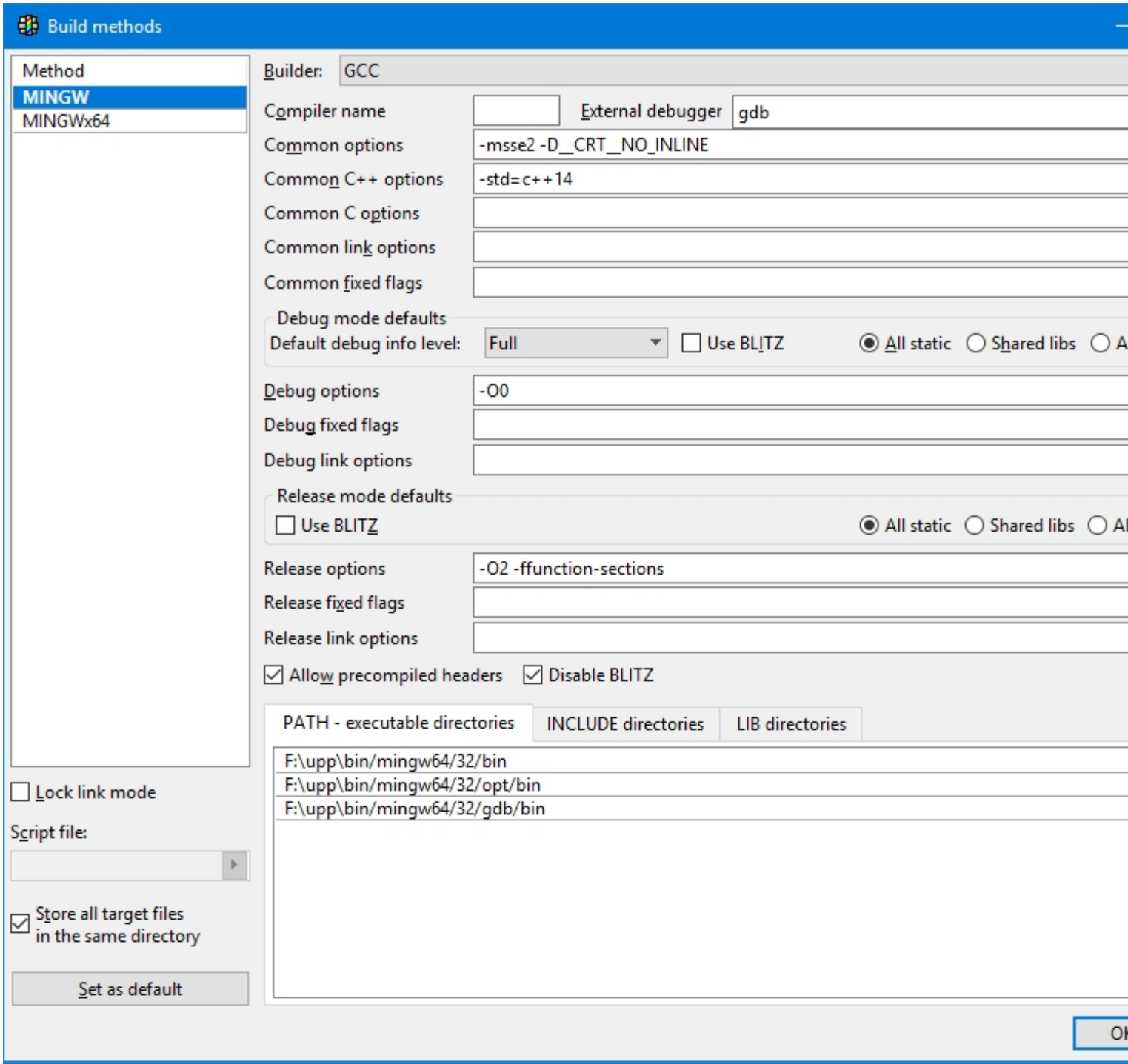
(): MyApps_MkLinks/MINGW.Debug.Debug_Full.Main.Noblitz\MyApps_MkLinks.o: In function `Z14ConsoleMainFn_
v':
MyApps_MkLinks/MyApps_MkLinks.cpp (40): undefined reference to `_imp___Z19CreateSymbolicLinkAPKcS0_m@12'
(): collect2.exe: error: ld returned 1 exit status

Build methods settings:

Toggle Spoiler

File Attachments

1) Build methotds.jpg, downloaded 1323 times


Build methods
Method
MINGW
MINGWx64
Builder: GCC
Compiler name
External debugger gdb
Common options -msse2 -D__CRT__NO_INLINE
Common C++ options -std=c++14
Common C options
Common link options
Common fixed flags
Debug mode defaults
Default debug info level: Full
Use BLITZ
All static
Shared libs
Debug options -O0
Debug fixed flags
Debug link options
Release mode defaults
Use BLITZ
All static
Shared libs
Release options -O2 -ffunction-sections
Release fixed flags
Release link options
Allow precompiled headers
Disable BLITZ
PATH - executable directories
INCLUDE directories
LIB directories
F:\upp\bin/mingw64/32/bin
F:\upp\bin/mingw64/32/opt/bin
F:\upp\bin/mingw64/32/gdb/bin
Lock link mode
Script file:
Store all target files in the same directory
Set as default
Build methods
Method
MINGW
MINGWx64
Builder: GCC
Compiler name
External debugger gdb
Common options -msse2 -D__CRT__NO_INLINE
Common C++ options -std=c++14
Common C options
Common link options
Common fixed flags
Debug mode defaults
Default debug info level: Full
Use BLITZ
All static
Shared libs
Debug options -O0

Subject: Re: CreateSymbolicLink doesn't compile using MinGW
Posted by Klugier on Tue, 26 Dec 2017 16:57:22 GMT
View Forum Message <> Reply to Message

Hello,

The compilation error occurred, because you are trying to use WinAPI without including header appropriate header. In your case it is winbase.h. Please notice that by including Core.h system header is never inherit. Core library must work in multi-platform environment, so doing such a thing would be bad.

Moreover when you decide to use CreateSymbolicLink your application will not compile on other platforms such as Linux without modification or writing ifdef blocks.

As far as I know U++ doesn't offer support for SymLink, but i might be wrong.

Sincerely,
Klugier

Subject: Re: CreateSymbolicLink doesn't compile using MinGW
Posted by _dka_ on Tue, 26 Dec 2017 17:13:10 GMT
View Forum Message <> Reply to Message

Thank you)

I found that in the Core.h file winbase.h is included under the condition
```
#ifdef COMPILER_MSC
    #include <winbase.h>
```
The question is closed.