Call MATLAB Functions from C/C++ Program

Pradeep Chatterjee
pchats@ieee.org

The step by step procedure is as follows
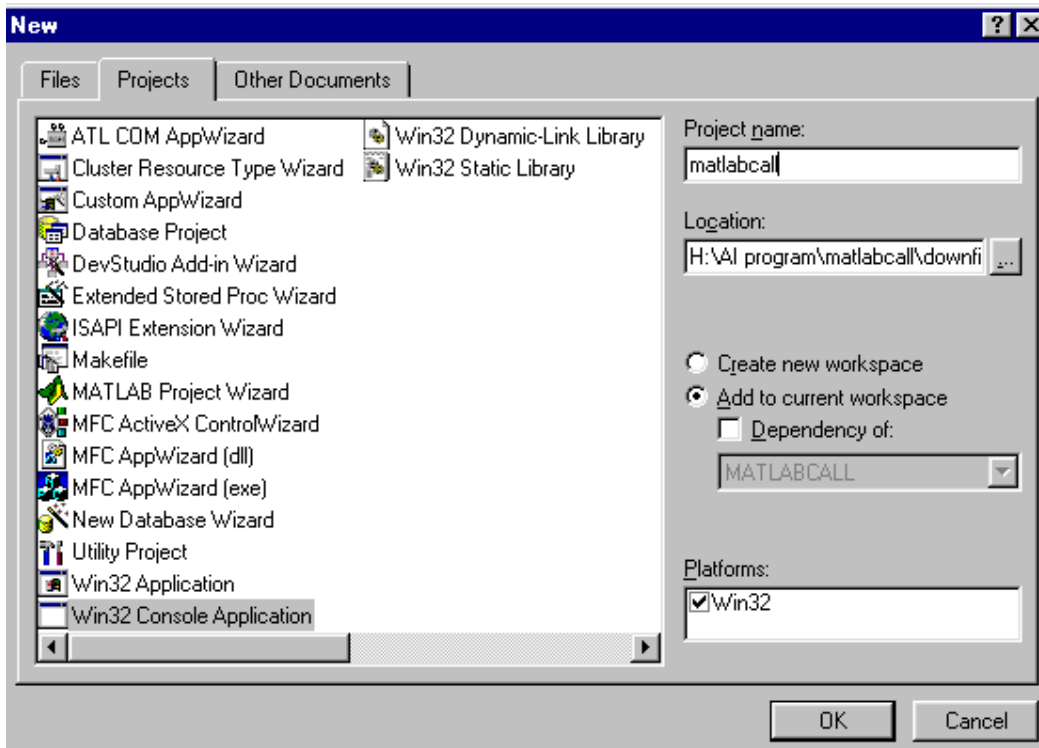
Software Used

Microsoft Visual C++ ver 6.0

MATLAB R12 Version 6

Step 1

Start Microsoft Visual c++, choose New Project, choose a win32 console application, (make sure the box in right hand bottom side) win32 is clicked. Name the project matlabcall.

Step2

 Next we need to add paths to the MATLAB Engine for the header and the library files. For that click on tools -> options, choose directory. For the include files add new path i.e. where the MATLAB include directory is stored. It is usually in the MATLAB folder-> extern-> include and

MATLAB folder ->extern ->lib ->win32 ->Microsoft ->msvc60. The window should look like this
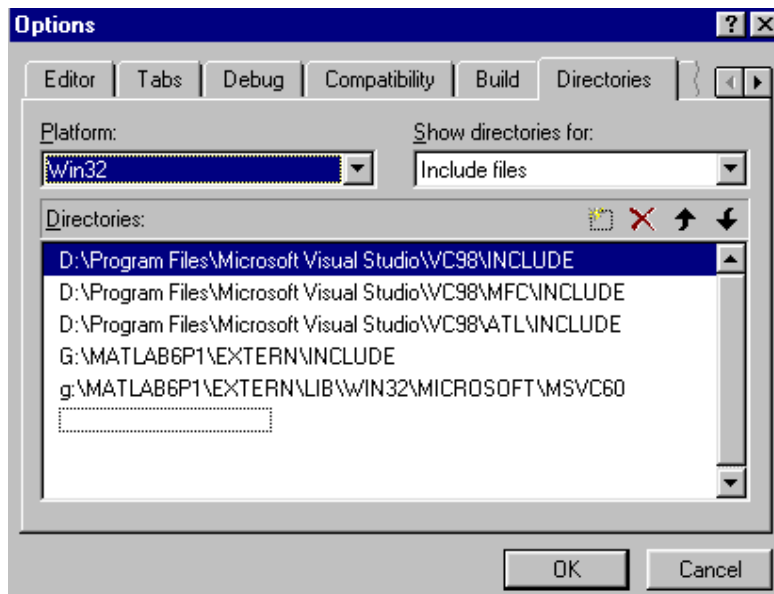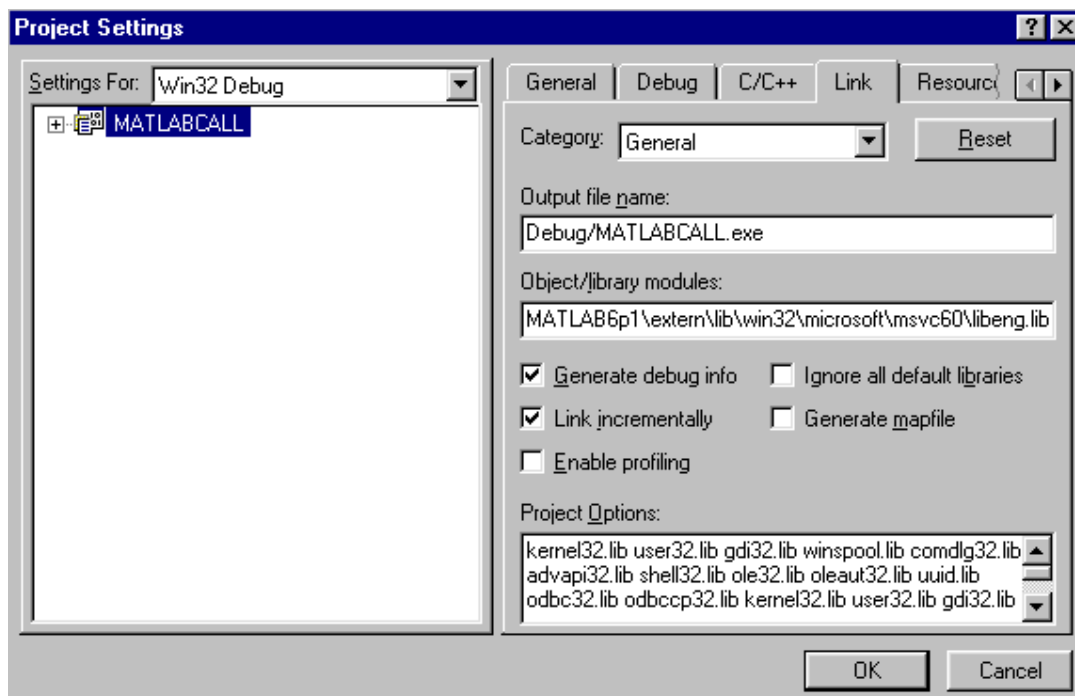

Step 3

Next click on Project -> Settings. In link tab and under the object/library modules add

 Libmx.lib libmex.lib libeng.lib

The window should look like this

Step 4

Choose File-> new -> C++ source file. Name the file matlabcall

The code is all in BLUE.

    a)   The header files need to written first

```
#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include"engine.h"     This file is needed to call the subroutines needed to call the MATLAB.
```

Next we need to declare the variables that we are going to use.

```
Engine *ep;
```

Step 5)

Write the code in C++ file as below

```
void main()

{
Engine *ep;


ep = engOpen("");

engEvalString(ep,"x = (1:1:10);" );

engEvalString(ep, "y = x.^2;");

engEvalString(ep, "plot(x,y);"   );

engClose(ep);

}
```

To start the MATLAB Engine . we typed in

```
ep = engOpen("");
```

This command initializes the MATLAB engine and returns ep to it.

To close the engine we typed.

engClose(ep);

Save the program, compile and build it.

The author acknowledges the work of Bharat Shah, which forms the basic framework of the method described above. Some changes have been incorporated in the above-mentioned process.

****