

# **MOBITEK**® SIM TOOL KIT

## APPLICATION PROGRAMMING INTERFACE

### version 6.0

## Manual for Software Developer

---

## Table of Contents

1. REVISION HISTORY.....	3
2. COPYRIGHT.....	3
3. TRADEMARKS.....	3
4. DISCLAIMER.....	3
5. ACKNOWLEDGEMENT.....	4
6. DEFINITION.....	4
7. INTRODUCTION.....	4
8. FEATURES.....	4
9. SYSTEM REQUIREMENTS.....	5
10. BENEFITS.....	5
11. LIMITATIONS.....	5
12. OVERVIEW OF SOFTWARE ARCHITECTURE.....	6
13. FLOW CHART.....	7
14. INSTALLATION GUIDE.....	8
15. STK API METHODS.....	9
(1) ActivateSTK() as Boolean.....	9
(2) CloseSTK() as Boolean.....	9
(3) DeactivateSTK () as Boolean.....	9
(4) MainMenuDisplay (MainMenuName As String, MainMenuList As String) as Boolean.....	9
(5) MainMenuSelect(ByVal ItemSelect As Integer) As Boolean.....	10
(6) SubMenuDisplay(SubMenuName As String, SubMenuList As String) As Boolean.....	10
(7) SubMenuSelect(ByVal ItemSelect As Integer) As Boolean.....	10
(8) InputRequestDisplay() As String.....	10
(9) InputSubmit(ByVal UserInputSubmit As String) As Boolean.....	10
(10) ResponseDisplay() As String.....	10
(11) SMSSentDisplay() As String.....	11
(12) STKStatusGet() as STKStatus.....	11
16. STK API FLOW CHART.....	12
17. SAMPLE CODE.....	13
18. DEPLOYMENT.....	13
19. SUPPORT.....	13
20. RELEASE NOTE.....	13
21. FAQ.....	15
Q: In the middle of my process, I want to reset/reboot the STK modem, what is the correct flow?.....	15
Q: Why do I receive a pop-up message – “Invalid GSM Modem connected!”?.....	15
Q: Why do I get “SendSMS() = False” and “ReadSMS() = False” even though “Modem.Init() = 1”?.....	15
Q: Why do I always get a “Fail” whenever I call any methods or functions?.....	15
Q: Why does the STK modem often hang (not responsive)?.....	16
Q: Can the STK API supports multiple STK modems?.....	16
22. APPENDIX 1: Referencing COM in .Net.....	17
23. APPENDIX 2: How to Use STK API in JAVA.....	18
System Requirements.....	18
Set-up Eclipse.....	18
Java Sample Code.....	23
24. APPENDIX 3: How to Decode Alpha Field Displayed in STK Menu.....	26

## 1. REVISION HISTORY

EDITION	ISSUED DATE	REMARK
1st	1st of April, 2006	Draft release
2nd	8th of April, 2006	Added SMSSentDisplay()
3rd	15th of June, 2006	Release version 1.3
4th	24th of July, 2006	Remove SMS API from manual
5th	27th of September, 2006	Release version 1.4
6th	26th of November, 2006	APPENDIX 1: Using COM Components from Visual Studio .Net Directly added
7th	30th of July, 2007	Minor modification
8th	3rd of September, 2007	Version 2.0 released
9th	31st of October, 2007	Version 2.1 released
10th	20th of December, 2007	Version 2.3 released
11th	27th of February, 2008	FAQ section added
12th	15th of March, 2008	Version 3.0 released
13th	3rd of April, 2008	Version 3.1 released
14th	6th of June, 2008	Version 4.0 released
15th	29th of July, 2008	Minor addition to section of "warranty and support"
16th	4th of December, 2009	Version 5.0 released
17th	5th of April, 2010	FAQ updated
18 <sup>th</sup>	12 <sup>th</sup> of August, 2013	Version 6.0 (beta) is released
19 <sup>th</sup>	19 <sup>th</sup> of December, 2013	"15. STK API FLOW CHART" amended
20 <sup>th</sup>	8 <sup>th</sup> of August, 2014	<ul style="list-style-type: none"> <li>• Version 6.1 is released</li> <li>• Version 6.2.1 is released</li> <li>• "RELEASE NOTE"</li> <li>• "APPENDIX 2: How to Use STK API in JAVA" added by Faez</li> <li>• "APPENDIX 3: How to Decode Alpha Field Displayed in STK Menu" added by Faez</li> </ul>

## 2. COPYRIGHT

© 2009 - 2014 MOBITEK System Sdn. Bhd. All rights reserved.

No part of this document may be reproduced, distributed, stored in a retrieval system or translated into any language, in any form or by any means, electronic, mechanical, magnetic, optical, photocopying, manual or otherwise, without the prior written permission of MOBITEK System Sdn. Bhd.

## 3. TRADEMARKS

**MOBITEK®** is a registered trademark owns by MOBITEK System Sdn. Bhd.

Product names, logos, brands and other trademarks referred in this document are the property of their respective trademark holders and are used only to directly describe the products being provided.

## 4. DISCLAIMER

MOBITEK makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose.

Furthermore, MOBITEK reserves the right to revise this publication and to make changes from time to time in

the contents hereof without obligation to notify any person of such revision or changes.

## 5. ACKNOWLEDGEMENT

We like to extend our appreciation to the following persons for their contribution in revising this manual:-

1. Mr. Muhamad Faezudin bin Hairuddin, UNIVERSITY SAINS MALAYSIA

## 6. DEFINITION

1. STK – SIM Tool Kit
2. STK MODEM – a GSM modem that is certified to use STK API.
3. SIM APPLICATION – an application that resides in the SIM card; usually the application is a “menu”. Example of SIM applications are top-up of prepaid account, mobile banking, information on demand, etc.
4. SOFTWARE APPLICATION – a Windows programme residing in computer.
5. SOFTWARE PROGRAMME – same meaning as “SOFTWARE APPLICATION”.

## 7. INTRODUCTION

**API TYPE:** ActiveX DLL component (Component Object Model) for Windows

**ActiveX Name:** MobitekSTK6.dll

**Version:** 6.2

**The API contains 5 classes:**

1. Modem
2. SIMToolKit
3. SMS
4. USSD
5. Phonebook

**Scope of Manual:** This manual only covers the **SIMToolKit** class. For information on the **Modem, SMS, USSD and Phonebook** classes, please refer to the manual, “SMS API v9 Manual”.

**Pre-requisite:** System integrator, and software developer must possess programming skill, and knowledge in making reference to DLL file.

## 8. FEATURES

1. Software developer can develop their own user interface or software application that can access and interact with the *SIM application* residing in the SIM card.
2. Software application can obtain menu items from *SIM application*.
3. Software application can select menu items.
4. Software application can obtain response from *SIM application*.
5. Software application can submit data to *SIM application*.
6. Software application can instruct *SIM application* to send out a SMS.
7. Software application can obtain status of *SIM application*.

8. Software application can stop a session of *SIM application*.

## 9. SYSTEM REQUIREMENTS

1. Any programming language that can use ActiveX DLL or COM such as:
  - a) Visual Basic
  - b) Visual Basic .Net
  - c) Visual C++
  - d) Visual Studio .Net
  - e) Cold Fusion
  - f) Any programming language that can call *MobitekSTK6.dll*
2. Operating System: Windows 98, 2000, XP, Server 2003, Server 2008, Vista, 7

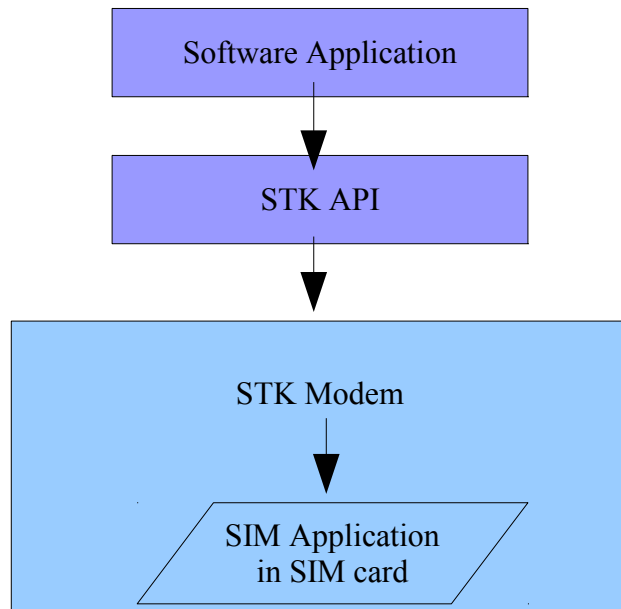
## 10. BENEFITS

1. To automate a top-up or reload process such as *Maxis e-load*, and *Digi Flexi e-load*.
2. To integrate multi-level marketing system with *SIM application*.
3. To integrate customer relationship management system with *SIM application*.
4. To integrate e-commerce system with *SIM application*.

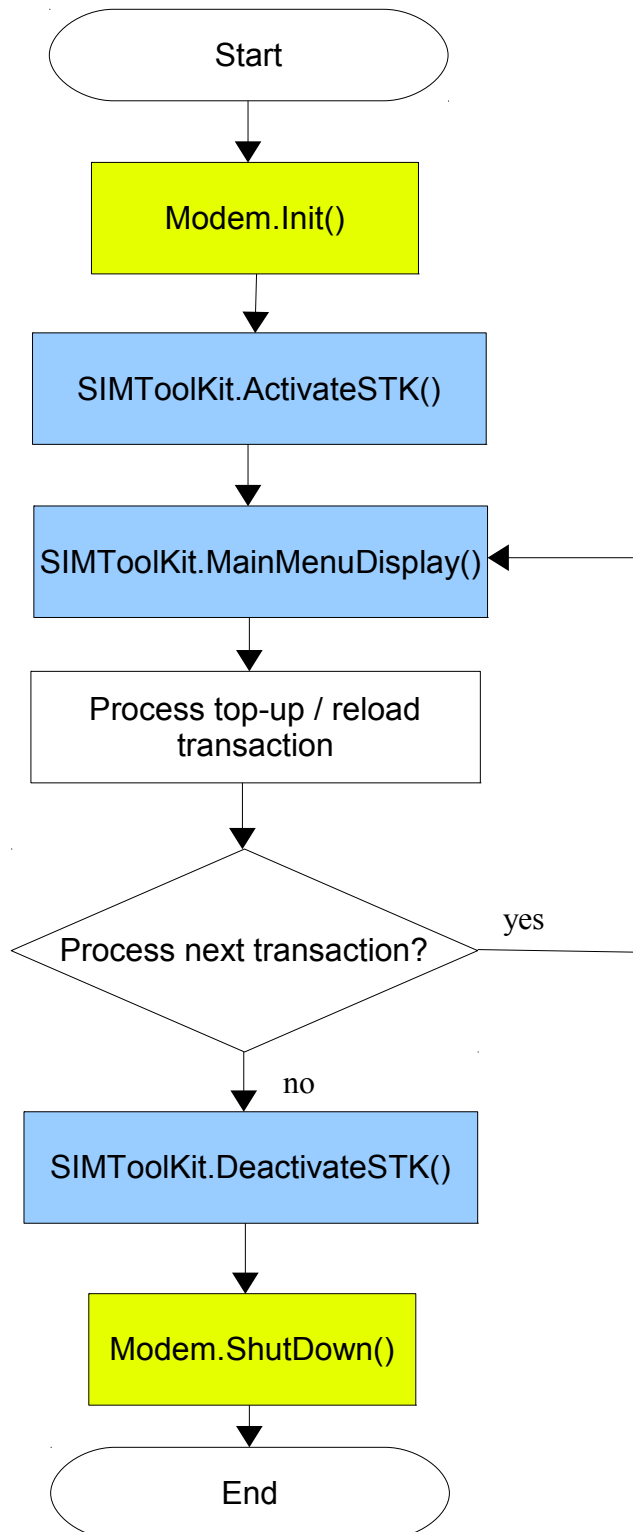
## 11. LIMITATIONS

1. STK API cannot load any application into SIM card.
2. STK API is not a development tool to develop *SIM application*.
3. STK API cannot change, modify, edit, or delete the menu in the SIM.
4. STK API does not work in *Vista* and *Windows 64 bit*.
5. STK Modem (and STK API) does not support USIM (3G SIM or 128k SIM).

## 12. OVERVIEW OF SOFTWARE ARCHITECTURE



### 13. FLOW CHART



## 14. INSTALLATION GUIDE

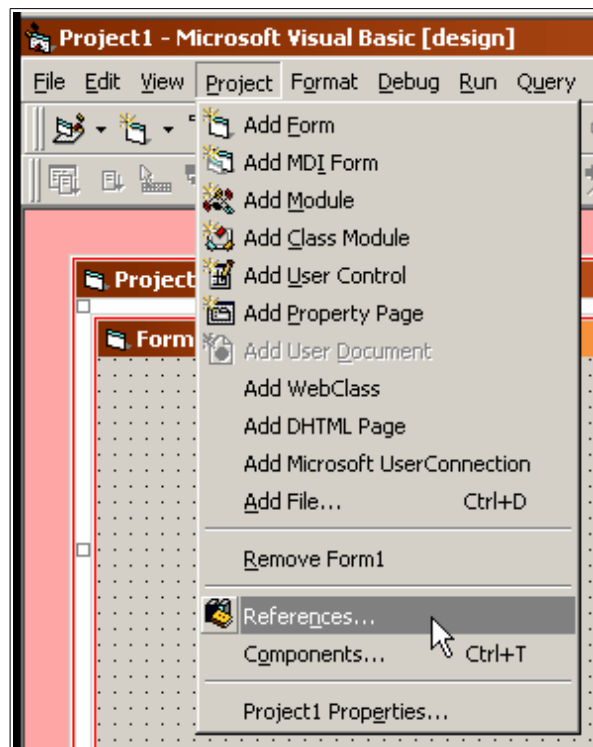
Run the "Setup.exe" to install "MobitekSTK5.dll" file. The dll file will be installed into "system32" folder".

Before using the STK API (ActiveX), please go through the following check list:

1. STK modem is properly set-up.
2. Identify the COM port where the STK modem is connected.
3. The IDE (VB6, VB.net, VC++, etc.) is properly configured to use the ActiveX (refer to your programming guide).

Below is a Visual Basic example on how to configure to use the "MobitekSTK5.dll":

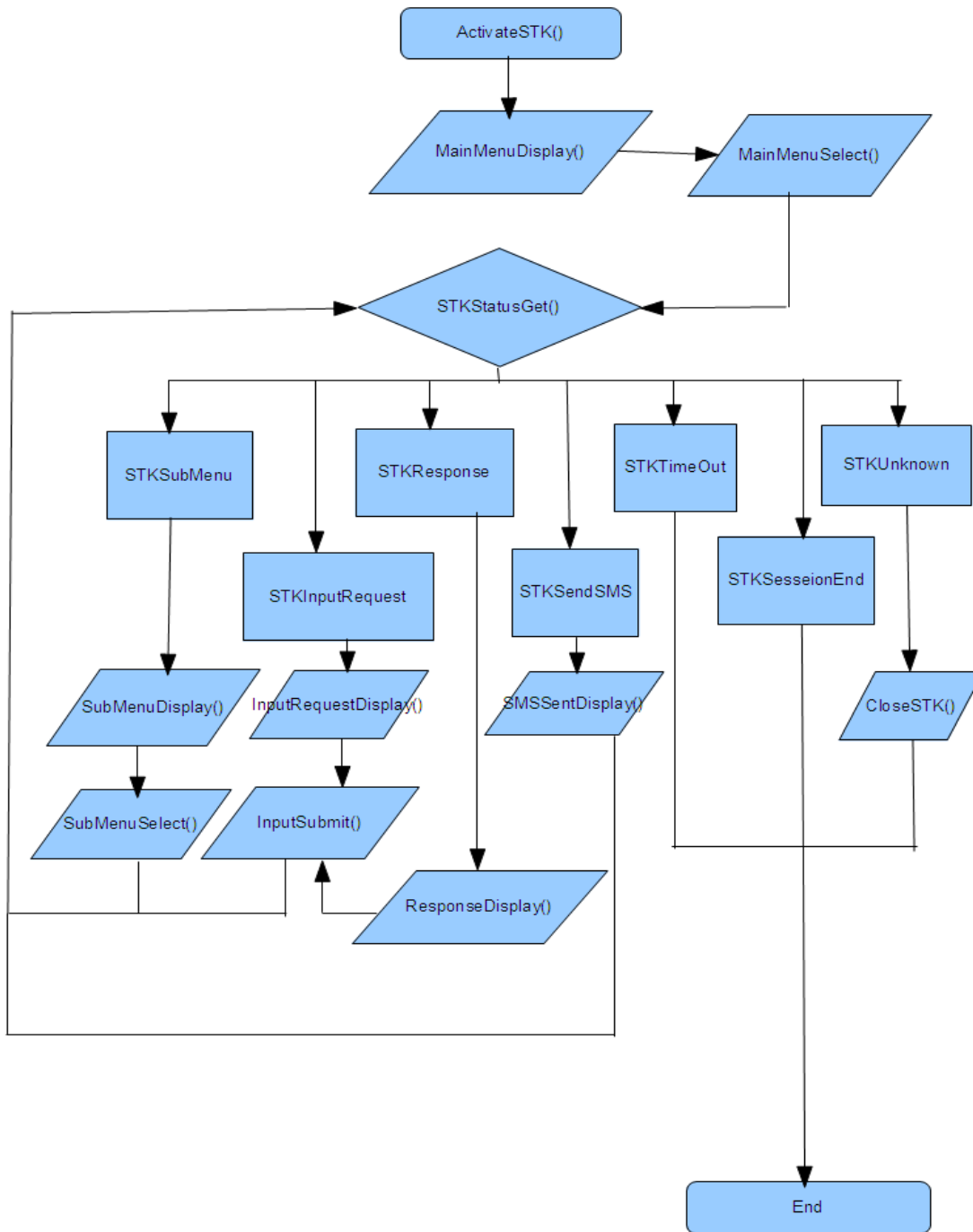
Goto "Project > References > MobitekSTK5"







### 15. STK API FLOW CHART



## 16. SAMPLE CODE

Please refer to the CD for VB sample code that uses STK API.

## 17. DEPLOYMENT

When you want to deploy your STK application on another PC, you need to run the set-up file (**setup.exe**) for **MobitekSTK6.dll** which is located in the CD folder.

## 18. SUPPORT

1. Support shall be in the form of e-mail support, and response time shall be within 3 working days. All e-mail must contain the following information:
  - a) Company name
  - b) Invoice number (can also be obtained from the white label located at the back of the modem)
  - c) Description of problem  
and send it to [support2007@mobitek.com.my](mailto:support2007@mobitek.com.my)
2. If the STK modem is not responsive (hang), please provide these information:-
  - a) Based on what observation that you conclude the modem is not responsive.
  - b) What did you do before it hangs?
  - c) What function / method or property did you call before it hangs?
  - d) What process was it working before it hangs?
  - e) Does your software have log file? If so, please include it.

## 19. RELEASE NOTE

### **VERSION 6.2**

1. Supports Java

### **VERSION 6.1**

2. Supports GL6100

### **VERSION 6.0**

3. USSD class upgraded, based on version 9
  - a) New function – *USSDListen()*

### **VERSION 5.0**

1. New class “Modem” added
2. Class “SMS” upgraded

### **VERSION 4.0**

1. This version only works with STK modem Type **Q24-STK4**.

2. STK API version 4 is similar to version 2, so you can re-use your process flow from version 2.
3. There are no new methods nor properties, so there is no need to re-write your code when upgrading from version 2.3 and 3.x.

### VERSION 3.1

1. This version only works with STK modem – **Wavecom Fastrack Supreme 10, Type S10-STK3**. Other types of STK modem will not work.
2. Method **MainMenuSelect()** improved. Previously all 1<sup>st</sup> try will get “MainMenuSelect() = False”.

### VERSION 3.0

1. This version only works with STK modem – **Wavecom Fastrack Supreme 10, Type S10-STK3**. Other types of STK modem will not work.
2. There are no new method nor property in *STK class*.
3. *Phonebook class* added.
4. *SMS class* has a new method – **GetSignalStrength()**. Refer to the SMS API Manual for more information.
5. Method **MainMenuDisplay()** improved. Auto-reconfiguration of modem is only required when necessary. Wait for STK's response then only display items in main menu.

### VERSION 2.3

1. There are no new method nor property.
2. Better handling of *Digi Flexi e-load* that returns non-standard response.
3. Automatically re-configure the STK modem in order to send and receive SMS.
  - a) The response time to obtain "True" or "False" for the **MainMenuDisplay()** will be longer in version 2.3 than in 2.1 as the function will automatically re-configure the STK modem. However, the re-configuration process will occur once only.

### UPGRADING FROM VERSION 4 TO 5

If you are upgrading from version 4 to 5, then you may need to re-write your code. The following table will guide you on methods or properties that has changed:

	STK API version 4	STK API version 5
<b>STK class</b>		All methods and properties are unchanged, i.e. same as version 4. You do not need to re-write the code when using STK.
<b>Initialise GSM Modem</b>	<b>SMS.ModemInit ()</b>	<b>Modem.ModemInit ()</b>
<b>SMS</b>		All methods and properties are unchanged, i.e. same as version 4. But with new methods:- SMS.SendSMSText() SMS.SendLMSText()
<b>USSD</b>		All methods and properties are unchanged, i.e. same as version 4.
<b>Disconnect GSM Modem</b>	<b>SMS.ModemClose()</b>	<b>Modem.ShutDown()</b>

## 20. FAQ

Last updated on 4<sup>th</sup> of December, 2009.

Q: In the middle of my process, I want to reset/reboot the STK modem, what is the correct flow?

A: The recommended flow is – half way through your code > **SIMToolKit.StopSTK()** > **SIMToolKit.DeactivateSTK()** > **Modem.Reboot()** > **Modem.Init()** > **SIMToolKit.ActivateSTK()**

The wrong flow is – half way through your code > **Modem.Init()**

If you do so, most likely you will get “Modem.Init() = SIMError”

Q: Why do I receive a pop-up message – “Invalid GSM Modem connected!”?

A: You are using a STK modem that is not compatible with STK API. You must use STK modem supplied by MOBITEK System. Other types of modem do not work.

Q: Why do I get “SendSMS() = False” and “ReadSMS() = False” even though “Modem.Init() = 1”?

A: The correct flow is – **Modem.Init()** > code running STK process > **SIMToolKit.DeactivateSTK()** > **SMS.SendSMS()** or **SMS.ReadSMS()** > **SIMToolKit.ActivateSTK()** > code running STK process

The wrong flow is – **Modem.Init()** > code running STK process > **SMS.SendSMS()** or **SMS.ReadSMS()**

I.e. you must deactivate the STK session before sending or reading SMS, otherwise you will get “**SMS.SendSMS() = False**” or “**SMS.ReadSMS() = False**”

Q: Why do I always get a “Fail” whenever I call any methods or functions?

A: Because your computer's processing speed is very, very much faster than SIM card's processing speed, and also faster than the communication speed between STK modem and server of GSM network operator. Therefore, your software application runs faster than *SIM application*, and if it is too fast then your software application may jam the *SIM application* resulting “False” whenever you called a function.

Suggested solutions are:-

a) Put a delay or wait in between calling of methods

e.g. **MainMenuSelect()** > **Wait** > **MainMenuSelect()** > **Wait** > **STKStatusGet()**

avoid – **MainMenuSelect()** > **MainMenuSelect()** > **STKStatusGet()**

b) Put a delay or wait between reload transaction

e.g. after you have completed a top-up for 1st customer > **Wait** > then process next customer

avoid – after you have completed a top-up for 1st customer > process next customer

### Q: Why does the STK modem often hang (not responsive)?

A: Your software application may jam the STK modem. It can be caused by any one of these:-

- a) 2 or more timers, each timer contains subroutine calling functions at the same time;
- b) 2 or more events, each event contains subroutine calling functions at the same time;
- c) 2 or more threads (asynchronous), each calling functions at the same time.

Suggested solution is to develop your software application in synchronous mode and in sequential mode.

### Q: Can the STK API supports multiple STK modems?

A: Yes. Our STK API (ActiveX-COM) is multi-threaded. There are 2 methods:-

**1st METHOD** – multiple applications calling 1 ActiveX-COM

e.g. *modem1.exe* calls *STKAPI.dll* to control STK modem no. 1 *modem2.exe* calls *STKAPI.dll* to control STK modem no. 2, etc.

The code for *modem1.exe* is the same as the code in *modem2.exe* You do not need to rewrite your code. You use back the same code, except on the part where you specify the COM port number.

**2nd METHOD** – 1 application creating 1 thread for each STK modem; you need to use .Net Framework such as VB.Net or C# or C++.Net to develop a multi-threaded application.

e.g. *modem.exe* creates thread no. 1 that calls *STKAPI.dll* to control STK modem no. 1  
*modem.exe* creates thread no. 2 that calls *STKAPI.dll* to control STK modem no. 2, etc.

## 21. APPENDIX 1: Referencing COM in .Net

The following article is from <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dndotnet/html/callcomcomp.asp>. It describes how you can call **ActiveX COM** from **.Net**:

*As a .NET developer, you also have the option of using COM components directly. At least, that's what it looks like, although you're programmatically still using a RCW to get to objects in unmanaged code. If you're working within a Visual Basic .NET project, you can follow these steps to add a reference to a COM component:*

1. Click **Project**, and then click **Add Reference**.
2. In the **Add Reference** dialogue box, click the **COM** tab.
3. Select the type library you wish to use from the list and click **Select**, or use the **Browse** button to locate a component that's not listed. The selected components will be added to the lower listview in the dialogue box.
4. Click **OK** to create RCWs for the selected type libraries in your Visual Basic .NET project.

*When you do this, you'll find that Visual Basic .NET actually creates a DLL in your project's /Bin folder, with a name derived from the original COM component name. For example, if you reference `BackEnd.dll` version 2.0 in this manner, Visual Basic .NET will create the RCW in the file `Interop.BackEnd_2_0.dll`.*

## 22. APPENDIX 2: How to Use STK API in JAVA

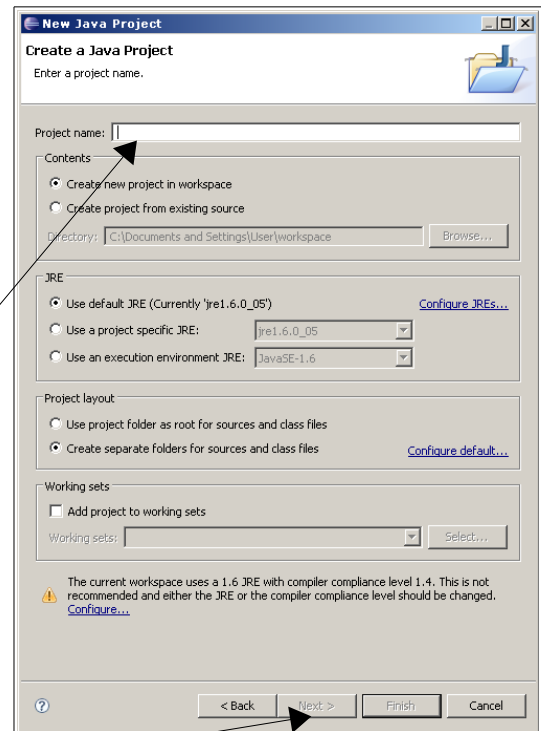
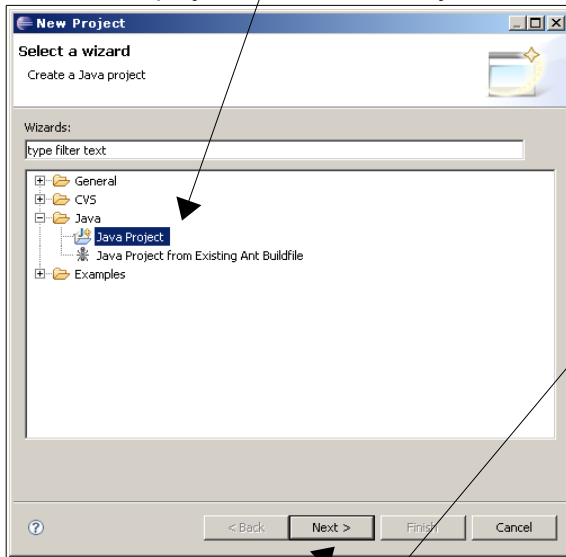
### System Requirements

Install the following:-

1. **SMS/GSM Modem**: follow the instruction in **SMS Gateway Development CD** to set-up the SMS/GSM modem.
2. **MOBITEK™ STK API** version 6.2 : follow the instruction in CD to install.
3. **JRE**: install JRE version 1.6 from the CD.
4. **JACOB**: unzip the file “jacob-1.14.3.zip” that is in CD, then copy these 2 files to “system32” folder:
  - a) jacob-1.14.3-x64.dll
  - b) jacob-1.14.3-x86.dll
5. **ECLIPSE**: install from CD; we choose ECLIPSE as the IDE for Java. You may use other IDE.

### Set-up Eclipse

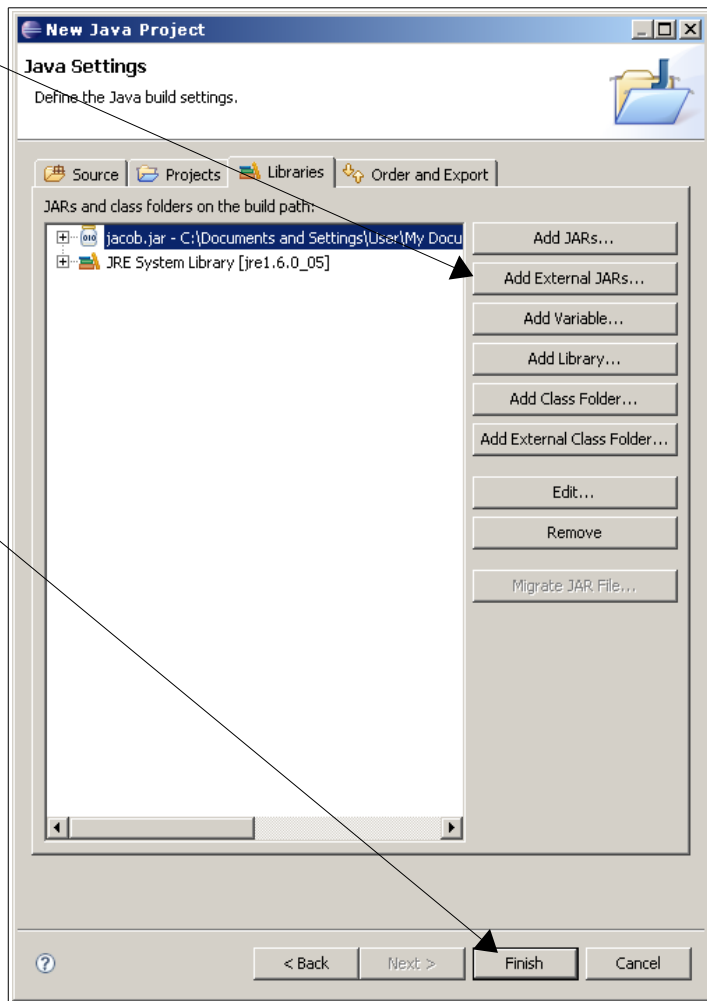
1. Add a new project, select “Java Project”



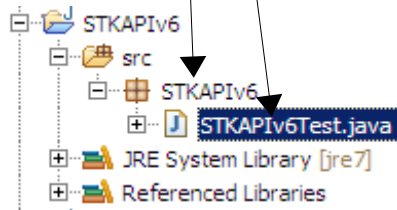
2. Click “Next”
3. Type “STKAPIv6” in “Project Name:”
4. Click “Next”



5. Go to "Libraries" tab
6. Click on "Add External JARs" button
7. Select the file "jacob.jar" (that was unzip from "jacob-1.14.3.zip")
8. The final outcome should be similar to figure right
9. Click on "Finish" button



10. Add a new package, named it "STKAPIv6"
11. Add a new class, named it "STKAPIv6Test.java"
12. And your final outcome should be similar to figure below:

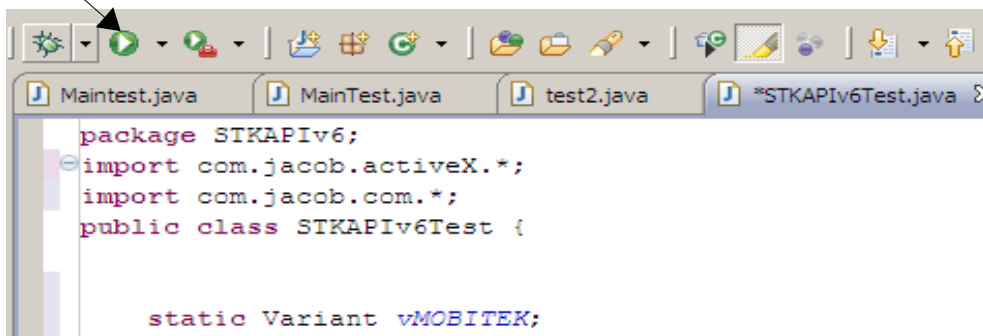


13. Copy the Java sample code (refer Below; also available in CD) and paste to "STKAPIv6Test.java"
14. In the sample code, change the COM port to the respective COM port:

```
vMOBITEK = Modem.invoke("Init", 1);
```

15. In the Java code, we already hardcoded a script just for the sake of testing the STK will function as below in Start() Function
  - a) Open STK Main Menu
  - b) Call sub menu, 254 → EasyAccess
  - c) Call sub menu, 1 → News & Sport
  - d) Call sub menu, 6 → News on Demand
  - e) Call sub menu, 5 → Sport
  - f) Call sub menu, 1 → Tennis
  - g) Send SMS Out
    - \* This STK Menu is using Maxis Telco (Malaysia Telco)
    - \* Please change accordingly to your telco or try to only call STK Main Menu First.

16. Click "Run" button



**17. And you will see the result in "Console" tab:**

```
GSM modem is connected to computer.
GSM modem is connected to GSM network.
The operator of the GSM network is MY MAXIS.
STK Activated : true
=====
EasyAccess
255:My Hotlink
254:News&Sports
253:Hotlines
252:Money
251:Downloads
250:At Play
249:Going Places
248:Community
247:My Favourites

>254 Selected!

News&Sports
1:News On Demand
2:EasyAlerts

>1 Selected!

1:Breaking News
2:General
3:Business
4:Technology
5:Entertainment
6:Sports

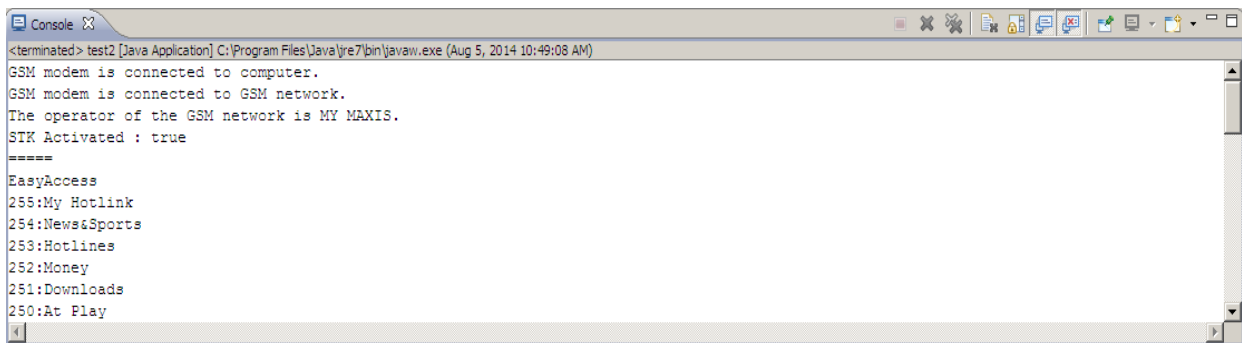
>6 Selected!

1:General
2:F1
3:Soccer
4:Golf
5:Tennis

>5 Selected!

1:Send
2:Back

>1 Selected!
User has allowed STK to send out SMS
=====
STK Deactivated : true
Connection between GSM modem and computer is closed.
```

A screenshot of a Java console window titled "Console". The window shows the following text output:

```
<terminated> test2 [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Aug 5, 2014 10:49:08 AM)
GSM modem is connected to computer.
GSM modem is connected to GSM network.
The operator of the GSM network is MY MAXIS.
STK Activated : true
=====
EasyAccess
255:My Hotlink
254:News&Sports
253:Hotlines
252:Money
251:Downloads
250:At Play
```

18. You have successfully used Java with **MOBITEK™ STK API** version 6.2 to:

- a) Initiate Modem
- b) Check the GSM operator name
- c) Activate the STK
- d) open and navigate through STK Menu
- e) send out SMS;
- f) Deactivate STK
- g) Close the modem connection

## Java Sample Code

```

package STKAPIv6;
import com.jacob.activeX.*;
import com.jacob.com.*;
public class STKAPIv6Test {

    static Variant vMOBITEK;
    static boolean bMOBITEK;
    static int iMOBITEK;
    static String sMOBITEK;

    static ActiveXComponent Modem = new ActiveXComponent
("MobitekSTK6.Modem");
    static ActiveXComponent STKAPI = new ActiveXComponent
("MobitekSTK6.SIMToolkit");

    public static void main(String[] args)
    {
        ModemInitialize();
        ActivateSTK();
        Start();
        DeactivateSTK();
        ModemDeinitialize();
        System.exit(0); //<-force close
    }

    private static void ModemInitialize() {

        vMOBITEK = Modem.invoke("Init", 1);
        iMOBITEK = vMOBITEK.getInt();
        switch (iMOBITEK)
        {
            case 0 : System.out.println("GSM modem is NOT connected
to computer!"); System.exit(0);
            case 1 : System.out.println("GSM modem is connected to
computer."); break;
            case 2 : System.out.println("GSM modem is NOT connected
to computer because a PIN is required!"); System.exit(0);
            case 3 : System.out.println("GSM modem is NOT connected
to computer because wrong PIN is entered!"); System.exit(0);
            case 4 : System.out.println("GSM modem is NOT connected
to computer because SIM card is blocked by network operator!"); System.exit(0);
            case 5 : System.out.println("GSM modem is NOT connected
to computer because SIM card has problem!"); System.exit(0);
        }

        vMOBITEK = Modem.invoke("IsConnectToGSM");

        bMOBITEK = vMOBITEK.getBoolean();
        if (bMOBITEK)
        {
            System.out.println("GSM modem is connected to GSM
network.");
            vMOBITEK = Modem.invoke("OperatorName");
            sMOBITEK = vMOBITEK.getString();
            System.out.println ("The operator of the GSM network is
" + sMOBITEK + ".");
        }
        else
        {
            System.out.println("GSM modem is NOT connected to GSM
network!");
            System.exit(0);
        }
    }

    private static void ActivateSTK() {
        // TODO Auto-generated method stub
    }
}

```

```

        vMOBITEK = STKAPI.invoke("ActivateSTK");
        System.out.println("STK Activated : " + vMOBITEK);
    }

    private static void Start() {
        // TODO Auto-generated method stub
        //SimToolKit Menu
        System.out.println("=====");
        //open main menu
        CallMainMenu();

        // Below are script please change accordingly or
        // Comment it to call only main menu

        MainMenuSel(254); // EasyAccess

        CallSubMenu();
        SubMenuSel(1);    // News&sport

        CallSubMenu();
        SubMenuSel(6);    // News On Demand

        CallSubMenu();
        SubMenuSel(5);    // Sport

        CallSubMenu();
        SubMenuSel(1);    // Tennis

        // End of Script

        AllowSendSMS();
        System.out.println("=====");
    }

    private static void CallMainMenu() {

        vMOBITEK = STKAPI.invoke("MainMenuDisplay");
        bMOBITEK = vMOBITEK.getBoolean();

        if(bMOBITEK==true)
        {
            String MMName =
                STKAPI.getPropertyAsString("MainMenuName");
            //property "MSG" is the sender's message
            String MMList =
                STKAPI.getPropertyAsString("MainMenuList");
            System.out.println(MMName+"\n"+MMList);
        }

    }

    private static void MainMenuSel( int num) {
        STKAPI.invoke("MainMenuSelect",num);
        System.out.println("\n>"+num + " Selected! \n");
    }

    private static void CallSubMenu() {
        vMOBITEK = STKAPI.invoke("SubMenuDisplay");
        bMOBITEK = vMOBITEK.getBoolean();

        if(bMOBITEK==true)
        {
            String MMName2 =
                STKAPI.getPropertyAsString("SubMenuName");
            //property "MSG" is the sender's message
            String MMList2 =
                STKAPI.getPropertyAsString("SubMenuList");
            System.out.println(MMName2+"\n"+MMList2);
            MMName2 = null;
        }
    }

```

```

        MMList2 = null;
    }

}

private static void SubMenuSel( int num) {
    STKAPI.invoke("SubMenuSelect", num);
    System.out.println("\n>"+num + " Selected! ");
}

private static void AllowSendSMS() {
    vMOBITEK = STKAPI.invoke("SMSSentDisplay");
    sMOBITEK = vMOBITEK.getString();

    if( sMOBITEK == null)
        System.out.println(sMOBITEK);
    else
        System.out.println("User has allowed STK to send out
SMS");
}

private static void DeactivateSTK() {
    // TODO Auto-generated method stub

    vMOBITEK = STKAPI.invoke("DeactivateSTK");
    System.out.println("STK Deactivated : " + vMOBITEK);
}

private static void ModemDeinitialize() {
    // TODO Auto-generated method stub
    //call API to close connection between GSM modem and computer,
the return value of the API call is assigned to "SMSAPIReturnValue"
    vMOBITEK = Modem.invoke("ShutDown");
    //since the return value of "ModemClose" is a boolean, so
extract the return value as a boolean and assign to "bSMS"
    bMOBITEK = vMOBITEK.getBoolean();
    if (bMOBITEK) System.out.println("Connection between GSM modem
and computer is closed." );
    else System.out.println("Connection between GSM modem and
computer CANNOT be closed!" );
}
}
}

```

### 23. APPENDIX 3: How to Decode Alpha Field Displayed in STK Menu

In countries where English is not the primary language, the STK Menu may display alphanumeric string instead of English text.

We provide VB.Net sample code to decode the alphanumeric string. The VB.Net Sample Code decodes the alphanumeric string into UCS2 and then decodes the UCS2 to display the unicode characters.

The sample code is in the MOBITEK Air-Time Reload Development Kit CD.

Example, the alphanumeric string displayed by Orange STK Menu is "810E00C4E5F2EEE9E5F220F4F2E1EEF3AE ", the VB.Net sample code will decode it and convert to "Dernier trans.". Refer image below.

