

# JPOS MSR Service Object Manual

*This Document System was last updated on December 9, 2010*

[Update history can be found here](#)

## Contents

JPOS MSR Service Object Manual .....	1
Installation .....	2
File List .....	2
Run POSTest .....	3
Sample Code for Using MSR JPOS .....	4
Revisions .....	5

# Installation

Below steps show how to install JDK and needed serial driver in the system.

1. To use Hyper-Terminal to verify your PC and MSR is worked normally.
2. Install JDK (www.sun.com), I assume you download jdk-1\_5\_0\_21-windows-i586-p.exe. This file is also can be found in [Software] folder.
3. Set environment variable:  
Control Panel -> System Properties -> Environment Variables  
-> New  
Variable name ="PATH"  
Variable value="C:\Program Files\Java\jdk1.5.0\_21\bin" <-----what your JDK installed path  
-> New  
Variable name ="JAVA\_HOME"  
Variable value="C:\Program Files\Java\jdk1.5.0\_21" <-----what your JDK installed path
4. install java comm API for windows:
  - I. The comm API is located in the [Libraries] folder: comm.jar, javax.comm.properties and win32com.dll.
  - II. Copy win32com.dll to your <JDK>\bin directory.  
\* DOS prompt command example:  
`copy win32com.dll "c:\Program Files\java\jdk1.5.0_21\bin"`
  - III. Copy comm.jar to your <JDK>\lib directory.  
\* DOS prompt command example:  
`copy comm.jar "c:\Program Files\java\jdk1.5.0_21\lib"`
  - IV. Copy javax.comm.properties to your <JDK>\lib directory.  
\* DOS prompt command example:  
`copy javax.comm.properties "c:\Program Files\java\jdk1.5.0_21\lib"`

**Note:** Above examples are the copy command under DOS prompt (if the jdk is located in C:\Program Files\Java\jdk1.5.0\_21)

## File List

Below is the JPOS needed files listed (located in [Libraries] folder):

- **comm.jar**: Java Communication API WIN32 communication library.
- **javax.comm.properties**: Used specify the devices and drivers that are available to the Java™ Communications API
- **jpos111.jar**: JPOS control file.
- **JposKeyHook.dll**: JNI hook key driver.
- **msr250so.jar**: MSR JPOS service library file.
- **postest.jar**: POSTEST java library.
- **win32com.dll**: Communication driver for comm.jar uses under Windows
- **xerces.jar**: JAVA document library file.

# Run POSTest

The POSTest program is using to read MSR track data.

Below shows how to run this program.

**Note:** If your MSR is not connected to COM1, then please modify the JPOS.XML to set Port property value to the port name where MSR is connected.

1. Run MSRPOS.bat
2. Click the MSR tab.
3. Press the Open button.
4. Press the Claim button, and then check the Device enabled and Data event enabled check box.
5. Check TRACK1 Enable
6. Check TRACK2 Enable
7. Check TRACK3 Enable
8. Swipe the card.
9. You will find the data.

The screenshot shows the JavaPOSTester in Progress application window. The window has a menu bar with 'File' and a toolbar with various device-related buttons. The 'MSR' tab is selected. Below the toolbar, there is a section for 'Logical name: gigaMSR' and 'JPOS\_S\_IDLE'. A row of buttons includes 'Open', 'Claim', 'Release', 'Close', 'Info', 'Statistics', 'Firmware', 'O/C/E', and 'EXIT'. On the left, there is a list of checkboxes for configuration: 'Auto disable', 'Data event enabled' (checked), 'Device enabled' (checked), 'Freeze events', 'Decode data' (checked), 'Parse decode data', 'Transmit sentinels', 'Per-track error reporting', 'Track 1 enabled' (checked), 'Track 2 enabled' (checked), 'Track 3 enabled' (checked), and 'Track 4 enabled'. A 'Clear Input' button is below these. The main area is titled 'MSR Reading' and contains a table with MSR data. The table has two columns: a small column with values 0 and a larger column with the corresponding MSR data. Below the table, there are labels for 'Account number:', 'Expiration date:', 'Title:', 'First name:', 'Middle initial:', 'Surname:', 'Suffix:', 'Service Code:', 'Track 1 discretionary data:', 'Track 2 discretionary data:', and 'Data Count: 0'. At the bottom right, there are 'Clear Fields' and 'Refresh Fields' buttons.

Track Data	Value
Track 1 Data:	0 B9999991234567890^STERLING/JOANNE^04121011445
Track 2 Data:	0
Track 3 Data:	0 i9999991234567890=00101220100005095016020000005030001041210123456789
Track 4 Data:	0

Account number: B9999991234567890  
Expiration date: 0412  
Title:  
First name: JOANNE  
Middle initial:  
Surname: STERLING  
Suffix:  
Service Code:  
Track 1 discretionary data:  
Track 2 discretionary data:  
Data Count: 0

# Sample Code for Using MSR JPOS

```
import jpos.*;
import jpos.config.*;
import jpos.config.simple.*;
import jpos.loader.*;
import jpos.loader.simple.*;
import jpos.profile.*;
import jpos.services.*;
import jpos.util.*;
import jpos.util.tracing.*;

public class JPOSMSRTester {
    public static void main(String args[]) {

        MSR msr = new MSR();

        //Open Device
        try
        {
            msr.open("MSR250");
        }
        catch(JposException e)
        {
            System.err.print(e);
        }

        //Claim device
        try
        {
            msr.claim(1000);
        }
        catch(JposException e)
        {
            System.err.print(e);
        }

        try
        {
            //Enable the device
            msr.setDeviceEnabled(true);
        }
    }
}
```

```

msr.setDataEventEnabled(true);

//Set wanted tracks, read all tracks
msr.setTracksToRead(0x0f);

//start reading
int i;
while(true)
{
    //waiting for card swipe
    i = msr.getDataCount();
    if(i>0)
    {
        //show track data
        System.out.println("Track1:");
        System.out.println(new String(msr.getTrack1Data()));
        System.out.println("Track2:");
        System.out.println(new String(msr.getTrack2Data()));
        System.out.println("Track3:");
        System.out.println(new String(msr.getTrack3Data()));
        break;
    }
}
//close device
msr.release();
msr.close();

}
catch(JposException e)
{
    System.err.print(e);
}
}
}

```

## Revisions

2010/12/09

\* Support USB HID Interface MSR reader

2009/1/8

\* Original Release