



## Using the Logitech|G Arx Control SDK with C#

---

© 2014 Logitech. Confidential

**The Logitech|G Arx Control SDK, including all accompanying documentation, is protected by intellectual property laws. All use of the Logitech|G Arx Control SDK is subject to the License Agreement found at the end of this document. If you do not agree to the terms and conditions of the License Agreement, you must immediately return any documentation, the accompanying software and all other material provided to you by Logitech. All rights not expressly granted by Logitech are reserved.**

## Contents

Overview .....	3
Making the Arx Control SDK work in your C# program .....	3
Steps .....	3

## Overview

The Logitech|G Arx Control Software Development Kit enables applications such as games to interact with the Arx Control app on mobile devices.

Arx Control allows games and third party developers to take advantage of an iOS/Android device as a secondary screen to display useful data from the game.

It is built as a C++ library, but it can be easily integrated in a C# assembly, using P/Invoke and function marshaling.

Please refer to the Logitech SDK's Doc\LogitechGArxControl.pdf for details on the SDK's functionality.

## Making the Arx Control SDK work in your C# program

The following steps show how to make the Logitech SDK work with a C# program. Please adapt these steps to your game for things to work.

### Steps

1. Create a SDK C# wrapper class as follows :

```
using System.Collections;
using System.Runtime.InteropServices;
using System.Collections.Specialized;
using System;

public class LogitechArx {

    public const int LOGI_ARX_ORIENTATION_PORTRAIT = 0x01;
    public const int LOGI_ARX_ORIENTATION_LANDSCAPE = 0x10;
    public const int LOGI_ARX_EVENT_FOCUS_ACTIVE = 0x01;
    public const int LOGI_ARX_EVENT_FOCUS_INACTIVE = 0x02;
    public const int LOGI_ARX_EVENT_TAP_ON_TAG = 0x04;
    public const int LOGI_ARX_EVENT_MOBILEDEVICE_ARRIVAL = 0x08;
    public const int LOGI_ARX_EVENT_MOBILEDEVICE_REMOVAL = 0x10;
    public const int LOGI_ARX_DEVICEYPE_IPHONE = 0x01;
    public const int LOGI_ARX_DEVICEYPE_IPAD = 0x02;
    public const int LOGI_ARX_DEVICEYPE_ANDROID_SMALL = 0x03;
    public const int LOGI_ARX_DEVICEYPE_ANDROID_NORMAL = 0x04;
    public const int LOGI_ARX_DEVICEYPE_ANDROID_LARGE = 0x05;
    public const int LOGI_ARX_DEVICEYPE_ANDROID_XLARGE = 0x06;
    public const int LOGI_ARX_DEVICEYPE_ANDROID_OTHER = 0x07;

    [UnmanagedFunctionPointer(CallingConvention.Cdecl)]
    public delegate void logiArxCB(int eventType, int eventValue,
    [MarshalAs(UnmanagedType.LPWSTR)]String eventArg, IntPtr context);

    public struct logiArxCbContext
    {
        public logiArxCB arxCallback;
        public IntPtr arxContext;
    }
}
```

```
}

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxInit(String identifier, String
friendlyName, ref logiArxCbContext callback);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxInitWithIcon(String identifier, String
friendlyName, ref logiArxCbContext callback, byte [] iconBitmap);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxAddFileAs(String filePath, String
fileName, String mimeType = "");

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxAddContentAs(byte[] content, int size,
String fileName, String mimeType = "");

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxAddUTF8StringAs(String stringContent,
String fileName, String mimeType = "");

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxAddImageFromBitmap(byte[] bitmap, int
width, int height, String fileName);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxSetIndex(String fileName);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxSetTagPropertyById(String tagId, String
prop, String newValue);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxSetTagsPropertyByClass(String
tagsClass, String prop, String newValue);
```

```

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxSetTagContentById(String tagId, String
newContent);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern bool LogiArxSetTagsContentByClass(String
tagsClass, String newContent);

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern int LogiArxGetLastError();

[DllImport("LogitechGArxControlEnginesWrapper.dll", CharSet =
CharSet.Unicode, CallingConvention = CallingConvention.Cdecl)]
public static extern void LogiArxShutdown();
}

```

2. Call the functions from the wrapper from your C# code as follows:

```

// Use this for initialization
void Start () {
    LogitechArx.logiArxCbContext contextCallback;
    contextCallback.arxCallBack = new LogitechArx.logiArxCB (this.SDKCallback);
    contextCallback.arxContext = System.IntPtr.Zero;
    bool retVal = LogitechArx.LogiArxInit("sdk.sample.test", "C#test", ref
contextCallback, null);

    if(!retVal){
        int retCode = LogitechArx.LogiArxGetLastError();
        Debug.Log("loading sdk failed:"+ retCode);
    }

static void SDKCallback(int eventType, int eventValue, System.String eventArg,
System.IntPtr context)
{
    if(eventType ==LogitechArx.LOGI_ARX_EVENT_MOBILEDEVICE_ARRIVAL)
    {
        //Send your files here
    }
    else if(eventType ==LogitechArx.LOGI_ARX_EVENT_MOBILEDEVICE_REMOVAL)
    {
        //Device disconnected
    }
    else if (eventType == LogitechArx.LOGI_ARX_EVENT_TAP_ON_TAG)
    {
        if(eventArg == "myBtn")
        {
            //Do something on this input
        }
    }
}
}

```

3. Copy Logitech SDK's Lib\GameEnginesWrapper\x86\LogitechGArxControlEnginesWrapper.dll to your C# 32 bit executable path
4. Copy Logitech SDK's Lib\GameEnginesWrapper\x64\LogitechGArxControlEnginesWrapper.dll to your C# 64 bit executable path
5. Compile and run your program

For questions/comments, email [devtechsupport@logitech.com](mailto:devtechsupport@logitech.com)