

Forum:[Hack Forums](#)

Thread: [Playing an XM Song](#)

requiredhq - October 26, 2019, 12:00 pm

Well, I've looked long and hard and the furthest I got was looking at a program irrklang. It didn't work.

How do I play an xm song in the background of my vb.net app, im using visual studio 2012 if that matters. It's in literally every keygen on the internet, yet there isn't a single tutorial other than this one horrible video on youtube. [XM Play using BassMod plugin - YouTube](#)

This is my last resort, I really want to add some background music to my changelog.. :l it's a small fun feature but a huge pain in the ass.

Pawnhub - October 26, 2019, 9:39 pm

[deployment - How to play background music in a C# Console Application - Stack Overflow](#)

Also check out the media.soundplayer class.

You can convert the code from C# to VB online and both share the same framework (in this case .net)

Eternal Darkness - October 26, 2019, 9:56 pm

<https://github.com/veso266/Chip-Player> ->

<https://github.com/veso266/Chip-Player/blob/master/ChipTune%20Player/PlayXM.cs>

More bassmod info:

<https://www.codeproject.com/Articles/120348/Play-XM-in-C-From-File-and-Memory>

requiredhq - October 26, 2019, 10:24 pm

(Oct 26, 2019 09:56 PM)Eternal Darkness Wrote: [\[CITE\]](#) [GitHub - veso266/Chip-Player: A Simple Chip Player using BASSMOD.dll to play ChipTunes](#) -> [Chip-Player/PlayXM.cs at master · veso266/Chip-Player · GitHub](#)

More bassmod info: [Play XM in C++ - From File and Memory - CodeProject](#)

This is good but it's not in visual studio 2012 vn.net, it's in like C to my knowledge. I know I need to reference the dll and the libraries will either be included or not depending on if it's for .net. Still a bit confused. Then how do I simply play a xm in the background. One command, upon trying to figure this out i know i need to dim some type of player then direct it to my.resources.file.xm but I know different programs have different commands like 2player or soundengine.

Eternal Darkness - October 26, 2019, 10:47 pm

(Oct 26, 2019 10:24 PM)requiredhq Wrote: [\[CITE\]](#) This is good but it's not in visual studio 2012 vn.net, it's in like C to my knowledge. I know I need to reference the dll and the libraries will either be included or not depending on if it's for .net. Still a bit confused. Then how do I simply play a xm in the background. One command, upon trying to figure this out i know i need to dim some type of player then direct it to my.resources.file.xm but I know different programs have different commands like 2player or soundengine.

The GitHub link is to a C# file, which is easily converted to VB.NET. VB.NET is what you are writing

it in, correct?

Some Converters:

<https://converter.telerik.com/>

<https://www.carlosag.net/tools/codetranslator/>

<https://codeconverter.icsharpcode.net/>

<https://www.developerfusion.com/tools/convert/csharp-to-vb/>

<https://www.dotnetspider.com/convert/Csharp-To-Vb.aspx>

https://www.tangiblesoftwareolutions.com/product_details/csharp-to-vb-converter.html

Likely you will need to touch up the output (conversion is not 100% accurate, but reasonably close).

Look at the other .cs files in the github repo in order to understand how to use the linked class...
The class allows you to play xm files, the actual usage is in Program.cs...

requiredhq - October 27, 2019, 3:33 am

(Oct 26, 2019 10:47 PM)Eternal Darkness Wrote: [\[CITE\]](#) The GitHub link is to a C# file, which is easily converted to VB.NET. VB.NET is what you are writing it in, correct?

Some Converters:

[Code Converter C# to VB and VB to C# Telerik](#)

[CodeTranslator: Code Translation From VB.NET C# TypeScript Java](#)

[Convert VB.NET to/from C# online - Roslyn Code Converter](#)

[Convert C# to VB.NET - A free code conversion tool - developer Fusion](#)

[Convert C# to VB.NET. Convert C# projects to VB.NET](#)

[C# to VB.NET Converter](#)

Likely you will need to touch up the output (conversion is not 100% accurate, but reasonably close).

Look at the other .cs files in the github repo in order to understand how to use the linked class...
The class allows you to play xm files, the actual usage is in Program.cs...

Okay, I'm running into a few issues. First, I put song.xml in resources. However I'm having an issue "loading up" the song.

```
Code Public Sub New(ByVal FileName As String)
    data = System.IO.File.ReadAllBytes(FileName)
End Sub
```

```
Public Sub New(ByVal XMData As Byte())
    data = XMData
End Sub
```

My Code

```
Code Dim Data As New String
Dim xmdata As New Byte
```

```
Data = System.IO.File.ReadAllBytes(My.Resources.song)
xmdata = My.Resources.song
```

Simply doesn't work. It's saying 1 dimensional array cannot be converted to byte, etc. Error being my.resources.song and data.

My next error comes with importing the dll and 'using it's library?', don't quote me on that but I think that's what I am trying to do. I've tried doing imports bassmod.dll, which gives me a namespace error. Here's the certain code that I'm trying to reference.

```

Code <DllImport("BASSMOD.dll")>
    Public Shared Function BASSMOD_Init(ByVal device As Integer, ByVal frequency As
UInteger, ByVal flags As UInteger) As Integer
    <DllImport("BASSMOD.dll", CharSet:=CharSet.Auto)>
    <MarshalAs(UnmanagedType.Bool)>
    Public Shared Function BASSMOD_Free() As Boolean
    <DllImport("BASSMOD.dll")>
    Public Shared Function BASSMOD_MusicLoad(
    <MarshalAs(UnmanagedType.Bool)> ByVal mem As Boolean,
    <[In]>
    <MarshalAs(UnmanagedType.LPArray)> ByVal XMfile As Byte(), ByVal offset As UInteger,
ByVal length As UInteger, ByVal flags As UInteger) As Integer
    <DllImport("BASSMOD.dll")>
    Public Shared Function BASSMOD_MusicPlay() As Integer

```

gigajew - October 27, 2019, 9:47 am

You're defining xmdata as one byte. You need to define it as an array of byte like this: Dim xmdata As Byte()

MavenGradle - October 27, 2019, 9:52 am

(Oct 26, 2019 09:39 PM)UnknownPawn Wrote: [\[CITE\] deployment - How to play background music in a C# Console Application - Stack Overflow](#)

Also check out the media.soundplayer class.

You can convert the code from C# to VB online and both share the same framework (in this case .net)

The System.Media.SoundPlayer does not play XM formats. I quote from the MS docs:

Quote:Controls playback of a sound from a .wav file.

<https://docs.microsoft.com/en-us/dotnet/api/system.media.soundplayer?view=netframework-4.8>

(Oct 27, 2019 03:33 AM)requiredhq Wrote: [\[CITE\]](#) Okay, I'm running into a few issues. First, I put song.xml in resources. However I'm having an issue "loading up" the song.

My Code

```

Code      Dim Data As New String
          Dim xmdata As New Byte

```

```

          Data = System.IO.File.ReadAllBytes(My.Resources.song)

```

```

          xmdata = My.Resources.song

```

Simply doesn't work. It's saying 1 dimensional array cannot be converted to byte, etc.error being my.resources.song and data.

Do you actually know the difference between a byte array and a byte? I recommend you to look up how to declare array variables.

Next up, in order to read a resource into a byte array, you do not use

Code System.IO.File.ReadAllBytes() which evidently takes in the path of a [FILE](#). A project resource is not always a file on your disk nor does providing this method with the actual resource work..

You could use something like this if you marked the file as a resource in your project:

```

Code Dim xmData() As Byte = My.Resources.I_am_an_idiot

```

```

Dim songLyrics As String = My.Resources.songlyrics

```

Pawnhub - October 27, 2019, 11:16 am

(Oct 27, 2019 09:52 AM)MavenGradle Wrote: [\[CITE\]](#) --snip--

To be fair unless the program has to take in XM by default like from a 3rd party

OP could use a utility like so. [Transform XM to WAV audio files online and for free!](#)

gigajew - October 27, 2019, 1:42 pm

Not Visual Basic but easy enough to convert or learn from <https://github.com/veso266/Chip-Player>

Eternal Darkness - October 27, 2019, 2:31 pm

(Oct 27, 2019 03:33 AM)requiredhq Wrote: [\[CITE\]](#) Okay, I'm running into a few issues. First, I put song.xml in resources. However I'm having an issue "loading up" the song.

```
Code Public Sub New(ByVal FileName As String)
    data = System.IO.File.ReadAllBytes(FileName)
End Sub
```

```
Public Sub New(ByVal XMData As Byte())
    data = XMData
End Sub
```

My Code

```
Code Dim Data As New String
Dim xmdata As New Byte
```

```
Data = System.IO.File.ReadAllBytes(My.Resources.song)
xmdata = My.Resources.song
```

Simply doesn't work. It's saying 1 dimensional array cannot be converted to byte, etc.error being my.resources.song and data.

My nexterror comes with importing the dll and 'using it's library?', don't quote me on that but I think that's what I am trying to do. I've tried doing imports bassmod.dll, which gives me a namespace `_error_`. Here's the certain code that I'm trying to reference.

```
Code <DllImport("BASSMOD.dll")>
Public Shared Function BASSMOD_Init(ByVal device As Integer, ByVal frequency As
UInteger, ByVal flags As UInteger) As Integer
<DllImport("BASSMOD.dll", CharSet:=CharSet.Auto)>
<MarshalAs(UnmanagedType.Bool)>
Public Shared Function BASSMOD_Free() As Boolean
<DllImport("BASSMOD.dll")>
Public Shared Function BASSMOD_MusicLoad(
<MarshalAs(UnmanagedType.Bool)> ByVal mem As Boolean,
<[In]>
<MarshalAs(UnmanagedType.LPArray)> ByVal XMfile As Byte(), ByVal offset As UInteger,
ByVal length As UInteger, ByVal flags As UInteger) As Integer
<DllImport("BASSMOD.dll")>
Public Shared Function BASSMOD_MusicPlay() As Integer
System.IO.File.ReadAllBytes returns Byte() not a String. It loads a file from a string, not a Resource
(info).
```

If you compile the GitHub link I can confirm it works. All about knowing how things work...

If this is too complicated you can use [LibVLC](#) as they support xm files. Might be slightly simpler, with a bigger dependency of course. Just hide the "player" from the UI...

Find it funny the same GitHub repo was linked...

Working VB.NET code (this is a console app OP, take the class and set up the main method however you need it implemented in your program). (If you want to test, De-base64 the github audio file, copy the dll file to the debug folder, and compile for x86/32-bit architecture). You can swap in your xm file too...

```
Code Imports System.Runtime.InteropServices
```

```
Module ThisModuleNameDoesntMatter
```

```
Class PlayXM
```

```
Private data As Byte()
```

```
Public Sub New(ByVal FileName As String)
```

```
data = System.IO.File.ReadAllBytes(FileName)
```

```
End Sub
```

```
Public Sub New(ByVal XMData As Byte())
```

```
data = XMData
```

```
End Sub
```

```
Friend Structure BASSMOD_CONSTANTS
```

```
Public Const BASS_MUSIC_LOOP As Integer = 4
```

```
Public Const BASS_MUSIC_RAMPS As Integer = 2
```

```
Public Const BASS_MUSIC_SURROUND As Integer = 512
```

```
Public Const BASS_MUSIC_CALCLEN As Integer = 8192
```

```
Public Const BASS_SYNC_END As Integer = 2
```

```
End Structure
```

```
<DllImport("BASSMOD.dll")> Public Shared Function BASSMOD_Init(ByVal device As Integer,  
ByVal frequency As UInteger, ByVal flags As UInteger) As Integer
```

```
End Function
```

```
<DllImport("BASSMOD.dll", CharSet:=CharSet.Auto)> Public Shared Function  
BASSMOD_Free() As Boolean
```

```
End Function
```

```
<DllImport("BASSMOD.dll")> Public Shared Function
```

```
BASSMOD_MusicLoad(<MarshalAs(UnmanagedType.Bool)> ByVal mem As Boolean, <[In]>
```

```
<MarshalAs(UnmanagedType.LPArray)> ByVal XMfile As Byte(), ByVal offset As UInteger, ByVal  
length As UInteger, ByVal flags As UInteger) As Integer
```

```
End Function
```

```
<DllImport("BASSMOD.dll")> Public Shared Function BASSMOD_MusicPlay() As Integer
```

```
End Function
```

```
Public Sub Play()
```

```
BASSMOD_Init(-1, 44100, 0)
```

```
BASSMOD_MusicLoad(True, data, 0, CUInt(data.Length),
```

```
BASSMOD_CONSTANTS.BASS_MUSIC_LOOP Or
```

```
BASSMOD_CONSTANTS.BASS_MUSIC_RAMPS Or
```

```
BASSMOD_CONSTANTS.BASS_MUSIC_SURROUND Or
```

```
BASSMOD_CONSTANTS.BASS_MUSIC_CALCLEN)
```

```
BASSMOD_MusicPlay()
```

```
End Sub
```

```
Public Sub [Stop]()
```

```
BASSMOD_Free()
```

```
End Sub
```

```
End Class
```

```
Sub Main()
```

```
Dim keypress As New ConsoleKeyInfo()
```

```
Dim XM As PlayXM
```

```
Try
```

```
XM = New PlayXM("i_loved_you_by_heart.xm")
```

```

    XM.Play()
Catch ex As Exception
    Console.WriteLine(ex.Message)
End Try
keypress = Console.ReadKey(True)
Select Case keypress.Key
    Case ConsoleKey.Escape
        XM.[Stop]()
End Select
Console.ReadLine()
End Sub

```

End Module

You should be able to use a resource even, like so: XM = New
PlayXM(My.Resources.resourceName)... Just make sure it is File-type:Binary (Properties of
Resources Tab)...

0xDEAD10CC - October 27, 2019, 9:29 pm

Personally I'd use minifmod.. This is the typical library used for most demoscene projects unless
you've developed your own minimalistic library which benefits the file reduction techniques used.

Anesthesia_Love - October 29, 2019, 6:28 pm

The Tutorial is pretty much on point on how to do this.

Whats so stupid is you need it to use x86 not 64 for it to even work properly. also you can play from
resources without an issue.

fisica638 - October 30, 2019, 10:48 am

It took me only a few minutes of basic documentation and 30m to do the program.

Well I have 2 ways to do it:

1) use the original api in c ++ and code a complete Wrapper on vb.net.

R : It will take me longer. So I did not. settle for the wrapper bug I found on the internet.

2) Use a Wrapper on .net that I found on the internet. Go a little bug.

R : Do not debug the program or it will give you an error. (It's the Wrapper, it seems that it
doesn't detect the api in c ++ or something)

You have to build it and look for it manually in the debug folder and execute it that way.

Well it works great. It is what you wanted not?

Link : github.com/salsoft2111995/XM-Player/releases/tag/1.0

Please Comment! Thank you. ?

requiredhq - November 4, 2019, 11:13 am

(Oct 30, 2019 10:48 AM)fisica638 Wrote: [\[CITE\]](#) It took me only a few minutes of basic
documentation and 30m to do the program.

Well I have 2 ways to do it:

1) use the original api in c ++ and code a complete Wrapper on vb.net.

R : It will take me longer. So I did not. settle for the wrapper bug I found on the internet.

2) Use a Wrapper on .net that I found on the internet. Go a little bug.

R : Do not debug the program or it will give you an `_error_`. (It's the Wrapper, it seems that it doesn't detect the api in c ++ or something)

You have to build it and look for it manually in the debug folder and execute it that way.

Well it works great. It is what you wanted not?

Link : github.com/salsoft2111995/XM-Player/releases/tag/1.0

Please Comment! Thank you. ?

Hey man looks awesome, good work. I'm having trouble importing the dlls into my own project, something about a com reference. I added them as existing items instead, used some of your code and am getting a namespace error on Imports Un4seen.BassMOD.BassMOD.

Any idea whats going on?

HFDone - November 4, 2019, 12:25 pm

you need to download from the link the first file which include the BassMOD dll and to add it to your project

fisica638 - November 4, 2019, 1:39 pm

(Nov 4, 2019 12:13 PM)requiredhq Wrote: [\[CITE\]](#) Any idea whats going on?

You are only going to add "Bassmod.net.dll" to the project, the other is not. the other stays with the .exe but don't add it to the project.

From what I can appreciate you didn't add it well. That's why he doesn't detect it.

Steps :

Step 1 and 2. Select "PROJECT" and go to the last option.

The Properties tab of your Project was opened. Great ! you're doing well.

Step 3 and 4. Select "References" and "Add" button.

That opened a dialog box.

Step 5 and 6. Go to "Browser" and press the "Browser" button.

It will open another dialog box. with which you are going to look for the DLL "BassMOD.Net.dll"

Step 7 and 8.You will search for the DLL, and press Add.

Step 9 and 10. Press OK (Make sure the DLL you just added is Framed.)

Ready You are done, now you can close the Properties tab and continue with your code.
Remember to put now if:

```
PHP Code
<?php
Imports Un4seen.BassMOD.BassMOD
```

The basic thing you learn in VB.NET, c#.net , VC++ is to use the IDE (VisualStudio) correctly.
You should know that already.

Regards . I hope I've helped.

requiredhq - November 6, 2019, 10:34 pm

(Nov 4, 2019 02:39 PM)fisica638 Wrote: [CITE](#) You are only going to add "Bassmod.net.dll" to the project, the other is not. the other stays with the .exe but don't add it to the project.

From what I can appreciate you didn't add it well. That's why he doesn't detect it.

Steps :

Step 1 and 2. Select "PROJECT" and go to the last option.

The Properties tab of your Project was opened. Great ! you're doing well.

Step 3 and 4. Select "References" and "Add" button.

That opened a dialog box.

Step 5 and 6. Go to "Browser" and press the "Browser" button.

It will open another dialog box. with which you are going to look for the DLL "BassMOD.Net.dll"

Step 7 and 8.You will search for the DLL, and press Add.

Step 9 and 10. Press OK (Make sure the DLL you just added is Framed.)

Ready You are done, now you can close the Properties tab and continue with your code.
Remember to put now if:

PHP Code

```
<?php  
Imports Un4seen.BassMOD.BassMOD
```

The basic thing you learn in VB.NET, c#.net , VC++ is to use the IDE (VisualStudio) correctly.
You should know that already.

Regards . I hope I've helped.

That solved the issue!

Dude you did amazing, I'm sure this will help anybody else with this issue. One last thing I am having trouble with

How do you pull the xm from your resources? Say I have song.xm in resources, I've tried
Code Dim DirXMfile As String = My.Resources.Song but it is saying it's read only.

After watching that video it says to do resource to array, has something like Code stream =
Reflection.Assembly.GetExecutingAssembly().GetManifestResourceStream("XXXX.song.xm")
but I'm not sure what that means or how to use it

See I am trying to play an xm in the background of a form when it opens.
Thanks for making this possible and all your help

fisica638 - November 7, 2019, 4:44 am

There is no need to extract the resource, you can read the file directly.

Example :

PHP Code

```
<?php  
Dim XmFileData As Byte() = My.Resources.sound  
Dim Inicializacion As Boolean = BASSMOD_Init(-1, 44100, 0)  
  
If Inicializacion = False Then  
    MsgBox("Error; Failed to Start BassMod")  
End  
End If  
  
BASSMOD_MusicLoad(XmFileData, 0, CUInt(XmFileData.Length),
```

BASSMOD_CONSTANTS.BASS_MUSIC_LOOP Or BASSMOD_CONSTANTS.
BASS_MUSIC_RAMPS Or BASSMOD_CONSTANTS.BASS_MUSIC_SURROUND Or
BASSMOD_CONSTANTS.BASS_MUSIC_CALCLEN)
BASSMOD_MusicPlay()

On the other hand If you really want to Extract the Resource "Sound.xml" You can do it like this:

PHP Code

```
<?php
```

```
Imports System.IO
```

PHP Code

```
<?php
```

```
Dim XmFileData As Byte() = Nothing
```

```
Dim DirXMfile As String = Path.GetTempPath & "sound.xml"
```

```
File.WriteAllBytes(DirXMfile, My.Resources.sound)
```

```
XmFileData = File.ReadAllBytes(DirXMfile)
```

If you want, you send me your Project and I help you. My Discord: Destroyer#3527

Eternal Darkness - November 7, 2019, 6:22 pm

(Nov 6, 2019 11:34 PM)requiredhq Wrote: [\[CITE\]](#) Dim DirXMfile As String = My.Resources.Song

Here you go:

(Oct 27, 2019 02:31 PM)Eternal Darkness Wrote: [\[CITE\]](#) You should be able to use a resource even, like so: XM = New PlayXM(My.Resources.resourceName)... Just make sure it is File-type:Binary (Properties of Resources Tab)...

You declare a "String" variable, what you really want is a Byte array as in "Byte()".

Try: Dim DirXMfile As Byte() = My.Resources.Song (make sure the resource is of type Binary)

requiredhq - November 7, 2019, 8:51 pm

(Nov 7, 2019 05:44 AM)fisica638 Wrote: [\[CITE\]](#) There is no need to extract the resource, you can read the file directly.

Example :

PHP Code

```
<?php
```

```
Dim XmFileData As Byte() = My.Resources.sound
```

```
Dim Inicializacion As Boolean = BASSMOD_Init(-1, 44100, 0)
```

```
If Inicializacion = False Then
```

```
MsgBox("_error_; Failed to Start BassMod")
```

```
End
```

```
End If
```

```
BASSMOD_MusicLoad(XmFileData, 0, CUInt(XmFileData.Length),
```

```
BASSMOD_CONSTANTS.BASS_MUSIC_LOOP Or BASSMOD_CONSTANTS.
```

```
BASS_MUSIC_RAMPS Or BASSMOD_CONSTANTS.BASS_MUSIC_SURROUND Or  
BASSMOD_CONSTANTS.BASS_MUSIC_CALCLEN)  
    BASSMOD_MusicPlay()
```

On the other hand If you really want to Extract the Resource "Sound.xml" You can do it like this:

PHP Code

```
<?php
```

```
Imports System.IO
```

PHP Code

```
<?php
```

```
Dim XmFileData As Byte() = Nothing
```

```
    Dim DirXMfile As String = Path.GetTempPath & "sound.xml"
```

```
    File.WriteAllBytes(DirXMfile, My.Resources.sound)
```

```
    XmFileData = File.ReadAllBytes(DirXMfile)
```

If you want, you send me your Project and I help you. My Discord: Destroyer#3527
I added you. The name is JBlanket.

I've run into another error, when I try to start the program it says there's an error on Code Dim Inicializacion As Boolean = BASSMOD_Init(-1, 44100, 0) and on running it says bassmod.dll not found. When I try to add bassmod.dll as a reference, it says it is not a valid assembly.

:|

Did some research and "Register for COM interop" is greyed out, if that means anything. Weird because I added the BassMOD.net.dll just fine and now I can't readd that one.

fisica638 - November 8, 2019, 4:11 am

(Nov 7, 2019 09:51 PM)requiredhq Wrote: [\[CITE\]](#) I've run into anothererror_, when I try to start the program it says there's an _error on

Code Dim Inicializacion As Boolean = BASSMOD_Init(-1, 44100, 0) and on running it says bassmod.dll not found. When I try to add bassmod.dll as a reference, it says it is not a valid assembly.

As I mentioned before. Do not add "bassmod.dll", that DLL only goes in the same folder next to the .EXE.

How to correct that error? :

1)Do not Debug the Program.

You have to build it and search and run it manually.

Find the .exe and open it.

(Nov 7, 2019 09:51 PM)requiredhq Wrote: [\[CITE\]](#) Did some research and "Register for COM interop" is greyed out, if that means anything. Weird because I added the BassMOD.net.dll just fine and now I can't readd that one.

As I said, the only dll you will add to the Project is: "BassMOD.net.dll" (Wrapper for the original dll)

The "bassmod.dll" DLL will only be placed in the same folder as the .exe

smackjack - September 1, 2020, 7:54 pm

it was already discussed...

.../showthread.php?tid=5451236

thx to terminated ?

But it is still interesting how you would get a single .exe file via vb.net

I mean there are a lot of keygens without that "bassmod.dll" file. So it has to be possible i guess...

fisica638 - September 2, 2020, 11:37 am

(Sep 1, 2020 07:54 PM)smackjack Wrote: [\[CITE\]](#) But it is still interesting how you would get a single .exe file via vb.net

I mean there are a lot of keygens without that "bassmod.dll" file. So it has to be possible i guess...

1) You can Protect the application with DotNetPatcher and merge the dependencies, that would leave only one .exe.

2) You can also write your own codec to reproduce that format, that would be a difficult task.