

## Music Scripting Guide for RPG Maker 2

---

Table of Contents:

I. Version History

II. Legal Information

III. Preface

IV. Terminology

V. Script Commands

VI. How to Script Custom Music

XX. Custom Music All The Time Tricks - courtesy of Doan the Nado, Vespuleth, and Nash

XIX. Frequently Asked Questions

---

### I. Version History

**Version 0.9:** Completed 2-07-07. Finished the guide up to a great stopping point where you can do almost everything - would like feedback from you guys! Some major things are missing and will of course be added (obviously, you can see the placeholders for them even) such as the Custom Music All The Time Tricks - courtesy of Doan the Nado, Vespuleth, and Nash and Tempo times for various SFXs. However, that is all that is not perfect, I think! So, um, yeah... Cool, huh? Let me know what you think so I can improve it, yes? No?.....Yes?!

---

### II. Legal Information

This guide is copyrighted by William Kirk. It is intended for private use only and cannot be used for profitable or promotional purposes regardless of the situation. Breaking this rule in any way, shape, or form is a direct violation of copyright law.

Unauthorized reproduction and distribution of this document, or any portion of it, may result in severe civil and criminal penalties. Any game titles, characters, and miscellaneous names are copyrighted by their respective companies.

If you are a webmaster of a site that wishes to post this document, send an email to [williampaladin@hotmail.com](mailto:williampaladin@hotmail.com) asking for permission. If permission is granted you can post it, but only **as long as its content is not changed in ANY WAY**. Failing to do so will result in losing

your permission permanently.

The newest version will always be available at <http://www.doanthenado.com/guides.htm> .

---

### III. Preface

I've written this guide because I believe it will help a great many people working with RPG Maker 2. It is designed to show and teach you exactly how to make your very own custom music with RPG Maker 2's Script Editor. I know it's awfully daunting at first, but I'll walk you through it and show you the ropes.

If you begin to have trouble on Step #1, I suggest you visit Neo Samurai's 'Beginner RPGM2 FAQ' and Doan the Nado's 'World Creation Guide' at <http://www.doanthenado.com/guides.htm> and doyleman's 'Script Tutorial' at [http://www.geocities.com/nate\\_dog7\\_7/ST.htm](http://www.geocities.com/nate_dog7_7/ST.htm) . I do not assume much of you, but I do assume that you have learned the common, basic uses of Scripts and Events (like basic townspeople) and the like.

In my opinion, Neo Samurai's 'Beginner RPGM2 FAQ', Doan's 'World Creation Guide', and doyleman's 'Script Tutorial' are the most thorough and straight-forward Guides for beginning RPGM2, and I am quite certain that after you do what is in their Guides you will have a much more pleasant time doing what's in my Guide.

Lastly, if you would like further help with RPG Maker 2, or actually anything game design/program/music-RPGM2-programming related, there's really only one place you should go:

**DOAN'S DOMAIN - BY FAR THE BEST PLACE ON THE NET!!!!!!!**

**Doan's Domain:** <http://doansdomain.proboards27.com/>

---

### IV. Terminology

In this Guide, you are bound to hear many of the following terms, and in order to understand this Guide, you will have to know what they mean. Fortunately for you, I've made a section specifically for the meaning of these terms so that if you find an unfamiliar term you can quickly and easily learn what it means from referring back to this section.

Also, unfortunately it will most likely be rather difficult to make a good song without being able to play at least one instrument. It is still possible, but it will take a MUCH longer time, because instead of just strumming on a guitar or jammin' on a piano and hearing the results instantly, you'll have to grudgingly program the Notes and Timing into RPGM2 then "Test Play" for the results which takes MUCH longer and is MUCH more confusing. What I mean by MUCH more confusing is that, for example, up until just recently after about two years of music-RPGM2-programming, I would often play something then have to try TWO TO SEVEN times to program

it into RPGM2 correctly. Ugh! Oh well, hopefully this Guide will help you, ya?

## **A. Sounds**

### **1. Notes**

A Note is a particular frequency that, when played at the same time as another sound of the same frequency, sounds good.

Total, there are twelve notes in RPGM2: A, A#, B, C, C#, D, D#, E, F, F#, G, and G#. But while these are the actual notes, RPGM2 shows them as: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, and 11. All you need to remember is that -12, 0, and 12 will sound good together because they are all of the same note. Likewise, -5 and 7 will sound good together. All you do is take the first number and add or subtract 12 to it. Pretty simple, right?

Lastly, note that this also applies when using different RPGM2 “Sound Effects” together. So, for example, a Flute sound of -5 will sound good when played at the same time as a Bass sound of -5 or 7. Pretty simple, right? And guess what, this is even the case with the “Sound Effects” you wouldn’t expect this to be the case with such as the animal sounds, percussion sounds, etc.!

The annoyance comes when you can’t fit your song into RPGM2 because of the lame 25 Note limit.

If that turns out to be the case, try shifting it so that, for example, instead of your song (theoretically speaking) ranging from -10 to +14 ranges from -12 to +12. (RPGM2 ranges from -12 to +12 for every “Sound Effect”, so you must fit it into there somehow)

Another alternative is to try and alter your song some so that different instruments come in and play different parts. This way instead of your Guitar needing to go up to +14 you could switch to the Harp and have the Harp go up to the same Note but at +2.

Also, perhaps putting the song aside for the moment and coming back to it after you’ve developed as a composer will result in you being able to find a nifty way to fit it into RPGM2. Either way, good luck!

### **2. Pitches**

Most of the time when Pitch is said it’s meant as an exact synonym for Note. The only difference is that a Pitch can be in between Notes and thus sound bad, but that will never be the case with RPGM2 seeing as they don’t have any  $1\frac{1}{2}$  Pitches available for sounds.

So in the end, when talking about RPGM2-music-programming sounds, Pitch = Note.

### **3. Melodies and Rhythms**

Melodies are what make music satisfying, and are the backbone of a song, while Rhythms are supporting beats that improve the song by adding something new throughout it. However, a decent song can still be made with a Rhythm as its backbone.

Over my three years of composing I’ve come to a much better understanding of what defines Melodies from Rhythms: A Melody satisfies in some indescribable way by the Notes and Timing

of it, while a Rhythm doesn't.

Take the Happy Birthday song for example. It inexplicably satisfies the listener by the change in Notes and Timing in the third and fourth repetition Measures, and is thus a Melody. Now imagine how it would be if the first two "Happy Birthday to you"s and the Notes of the corresponding words were repeated again in those third and fourth repetitions - that's a Rhythm.

Either way, don't worry about the difference between Melodies and Rhythms! If you don't get it yet, you'll come to do so with time as a composer. I'm just trying to share a little of what I've learned with you.

All you need to know about Melodies and Rhythms for RPGM2-music-programming is that those are the words I will be using. You'll also have to have a general idea of what they are and know that your song should originate from some sort of Main Melody/Rhythm (which is interestingly, also called a Theme \*smiles and giggles\* ).

## **B. Timing**

### **1. Measures**

Well, it's good that I get to this after Melodies and Rhythms seeing as the answer to the common question "How long is a Measure, usually?" is 1/4 of your Main Melody/Rhythm.

A Measure is tough to describe so as that someone who's never spent any time studying music before can understand, but the best I can do is that it's a unit of measurement for music: it measures the music by a certain time amount relating to it's beat (for example, the most basic beat is one beat per measure). Hooie, I think it'll become more clear when we get to the actual programming part of this Guide.

### **2. Frame**

Perhaps the most important word of all to know, but actually it's a word from RPGM2 so most people probably know what it is already. Note that the word Frame is commonly abbreviated as F by both RPGM2, **Doan's Domain** members, and will be by me in the sample Scripts I provide.

In RPGM2, 1 Frame is equal to 1/30th of a second, and thus 30 Frames is equal to a second. When we program our custom music in the Script Editor of RPGM2, we do it with "Sound Effects" (from here on often abbreviated as SFX) and Wait Commands. The SFX commands play a sound, then the Wait command tells RPGM2 to wait a certain number of Frames before playing the next sound. Make sense?

### **3. Notes**

A Note, in reference to Timing instead of Sounds, is when a Sound (aka. a Note) is played. For example, if you (or RPGM2) were told to play four Quarter-Notes in one Measure of 32 Frames, you'd play one Sound per 8 Frames. If you were using a percussion sound for this it would produce a beat of bum-bum-bum-bum. Right, four per Measure? Or, if you (or RPGM2) were told to play two Half-Notes in one Measure of 32 Frames, you'd

play one Sound per 16 Frames. And, if you were using a percussion sound for this it would produce a beat of bum...bum... Two per Measure. Startin' to make a little sense, huh?

### **C. The Sound Limit**

This is the real pooper... Whenever (I forget the exact number, about 5 or so) SFX have played, the next SFX that gets played will automatically force the first of those 5 SFX to stop playing. The problem with this is that it makes it a pain in the butt to add a lot of layers to a song, and to use long-lasting sounds such as the Bass, Ocarina, Flute, Violin, etc. (as opposed to quick sounds such as the Guitar 1, Piano, Harp, Banjo, etc.). And sadly, there's no way around it. Just try to plan ahead, and expect a lot of your songs to lack one thing or another (for me usually bass, but sometimes even bass and percussion)... This is what I'm referring to when I say The Sound Limit.

---

## **V. Script Commands**

### **A. Call [Script]**

This command will execute ALL the commands in the Script you choose. I'm sure you can imagine how useful this can be both for this and other things with RPGM2.

### **B. Other: Wait**

This command is rather simple. It tells RPGM2 to wait a certain number of Frames until performing the next command.

### **C. Sound: SFX**

This command plays one of RPGM2's various SFX (Sound Effects).

Pitch:

This is what RPGM2 Pitch (-12 to +12) you want the SFX to be. For more help refer to IV. A. 1. Notes and IV. A. 2. Pitches.

Tempo:

This is how long you want the SFX to continue playing the sound.

For most SFX this doesn't matter very much, but for the long-lasting sounds such as the Bass, Ocarina, Flute, Violin, etc. this is extremely important (as I'm sure you'll find out).

Nonetheless, in order to make your custom music sound as good as possible it's crucial to find the perfect Tempo for long-lasting SFX and a decent Tempo for quick SFX (such as the Guitar 1, Piano, Harp, Banjo, etc.).

Lucky for you, I've put my Tempos from my custom music in this guide for you right here! Unfortunately for you, the multiples of 5 Frames (5, 10, 15, etc.) ones aren't quite perfect (and perhaps a few other instruments), but at least they'll give you a point of reference.

Anywhoo, here you go:

---

## VI. How to Script Custom Music

### A. The Basics

We've already touched upon it a little. Basically, how you script custom music in RPGM2 is by combining Wait commands and Sound Effects so that the SFX script commands tell RPGM2 what sounds to make and the Wait commands tell RPGM2 how long to wait in between playing said Sound Effects. (Frames is our unit of measurement in time, and 30 Frames is equal to one second.)

Take this script for instance:

```
SFX: Guitar 1 Pitch 0 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 0 Tempo 300  
Wait 8F
```

When RPGM2 runs this Script, first it will play a Guitar 1 Sound Effect, then Wait 8 Frames, then play another Guitar 1 Sound Effect, then wait another 8 Frames. Pretty simple, right? ;)

### B. Organization

Doing it this way will make it MUCH, MUCH easier and quicker to make changes and additions to your work as well as save you a whole poopload of memory, so it REALLY is best you set-up your scripts like this. Read through the scripts and what I say afterwards.

NOTE: I know this song \*\*\*\*s huge poopberries. I just made it up on the spot since I was in a hurry to finish since several people needed help and wanted the FAQ done.

```
Song Main Script_  
Call [Instrument1 Main Script]  
Call [Instrument2 Main Script]
```

```
Instrument1 Main Script_  
Call [Instrument1 Verse Measure1]  
Call [Instrument1 Verse Measure2]  
Call [Instrument1 Verse Measure1]  
Call [Instrument1 Verse Measure3]  
Call [Instrument1 Chorus Measure1]  
Call [Instrument1 Chorus Measure2]
```

Call [Instrument1 Chorus Measure1]  
Call [Instrument1 Chorus Measure3]

Instrument1 Verse Measure1\_  
SFX: Guitar 1 Pitch 0 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 0 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 0 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 0 Tempo 300  
Wait 8F

Instrument1 Verse Measure2\_  
SFX: Guitar 1 Pitch 2 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 2 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 2 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 2 Tempo 300  
Wait 8F

Instrument1 Verse Measure3\_  
SFX: Guitar 1 Pitch 5 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 5 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 5 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 5 Tempo 300  
Wait 8F

Instrument1 Chorus Measure1\_  
SFX: Guitar 1 Pitch 7 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 7 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 7 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 7 Tempo 300  
Wait 8F

Instrument1 Chorus Measure2\_  
SFX: Guitar 1 Pitch 9 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 9 Tempo 300  
Wait 8F

SFX: Guitar 1 Pitch 9 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 9 Tempo 300  
Wait 8F

Instrument1 Chorus Measure3\_  
SFX: Guitar 1 Pitch 12 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 12 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 12 Tempo 300  
Wait 8F  
SFX: Guitar 1 Pitch 12 Tempo 300  
Wait 8F

Instrument2 Main Script\_  
Call [Instrument2 Verse Measure1]  
Call [Instrument2 Verse Measure2]  
Call [Instrument2 Verse Measure1]  
Call [Instrument2 Verse Measure2]  
Call [Instrument2 Verse Measure1]  
Call [Instrument2 Verse Measure2]  
Call [Instrument2 Verse Measure1]  
Call [Instrument2 Verse Measure3]  
Call [Instrument2 Chorus Measure1]  
Call [Instrument2 Chorus Measure2]  
Call [Instrument2 Chorus Measure1]  
Call [Instrument2 Chorus Measure3]

Instrument2 Verse Measure1\_  
SFX: Snare 1 Pitch 0 Tempo 150  
Wait 16F  
SFX: Snare 1 Pitch 0 Tempo 150  
Wait 8F  
SFX: Snare 1 Pitch 0 Tempo 150  
Wait 8F

Instrument2 Verse Measure2\_  
SFX: Snare 1 Pitch 2 Tempo 150  
Wait 16F  
SFX: Snare 1 Pitch 2 Tempo 150  
Wait 8F  
SFX: Snare 1 Pitch 2 Tempo 150  
Wait 8F

Instrument2 Verse Measure3\_  
SFX: Snare 1 Pitch 5 Tempo 150  
Wait 16F

SFX: Snare 1 Pitch 5 Tempo 150  
Wait 8F  
SFX: Snare 1 Pitch 5 Tempo 150  
Wait 8F

Instrument2 Chorus Measure1\_  
SFX: Snare 1 Pitch 7 Tempo 150  
Wait 16F  
SFX: Snare 1 Pitch 7 Tempo 150  
Wait 8F  
SFX: Snare 1 Pitch 7 Tempo 150  
Wait 8F

Instrument2 Chorus Measure2\_  
SFX: Snare 1 Pitch 9 Tempo 150  
Wait 16F  
SFX: Snare 1 Pitch 9 Tempo 150  
Wait 8F  
SFX: Snare 1 Pitch 9 Tempo 150  
Wait 8F

Instrument2 Chorus Measure3\_  
SFX: Snare 1 Pitch 12 Tempo 150  
Wait 16F  
SFX: Snare 1 Pitch 12 Tempo 150  
Wait 8F  
SFX: Snare 1 Pitch 12 Tempo 150  
Wait 8F

So, as you see, when you set it up this way, when you go back and edit your songs you won't have to change the part of the song that is now in the script titled (for example) Instrument2 Verse Measure1 a WHOPPING four times!!! This will SIGNIFICANTLY save you time, memory, and effort down the road as well as allow you to customize your songs with greater ease.

And to explain how this allows you to customize your songs with greater ease, you could have an instrument and do this in its Main Script (assuming it had a song structure of two verses, three choruses, and a bridge):

```
InstrumentPoop Main Script_  
Wait X Frames (X representing number of Frames in first Verse plus number of Frames in first Chorus)  
Call [InstrumentPoop Verse]  
Call [InstrumentPoop Chorus]  
Call [InstrumentPoop Bridge]  
Call [InstrumentPoop Chrous]
```

Then later, if you tried that out in "Test Play" and found that you thought that InstrumentPoop Chorus should also play during the first Chorus, you can quickly and easily make that change in

just two little edits (changing the number of Frames in the Wait command and adding a Call [InstrumentPoop Chorus] between the Wait and Call [InstrumentPoop Verse] commands) as opposed to having to look through a huge script and figure out where to add in the proper instrument commands to make InstrumentPoop Chorus play during the first Chorus too.

---

—

**XX. Custom Music All The Time Tricks** - courtesy of Doan the Nado, Vespueth, and Nash

---

—

### **XIX. Frequently Asked Questions**

1. I tried scripting custom music but when I went to listen to it in Test Play all I heard was this insanely loud, jumbled sound. I looked through my scripts and as far as I know there's nothing wrong with them. What did I do wrong?

Ah yes, you always want to be on Script: Apply In Order. It runs on Apply In Order automatically, so as long as you don't put the Script: Apply Together command before your custom music you'll be fine. The exception to this is Doan and Ves's cool Custom Music All The Time Trick which is detailed above, but otherwise you'll want to be on Script: Apply In Order. For an explanation on this press start on Script: Apply In Order for RPGM2's in-game help of it, check out Dungeon Warden's "Advanced RPGM2 FAQ" found at <http://doanthenado.com/guides.htm> , or ask at **DOAN'S DOMAIN, THE BEST PLACE ON THE NET!!!!**

2. Poop.

---

—