SPJHELPER Version 6.0 ©2014, Fred Miller

An Aid in the Preparation of SPJ Files for Digitrax SoundFX® Decoders

Development of custom software to load into the Digitrax SoundFX® decoder is not a trivial task. Even those users who have some programming experience have found the development of the necessary macro assembly language code, using the Digitrax Sound Definition Language (SDL) a challenging effort. The SPJHELPER© was developed to aid in the construction of those programs.

Version 6.0 of S supports the newer Digitrax Series-6 Premium 4-voice, 16 bit decoders as well as the 3-voice, 8 bit Legacy and Standard decoders.

CONTENTS

APPENDICES

Overview of the SPJHELPER© Software

SPJHELPER© actually has two different personalities.

- It has functional support for the *non-techie* to enable creation of all the components necessary for the Digitrax SoundLoaderV2™ software to load into a Digitrax SoundFX® decoder, and

- Alternatively, SPJHELPER© has easily accessible tools for the more *experienced programmer* to build the necessary files with ease.

Installation and operation SPJHELPER© also includes easy access to optional installation of support software including a text file editor (NotePad+++), a sound file editor (WavePad or Audacity), the MicroChip Macro Assembler, and the Digitrax SoundLoaderV2™. All of these programs have been released by their developers without charge. SPJHELPER© can install those software tools with a click of a button.

Detailed instructions for the operation of SPJHELPER© are presented later in this document. However, it will be useful to better understand the Digitrax decoders with the following overviews

Digitrax SoundFX® decoders

The newly released Digitrax Series 6 Premium SoundFX® decoders are capable of playing sounds in four “voices.” (The Series 6 Standard and the Legacy Digitrax decoders can only play sounds in three "voices.") That means the decoders can play up to four (or three for Legacy) different sounds at the same time. The program running inside the decoder can initiate these sounds based on a number of internal or external activities, for example pressing a throttle function key. The sounds defined in that program have a priority for playing, i.e., within one of the voices, a higher priority sound could interrupt a lower priority sound. Care must be exercised in setting those priorities

In addition, the program running in the decoder could have multiple “schemes” or personalities. Only one is active at a time but can be changed with a Configuration Variable (CV) setting. Many of the original Digitrax SoundFX® decoders were released with a sound program of two schemes, one a generic diesel locomotive and the other a steam locomotive. Series 6 decoders have 8 schemes. Changing the Scheme Selection CV selected the “personality” of the decoder, e.g., from diesel to steam. Other sound programs have been developed which provide variations on the operating characteristics and sounds by changing the Scheme Selection CV. The *non-technie personality* of SPJHELPER© does not support development of multiple Scheme projects, however the more *complex personality*, where the user is developing full code, can support multi-scheme projects.

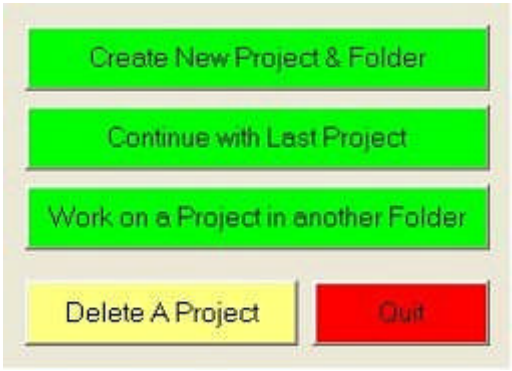
Digitrax SoundFX® decoders are sold with pre-loaded sounds appropriate for generic locomotives. The Digitrax web site also offers a selection of sound projects to download from their Sound Depot. It is possible to customize a decoder's sound scheme by using sound recordings in .wav format. You can replace any locomotive sound part (a chuff, or a brake squeal, etc) with an actual sound recording. This is done with the PC-based program SoundLoaderV2™ in conjunction with a Digitrax PR3 programmer. See Appendix C for additional information about construction of sound files.

Of particular interest, however, is that the operating characteristics of the decoders can be established by loading a new program to run in the decoder. This is accomplished by developing a file called the Sound Project File or SPJ. The Digitrax SoundLoaderV2™ program is used to download the necessary components of the SPJ files to the sound decoder in the locomotive on a programming track via the Digitrax PR3. The SPJ file is actually a collection of files including the sound files and the operating program. Appendix A describes the contents of Sound Project Files (.SPJ). SoundLoaderV2™ software has other useful capabilities. These are further described in Appendix B.

Operating the SPJHELPER© Software

At initial setup SPJHELPER© scans the user's PC for the four software tools used. If the recommended software is not found, the user is given the opportunity to install that software, or a substitute. For example the recommended editor is NotePad+++, however the MS NotePad or MS WordPad could be substituted. See further information about the Support Software in Appendix E. These selections can be changed later .In addition the user is provided an opportunity to install sample projects which were developed by SPJHelper©.

Whenever SPJHelper© is run the user is prompted to select the *complexity* of the functions desired. A selection of "Simple*, non techie*" or "*Nittie Gritties*…" sets the functions that will be made available.



After specifying the desired complexity, the availability of the *Create New*, *Continue Last* or *Another Project* function buttons will be based on whether SPJHELPER© has been used with a previous project, or not. If none were found, a new project can be started in a new sub-folder within the recommended SPJ_Projects folder, or previous projects can be located on the users PC and import files into sub project folders.

SPJHELPER© likes to work with project sub-folders within the overall *C:\SPJ_Projects* folder. The project sub-folders will be created with the same name as the project under development. The files prepared by the software will have the same base name with applicable file extents. For example, for a project called "DIESEL" the files created will be DIESEL.asm, DIESEL.map, etc., all saved in the project folder C:\SPJ_Projects\DIESEL\

Developing SPJs using the Simple Design Tool

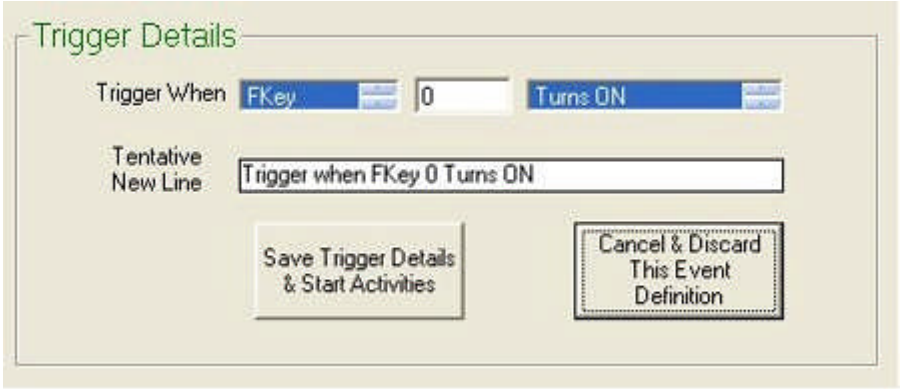
```
==============================
E:8 Trigger when FKey 2 While On
Set Volume by value in CV 141
Play Diesel_horn_begin.wav
Play Diesel_horn_cont.wav
Play Diesel_horn_end.wav
==============================
```

The DESIGN tool is based on development of descriptions, or scripts, associated with triggered Events. Those Events can be triggered by *Function Keys, Timers, Power-On, CV Reset activity*, etc. Once the Event triggering parameters are selected, the activities that will occur with the triggered event are described. Activities can include *Playing* a sound clip*,* setting *Timers, Delaying* a specified period of time, etc.

SPJHELPER© displays a screen with three or four columns for the three or four “voices” and 10 cells for events within each voice, (30 to 40 events within the project). Provision is made to select either a Legacy/Series-6 Standard (8-bit 3 Voice) project, or a Series-6 Premium

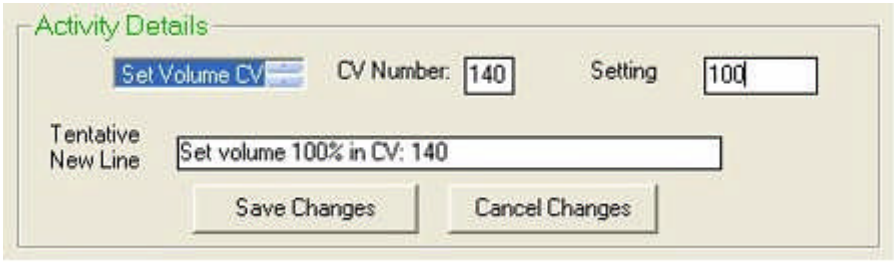
Right-Mouse clicking on any cell will bring up a menu, enabling the user to ADD a new EVENT, if none existed in that cell, or Edit, delete or show details about the Event clicked on.

The ADD or EDIT Event Screens provide a selection of available Triggers that may be used to define the Event script.



When a Trigger Event has been described, the ADD or EDIT Event Screens provide a selection of

available Activities that may be used to define the Event script.

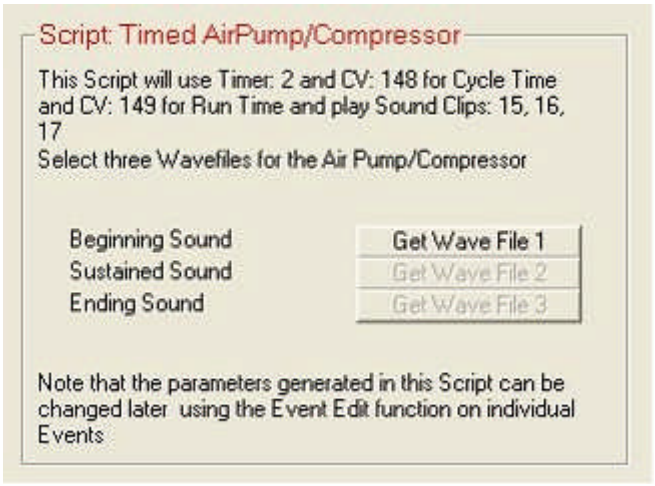


Once Events have been developed, they can be dragged (Left-Mouse) to any available cell including across Voice Columns.

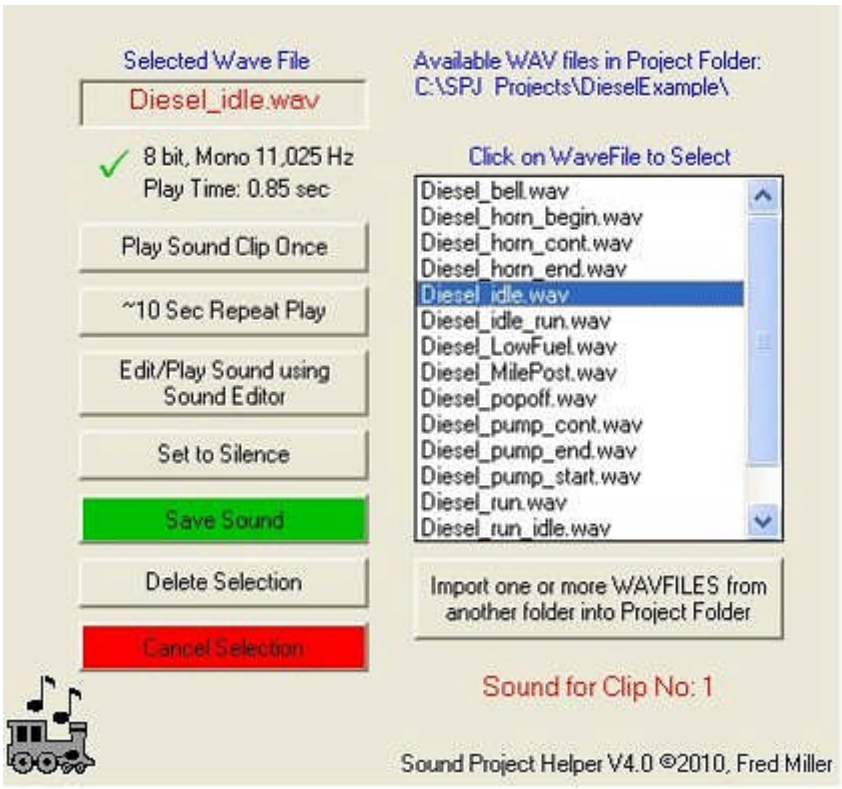
A selection of 11 “pre-packaged” scripts are available in SPJHELPER©. Some of the pre-packaged scripts will generate multiple Events and place those Events in the appropriate cells and Voices.

Additional scripts may be added in later versions of SPJHELPER©.

The Pre-Package script tool prompts the user for certain parameters, e.g., identifying the Sound Clips to use, but makes assumptions about most of the parameters necessary to develop one or more Events and associated Activities. The resulting events can be subsequently edited, if desired, using the Event Edit facility described above.



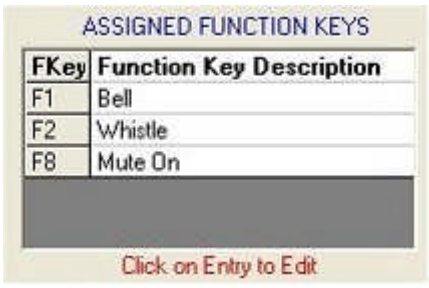
When referencing Sounds in any of the Activities of an Event Script, the user is given the opportunity to select Sound Wavefiles from any location in the user's PC, and transfer them to the Project Folder.



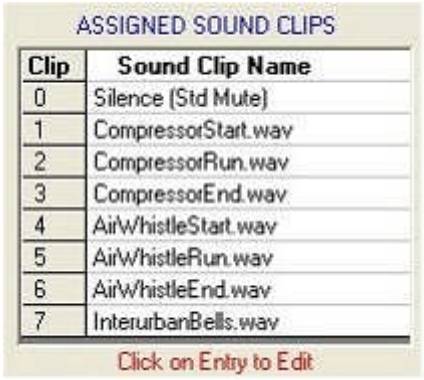
The Sound Wavefile will be checked for Sound-Loader® acceptable formats (8, 12 or 16 bit, Mono 11,025 hz - see Appendix C) and the total playing time is displayed.

A selected Sound Wavefile can also be played through the PC speakers directly from SPJHELPER©. The file can also be edited using the WavePad (or equivalent) audio editing software.

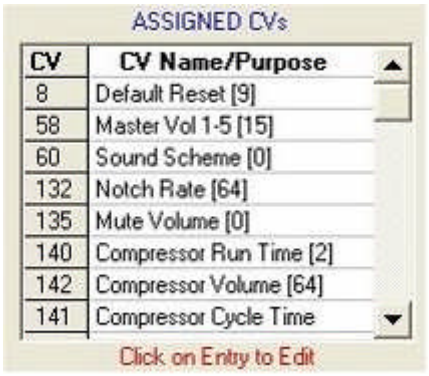
As the Event Script is developed using the *Add Event* or *Pre-packaged Scripts*, any CVs, Sound Clips, or Function Keys referenced, will be listed in the left column of the Design Screen.



Left-Mouse clicking on any of those left-column descriptions will enable editing of the descriptive information. For example clicking on a Function Key description will let the user change the default name (F2) to something more descriptive (Whistle). After the text has been changed, an [Enter] key will save the entry. Deleting all of the text (and an [Enter] key) will delete that entry.

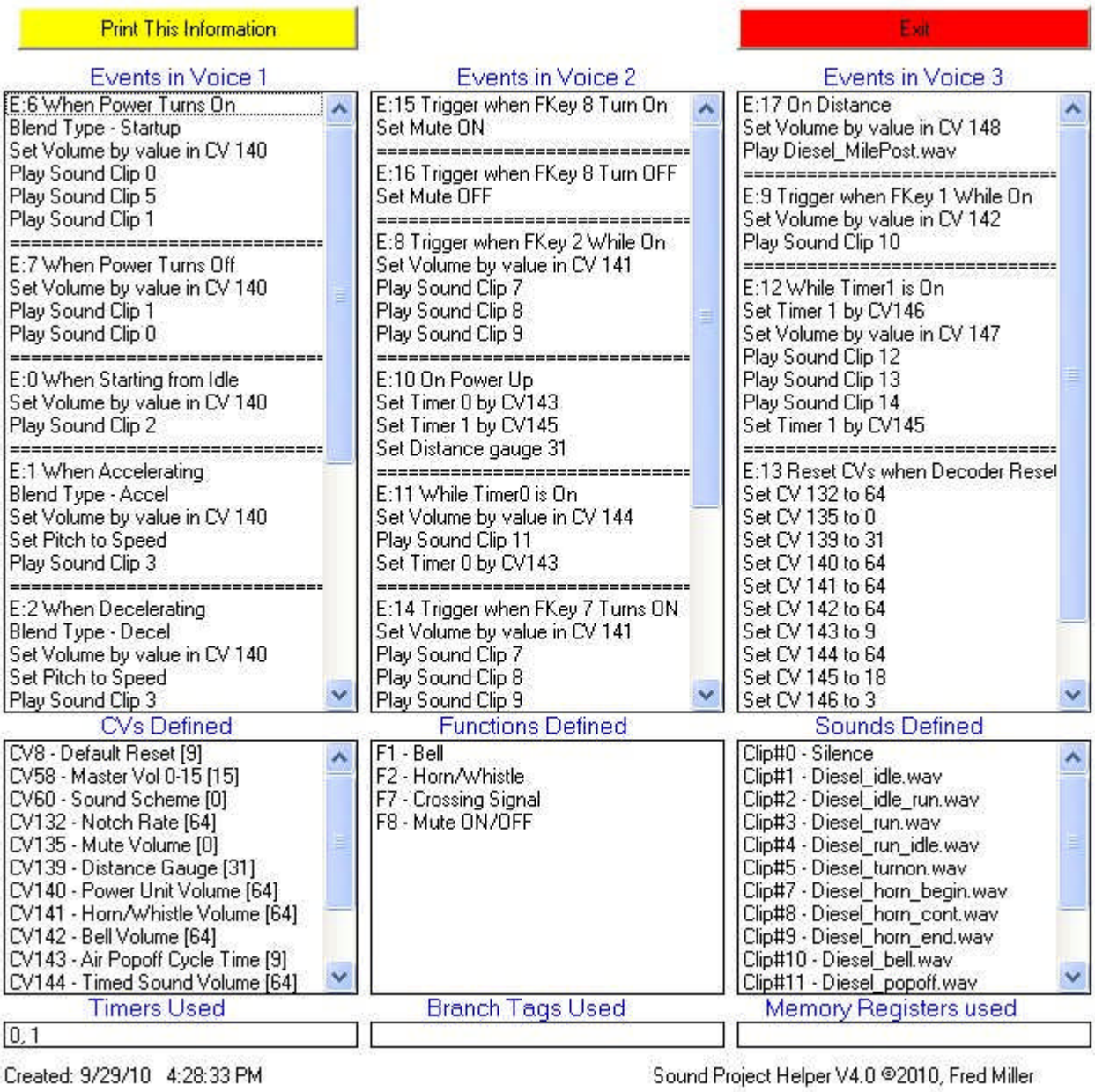


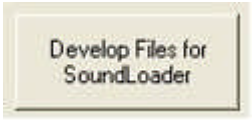
Left-Mouse clicking on an Assigned Sound Clip description will bring up the Sound Menu described above. That menu will let the user *find, play*, or *edit* a sound clip. Saving the clip from that pop-up menu will place the name in the left-column.



CV names can similarly be changed to something more descriptive. It is useful to also indicate the default value for this CV by typing the value in square brackets. This will be useful when running the SoundLoaderV2™ program. After the text has been changed, an [Enter] key will save the entry. Deleting all of the text (and an [Enter] key) will delete that entry.

As the user is developing the Event Scripts, it might be useful to display and/or print out the details of the design in progress.





When the Design is complete, a click on the Develop Files button will initiate the process of building the necessary files for loading into the SoundLoaderV2™ software. This will include the completed software file (.SDF) created from the Macro Assembler, the Sound Handle File (.MAP) and the Project Descriptive File (.TXT)

When this process is completed, SPJHELPER© will automatically start SoundLoaderV2™ with all of the files loaded. No additional "importing" is necessary, but the project must be "downloaded" to the SFX decoder using the big green button in SoundLoaderV2™.

File View COM Port SoundTest Decoder Address CV Editor IPL Help

Sound Assignments (Left Double-Click to Play or Assign, Right Click to Edit)

001	Steam_Chuff1	Steam_Chuff1.wav	0.454 Sec	5008	1390h	11KS/S, 1Ch,	16Bit, (16 bit DL)
002	Steam_Chuff2	Steam_Chuff2.wav	0.476 Sec	5248	1480h	11KS/S, 1Ch,	16Bit, (16 bit DL)
003	Steam_Chuff3	Steam_Chuff3.wav	0.449 Sec	4956	135Ch	11KS/S, 1Ch,	16Bit, (16 bit DL)
004	Steam_Chuff4	Steam_Chuff4.wav	0.456 Sec	5032	13A8h	11KS/S, 1Ch,	16Bit, (16 bit DL)
005	Steam_airpump	Steam_airpump.wav	0.979 Sec	10797	2A2Dh	11KS/S, 1Ch,	8Bit, (8 bit DL)
006	Steam_blow_start	Steam_blow_start.wav	0.351 Sec	3872	F20h	11KS/S, 1Ch,	8Bit, (8 bit DL)
007	Steam_blow_run	Steam_blow_run.wav	0.655 Sec	7232	1C40h	11KS/S, 1Ch,	8Bit, (8 bit DL)
008	Steam_blow_end	Steam_blow_end.wav	0.3 Sec	3313	CF1h	11KS/S, 1Ch,	8Bit, (8 bit DL)
009	Steam_Bell	Steam_Bell.wav	1.804 Sec	19892	4DB4h	11KS/S, 1Ch,	16Bit, (16 bit DL)
010	Steam_Whistle_start	Steam_Whistle_start.wa	0.691 Sec	7626	1DCAh	11KS/S, 1Ch,	16Bit, (16 bit DL)
011	Steam_Whistle_run	Steam_Whistle_run.wav	2.328 Sec	25674	644Ah	11KS/S, 1Ch,	16Bit, (16 bit DL)
012	Steam_Whistle_end	Steam_Whistle_end.wav	1.709 Sec	18844	499Ch	11KS/S, 1Ch,	16Bit, (16 bit DL)
013	Steam_Water_start	Steam_Water_start.wav	0.975 Sec	10752	2A00h	11KS/S, 1Ch,	16Bit, (16 bit DL)
014	Steam_Water_run	Steam_Water_run.wav	1.875 Sec	20676	50C4h	11KS/S, 1Ch,	16Bit, (16 bit DL)
015	Steam_Water_end	Steam_Water_end.wav	2.348 Sec	25888	6520h	11KS/S, 1Ch,	16Bit, (16 bit DL)
016	Steam_Boiler	Steam_Boiler.wav	0.732 Sec	8080	1F90h	11KS/S, 1Ch,	8Bit, (8 bit DL)
017	Steam_LowWater	Steam_LowWater.wav	1.781 Sec	19645	4CBDh	11KS/S, 1Ch,	8Bit, (8 bit DL)

TRIGGERS AND ACTIVITIES FOR SIMPLE DESIGN

The following EVENT TRIGGERs are available in the Simple Design Mode.

Triggering Events	Comments
F Key	Function Keys 0-23, ON or OFF
Power	Decoder Selected ON or OFF
CV Reset	CVs reset to Factory Defaults
Speed Chg	Idle, Accel, Run, Moving, etc.
Dir Chg	Throttle Direction change
Timer	Internal Timers 0-5
CAM	External Trigger
Clock	64 msec clock tick
OPS Pgm	OPS Mode programming change
Distance	Increment to Distance Notation

The following EVENT ACTIVITIES are available in the Simple Design Mode.

Event Activities	Comments
Play Sound	Play Sound Clips
Set Vol by CV	Set Volume using a CV
Set Default Vol	Reset Volume to Default value
Set Pitch	Set Pitch relative to speed
Delay by Secs	Delay for 0.1 to 6.0 seconds
Delay by CV	Delay by value in CV
Set Timer by CV	Set Timer with value in CV
Set Timer by Val	Set Timer 0.1 to 17.0 minutes
Set Vol CV	Set value in CV for Volume
Set other CV	Set value in other CV
Test CV (*)	Test value in CV
Set Memory	Set User Memory slot 0-5
Add to Memory	Add/Sub from Memory
Test Memory (*)	Test for value in User Memory
Branch to Tag	Branch to a defined Tag
Set Tag	Set a Tag for Branch
Blend	Blend sound based on Speed
Random Do Next	Randomly do next activity
Set Distance Gauge	Turn on/set/off Distance Gauge
Enable Motor	Ensure motor is engaged

(*) CV or Memory values are tested against a provided value. If equal, the next activity is skipped. This is typically used with Branches and Tags to provide alternate activities in an event depending upon tested values. The use of tests and branches is shown in some of the included Sample Projects.

Tips on Developing Event Scripts

As indicated earlier, the Digitrax SFX decoder programming is made up of triggering Events and a series of one or more Activities to be performed when such an Event is triggered.

When building the Event Scripts, SPJHELPER© provides an easy selection of Triggers and the criteria defining the trigger action. For example one simple Triggered event might be something like "*When F1 is Turned ON*." The associated script activities might be something like "*Set Volume using CV_140*" followed by an activity like "*Play Sound Clip 1*." A variety of Triggered Events and Script activities are provided along with the parameters to particularize them for the desired performance.

It is generally suggested that sound events associated with moving, e.g., diesel sounds, steam engine chuffs, etc. be developed in Voice 1. This gives them the highest priority. On the other hand, sound events that are triggered by a timer, e.g., Air Pumps and Compressors, should be placed in Voice 3 or 4.

Typically Voice 2 is used for Function key triggered activities but care must be taken in which sounds might be playing at the same time since only ONE sound can be playing from each voice concurrently.

Some of the more complex scripts might require setting "logical switches" and then testing those "switches" in subsequent events. SPJHELPER© provides for script activities to set Memory Registers ("switches") and then test and branch to other locations in a script to tailor the processes. Testing CV values or Memory values causes the next activity to be skipped if the test was equal.

The sample sound projects provided with SPJHELPER© demonstrate many of the script capabilities.

MORE COMPLEX TOOLS FOR THE PROGRAMMER

Running SPJHELPER© in the more complex (Nitty Gritty) mode, the programmer-oriented-individual is provided easy access to:

- An Editor for developing or modifying the Assembly Language program (ASM),
- An Editor for building or modifying the Project Description File (TXT),
- The Micro Assembler to compile the program code into a HEX file
- A tool for automatically creating the Sound Handle file (MAP) from the .ASM file
- An Editor for developing or modifying the Sound Handle file (MAP)
- A facility to locate and import ASM files from elsewhere on the PC
- A Sound File Editor program (e.g., WavePad) to edit Sound Wavefiles
- A tool to “extract” all components from an existing SPJ file including Sound Files, Map Files, Text Descriptive File and the SDF code file
- The Extract SPJ tool can also convert the SDF code file into text ASM file, suitable for modification and subsequent assembly with the macro assembler
- A tool to “analyze” all components of an SPJ file in the current project folder. Component sizes and characteristics are displayed (or printed) including sizes for the full project
- Direct access to the SoundLoaderV2™

Status messages will be provided as the programmer works through the compilation.

Unlike the case in the *Simple mode* of SPJHELPER© operation, in this *Nitty Grittie programmer mode* the various files created must be individually loaded into the SoundLoaderV2™ program using that program’s IMPORT commands. Sound files must also be individually associated with the Handles listed from the MAP file. If an SPJ file exists in the current project folder, it will be loaded but changed components will need to be replaced in SoundLoaderV2™

APPENDIX A - Components of the Sound Project File (SPJ)

The Digitrax Sound Project files (SPJ) cannot be viewed or retrieved from a sound decoder. To see the components of a project, an original SPJ file must normally be loaded into the Digitrax SoundLoaderV2™ Software. See note below about capabilities in the Complex Mode of SPJHELPER©.

The SPJ "file" actually contains four different file types:

- The actual sound clips (in a specific Microsoft WAV file format)

- A MAP file which matches the sound clips (in the sequence shown by the SoundLoaderV2™) to the identification in the SDF code. The MAP files from a SPJ can be exported using SoundLoaderV2® and then viewed in a text editor like NOTEPAD+++.

- The SDF code, which makes it all run, is a version of the compiled HEX file from an Assembly Language (ASM) file using a MacroAssembler such as is available from MicroChip Technologies (and included in their free MPLAB IDE). Note this whole process is developing the programming code, which is loaded into the micro controller within the sound decoder.

- A descriptive Text file (TXT) which serves two purposes:

    1. Readable documentation about the SPJ, and

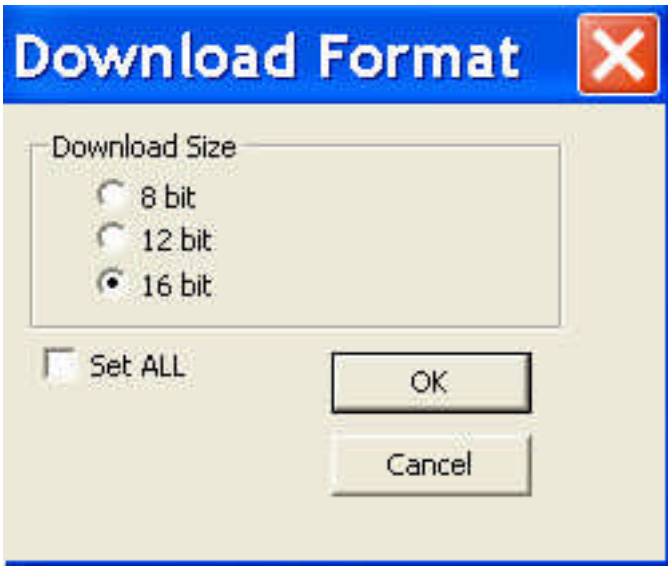
    2. Descriptive information, which pops up when using the SoundLoaderV2® software, e.g., the pop up descriptions of the F keys and the default CV values. The Descriptive Files can be viewed directly in SoundLoaderV2™ (View pull down menu) and can also be exported for modification.

Note than only the .TXT Descriptive file and the .MAP Handle file are normally accessible from an existing Sound Project using SoundLoaderV2™. However SPJHELPER© does provide a tool to extract these and the other components (WAV files, translated SDF file, etc.) from existing Sound Projects. These can be subsequently modified to create a new Sound Project.

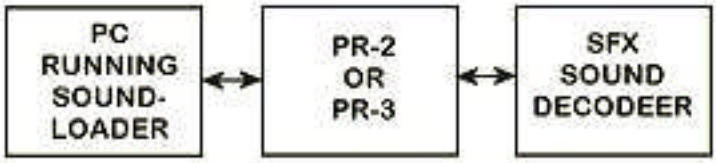
APPENDIX B - SoundLoader® Software

The Digitrax SoundLoader's main function is to download components of the Sound Project Files (.SPJ files) to the sound decoder in the locomotive on a programming track via the PR2 or PR3. Appendix A describes the contents of Sound Project Files (.SPJ file). SoundLoaderV2™ has other capabilities, too:

- SoundLoaderV2™ can be used to install/replace customized sounds in place of the existing project sounds.
- Once you've overlaid new custom sounds, you can save the sound project file with a new name, thus creating a 'custom' sound project file that you can share with others.
- SoundLoaderV2™ can "play" the sounds in a connected sound decoder using a simulated on-screen throttle, complete with F keys, direction and speed controls.
- The program is also handy for changing CVs in the sound decoder.
- SoundLoaderV2™ provides a facility (when using Series 6 Premium decoders) to change a 12 or 16 bit wavefile to a lesser bit structure. For example "downloading" a 16 bit wavefile in 12 bits or 8 bits saves flash memory space in the decoder. The loss in sound quality might be justified to gain additional space for large projects.



SoundLoaderV2™ runs on a PC and connects to your sound decoder using the Digitrax PR2 or PR3 programmer and a programming track. The PR2 programmer has 2 plugs, an RJ12 that connects the PR2 to your programming track and a 12-15V DC power supply (PS14 is available from Digitrax), and a DB25 serial plug that connects to your PC. A DB25 to DB9 connector is also included with the PR2 for PCs that only support the DB9 configuration. A USB adapter can be used with the PR2 if a serial port is not available. The newer PR3 already has the USB connectivity.



APPENDIX C. Sound File Sizes and Restrictions

The micro chip and the control software used in the Digitrax SoundFX® Decoders cannot address files greater than 1,048,575 bytes (1024K) or ~95 seconds of play at the prescribed 11 MHz wave file sampling rate. Note that the Legacy and Series 6 Standard decoders cannot address files greater than 131,071 bytes (128K) for ~12 seconds of play. If a file of greater length is loaded, the sounds for that clip AND all of the clips defined (in the sequential list) after that clip will not play. This is probably because the pointers get all out of whack (technical term). However, sound clips defined earlier in the list will play, assuming the total memory size hasn't been maxed out.

TIPS:

- Don't use a sound file of greater than 128K (131,072) bytes for a Legacy or Series 6 Standard decoder, or 1024K (1,048,575) bytes for a Series 6 Premium decoder. The file sizes are shown in the SoundLoaderV2™ sound clip list, or in various capabilities of the SPJHELPER© including the Complex Mode "Analyze SPJ" function and "Edit Sounds" function of the Simple Mode.

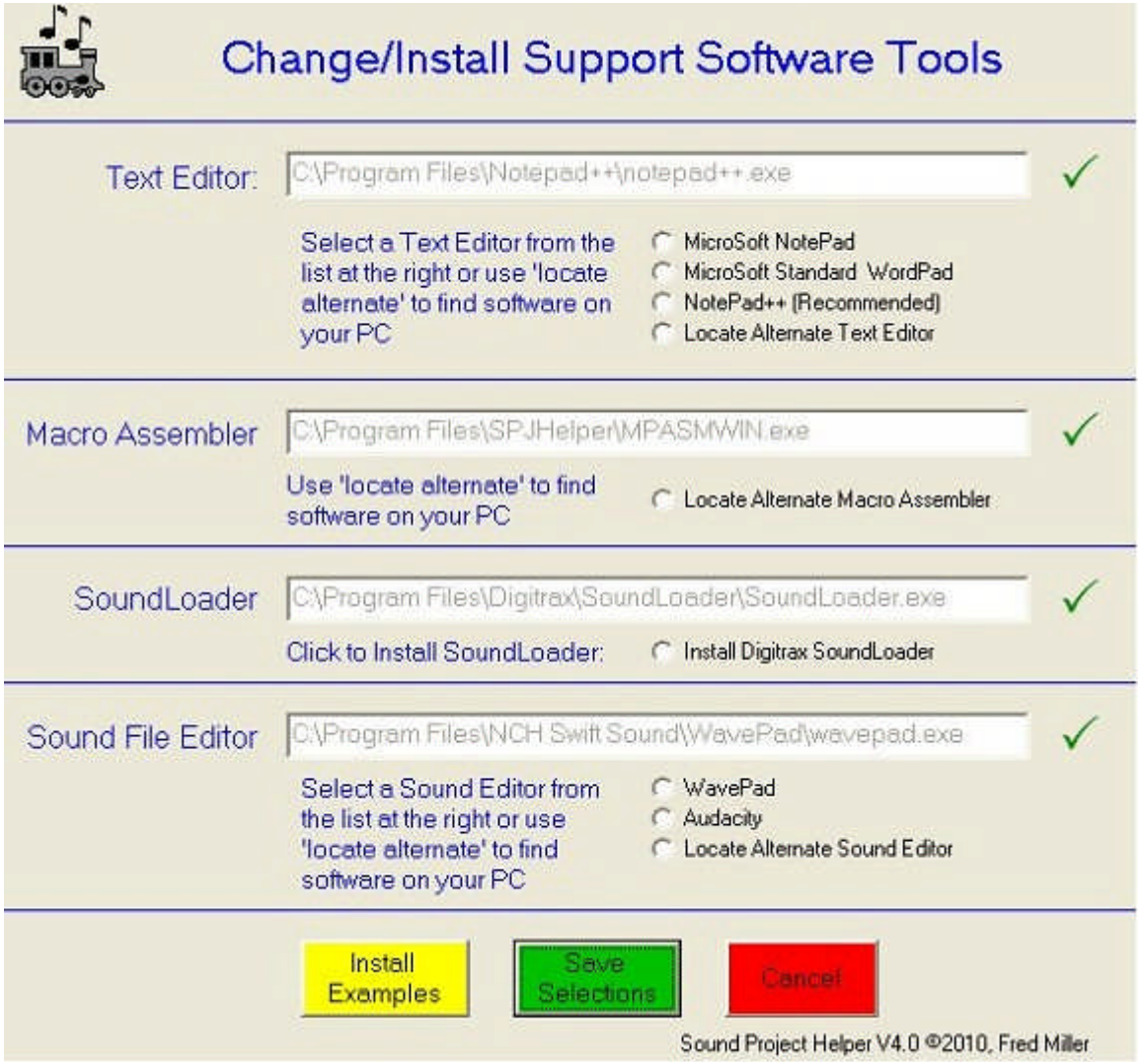
- Make sure, after loading a project into a decoder, that the amount of "Available Free Flash" is NOT a negative number. Click on the "Get more info" button in lower left to assure a good reading. If it's negative, the project is too big. The Analyze SPJ function of SPJHELPER© gives a good analysis of sound file and other component sizes.

- If you are having sound problems (and you don't have any of the above problems), try doing a "Manual Erase" of the decoder memory, cycle the power to the decoder (or PR 2/3), and then check to see it the "Available Free Flash" memory is the same as the total "Flash Size" shown above. Then do a new clean project load. Note that the accumulative size of Sound Clips and the SDF program may be shown a couple of K bytes larger since the decoder flash memory storage puts files on even 256 byte boundaries. This is demonstrated in the Analyze SPJ function of SPJHELPER©.

- The SFX064, SFX004 (SoundBug) and other Legacy decoders as well as the Series 6 Standard decoders contain 4 Megabits of memory. That's 524,288 bytes of Flash Memory. The SFX0416 and the Series 6 Premium decoders contain 16 Megabits - or 2,097,152 bytes, four times as much Flash Memory for those 'bigger projects." (Max individual sound file sizes still 128K or 1024K as noted above).

Appendix E

The Support Software included with the SPJHELPER© installation can be installed right from the SPJHELPER© Software. Alternate software can be included by changing the locations as shown below. A click on the [Change Support Software] button brings up the following screen:



Tips on Installation of Support Software:

- NotePad+++ It is suggested that the standard installation be accepted, i.e., accept the License Agreement, install to the default folder. However it is recommended that the additional offered software components be unchecked as well as the offer to “launch” NotePad+++ at the end of the installation

- MPASM Macro Assembler The standard installation provided by Microchip on their website is an extensive download and installation, putting support for all of the Microchip products. If the complete MPLAB Suite of software has not already been installed, SPJHELPER will install the one necessary program with out all of the extra software and supporting files. No options or selections will be necessary.

- WavePad or Audacity It is suggested that the standard installation be accepted, i.e., accept the License agreement and install into the default folder. However unchecking the “Related Products” will result in a smaller installation. Uncheck “launch” at the end of installation. WavePad may connect to their website after installation but just close that out.

- SoundLoaderV2™: No special options, other than accepting License Agreement and installation into standard default folder. No Desktop Icon is necessary so that option can be unchecked.

APPENDIX D - Modifying Sound Files in a Sound Project

There are two kinds of sounds in SoundFX® decoders, simple sounds, and sequenced sounds. A simple sound is a sound that always sounds the same and always has the same length. An example of a simple sound is a bell. The striker strikes the bell and it rings for a certain length of time.

To prepare a simple sound, using any sound editor software. SPJHELPER© loads and executes the WavePad program:

- Load a sound file and Isolate the portion of the sound to be used by trimming the excess time from the recording by carefully marking the beginning and the end of the sound you want to hear. Save this trimmed file as a .wav file. Make sure the saved .wave file format is "8 bit" and 11 KHz. (Series 6 Premium decoder projects can accept 12 or 16 bit formats but at the same 11KHz sampling rate.)

- Use the SoundLoaderV2™ software to load and associate the sound with the sound "handle". Locate the sound "handle" you want to change in the list and "right-click" on that entry - one of the menu choices will be "Assign Sound File" - Select this option.

- Navigate (browse) to the new wavfile you've created, select the file and click the 'Open' button. You've now successfully modified your Project file. Save your customized sound project file (.spj) with a *new* file name.

You can repeat the above steps and replace as many (or all) of the sound types as desired in the original sound project file. Once you've created this new sound project file you can download it directly to your Locomotive using SoundLoader's "Program" button

The other type of sound SoundFX® decoders support is sequenced sound. A sequenced sound is a sound that is made up of three parts: an Attack sound, a Sustain sound, and a Decay sound.

- *Attack* is the starting sound
- *Sustain* is the running part of the sound
- *Decay* is the 'end' sound

An example of a sequenced sound is a horn or whistle. Blowing the horn for 15 seconds requires an Attack sound that *begins* the sequence, a Sustain sound that *prolongs* the sound for as long as desired, and finally *ends* with a Decay sound. In actual practice, file sizes for the beginning and end of the sound, the Attack and Decay, may possibly be larger than the Sustain because the Sustain is simply a small snippet of sound repeated as long as needed. Examples of this type of sound are the Whistle, Water Pump and Horn.

Developing the individual components of this type of sound takes a bit of practice. The general idea is to take a full recording of the sound, for example whistle. Then try to find a portion of the sound that seems to repeat. Save this as the *Sustain* part making sure that when it is played in a loop there are no obvious clicks. Then try to locate portions of the sound that seem lead into that sound (Attack) and end that sound (Decay). These are saved as separate sound clips.

APPENDIX F Credits.

The SPJHELPER© Software was developed by Fred Miller. This Version is just the beginning. Additional functions may be installed in later versions.

The author wishes to thank AJ Ireland of Digitrax, Inc. for his help and encouragement in the construction of this software. Thanks also go to John McMasters for sharing his experiences with the new Series 6 Premium decoders.

The software referenced and presented for installation by SPJHELPER© is made available without charge from the websites of the following organizations:

NOTEPAD+++ Free Software Foundation, Inc. (http://notepad-plus.sourceforge.net/)

AUDACITY (http://audacity.sourceforge.net/)

WAVEPAD NCH Software (www.nch.com.au/software)

MPASM™ Microchip Technology, Inc. (http://microchip.com)

SOUNDLOADERV2™ Digitrax, Inc. (http://www.digitrax.com)