

Fsk Encoder Ihx8ReaderExtension user guide

© The Fsk Encoder project

October 2025

Contents

1 Description	2
2 GUI	2
3 Installation	3
4 Configuration	3

Introduction

The Fsk Encoder application is a usefull tool for SW development in retro computing enviornment.

It's capable of converting binary code or data files into FSK encoded sound samples which can be played on the computers sound card. Together with an appropriate interconnection cable the sound output of the host computer can be connected to the sound input of a retro computer system to upload the data.

This extension (*Ihx8ReaderExtension*) enables the FskEncoder application to read binary files.

1 Description

8-bit Intel Hex formatted files (IHX-8) contain programm code or data in record oriented structure generated eighter by a programm in the used toolchain or as a result of a memory dump i.e. from a EPROM programmer. The details of this strucure is not part of this document.

IHX-8 files contain in the data records a address field, which is read and used by the Ihx8readerExtension in two ways:

First, the address inforatoin is used to figure out the start address of a memory block and it's size and second, it is used to distinguish different memory blocks in a file. A memory block ends respectively a new one starts if the address of a record is greater than the one of the last record plus the record size.

Each distinct memoryblock results to a memory region which can be selected for an upload.

2 GUI

The Ihx8ReaderExtension's specific GUI panel is shown in the upper part of the *Target specific information area*.

Because all required information can be obtained from the source file, the only interaction is the slelction of the candidate regions.

The layout of the panel is shown in the picture beside with an exemple that consists of three memory regions.

Region	Select	Start Adr	End Adr	Size
Region 1	<input type="checkbox"/>	0x0000	0x001F	0x20
Region 2	<input type="checkbox"/>	0x1000	0x103F	0x40
Region 3	<input type="checkbox"/>	0x2000	0x207F	0x80

For each memory block of the source file, a region is created and it's metadata is displayd in the related fields '*Start Adr*' - which is taken directly from source file, '*End Adr*' and '*Size*' - which are both calculated based on the file content.

Each *region* has a 'Select' checkbox which controls whether the region should be candidate of an upload task or not. Multiple selections are treated as subsequent uploads and the upload is done automatically one by one in the order of the sections.

Some target systems cannot handle an upload of multiple regions in one single task, they must be restarted for each region. This behaviour is supported by FskEncoder by displaying a Yes/No dialogue box for each selected region. The upload starts when the [Yes] button is clicked and can be skipped by clicking the [No] button.

3 Installation

The Ihx8ReaderExtension comes as *Ihx8ReaderExtension-m.s.b.zip* file containing all the necessary folders and files, which must be unpacked in the installation folder of the FskEncoder application.

Note: The .zip file contains a version code consisting of

- m the main line of the build,
- s the stream of the build and
- b the build number.

The main line of the extension **must** match the one of the application to function properly.

The extension .zip file provides configuration snippets which must be merged into the FskEncoder configuration to make the extension accessible (see next section).

4 Configuration

After unpacking the .zip file, three new files appear in the installation directories:

1. `./bin/Ihx8ReaderExtension.bat` containing the Java CLASS_PATH extension and must be merged into the `./bin/FskEncoder.bat` file,
2. `./cfg/Ihx8ReaderExtension.properties` containing a template for the configuration of the `./cfg/Plugin.properties` file.
3. And at least the extension file *Ihx8ReaderExtension.jar* in the `./extension` directory.

Both, the .bat and .properties files contain instructions on how to merge the extensions into the FskEncoder configuration.