Building SVI FF DDs

# Background

FF DD has a .sym file (DD4 format) or a .sy5 file (DD5 format). These text files contain mapping of symbols to numeric ids that must match these definitions in the firmware.

So, if the new firmware wants to claim backward compatibility to the previous DD via COMPATIBILITY\_REV, it is necessary (but not sufficient) that old .sy\* definitions remain intact in the new .sy\* files.

# Under the hood

The .sy\* files combine symbols unique to the device (also put in SYMBOLS.TXT), and symbols from the standard DD library.

One takeaway is that if we change the version of standard library, we need confirmation from FieldComm whether it maintains compatibility.

The other takeaway is that to build a compatible DD, the old SYMBOLS.TXT content must remain intact in the new SYMBOLS.TXT

# Implementation

In SVI FF, DDs are built as part of REQIMPORT pass of the FFP build process. The makefile containing the build commands is currently forsofting.mak in FD-SW root directory. The commands are wrapped in a macro DD\_command which currently can be mapped to macros DD\_devel or DD\_only.

## DD\_devel

This option builds a fresh DD without any regard to any previous build or its DD. It is intended to start a new project or redesign the device significantly.

After the first pass, if compatibility with the previous builds is to be maintained, it is necessary to remap DD\_command.

### Instructions for changing DD\_command to DD\_only

1. Edit forsofting.mak to read DD\_command=$(DD\_only)
2. Open command line and change to directory target\appl\fbif\ddl\0008 relative to FFP root directory
3. Issue commands   
   sort xxxx.sym > xxxx.sym.ref  
   sort xxxx.sy5 > xxxx.sy5.ref  
   where xxxx stands for the actual DD basename.  
   NOTE. If .ref files existed before, check them out first.
4. Copy SYMBOLS.TXT from ..\FFTokenizer\release\004745\0008 to target\appl\fbif\ddl\0008
5. Test the build
   1. Save xxxx.sy\*, xxxx.ff\* and SYMBOLS.TXT
   2. Run  
      gnumake proj=FFP REQIMPORT
   3. Compare new and saved DD files. There must be no differences
6. Check in SYMBOLS.TXT, \*.ref files, and forsofting.mak.

## DD\_only

This option builds a DD based on SYMBOLS.TXT of the previous DD, whether taken from a previous release or the previous build. It compares existing and newly generated \*.ref files and fails on mismatch

### How to use

If you continue working from [fresh build](#_DD_devel), everything is in place.   
If you are borrowing DDs from a previous release, follow these steps:

1. Copy old SYMBOLS.TXT to target\appl\fbif\ddl\0008 relative to FFP root directory
2. Copy old yyyy.sy\* files to the same directory
3. Issue commands  
   sort yyyy.sym > xxxx.sym.ref  
   sort yyyy.sy5 > xxxx.sy5.ref  
   where xxxx stands for the actual DD basename.
4. Check in SYMBOLS.TXT and \*.ref files

If you are removing or modifying existing items, compatibility is lost. Make sure COMPATIBILITY\_REV is set to the current device revision.

You can switch to DD\_devel and generate fresh artifacts, or you may choose to replace your \*.ref files and SYMBOLS.TXT from ..\FFTokenizer\release\004745\0008 to target\appl\fbif\ddl\0008. Follow the instructions for adding items, below.

If you are adding items, compare the differences in SYMBOLS.TXT and \*.ref in the previously mentioned Tokenizer and project directories. If upon inspection you find that all changes are intended, replace SYMBOLS.TXT and \*.ref in the project directory and check in.

Hint: If you intend to add stuff, you know the build will initially fail. To force it complete, *temporarily* uncomment the line   
export igdiff=-  
in FD-SW\includes\project\_FFP\settings.mak