GottFA80

Gottlieb System80 MPU based on FPGA

Software Version 0.7

user manual

ralf@lisy.dev

v1.01 24.11.2020

Table of contents

Important remark

By using GottFA80 it is possible to damage your pinball machine. As this is a private project with NO commercial interest the author accepts no liability for any damage that may arise by using GottFA80!

1. Introduction

- GottFA80 is a 100% hobby project. This makes the solution cheap, depending on where you buy your components it is possible to create your Gottlieb replacement MPU for less than 50€.

What do you need?

- Basic soldering skills
- Possibility to read/write micro SD cards
- A PC with an USB port in order to be able to program the FPGA
- Gottlieb rom images (not included due to Copyright limitation)

2. Quickstart

1. Download latest versions of the SD card Image and the FPGA program from lisy.dev
2. Write the image to a SD card
3. Add Gottlieb roms
4. Program the FPGA
5. Configure switch ‘game select’ according to your pinball (Appendix A)
6. Replace your original System80 MPU with GottFA80
7. Switch the Game ON
8. Enjoy

3. Put the GottFA80 Image to the SD card

On my website you will find the latest version of GottFA80 an (zipped) image file. After unpacking the image can be put on a SD card. I do recommend using Win32DiskImager for doing that.

3.1.Win32DiskImager

- Insert the SD card into your SD card reader. You can use the SD card slot if you have one, or an SD adapter in a USB port. Note the drive letter assigned to the SD card. You can see the drive letter in the left hand column of Windows Explorer, for example **E:**
- Download the Win32DiskImager utility from the Sourceforge Project page as an installer file, and run it to install the software.
- Run the `Win32DiskImager` utility from your desktop or menu.
- Select the LISY image file you extracted earlier.
- In the device box, select the drive letter of the SD card. Be careful to select the correct drive: if you choose the wrong drive you could destroy the data on your computer's hard disk! If you are using an SD card slot in your computer, and can't see the drive in the Win32DiskImager window, try using an external SD adapter.
- Click 'Write' and wait for the write to complete.
- Exit the imager and eject the SD card.

4. Installation

GottFA80 boards have the same connectors and same mounting holes as the original Gottlieb System80 MPUs, so replacing of the board can be done in seconds.

5. Dip Switch Settings

5.1. S1 Dips 1…6 : game select

Here you can select what game GottFA80 should run. This depends on the roms placed on the SD card. See Appendix A for a full list and Chapter 'SD card' for an explanation of the structure of the SD card content.

5.1.1. S1-Dip7 -> Save nvram content to eeprom

GottFA does use an eeprom to save the nvram content (e.g. Highscores and extended settings). Each game has its own area on the eeprom. For bench testing dip2 can be set momentarely to ON, which results in GottFA saving the current nvram content to eeprom. During normal gameplay nvram content is saved automatically by test switch, game over relay and credit button.

5.1.2. S1-Dip8 -> init nvram

With dip8 to ‚ON' GottFa during boot will initialize the nvram ram for the selected game to zero. This is useful if you want to reset ALL ram content.

6. boot sequence

6.1. phase 1: init (not implemented yet!)

Immediately after switching on the pinball with GottFA80 inserted you will see the following output on the display of your pinball machine

Player 1: version of the FPGA program running

Player 2: value of selected game on S1

Player 3: lisy.dev unique identifier for FPGA based MPUs

Player 4: version of image on inserted SD card (not implemented yet)

6.2. phase 2: SD card read

GottFA80 tries to read the SD card content, if this fails the red LED 'SD card errror' will go ON. On success the current version of the SD card image is shown at display 4.

6.3. phase 3: program execution

The code indicated by the Dip switch 'game select' is red from the SD card and executed. If the code runs (regular interrupts are seen) the yellow 'ON' LED will go ON.

7. programming the FPGA

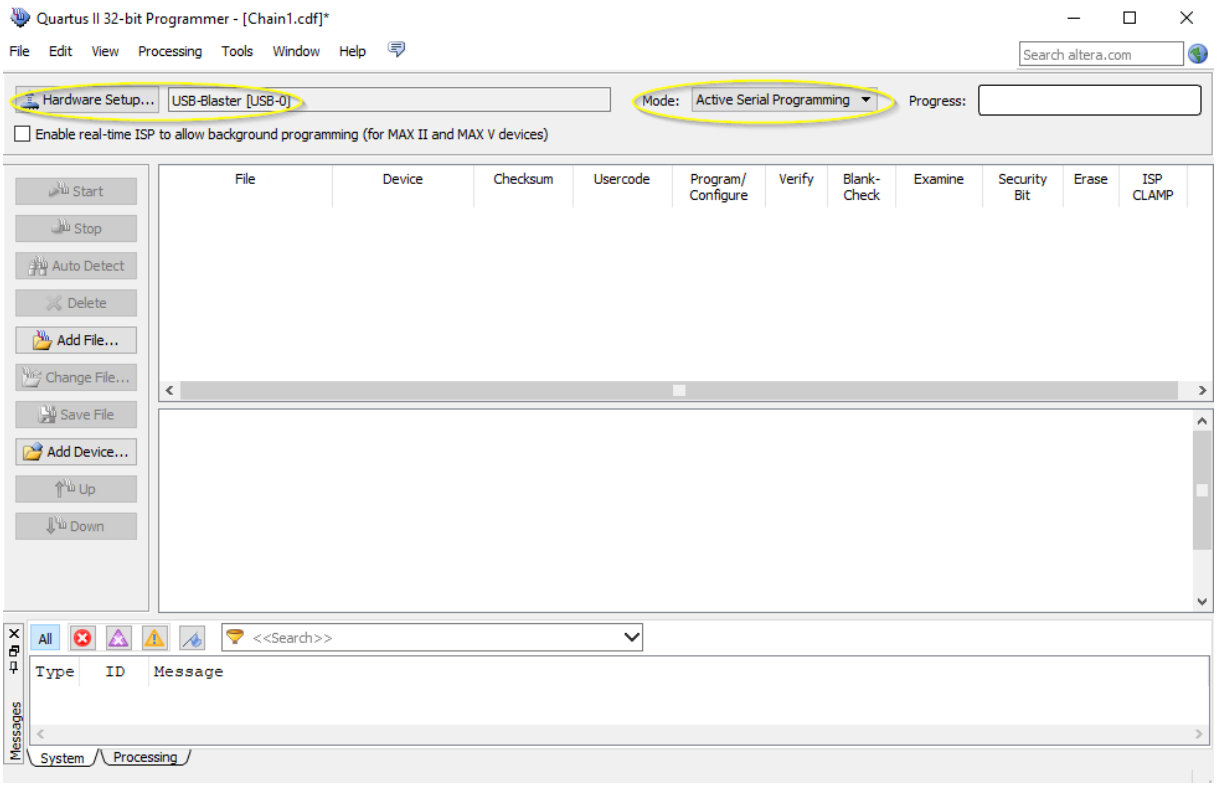
To program the FPGA you need the Quartus Programmer. It can be downloaded from the Intel Website for free. You just need to create an user account.

https://fpgasoftware.intel.com/13.0sp1/?edition=web

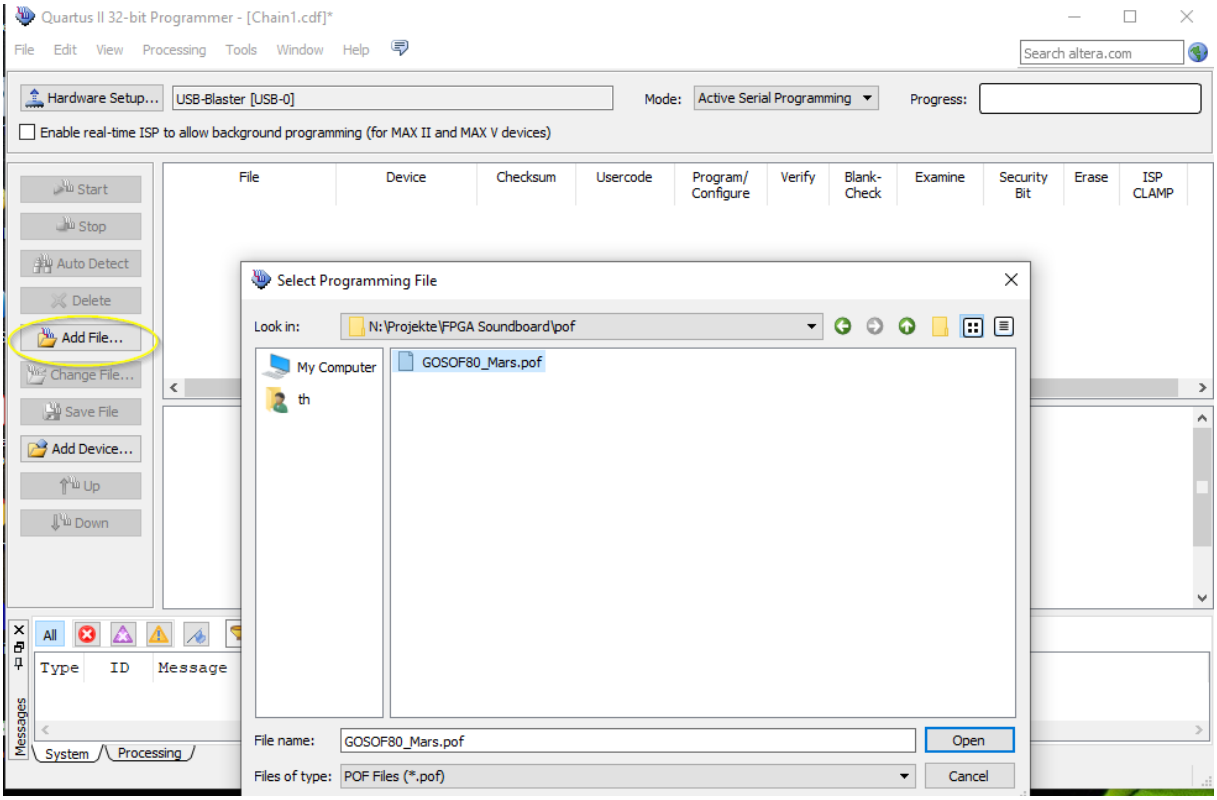
Note: Use version 13.0sp1! **Do not use newer versions of the programmer.** The used FPGA for GottFA is quite old (and therefor cheap) but is not supported by latest versions of the programmer.

1) Connect the USB Blaster to the PC

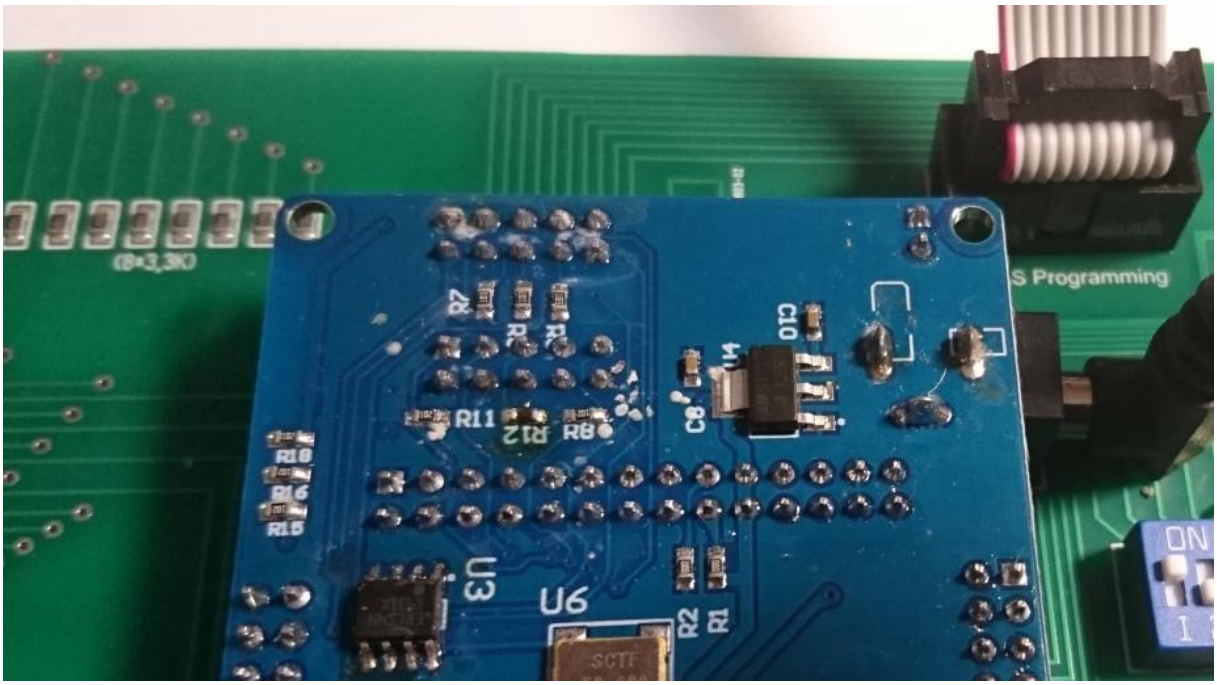
2) Start the programmer, make sure in the Hardware setup ‚USB-Blaster' is visible and set the Mode to ‚Activeserial Programming'



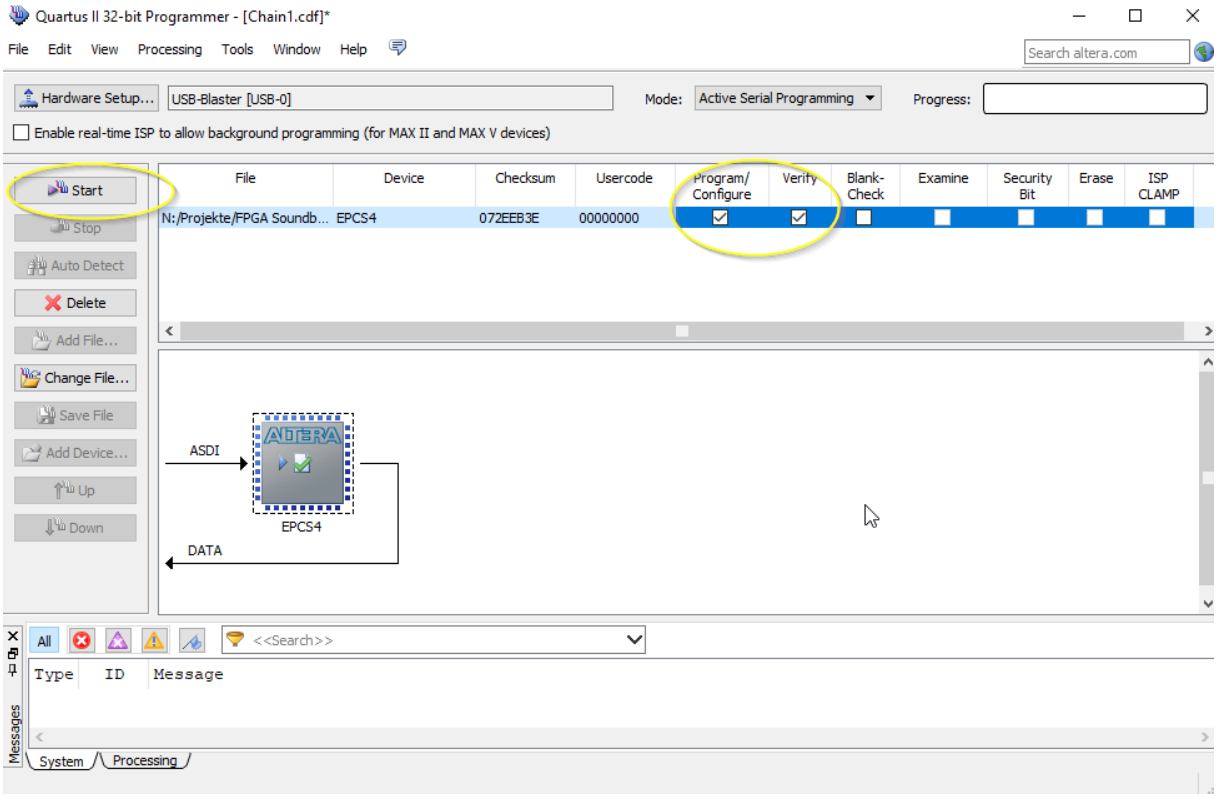
3) Select ‚Add File‘ and choose the right ‚pof‘ file for your game



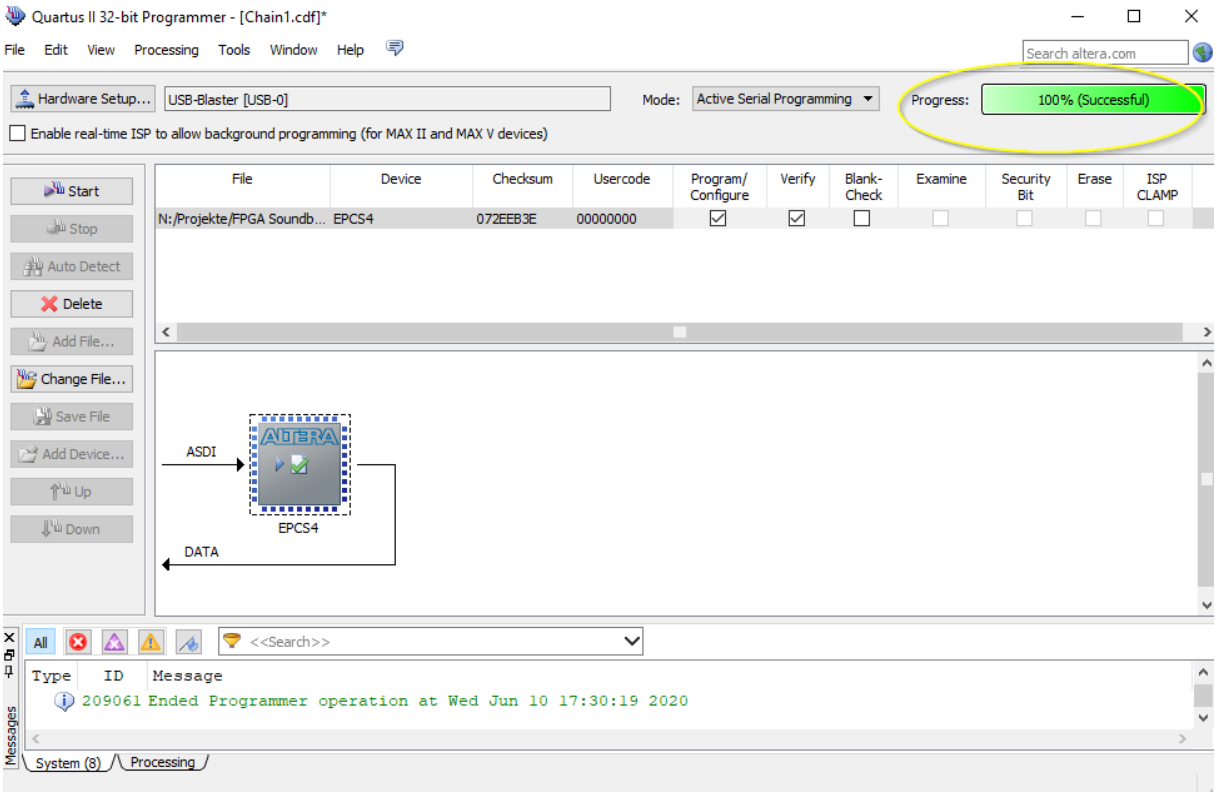
4) Power the FPGA with 5 Volt (middle pin ‚+‘, outer pin ‚-‚) and connect the USB Blaster to the ‚AS programming‘ connector on the GottFA PCB



5) Select ‚Program/Configure‘ and , Verify‘



6) Push start, wait for progress 100%



Thats it!

8. structure of SD card

Due to limitations of the SD card read routine in the FPGA (it does read fix sector numbers instead of looking for filenames) it is necessary to use my SD-card image (128 Mbyte). You can write the image to a SD-card of your choice.

8.1. rom file structure for GottFA80

Each rom file has a size of 16Kbyte and must have the 'game rom' within the first 8 Kbyte and the 'System rom' in the second 8 Kbyte. All Gottlieb Games have a 8Kbyte system rom while the size of the game roms depends on the game. Early Gottlieb games will come with 1Kbyte or 2Kbyte roms, meaning you have to fill the gaps.

Example for Black Hole:

Gottlieb Black Hole has a 2Kbyte game.rom (668-4.cpu) und two systems roms 4Kbyte each (U2_80.bin & U3_80.bin). To create a rom file for GottFa (GottFA80_BH.img) you can use the following command on a windows system:

Copy /b 668-4.cpu + 668-4.cpu + 668-4.cpu + 668-4.cpu + U2_80.bin + U3_80.bin GottFA80_BH.img

The multiple copies of 668-4.cpu are just to fill the 6Kbyte gap between the game rom and the system rom.