# An Example of PCB Reverse Engineering – Reconstruction of Digilent JTAG SMT3 Schematic

Matěj Bartík
CTU FIT
matej.bartik@fit.cvut.cz

Tomáš Beneš
CTU FIT
benesto3@fit.cvut.cz

Karel Hynek
CTU FIT
hynekkar@fit.cvut.cz

***Abstract*—This paper presents a successful reverse engineering process of Digilent JTAG-SMT3-NC module, revealing the identity of all key components. The reconstruction required a deep knowledge of PCB (Printed Circuit Board) design and manufacturing process and knowledge of (elementary) function principles and behavior of the examined device. We were able to reveal 80% of schematic via analysis of publicly available resources such as original high-resolution images and BOM (Bill of Material) fragments. The remaining 20% were obtained by non-invasive test equipment such as multi-meter and microscope. The reconstructed schematic has been verified by designing our own PCB implementing the original SMT3 function.**

***Index Terms*—Xilinx; FPGA; JTAG; Digilent; SMT3; Reverse; Engineering; PCB; FTDI;**

## I. Introduction

A JTAG (Joint Test Action Group) [1] adapter is probably the most popular way for uploading a user code into a programmed device such as processors or any other programmable logic. It is nearly 30 years, when FPGAs/CPLDs development boards were used for education of the new generation of digital logic engineers, making a solderless breadboard and 74xx logic gates outdated and obsolete. However, these development boards were supplied without on-board JTAG adapter (as is common in the present) which has to be bought separately. That was the era of Xilinx Parallel Platform Cable III (DLC5) [2], which was connected to a PC via a parallel port, and much cheaper homemade DLC5 clones.

Fig. 1. Xilinx Platform Cable USB (DLC9)

The scenario became the same for DLC5 successors such as Xilinx Platform Cable USB I [3] (see Fig. 1) and II [4] (DLC9 and DLC10 respectively). The schematic and other resources of the DLC9 leaked nearly ten years ago [5] which allows to produce significantly cheaper clones (approximately 30 USD) compared to genuine DLC10 (225 USD). The price of a DLC9 clone [5] cannot be lowered further regarding two fundamental and expensive components: CY7C68013A USB interface (∼9 USD) and Xilinx XC2C256 CPLD (∼18 USD).

The DLC9 design was the last JTAG adapter made by Xilinx, which was used on development boards (Spartan-3E Starter Kit [6] for example). The era of the third party (Digilent) JTAG adapters has begun with the introduction of Digilent Basys [7] and Nexys [8] boards which are using proprietary JTAG adapter based on an MCU (Micro Controller Unit). This concept has been abandoned in a few years after the introduction of Xilinx Vivado environment and 7-Series FPGAs resulting into the first standalone JTAG adapter (JTAG-SMT1 [9], introduced 2012) which uses cheap and widely available ICs made by FTDI such as FT2232H [10] or FT232H. The latest and the most powerful JTAG adapter (JTAG-SMT3-NC [11]) uses FT2232H as well and costs 60 USD. Therefore our motivation is:

- We don't like *"This Page Intentionally Left Blank"* text in a development board schematic [12] and the "Security through Obscurity" principle at all.
- The obscured schematic does not allow implementing any driver/tool for any platform, which is not officially supported.
- It also does not allow to repair broken boards used in education, which is also the important factor for us.
- Why nobody made an attempt to make a clone like DLC5 and DLC9 in last seven years?
- We were interested in what makes the JTAG adapter that expensive when FTDI FT2232H costs 6 USD or even less in mass quantities. Intel/Altera's USB Blaster clones [13] are available for less than 3 USD including shipping.

## II. State of the Art & Analysis

We chose the Digilent SMT3 adapter to be reverse engineered because of these facts:

- It is the latest model thus we assume the longest support.
- It seems to be used on various development boards such as Digilent ZYBO [15], Basys3 [16], Cmod S7, Cora Z7, and Arty.
- We favor UART [10] interface rather than SPI [10] interface support (SMT2 [14]) which allows saving of one USB connector and one USB interface IC.
- Availability of SMT3 images in quite good resolution [11].
- Availability of Basys 3 images in high resolution [17].

978-1-7281-5709-2/19/$31.00 ©2019 IEEE

- We assumed some of the components and principles have been used (recycled) in SMT2 [14] and SMT1 [9] adapters.

### A. *The JTAG Adapter Behavior from High-Level Perspective*

The first important step is to determine how any JTAG adapter (including Digilent and Xilinx) works and looks like. It is a black box with typically two interfaces: one for connecting a PC (USB or parallel port) and the JTAG interface.

The JTAG interface has a set of signals (see Table. I) [1] with defined electrical properties. Each of these signals has the signal direction defined (input or output). A typical development board has a standalone JTAG connector besides on-board JTAG circuitry. Thus JTAG signals are push-pull topology usually, therefore they shouldn't be connected together (to another JTAG adapter) to prevent a short circuit. Connecting such signals together requires multiplexers or buffers with tri-state outputs (controlled by an "output enable" signal). The source of the output enable signal is adapter specific and has to be determined for proper operation besides other auxiliary signals like Zynq SRSTn [18] function (PS_SRST_B) in case of the particular example of the SMT3 module.

TABLE I
JTAG INTERFACE SIGNALS FROM PERSPECTIVE OF A JTAG ADAPTER [11]

| Signal | Common Topology | Direction | Has to be tri-stated |
|---|---|---|---|
| TCK | push-pull | output | Yes |
| TDI | push-pull | output | Yes |
| TDO | push-pull | input | No |
| TMS | push-pull | output | Yes |
| TRSTn | open-drain (push-pull) | output | No (N/A) |

### B. *SMT3 Adapter First Look*

Most of the area of the SMT3 (see Fig. 2) is occupied by FTDI FT2232H [10] IC (QFN64 package) which seems to be the most important IC on-board. The PCB itself is quite small containing ICs which will be identified later (assuming level shifters). Besides these ICs, there are several resistors, capacitors, inductors, two LEDs and oscillator. Thanks to the PCB manufacturing process, there is a good contrast between traces and void areas which allows to track a lot of signals just by following them on the PCB image. The second important thing is the fact the all vias are unmasked and they are easy to observe. It seems there are no blind or burried vias. We assume the PCB has 6 layers because the bottom side is any without traces (just SMT3 signal pads summarized in Table II) and two (or even three) layers are probably dedicated to the power distribution. It seems the SMT3 is intended to be powered permanently using the *VBUS_DETECT* input to wake it up when USB cable is being attached.

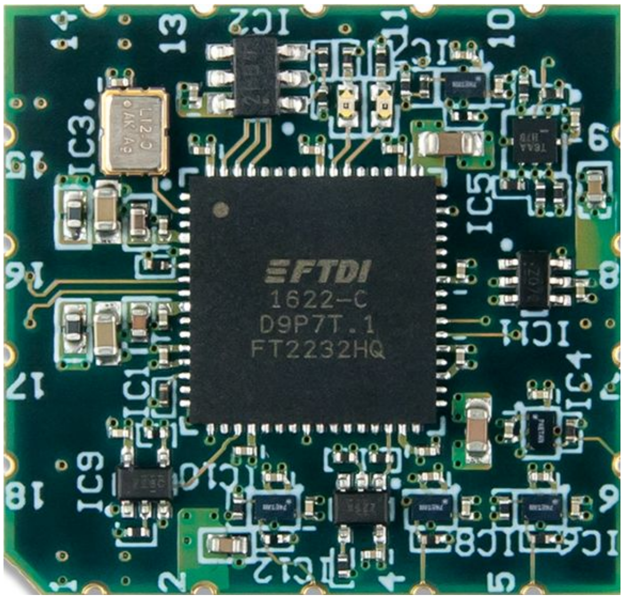


Fig. 2. Digilent JTAG-SMT3-NC top view [11]

TABLE II
SMT3 ADAPTER PINOUT [11]

| Pin Number | Pin Name | Pin Number | Pin Name |
|---|---|---|---|
| 1 | GND | 10 | TCK |
| 2 | VREF_UART | 11 | TDO |
| 3 | CTS | 12 | TDI |
| 4 | RTS | 13 | TMS |
| 5 | RXD | 14 | GND |
| 6 | TXD | 15 | VDD |
| 7 | GND | 16 | USB_DM |
| 8 | VREF_JTAG | 17 | USB_DP |
| 9 | PS_SRST_B | 18 | VBUS_DETECT |

### C. *FTDI FT2232H Analysis and Experiments*

The next step was to analyze the FTDI FT2232H behavior [10]. FT2232H is a successor of the famous FT232R and another ICs for an USB to UART conversion (or to other interfaces respectively). FT2232H has two independent MPSSEs (Multi-Protocol Synchronous Serial Engine) blocks which can be used to create an UART/FIFO/JTAG/I$^2$C/SPI interfaces. We had to analyze the FT2232H datasheet to understand how the FT2232H operates, looking for answers to these questions:

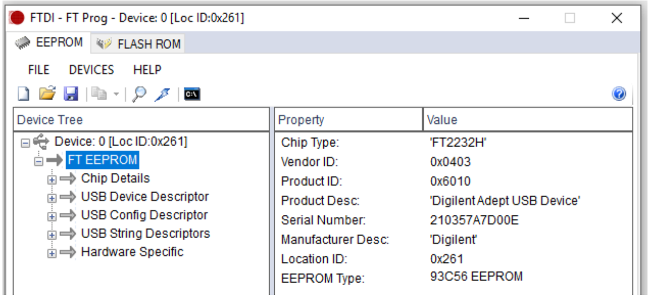
- Which MPSSE is used for the JTAG and the UART?
- What is a typical use case for supporting these interfaces?
- What are the differences between fundamental schematics and the recovered schematic of SMT3?
- Can the behavior be changed significantly by EEPROM configuration?
- What is the content of the original/SMT3 EEPROM?
- Can we convince an FT2232H evaluation board (FT2232H Mini-Module – see Fig. 3) [19] to act like SMT3 in Xilinx Vivado?

We assumed the main differences between the SMT3 and FT2232H Mini-Module board will be the EEPROM content and the presence of level shifters which are not an issue in case the Mini-Module board will be wired directly to a standalone JTAG port and will have the VREF voltage of a connected FPGA development board the same as VCCIO voltage used in FT2232H. We used a ZYBO board thus VCCIO and VREF are

Fig. 3. FTDI FT2232H Mini-Module [19]

the same (+3.3 V). Thus JTAG signals are wired directly to the connector and the on-board JTAG circuitry should be in high-impedance state, there is no need to worry about using "output enable" function (or other auxiliary signals respectively) on the Mini-Module.

The next step was to analyze the behavior of the ZYBO on-board JTAG adapter which seems to be functionally equivalent to the SMT3. We connected the ZYBO board to a PC via USB cable to read the information provided by the driver. We discovered the SMT3 uses the same VID (Vendor ID) and PID (Product ID) as the original FT2232H (see Fig. 4). Regarding this fact, we decided to read the EEPROM (93LC56B [20]) content via FTDI FT_PROG [21] tool. We discovered the main differences are customized "strings" in the EEPROM content. We found texts "Digilent Adept USB Device", "Digilent" and "Zybo" which are related to the tested board. We determined the JTAG function is located at MPSSE A thus MPSSE B is configured to operate as UART interface with VCP (Virtual COM Port) driver enabled.



| Property | Value |
|---|---|
| Chip Type: | 'FT2232H' |
| Vendor ID: | 0x0403 |
| Product ID: | 0x6010 |
| Product Desc: | 'Digilent Adept USB Device' |
| Serial Number: | 210357A7D00E |
| Manufacturer Desc: | 'Digilent' |
| Location ID: | 0x261 |
| EEPROM Type: | 93C56 EEPROM |

Fig. 4. Example of FT_PROG GUI with SMT3 connected

We decided to make an experiment with the **libusb** library [22] which allows to "spoof" device properties of an existing USB device to be replaced with a custom user defined properties. We forced the Mini-Module board to enumerate as ZYBO board by using information we read out of system logs and drivers (mostly the custom device strings identified earlier). Therefore the Mini-Module was recognized as a JTAG adapter by Xilinx Vivado. We connected the Mini-Module to the ZYBO board (via the dedicated JTAG 6-pin header) and we were able to establish a JTAG connection between Xilinx Vivado and the Xilinx Zynq 7010 SoC (chip used on the ZYBO board) successfully.

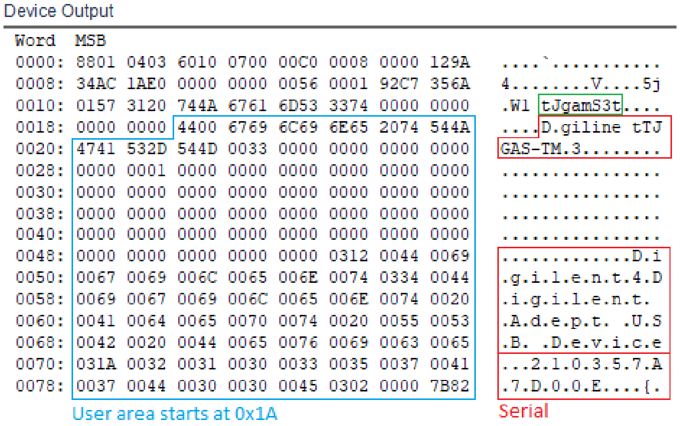

```
Device Output
 Word  MSB
 0000: 8801 0403 6010 0700 00C0 0008 0000 129A   ....`...........
 0008: 34AC 1AE0 0000 0000 0056 0001 92C7 356A   4........V....5j
 0010: 0157 3120 744A 6761 6D53 3374 0000 0000   .W1 tJgamS3t....
 0018: 0000 0000 4400 6769 6C69 6E65 2074 544A   ....D.giline tTJ
 0020: 4741 532D 544D 0033 0000 0000 0000 0000   GAS-TM.3........
 0028: 0000 0001 0000 0000 0000 0000 0000 0000   ................
 0030: 0000 0000 0000 0000 0000 0000 0000 0000   ................
 0038: 0000 0000 0000 0000 0000 0000 0000 0000   ................
 0040: 0000 0000 0000 0000 0000 0000 0000 0000   ................
 0048: 0000 0000 0000 0000 0000 0312 0044 0069   .............D.i
 0050: 0067 0069 006C 0065 006E 0074 0334 0044   .g.i.l.e.n.t.4.D
 0058: 0069 0067 0069 006C 0065 006E 0074 0020   .i.g.i.l.e.n.t.
 0060: 0041 0064 0065 0070 0074 0020 0055 0053   .A.d.e.p.t. .U.S
 0068: 0042 0020 0044 0065 0076 0069 0063 0065   .B. .D.e.v.i.c.e
 0070: 031A 0032 0031 0030 0033 0035 0037 0041   ...2.1.0.3.5.7.A
 0078: 0037 0044 0030 0030 0045 0302 0000 7B82   .7.D.0.0.E....{.
```

Fig. 5. Example of SMT3 EEPROM content

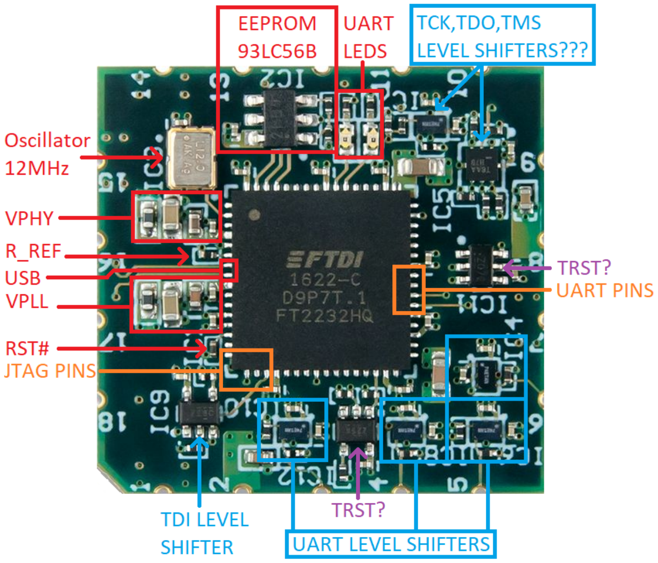


Fig. 6. Identified areas after FT2232H analysis

## *D. Gathering and Analyzing Publicly Available Information*

In this section, we would like to provide some insight into a PCB design process including some general rules, which are followed by most PCB designers. These rules are affecting components placement and the entire function of a device.

- Decoupling capacitors should be as close as possible to power supply pins. Bulk capacitors should be placed on main voltage rails as well.
- Components that are related to each other should be close to each other to minimize the traces length.
- Recycle an already proven schematic to save design time. Minimize the number of component vendors/suppliers to make assembly process simple.
- Proper documentation is a must. Otherwise it can't leak out, at least part of it.

We also utilized the fact we have ZYBO, ZedBoard and Basys3 boards available for possible detailed examination.

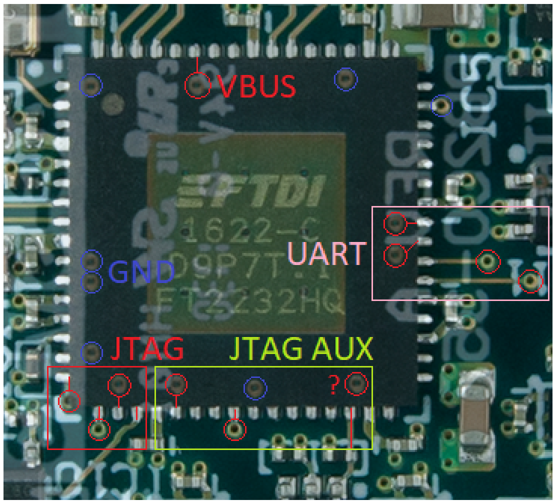


Fig. 7. PCB vias (and related traces) underneath the FT2232H chip

*1) SMT3 PCB Layout Analysis:* We had to determine the differences of a common FTDI FT2232H board layout and Digilent SMT3 to identify additional components such as level shifters (which we assumed earlier), power regulators, etc. It is essential to identify all components which are not in the FT2232H reference schematic. We followed the *"USB Self Powered Configuration"* schematic [10], which is probably the closest to the SMT3. We analyzed the SMT3 PCB from a PCB designer perspective and we estimated several areas which are added to the reference schematic (see Fig. 6). Regarding the facts the PCB vias are unmasked (thus easy to see) and we already determined the JTAG interface uses MPSSE A and UART interface uses MPSSE B, we combined SMT3 images (the top and the bottom view) into a semi-transparent image. This image reveals which FT2232H pins are bonded out therefore they should have some kind a purpose/function for proper SMT3 operation (see Fig. 7). The analysis summary:

- 9 unidentified ICs, however most of them are probably level shifters or logic gates with level shifting capability,
- It seems there are 5 same ICs in DSBGA(-6) package, which might be manufactured by the Texas Instruments. 5 is also the number of signals (TDO + UART signals) which do not need to be tri-stated,
- The IC in the bottom left corner (SC70-5 package) has a visible trace which leads directly to the TDI pin thus it should has a connection to a pin with "output enable" function besides TDI level shifting feature,
- The remaining ICs in SC70-5 and SC70-6 packages seem to be related to the PS_SRST_B function due to their proximity to the PS_SRST_B SMT3 pin (respectively to the JTAG AUX area),
- 3 unidentified bonded out FT2232H pins (plus 1 unidentified via), it is expected at least one of them is dedicated to the "output enable" function.

*2) Basys3:* The documentation for Digilent Basys3 board doesn't include a lot information, however one of the board images is found quite useful. The image [17] of both sides of Basys3 board is in the resolution higher than 4K therefore we can analyze most of the traces and components by using a computer screen only. Basys3 was designed to be simple and cheap thus only absolutely necessary components are used. Therefore the JTAG area (see Fig. 8) can be easily identified. It is obvious there is a tri-state buffer (DFN/QFN14 package) manufactured by Fairchild (part of the ON Semi at this moment). It is expected to be in quad-channel configuration because buffers are available in powers of two usually. We were not able to determine if the IC is uni- bi-directional or has single- or dual-supply. These facts seems to be unimportant, however they limited the number of ICs to be explored significantly.

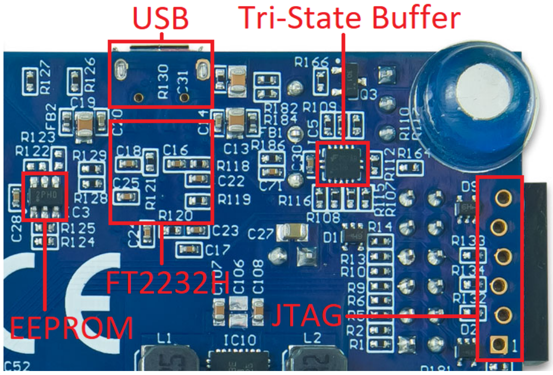


Fig. 8. Basys3 JTAG area [17]

*3) ZedBoard and SMT2:* The ZedBoard [23] was the first Zynq based board dedicated for enthusiasts (cheaper than Xilinx ZC702 and ZC706 boards) which was developed in cooperation of Avnet, Digilent and Xilinx. The Avnet was responsible for the assembly process and most of the documentation [24] is available including manufacturing data (gerbers and drill files) and complete BOM. The JTAG circuitry (SMT2 [14]) is also obscured in the schematic, however there is a notice [25] that can be considered to be a goldmine because it reveals the identity of all ICs used by the Digilent SMT2 JTAG adapter:

> *"The ZedBoard contains a proprietary USB-JTAG circuit which is obscured on this sheet. This circuit contains IC19, IC20, IC21, IC22, IC23, IC24, JP13, and J17, along with a number of resistors and capacitors. The firmware for this circuit is not publicly available. However, the firmware does come pre-programmed in the SMT2 USB-JTAG module available from Digilent and Avnet."*

TABLE III
ZEDBOARD (SMT2) BOM FRAGMENT [24]

| RefDes | Part number |
|---|---|
| IC19 | 74LCX126BQX, quad buffer, LV N-Inv, DQFN14 |
| IC20 | NC7WZ07, dual non-inverting buffer, SC-70-6 (SOT-363) |
| IC21 | NC7SZ125M5X, Tri-State Buffer UHS, SOT-23 |
| IC22 | 767-12-69, 12MHz 3.3V HCMOS Oscillator, HC5 |
| IC23 | FT232HQ, USB HS to UART/FIFO SPI/JTAG/I2C, QFN-48 |
| IC24 | 93LC56BT-I/OT, EEPROM 2Kbit 3MHz, SOT23-68 |

We assumed the mentioned components (see Tab. III) might be used for the SMT3 as well. We studied also the SMT2 datasheet [14], which includes structures used for input and output pins of SMT2 (see Fig. 9). It is a decent clue, however we believe the SMT3 uses a pull-up or pull-down resistor on every single input and output (as well control pins) of the level shifters to define default logic levels while FT2232H is being powered up. The 74LCX126 component [26] is actually the same component used in Basys3 JTAG area thus we assume the X126 IC class might be used for implementing SMT3 JTAG tri-stated output signals.

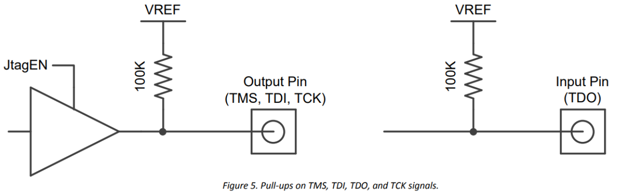


Fig. 9. SMT2 JTAG pull-ups [14]

*4) Microscope Analysis:* We finally decided to buy an SMT3 module for evaluation regarding the fact all mentioned previous steps gave us enough insight into the possible SMT3 operation. It was essential to reveal the identity of all remaining components. We used a common binocular microscope to read top marking code to successfully identify every single IC (see Tab. IV). The following step was to determine the most probable purpose of passive components such as resistors and capacitors. The datasheets of used ICs were used to estimate the purpose of these passive components (high-lighted with specific colors – see Fig. 10) depending their physical placement and particular shape of PCB traces near by:

TABLE IV
SMT3 IC Top Markings

| Package | Top Marking | Part Number | Purpose |
|---|---|---|---|
| DSBGA-6 | 834**TA2** [27] | SN74LVC1T45YZPR | UART & TDO |
| SC70-5 | **Z26**Z [28] | NC7SZ126M5X | TDI |
| UQFN-8 | **T6**AB FNB [29] | NC7WZ126L8X | TCK & TMS |
| SC70-5 | **Z25**Z [30] | NC7SZ125M5X | SRSTn? |
| SC70-6 | **Z07**9 [31] | NC7WZ07P6X | SRSTn? |

- (PINK) bulk capacitors (VCCIO, UART & JTAG VREF),
- (BLUE) decoupling capacitors for each power supply pin of an IC,
- (YELLOW) FT2232H reference schematic capacitors,
- (RED) two pull-ups are used for UART inputs (one pull-up for outputs respectively),
- (LIME) JTAG pull-ups, one resistor placed of each side of a level shifter,
- (PURPLE) "output enable" pull-down,
- (BROWN) unidentified yet, might to be related to the SRSTn function,
- All remaining components were identified in the FT2232H analysis step (see Fig. 6).

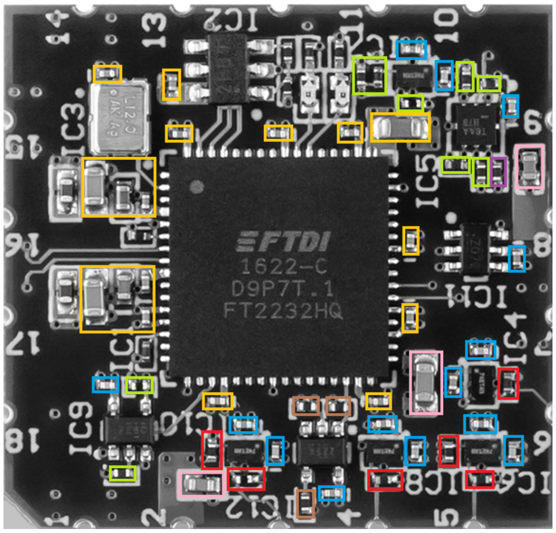


Fig. 10. SMT3 passive components

*5) Multi-Meter Probing:* The number of unidentified components, traces and vias has been significantly reduced by all previous steps. However we still had to determine the purpose of two remaining ICs (NC7SZ125 & NC7WZ07), three resistors, three bonded FT2232H pins and a single via. We decided to use a multi-meter in a *continuity mode* [32], which is intended to determine which component pads are connected (shorted) together (the multi-meter beeps when a short circuit occurs). Testing all possible combinations will be challenging and time consuming process, however we can utilize the fact the number of combinations has been limited down already and it is no longer an issue.

We decided to start with the **SRSTn** SMT3 module output pad which is connected to the output of the NC7WZ07 (VCC pin is connected to the JTAG VREF). Therefore the respective NC7WZ07 input pin seems to be connected to the output pin (Y) of the NC7SZ125 and a pull-up resistor (to the FT2232H VCCIO). There is a visible trace connecting the NC7SZ125 OE# input to the FT2232H pin ACBUS4. We found the NC7SZ125 A input is connected to the FT2232H pin ACBUS5 (probably uses the last undetermined via). Both NC7SZ125 inputs are pulled up to the VCCIO (probably acts as XOR gate during power-up phase therefore the SRSTn feature is inactive).

That means only two bonded FT2232H pins remain to be identified. It was assumed one pin behaves as "output enable" function. We knew the "output enable" signal is connected to the NC7SZ126 (pin 1) therefore it was easy to confirm the FT2232H pin ADBUS7 serves this purpose. There were no clues where the last unrecognized FT2232H pin is connected to thus we had test all possibilities. We started with testing every single pin of FT2232H to be connected to the ADBUS4 therefore we identified it is connected to the ADBUS0 (the FT2232H JTAG TCK signal).

*6) SMT3 EEPROM Content:* The last step of SMT3 analysis was to read out the EEPROM content. Thus SMT3 has no USB connector, we designed a breakout board (see Fig. 11).

Beside the SMT3, there is power regulator and plenty of connectors and test points to have access to every pin of the SMT3. The SMT3 EEPROM content (see Fig. 4 and Fig. 5) is very similar to the ZYBO content thus we discovered the SMT3 related data partions swiftly. We assume there might be some kind of protection to prevent change of a SMT3 serial number or any EEPROM data.

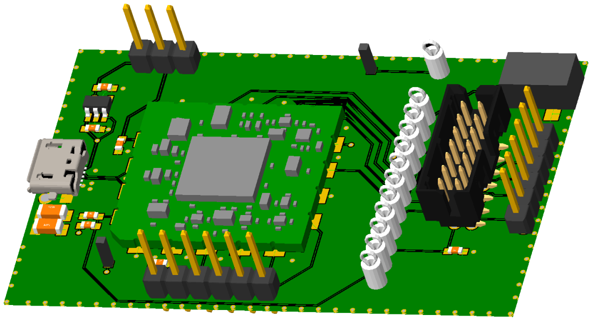


Fig. 11. The SMT3 breakout board – 3D render

## III. The SMT3 Clone Implementation and Testing

We decided to design own our PCB (Fig. 12) based on the gathered information to prove our assumptions. The very first step was to load the SMT3's EEPROM content into the clone's EEPROM. The clone was successfully recognized by Xilinx Vivado and operates in the same way as the genuine SMT3 adapter. Therefore the recovered schematic works (see Fig. 13). The second step was to test the serial number can be changed (clones using the set of compromised serial numbers can be blocked easily by drivers or by Vivado) so we changed the last character of the serial number. The clone was recognized by FT_PROG and operating system, however it wasn't detected by Vivado. We assumed the EEPROM content by some kind of checksum like CRC-16 (Cyclic Redundancy Check) [33]. The FT2232H datasheet doesn't mention possible location and used function of such checksum. However an open source *libFTDI* library [34] describes the algorithm to calculate the checksum (and the checksum address in EEPROM). Therefore we wrote the modified checksum and the clone was recognized by Vivado again. We made several tests using random serial numbers to find out a possible prefix part of the serial number. We estimated it is safe to change last 32 bits of the serial number without any impact to the clone's function. This allows to produce 4 billions of clones.

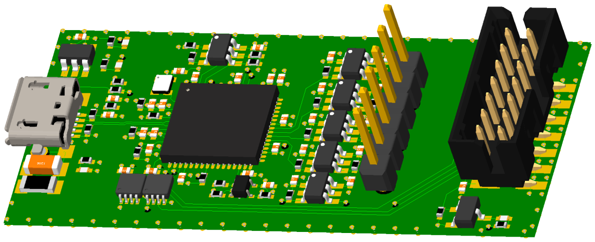


Fig. 12. The SMT3 Clone – 3D render

## IV. Improving SMT3 Resilience Against Reverse Engineering

The SMT3 primary function is implemented in the FTDI FT2232H IC and does not require any firmware decompilation to find out how it works. The (in)security of SMT3 relies on the obscurity of the electrical scheme and the EEPROM content. This practice can be considered as insufficient and does not prevent a device to be reverse engineered in a reasonable time (and effort) especially from the perspective of the SMT3 price. The trade off between device price and reverse engineering effort is the key aspect [40]. On the other hand, the FT2232H circuit does not provide any security features to protect the EEPROM against copying or modification.

We propose several ways how to increase the SMT3 immunity against reverse engineering:

- Using blind or buried vias will make a PCB analysis harder because it requires an X-ray to spot traces. Embedding components [36] into a PCB can be also used.
- It is harder to reverse engineer old MCU based JTAG adapters because of firmware presence.
- The serial numbers should not be a simple increment. Using a polynomial generated numbers will be harder to guess because the sparse distribution.
- The checksum function is publicly known and it is linear function. Using the checksum is viable for ensuring the EEPROM content is valid, however adding a secret and using a non-linear function will be much better. A larger EEPROM (93LC66) might be used to store a digital signature (DSA or ECDSA algorithm [37]).

The EEPROM related improvements can be used to confirm the EEPROM memory integrity. Devices without a valid signature can be rejected by the Digilent driver in Vivado. This solution does not prevent cloning of the EEPROM, but it might limit the mass production of the cloned devices (it will be harder to change the serial number). However any of these techniques can be overcome by modifying a DLL library (FTDI or Vivado), but these countermeasures could increase the difficulty of reverse engineering process significantly.

## V. Reverse Engineering Legality

The attitude [38] we used (observation of physical properties, deep analysis of datasheets, usage of IC chip vendor software and deep analysis of open source software) for this particular example of reverse engineering has not involved any kind of invasive software reverse engineering techniques such as disassembling or code analysis. SMT3 does not seem to be using any kind of any executable firmware due to the capacity of the used EEPROM. All observations were made by a publicly available software provided by an IC vendor or by an open-source software such as Linux kernel, libusb and libFTDI libraries provided under (L)GPL license [39]. We assume the act of PCB reverse engineering process and the schematic recovery process comply with current legislation [35], [38], [40]–[42] especially considering the fact the probable Digilent JTAG-SMT3-NC schematic does not seem to be innovative enough (recommended reference schematic of each component were used) thus the design seems to be inspired by the Xilinx DLC9 datasheet. The recovered schematic may contains errors despite the schematic has been inspired by the Digilent JTAG-SMT3-NC.

## VI. Conclusion

We successfully reverse engineered the Digilent JTAG-SMT3-NC adapter, which resisted for at least seven years. The SMT3 predecessor, the Xilinx Platform Cable USB was reverse engineered in five years despite its higher complexity. We used publicly available information only to estimate the SMT3 function, to reconstruct the schematic and to determine the purpose of data stored in the SMT3's EEPROM. The developed clone was tested in Xilinx Vivado and it's fully compatible with the SMT3. We assume the SMT3 principles are used in other Digilent JTAG adapters, therefore we are able to repair damaged boards or to program such boards from currently unsupported systems.

## Acknowledgment

This research has been partially supported by the CTU project SGS17/213/OHK3/3T/18 ”Dependable and attack-resistant architectures for programmable devices”.

## References

[1] Technical Guide to JTAG, XJTAG [Online]. Available: https://www.xjtag.com/about-jtag/jtag-a-technical-overview/

[2] Overview of Xilinx JTAG Programming Cables and Reference Schematics for Legacy Parallel Cable III (PC3) (XTP029), Xilinx [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/xtp029.pdf

[3] Platform Cable USB (DS300), Xilinx [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds300.pdf

[4] Platform Cable USB II (DS593), Xilinx [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds593.pdf

[5] Xilinx USB-JTAG-Adapter [Online]. Available: https://www.mikrocontroller.net/articles/Xilinx_USB-JTAG-Adapter

[6] Spartan-3E FPGA Starter Kit Board User Guide (UG230) [Online]. Available: https://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf

[7] Digilent Basys Board, Digilent [Online]. Available: https://reference.digilentinc.com/_media/basys:basys_ds.pdf

[8] Digilent Nexys Board Reference Manual, Digilent [Online]. Available: https://reference.digilentinc.com/_media/nexys:nexys_rm.pdf

[9] JTAG-SMT1 Programming Module for Xilinx FPGAs, Digilent [Online]. Available: https://reference.digilentinc.com/_media/jtag_smt1/jtag-smt1_rm.pdf

[10] FT2232H Dual High Speed USB to Multipurpose UART/FIFO IC, FTDI [Online]. Available: https://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT2232H.pdf

[11] JTAG-SMT3-NC Reference Manual, Digilent [Online]. Available: https://reference.digilentinc.com/reference/programmers/jtag-smt3/reference-manual

[12] ZYBO Schematic, Page 7, Digilent [Online]. Available: https://reference.digilentinc.com/_media/zybo:zybo_sch.pdf

[13] USB Blaster (ALTERA CPLD / FPGA download cable), Aliexpress [Online]. Available: https://tinyurl.com/y38zazqm

[14] JTAG-SMT2 Programming Module for Xilinx FPGAs, Digilent [Online]. Available: https://reference.digilentinc.com/_media/jtag_smt2/jtag-smt2_rm.pdf

[15] ZYBO FPGA Board Reference Manual, Digilent [Online]. Available: https://reference.digilentinc.com/_media/reference/programmable-logic/zybo/zybo_rm.pdf

[16] Basys3 FPGA Board Reference Manual, Digilent [Online]. Available: https://reference.digilentinc.com/_media/reference/programmable-logic/basys-3/basys3_rm.pdf

[17] Basys3 Hi-Res image file, Digilent [Online]. Available: https://reference.digilentinc.com/_media/basys3-frontbackviews.jpg

[18] Zynq-7000 SoC Technical Reference Manual (UG585) [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf

[19] FT2232H Mini Module USB Hi-Speed FT2232H Evaluation Module Datasheet, FTDI [Online]. Available: https://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232H.pdf

[20] 2K Microwire Compatible Serial EEPROM, Microchip [Online]. Available: http://ww1.microchip.com/downloads/en/devicedoc/21794f.pdf

[21] FT_PROG 3.3.88.402 - EEPROM Programming Utility, FTDI [Online]. Available: https://www.ftdichip.com/Support/Utilities.htm#FT_PROG

[22] A cross-platform user library to access USB devices [Online]. Available: https://libusb.info/

[23] ZedBoard, Avnet [Online]. Available: http://zedboard.org/product/zedboard

[24] ZedBoard Resources, Avnet [Online]. Available: http://zedboard.org/support/documentation/1521

[25] ZedBoard Schematics Rev. D.2, Page 12, Avnet [Online]. Available: http://zedboard.org/sites/default/files/documentations/ZedBoard_RevD.2_Schematic_130516.pdf

[26] 74LCX126: Low Voltage Quad Buffer with 5V tolerant Inputs and Outputs, ON Semi [Online]. Available: https://www.onsemi.com/pub/Collateral/74LCX126-D.pdf

[27] SN74LVC1T45 Single-Bit Dual-Supply Bus Transceiver With Configurable Voltage Translation and 3-State Outputs, TI [Online]. Available: http://www.ti.com/lit/ds/symlink/sn74lvc1t45.pdf

[28] NC7SZ126 TinyLogic UHS Buffer with Three-State Output, ON Semi [Online]. Available: https://www.onsemi.com/pub/Collateral/NC7SZ126-D.PDF

[29] NC7WZ126: TinyLogic UHS Buffer with 3-STATE Output, ON Semi [Online]. Available: https://www.onsemi.com/pub/Collateral/NC7WZ126-D.pdf

[30] NC7SZ125 TinyLogic UHS Buffer with Three-State Output, ON Semi [Online]. Available: https://www.onsemi.com/pub/Collateral/NC7SZ125-D.PDF

[31] NC7WZ07: TinyLogic UHS Dual Buffer (Open-Drain Outputs), ON Semi [Online]. Available: https://www.onsemi.com/pub/Collateral/NC7WZ07-D.pdf

[32] How to Use a Multimeter – Continuity [Online]. Available: https://learn.sparkfun.com/tutorials/how-to-use-a-multimeter/all

[33] Catalogue of parametrised CRC algorithms with 16 bits, CRC RevEng [Online] Available: http://reveng.sourceforge.net/crc-catalogue/16.htm

[34] libFTDI - FTDI USB driver with bitbang mode, Intra2nNet [Online] Available: https://www.intra2net.com/en/developer/libftdi/index.php

[35] M. Fyrbiak et al., ”Hardware reverse engineering: Overview and open challenges,” 2017 IEEE 2nd International Verification and Security Workshop (IVSW), Thessaloniki, 2017, pp. 88-94. doi: 10.1109/IVSW.2017.8031550

[36] V. Solberg, ”Embedding Passive and Active Components: PCB Design and Fabrication Process Variations”, [Online] Available: http://www.circuitinsight.com/pdf/embedding_passive_active_components_ipc.pdf

[37] FIPS 186-4, Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8900, doi: 10.6028/NIST.FIPS.186-4

[38] M. G. Rekoff, ”On reverse engineering,” in IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-15, no. 2, pp. 244-252, March-April 1985. doi: 10.1109/TSMC.1985.6313354

[39] Licenses & Standards, Open Source Initiative [Online] Available: https://opensource.org/licenses

[40] P. Samuelson, ”The Law and Economics of Reverse Engineering”, 111 Yale L.J. 1575 (2001)

[41] Samuelson, P.. A Turning Point in Copyright: Baker v. Selden and Its Legacy. UC Berkeley: Center for the Study of Law and Society Jurisprudence and Social Policy Program. 2004 [Online] Available: https://escholarship.org/uc/item/7qp3n8d1

[42] Reverse engineering and Intellectual property. [Online] Available: https://electronics.meta.stackexchange.com/questions/3366/reverse-engineering-and-intellectual-property

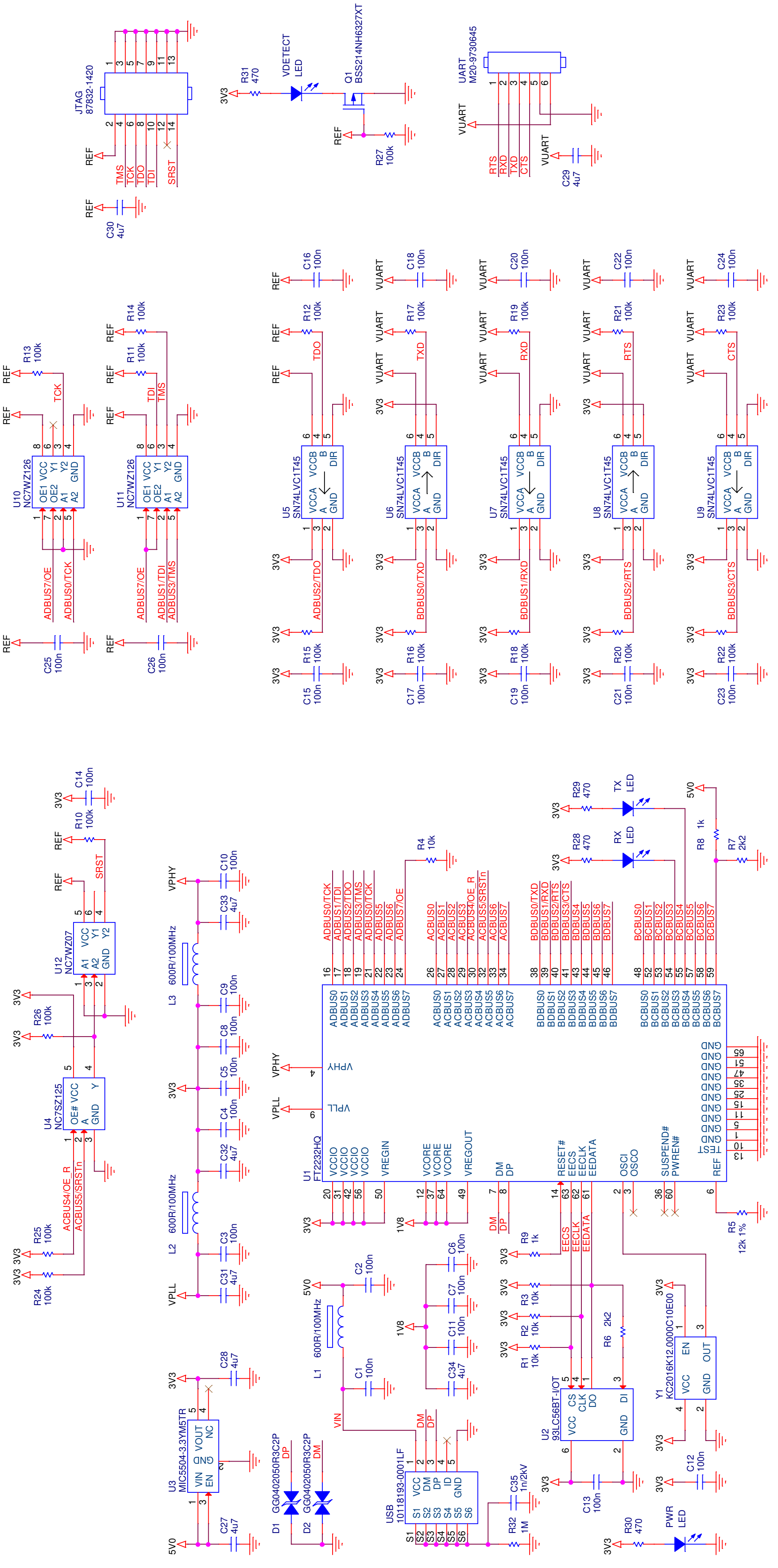


Fig. 13. Schematic of the JTAG-SMT3-NC Clone