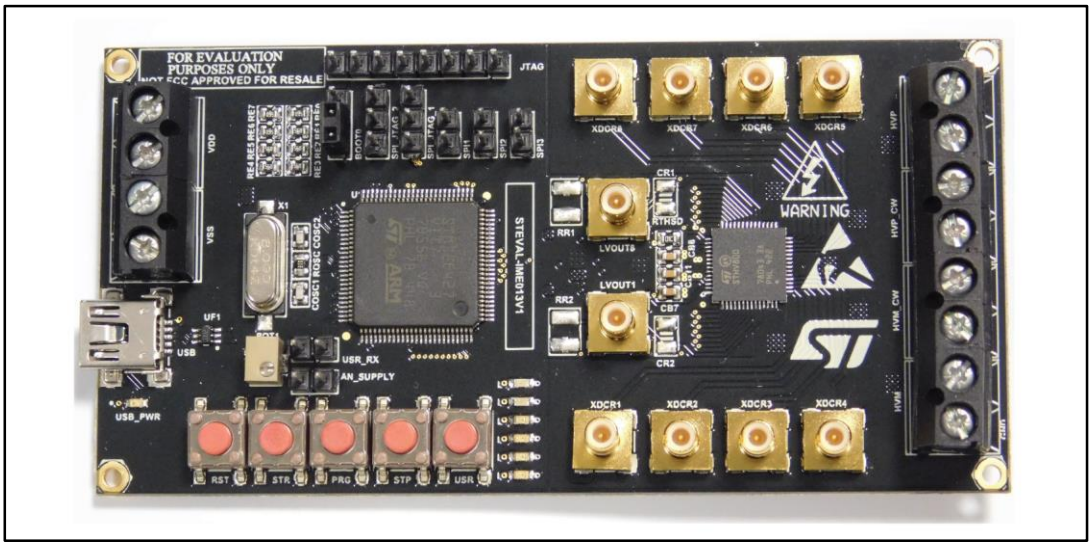
UM1977
User manual

STEVAL-IME013V1 evaluation board based on the STHV800 ultrasound pulser

Introduction

The STEVAL-IME013V1 is an evaluation board designed around the STHV800 ultrasound pulser IC, a state of the art device for ultrasound imaging applications. The system can drive eight transducers as 8-channel transmitters and display the output waveforms directly on an oscilloscope by connecting the scope probe on the relative BNCs. Four preset waveforms are available to test the HV pulser under different conditions. It is also possible to change these preset waveforms via a PC GUI (see STSW-IME011 on *www.st.com*).

Figure 1: STEVAL-IME013V1 evaluation board



Contents

1 Board features

- Suitable for ultrasound imaging applications
- 8 monolithic channels, 3 level high voltage pulser
- Integrated T/R switch
- On-board equivalent piezoelectric load implemented through an R/C equivalent network
- USB interface is available to upload customized output waveforms
- Built in microcontroller Flash memory available for storing customized waveforms
- High voltage screw connectors to power the STHV800
- Automatic lockout overvoltage protection
- 7 LEDs to check EVAL BOARD status and proper operation
- Human machine interface to select, start and stop the stored output waveforms

2 Getting starting

The STEVAL-IME013V1 is shipped by STMicroelectronics ready to use. The user only has to:

1. Plug the right power supply to the board (see *Section 2: "Board features"* for further details)
2. Connect the BNC to the oscilloscope to visualize the generated waveforms once you start the program
3. Check that the LED PROGRAM 1 (L1) turns on
4. Select the waveform with the program button. The corresponding program LED (L1-L4) turns on
5. Press the start button to run the selected program, the start LED L5 turns on. After program ends, the microcontroller returns to the idle state (LED L5 is off)
6. If a continuous wave program is selected, only the stop can halt program execution. The microcontroller returns to the idle state and the STOP LED (L5) turns off
7. To run the same program again, restart from step 5. To run another program, restart from step 4
8. Overvoltage protection will suspend pattern generation and the red LED L6 switch on if the HV supply exceeds 90 V. Pattern generation will restart after the HV supply voltage drops back into the allowed range.

3 Hardware layout and configuration

The STEVAL-IME013V1 evaluation board is designed around the STHV800. The hardware block diagram (*Figure 2*) illustrates the main connection between STHV800 and the STM32F4. *Figure 3: "STEVAL-IME013V1 board layout"* can help you to locate connectors, LEDs and features on the board.

Figure 2: Hardware block diagram

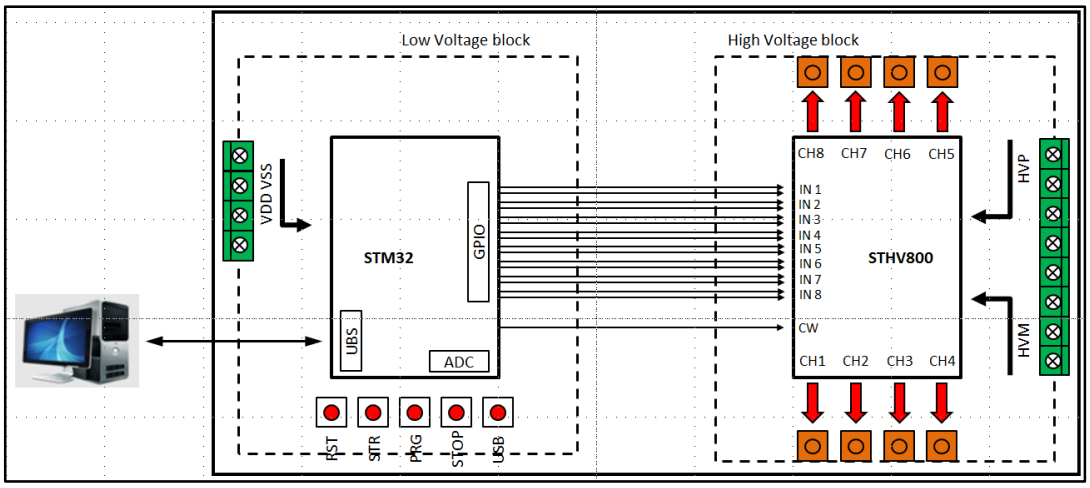
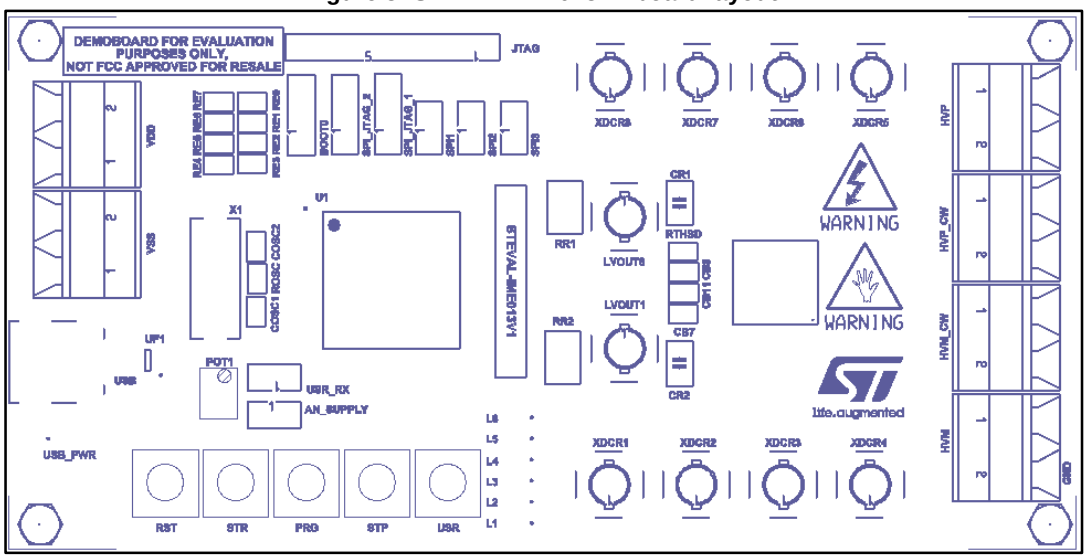


Figure 3: STEVAL-IME013V1 board layout



3.1 Power supply

The low voltage block of the STEVAL-IME013V1 board is designed to be powered:

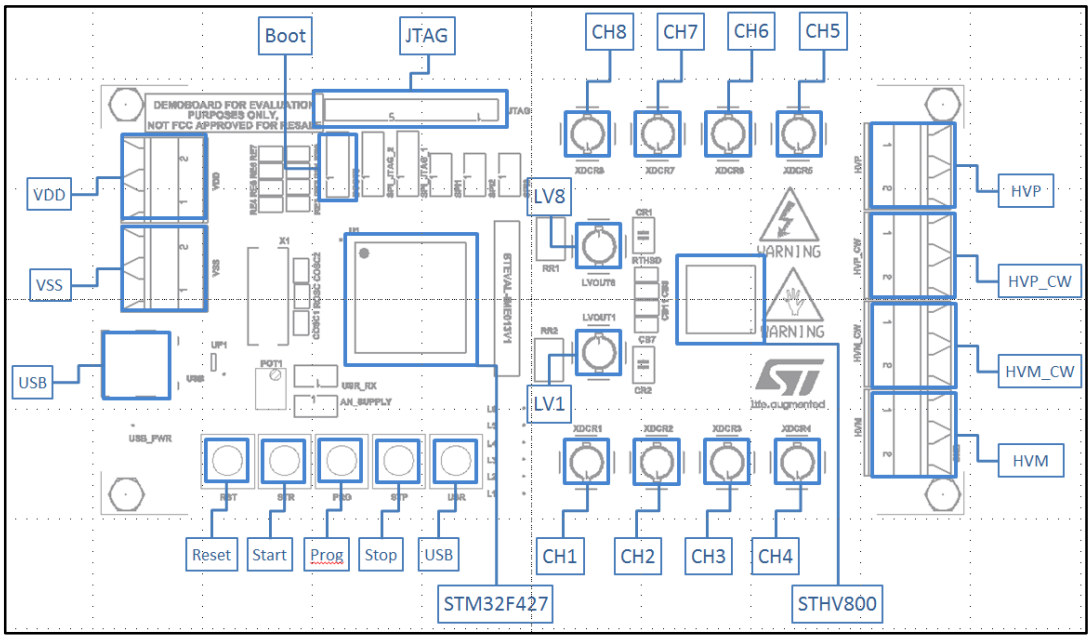
- during programming and when the board is connected to the PC:
  - 5 V DC through USB Mini B connector to supply the STM32F4
- during pattern generation and when high voltage is powered on:
  - 5 V DC connected to VDD to supply STM32F4 and STHV800 through an LDO
  - -5 V DC connected to VSS to supply STHV800 through an LDO

the USB link must be removed when high voltage is connected to the board

The high voltage block of the STEVAL-IME013V1 is designed to be powered by:

- VDD: Positive supply voltage, 5 V (2 - VDD conn.)
- GND: Ground (1 – VDD conn. And 2 – VSS conn.)
- VSS: Negative supply voltage -5 V (1 - VSS conn.)
- HVM: TX high voltage negative supply
- HVM_CW: Continuous high voltage negative supply
- GND: Ground
- HVP_CW: Continuous high voltage positive supply
- HVP: TX High voltage positive supply

Figure 4: STEVAL-IME013V1 board connections



3.2 MCU

The STM32F427 is fully dedicated to generate the bitstream on its GPIO pins to drive the pulser output channels. It is pre-programmed as a DFU (device firmware upgrade) device to upgrade internal FLASH memory. The STM32F427 manages all the DFU operations, such as the authentication of product identifier, vendor identifier and Firmware version.

The MCU drives the pulser channels through the use of different General Purpose I/O (GPIO) pins. You can simulataneously drive 1 to 16 different pins by simply writing a 16-bit word in the GPIO Output Data Register (ODR). The board can be connected to a PC through via USB and the required pattern can be transmitted as a sequence of states for each pulser channel and of durations for each state. Here, all the durations are expressed in units of MCU system clock cycles. Once the information is received, the channel states are converted into 16-bit words for the GPIO peripheral and they are stored in the embedded Flash, together with the timing information.

Once programmed, the PC connection is no longer required as the board can act as a stand-alone device. Different patterns can be stored selected for use at run-time.

The same MCU can perform two different solutions to ensure real-time execution.

The first solution involves the use of the STM32 Direct Memory Access (DMA) peripheral.

The DMA is able to transfer data from memory to any peripheral register, GPIO included, without taking into account the MCU core. To trigger the DMA transfer, a general purpose TIMER is used. The timer works at the system clock frequency, basically acting as a counter, and the reload value (the value at which the counter returns to zero) is stored in the Auto Reload Register (ARR).

The timer triggers two different DMA channels in two different moments:

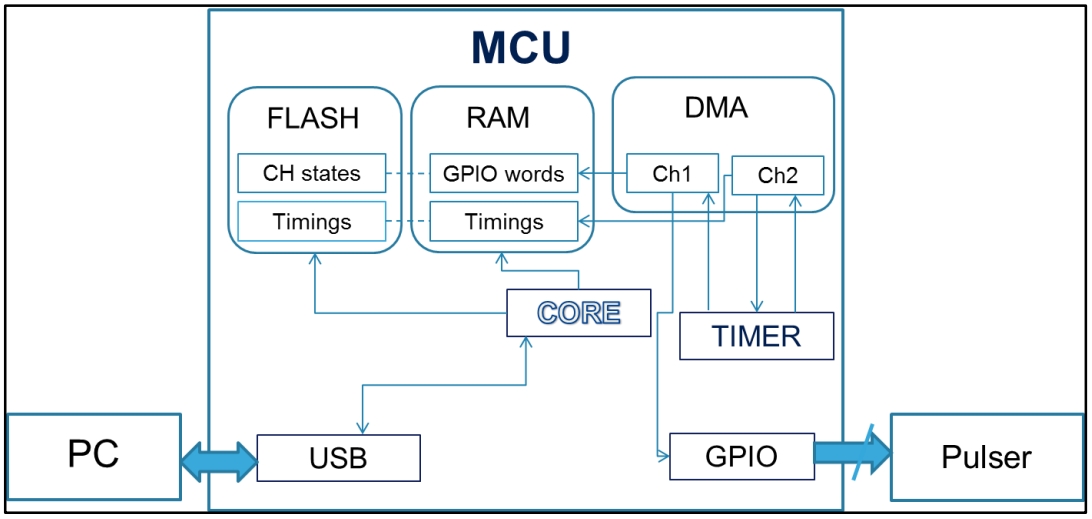
1. the first channel is triggered at each reload event
2. the second is triggered at a constant time after reload.

Once the first trigger is received, the first channel transfers the new GPIO word into the ODR. At the second trigger, the second channel transfers the new duration information into the ARR. The timer preload feature is enabled, so that the new ARR value is effective only at the next reload. Since the time needed by the first DMA channel to update the ODR is a constant, considering the reload trigger as the starting point, the time that elapses between two different GPIO updates is simply given by the ARR value.

The Circular buffer feature of the DMA can be enabled to allow the automatic regeneration of the same pattern at each end.

This solution has the advantage of being fully managed by hardware and the MCU core remains free to handle user tasks. The main drawback is that each timing value between two subsequent states cannot be lower than a minimum value in order to guarantee enough time for both the two DMA channels to perform their transfers.

Figure 5: Solution 1



The second solution is designed to overcome the minimum duration requirement of the DMA solution and involves the MCU core directly.

During run-time, the core generates the binary assembly code it needs to load and store each word in the ODR. Unnecessary instructions like control loops are avoided, so the code is simply a succession of simple load/store instructions. To adapt the timing to the pattern needs, dummy instructions are inserted in the assembly code.

To avoid wasting time to load each word from memory, the word is inserted as a literal in the assembly instruction itself. This means that a 32-bit instruction is needed instead of a 16-bit equivalent. In order to avoid any latency due to the instruction fetch from Flash, the

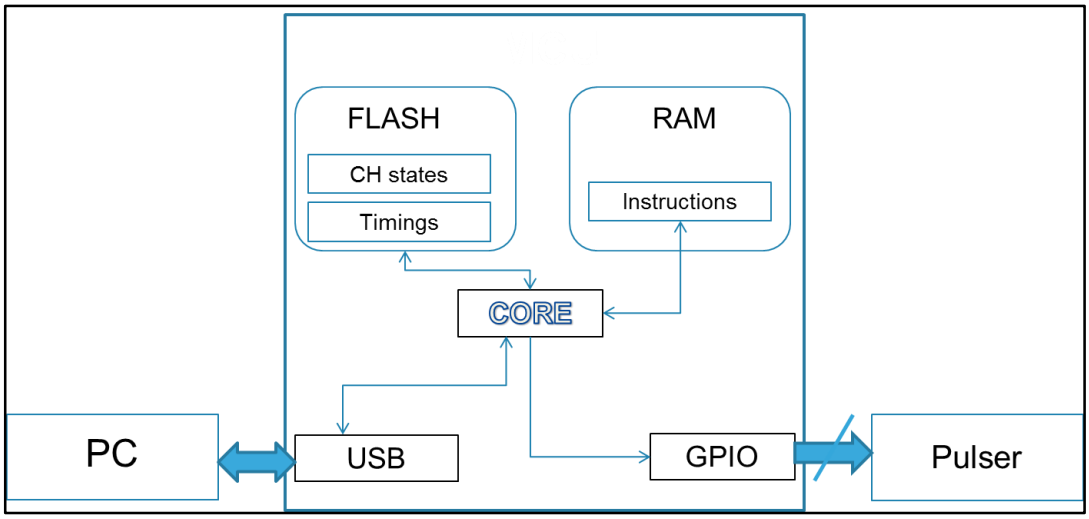
code is executed from the embedded RAM. Moreover, the RAM is configured to be accessed by the core through a different bus from the bus used to access the ODR.

These solutions render it possible to achieve a minimum two system clock cycle time before two updates and still maintain a one system clock cycle resolution. For instance, if you consider a STM32F4 clocked at 168 MHz, the minimum timing you can achieve is 12 ns and you can set the duration of each state with a resolution of 6 ns.

For a repetitive pattern, a branch instruction is added at the end of the routine to restart the pattern generation. In this case, the clock cycles needed for the branch instruction must be considered for the last state.

The main drawback of this solution is that the MCU core is 100% occupied by the pattern generation, even if it can be still called by peripheral interrupts to stop the pattern generation to perform other tasks.

Figure 6: Solution 2



3.3 Stored patterns

STEVAL-IME013V1 can store four patterns in the MCU Flash memory in order to show the achievable performance at the pulser outputs. Four selectable patterns are already stored in the STM32 Flash memory as the default, ready-to-use set. A detailed description of the programs settings is listed below.

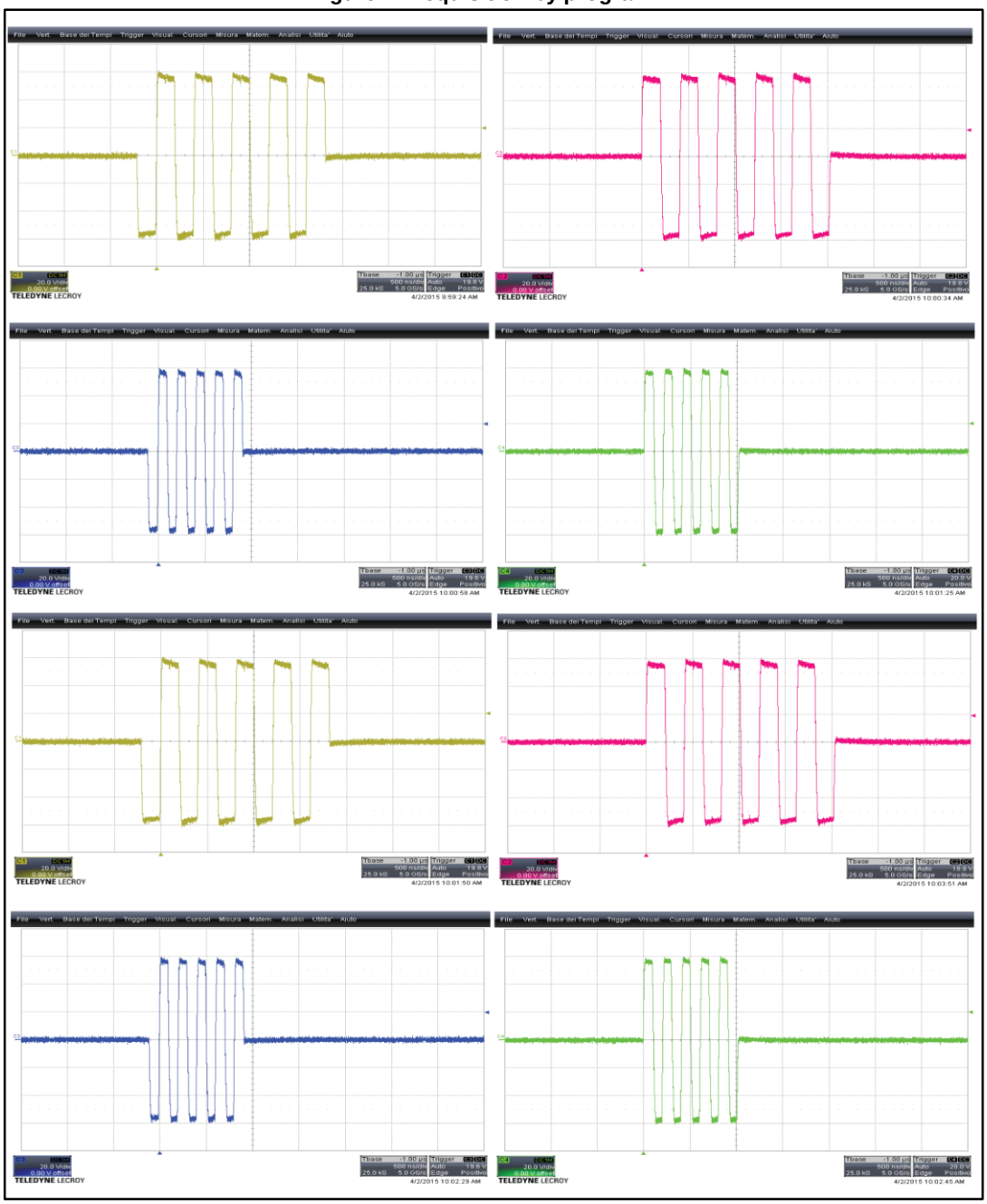
Program 1, CW=0

Table 1: Program 1

PW 5 pulses - HV=±60 V; Load: 270 pF//100 Ω						
	Mode	**Freq. (MHz)**	**No of pulses**	**Initial pulse**	**H-Bridge**	**PRF**
Ch 1	PW	2.5	5	Negative	TX	150 µs
Ch 2	PW	2.5	5	Positive	TX	150 µs
Ch 3	PW	5	5	Negative	TX	150 µs
Ch 4	PW	5	5	Positive	TX	150 µs
Ch 5	PW	2.5	5	Negative	TX	150 µs
Ch 6	PW	2.5	5	Positive	TX	150 µs
Ch 7	PW	5	5	Negative	TX	150 µs
Ch 8	PW	5	5	Positive	TX	150 µs

- XDCR_1, XDCR_2, XDCR_5, XDCR_6: Pulse Wave mode, TX switching, 5 pulses, time-period TP=400 ns and PRF=150 µs
- XDCR_3, XDCR_4, XDCR_7, XDCR_8: Pulse Wave mode, TX switching, 5 pulses, time-period TP=200 ns and PRF=150 µs

Figure 7: Acquisition by program 1



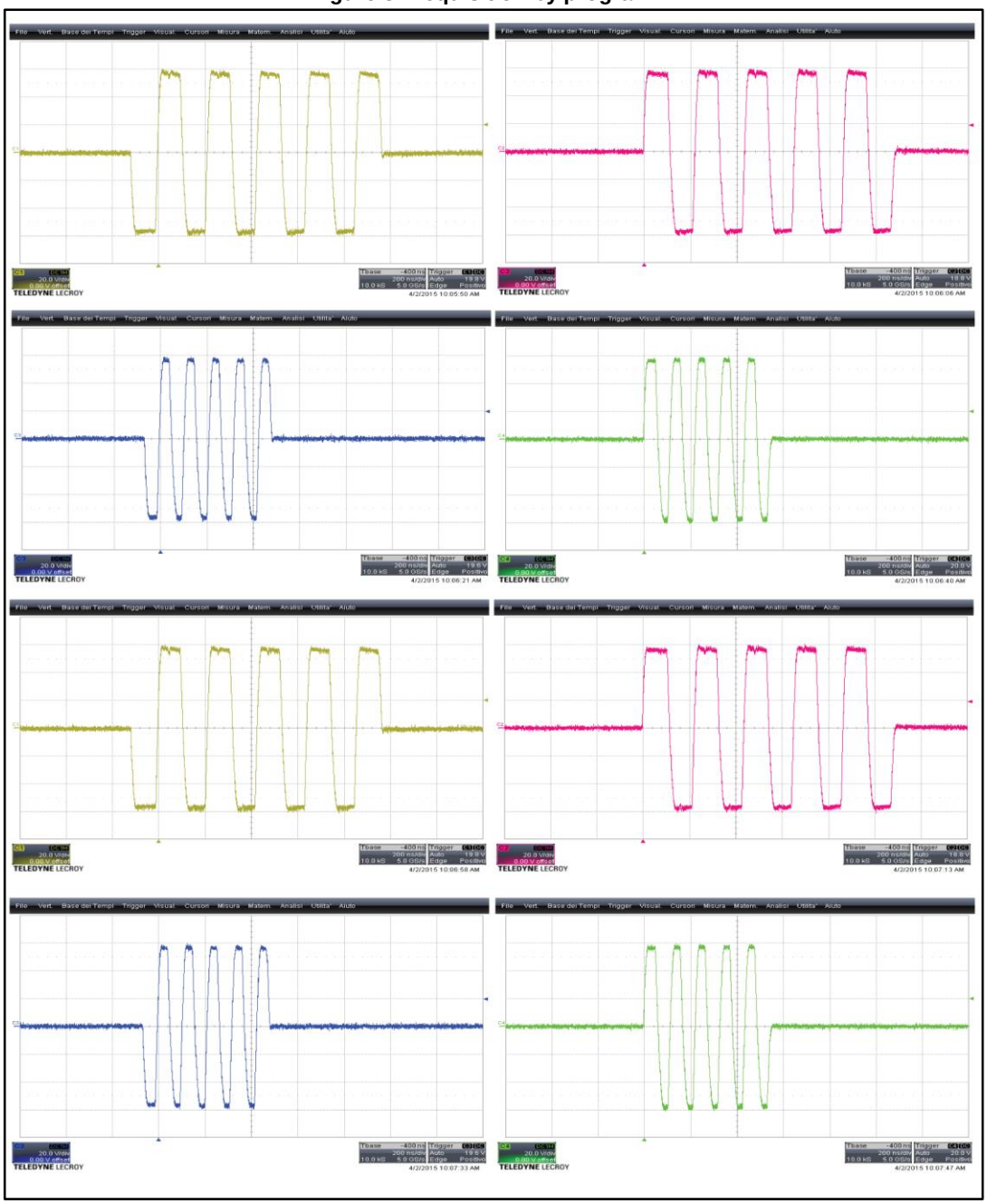
Program 2 CW=0

- XDCR_1, XDCR_2, XDCR_5, XDCR_6: Pulse Wave mode, TX switching, 5 pulses, time-period TP=200 ns and PRF=150 µs
- XDCR_3, XDCR_4, XDCR_7, XDCR_8: Pulse Wave mode, TX switching, 5 pulses, time-period TP=100 ns and PRF=150 µs

Table 2: Program 2

PW 5 pulses - HV=±60 V; Load: 270 pF//100 Ω						
	Mode	**Freq. (MHz)**	**No. of pulses**	**Initial pulse**	**H-Bridge**	**PRF**
Ch 1	PW	5	5	Negative	TX	150 µs
Ch 2	PW	5	5	Positive	TX	150 µs
Ch 3	PW	10	5	Negative	TX	150 µs
Ch 4	PW	10	5	Positive	TX	150 µs
Ch 5	PW	5	5	Negative	TX	150 µs
Ch 6	PW	5	5	Positive	TX	150 µs
Ch 7	PW	10	5	Negative	TX	150 µs
Ch 8	PW	10	5	Positive	TX	150 µs

Figure 8: Acquisition by program 2



Program 3

- XDCR_1, XDCR_2, XDCR_5, XDCR_6: Continuous Wave mode, TX-CW switching, time-period TP=400 ns
- XDCR_3, XDCR_4, XDCR_7, XDCR_8: Continuous Wave mode, TX-CW switching, time-period TP=200 ns

Table 3: Program 3

Continuous Wave – HV_CW=±10V; Load: 270 pF//100 Ω					
	Mode	**Freq. (MHz)**	**No. of pulses**	**Initial pulse**	**H-Bridge**
Ch 1	CW	2.5	Continuous wave	Positive	TX-CW
Ch 2	CW	2.5	Continuous wave	Negative	TX-CW
Ch 3	CW	5	Continuous wave	Positive	TX-CW
Ch 4	CW	5	Continuous wave	Negative	TX-CW
Ch 5	CW	2.5	Continuous wave	Positive	TX-CW
Ch 6	CW	2.5	Continuous wave	Negative	TX-CW
Ch 7	CW	5	Continuous wave	Positive	TX-CW
Ch 8	CW	5	Continuous wave	Negative	TX-CW

Figure 9: Acquisition by program 3

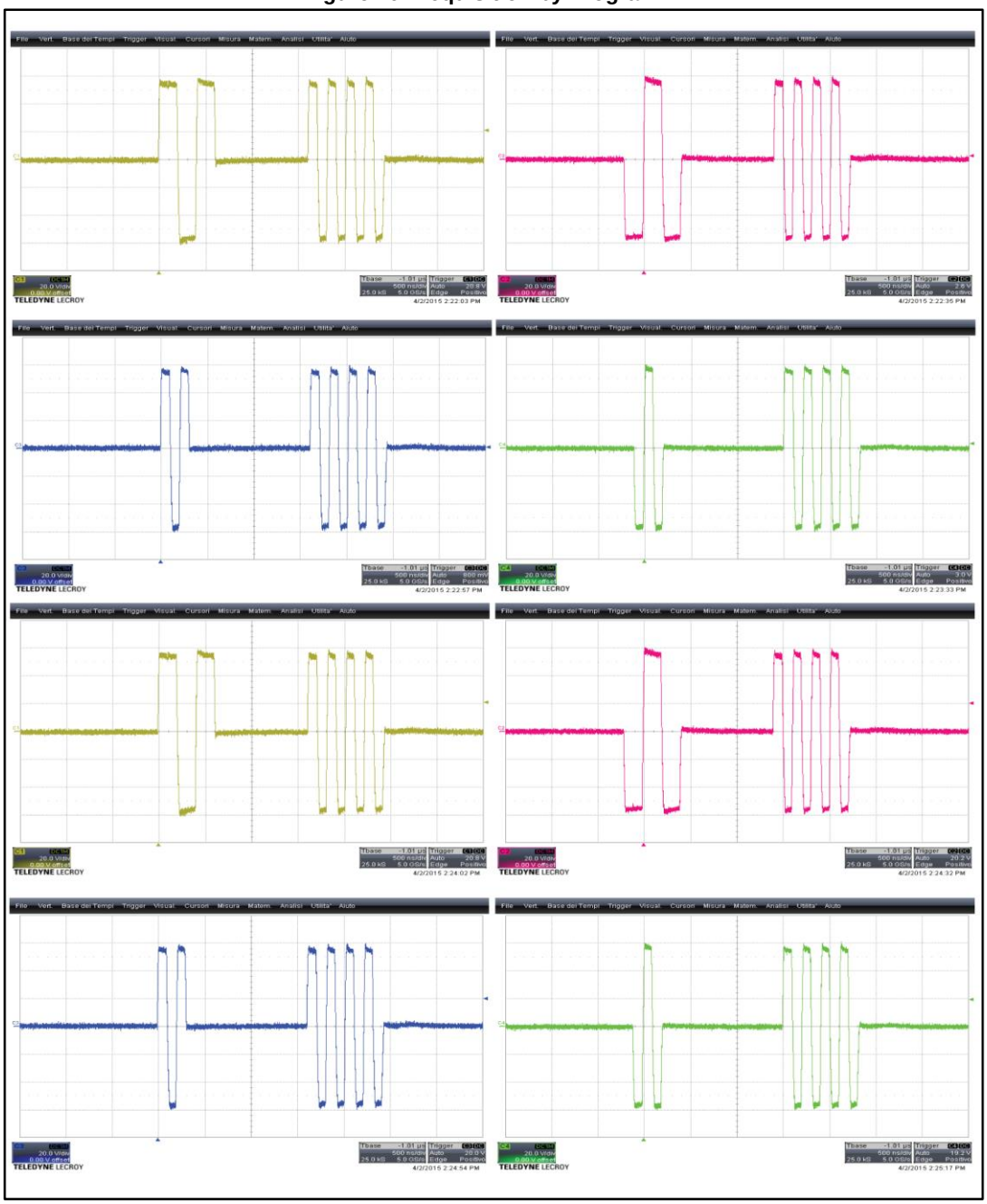
Program 4 CW=0

- XDCR_1, XDCR_2, XDCR_5, XDCR_6: Pulse Wave mode, TX0 switching, 1.5 pulses, time-period TP=400ns and consequently TX1 switching, 5 pulses, time-period=200 ns and PRF=150 µs
- XDCR_3, XDCR_4, XDCR_7, XDCR_8: Pulse Wave mode, TX0 switching, 1.5 pulses, time-period TP=200ns and consequently TX1 switching, 5 pulses, time-period=200 ns and PRF=150 µs.

Table 4: Program 4

PW - HV=±60 V; Load: 270 pF//100 Ω						
	Mode	**Freq. (MHz)**	**No. of pulses**	**Initial pulse**	**H-Bridge**	**PRF**
Ch 1	PW	2.5 - 5	3 half pulse then 4 pulse	Positive	TX	150 µs
Ch 2	PW	2.5 - 5	3 half pulse then 4 pulse	Negative	TX	150 µs
Ch 3	PW	5	3 half pulse then 4 pulse	Positive	TX	150 µs
Ch 4	PW	5	3 half pulse then 4 pulse	Negative	TX	150 µs
Ch 5	PW	2.5 - 5	3 half pulse then 4 pulse	Positive	TX	150 µs
Ch 6	PW	2.5 - 5	3 half pulse then 4 pulse	Negative	TX	150 µs
Ch 7	PW	5	3 half pulse then 4 pulse	Positive	TX	150 µs
Ch 8	PW	5	3 half pulse then 4 pulse	Negative	TX	150 µs

Figure 10: Acquisition by Program 4



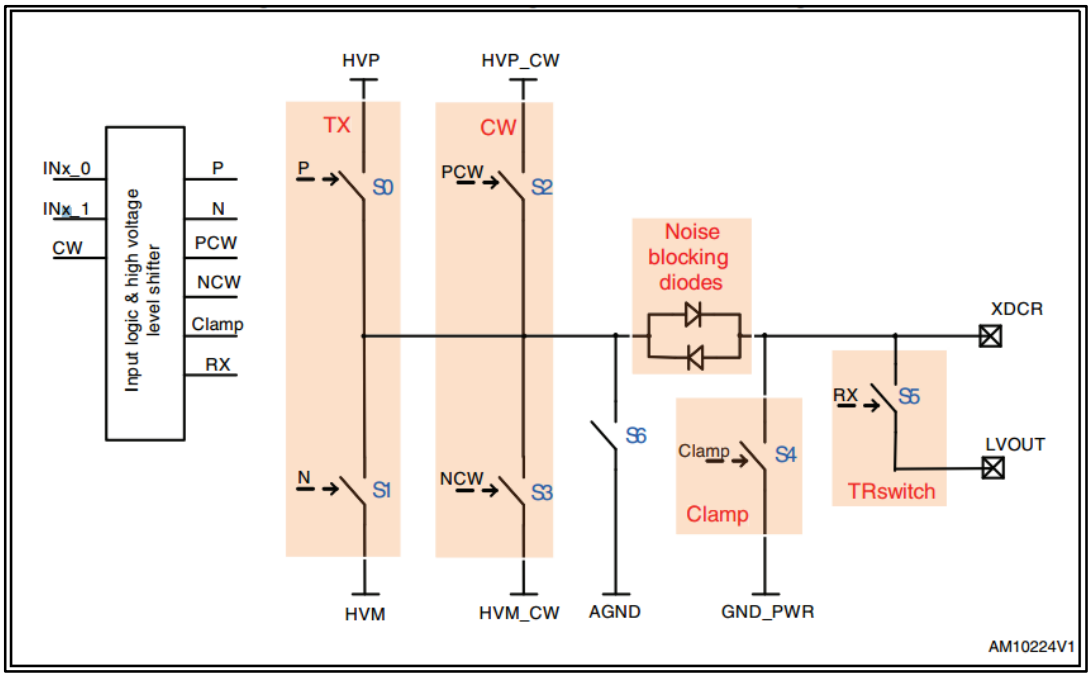
The board can be connected to a PC via USB cable and patterns can be edited through a user inetrface.

Remove the USB link when the high voltage is connected to the board.

3.4 STHV800 stage

The STHV800 high-voltage, high-speed pulser generator features eight independent channels. It is designed for medical ultrasound applications, but can also be used for other piezoelectric, capacitive or MEMS transducers. The device contains a controller logic interface circuit, level translators, MOSFET gate drivers, noise blocking diodes and high-power P-channel and N-channel MOSFETs as output stages for each channel. It also includes clamping-to-ground circuitry, anti-leakage, an anti-memory effect block, a thermal sensor and a HV receiver switch (HVR_SW) that ensures strong decoupling during the transmission phase. The STHV800 also features self-biasing and thermal shutdown blocks.

Figure 11: STHV800 single channel block diagram



Each channel can support up to three active output levels with one half-bridge. Each channel consists of two supplied output stages for pulsed wave (PW) and continuous wave (CW) operations. The PW output stage is able to provide up to ±2 A peak output current while, to reduce power dissipation and jitter during continuous wave mode, the fully optimized CW output stage delivers up to ±0.3 A.

For further information, please refer to the STHV800 datasheet.

3.5 Operating supply conditions

Table 5: DC working supply conditions

Operating supply voltage					
Symbol	**Parameter**	**Min.**	**Typ.**	**Max.**	**Value**
VDD	Positive supply voltage	5	6	10	V
VSS	Negative supply voltage	-5	6	-10	V
HVP	TX high voltage positive supply			95	V
HVM	TX high voltage negative supply	-95			V
HVP_CW	CW high voltage positive supply			95	V
HVM_CW	CW high voltage negative supply	-95			V

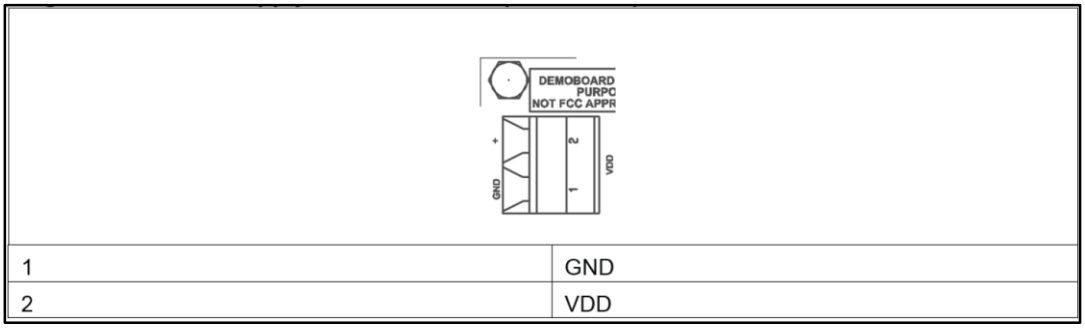
While HVM / HVM_CW and HVP / HVP_CW are fully independent on the board, the relationships HVM_CW ≥ HVM and HVP_CW ≤ HVP must be respected during operation.

4 Connectors

4.1 Power supply

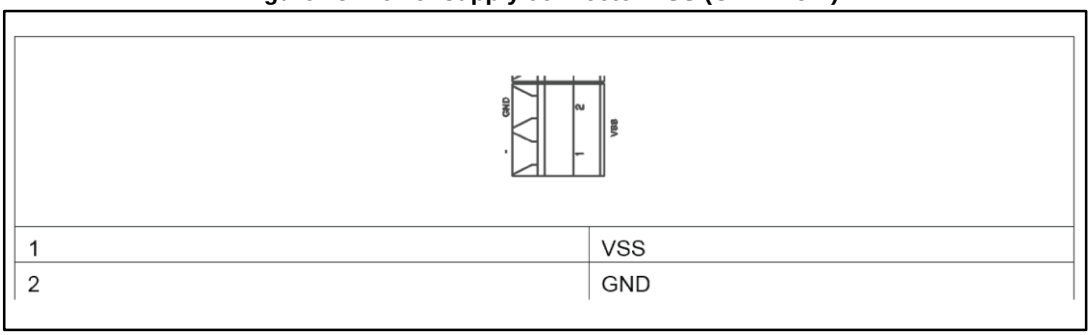
The STEVAL-IME013V1 board must be powered through the screw connectors shown in following figures.

Figure 12: Power supply connector VDD (+5 V - GND)



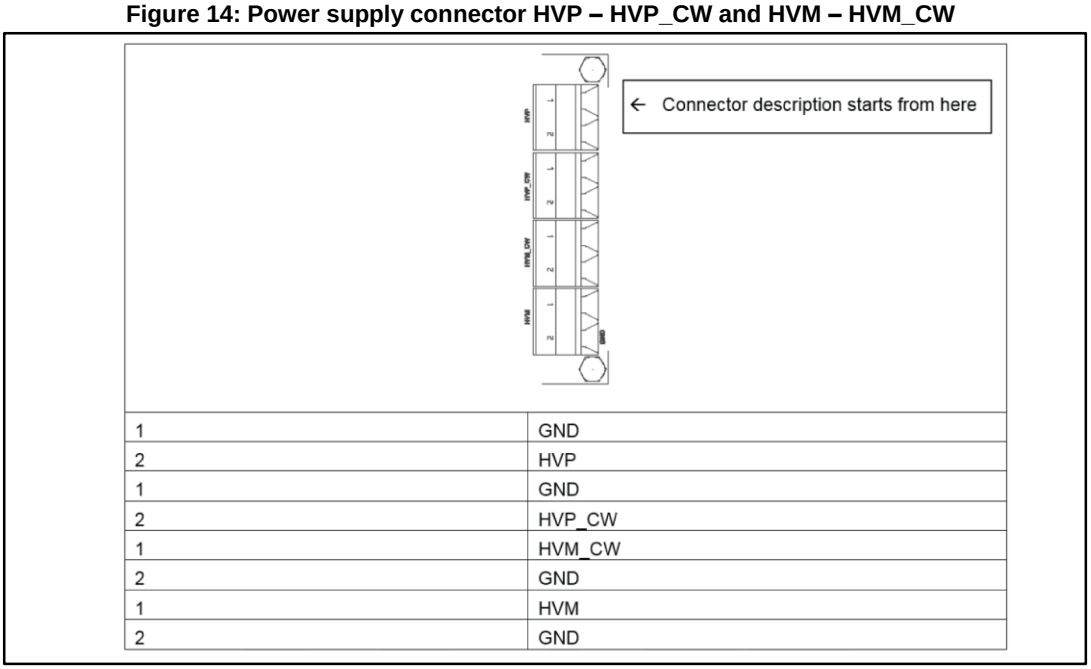
1	GND
2	VDD

Figure 13: Power supply connector VSS (GND - -5 V)



1	VSS
2	GND

Figure 14: Power supply connector HVP – HVP_CW and HVM – HVM_CW



1	GND
2	HVP
1	GND
2	HVP_CW
1	HVM_CW
2	GND
1	HVM
2	GND

4.2 Power-up sequence

1	VDD
2	VSS

The only recommendation for powering up the board is for low voltages as shown in above. Other voltages are fully power-up/power-down free, so there is no recommended sequence to follow.

4.3 MCU

Figure 15: USB mini-B connector (CN1)

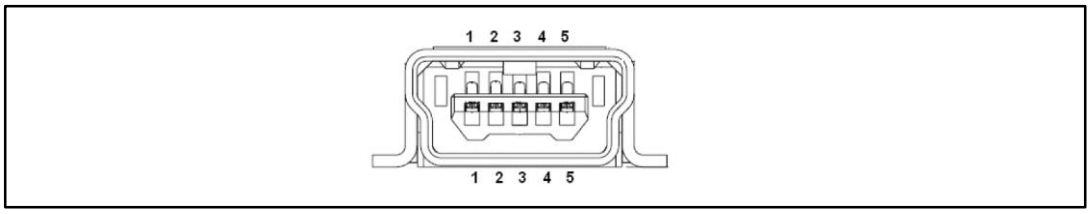
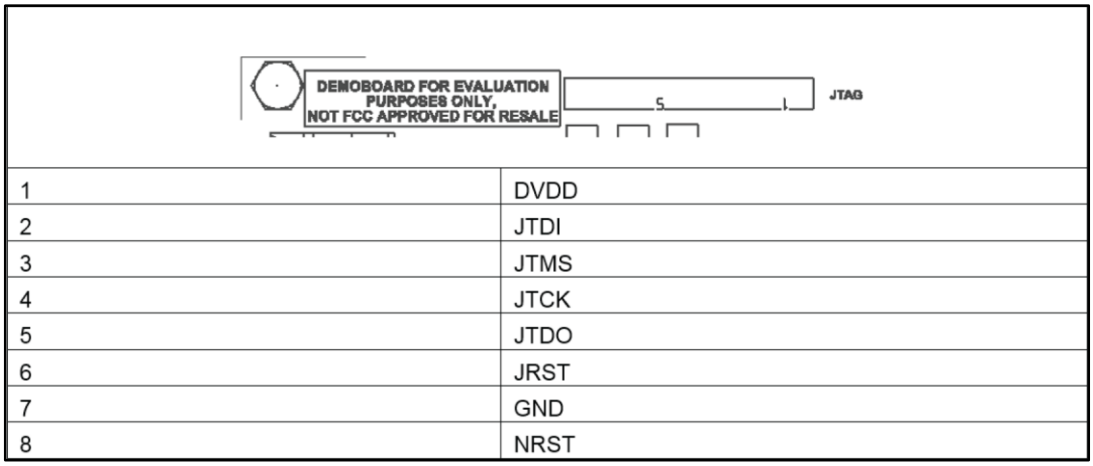


Table 6: USB mini B connector pin out

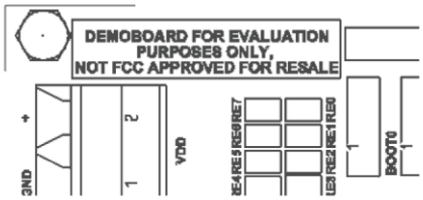
PIN number	Description
1	VBUS (power)
2	DM (STM32 PA11)
3	DP (STM32 PA12)
4	N.C.
5	Ground

Figure 16: JTAG connector



1	DVDD
2	JTDI
3	JTMS
4	JTCK
5	JTDO
6	JRST
7	GND
8	NRST

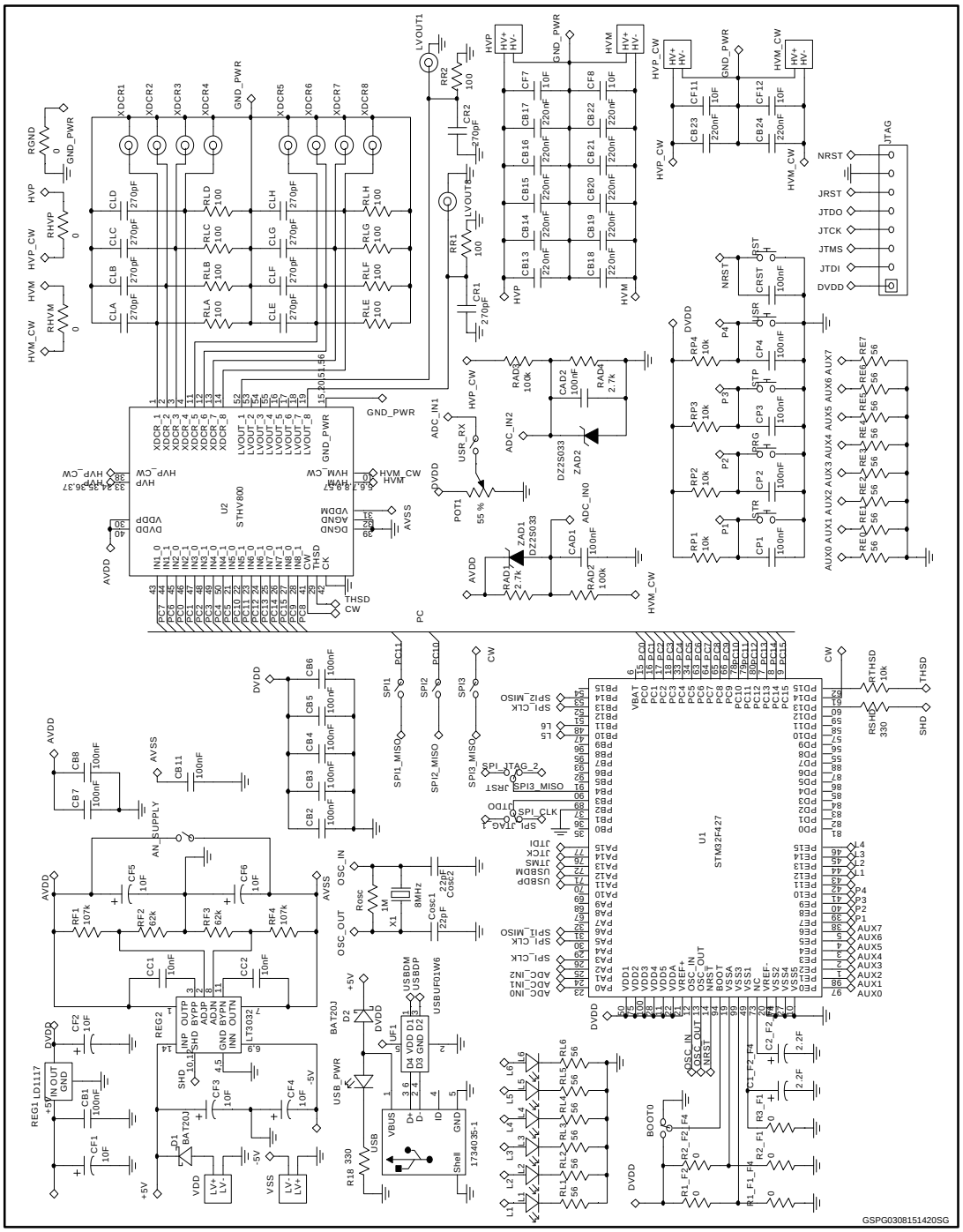
Figure 17: Boot connector



1	GND
2	BOOT0 (boot from flash memory)
3	DVDD (DFU mode)

5 Schematic

Figure 18: STEVAL-IME013V1 circuit schematic



6 Revision history

Table 7: Document revision history

Date	Version	Changes
02-Dec-2015	1	Initial release.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.