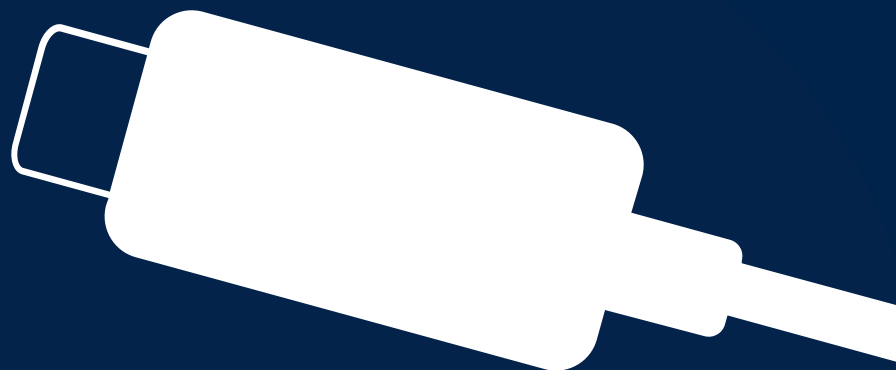




life.augmented



STUSB

STSW-STUSB006 Quick Start

STUSB4761 NVM customization through
I²C using NUCLEO-F072RB

v1.0



Table of Content

Part 1

Preparing Binary for Nucleo

Part 2

Connecting Nucleo for STUSB4761 Flashing

PART 1:

Preparing Binary for Nucleo





Software prerequisite

- IAR, Keil or ARM GCC are needed to recompile project
- Project compilation is mandatory if default NVM content is not suitable for application



Firmware preparation vs NVM content

- Project is delivered assuming default NVM configuration (as per datasheet):

The screenshot shows a file explorer window with the following structure:

- STSW-STUSB006
 - Drivers
 - EWARM
 - settings
 - STSW-STUSB006
 - Inc
 - MDK-ARM
 - Src

The right pane shows a table of files:

Name	Date modified	Type	Size
gpio.h	9/11/2019 3:40 AM	C/C++ Header	3 KB
i2c.h	9/11/2019 3:40 AM	C/C++ Header	4 KB
main.h	9/30/2019 7:48 AM	C/C++ Header	4 KB
mxconstants.h	3/28/2018 9:16 AM	C/C++ Header	3 KB
stm32f0xx_hal_conf.h	9/11/2019 3:40 AM	C/C++ Header	13 KB
stm32f0xx_it.h	9/11/2019 3:40 AM	C/C++ Header	3 KB
STUSB_NVM.h	9/18/2019 8:48 AM	C/C++ Header	1 KB
tim.h	9/11/2019 3:40 AM	C/C++ Header	4 KB
usart.h	9/11/2019 3:40 AM	C/C++ Header	4 KB
USB_PD_defines.h	9/30/2019 8:05 AM	C/C++ Header	4 KB

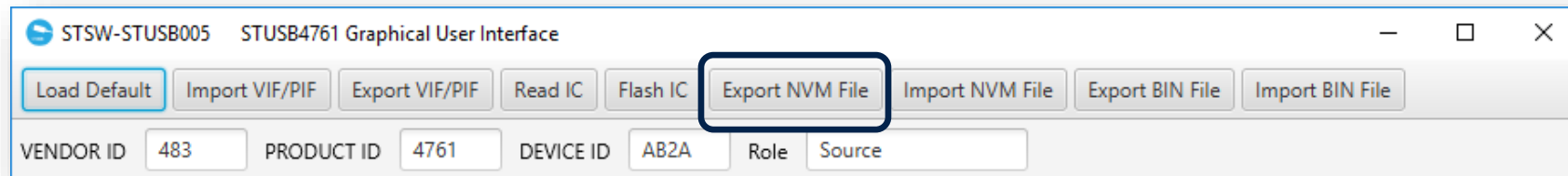
- Default [STUSB_NVM.h](#) file content:

```
uint8_t Sector0[8] = {0x83, 0x04, 0x61, 0x47, 0x2A, 0xAB, 0x98, 0x96};  
uint8_t Sector1[8] = {0x04, 0x08, 0x2C, 0x77, 0x2C, 0x74, 0x2C, 0x74};  
uint8_t Sector2[8] = {0x2E, 0x74, 0x2D, 0x74, 0x23, 0x74, 0xC0, 0x03};  
uint8_t Sector3[8] = {0x59, 0x38, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00};  
uint8_t Sector4[8] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x61, 0x01};
```



STUSB_NVM.h update

- NVM content is updated by modifying [STUSB_NVM.h](#) file
- STUSB4761 parameters can be customized to fit with application requirements using [STSW-STUSB005](#) Graphical User Interface and exporting custom [STUSB_NVM.h](#) file:



→ New [STUSB_NVM.h](#) file must be imported in STUSB-STUSB006 library, replacing the default file.

→ project compilation is mandatory to get updated binary file



IAR compilation

- IAR version used: 8.20 or later
 - Open project: Project.eww
 - After [STUSB_NVM.h](#) update, compile
 - New binary file is located here:

The screenshot shows a file explorer window with the path "STSW-STUSB006 > EWARM > STSW-STUSB006 > Exe". The window contains a table of files:

Name	Date modified	Type	Size
NVM_Flasher_STM32F072RB.bin	10/16/2020 9:21 AM	BIN File	20 KB
NVM_Flasher_STM32F072RB.out	10/16/2020 9:21 AM	OUT File	262 KB
NVM_Flasher_STM32F072RB.sim	10/16/2020 9:21 AM	SIM File	20 KB



Keil compilation

Keil version used: uVision 5

- Open project: NVM_Flasher_STM32F072RB.uvprojx
- After STUSB_NVM.h update, compile
- New binary file is located here:

A screenshot of a file explorer window showing the output of a Keil compilation. The path is STSW-STUSB006 > MDK-ARM > STSW-STUSB006 > Exe. The table below lists the generated files.

Name	Date modified	Type
<input type="checkbox"/> NVM_Flasher_STM32F072RB.axf	10/15/2020 4:03 PM	AXF File
<input type="checkbox"/> NVM_Flasher_STM32F072RB.bin	10/15/2020 4:03 PM	BIN File
<input type="checkbox"/> NVM_Flasher_STM32F072RB.hex	10/15/2020 4:03 PM	HEX File



GNU Arm Embedded Toolchain:

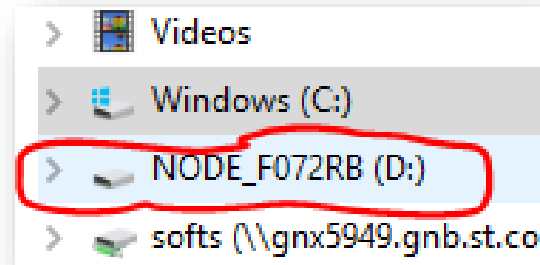
- version used: 9 2020-q2-update with GNUmake make-4.3
- Make using [Makefile](#) in STSW-STUSB006/MAKEFILE
- After [STUSB_NVM.h](#) update, compile
- New binary file is located here:

Name	Date modified	Type	Size
NVM_Flasher_STM32F072RB.bin	10/16/2020 11:06 AM	BIN File	18 KB
NVM_Flasher_STM32F072RB.elf	10/16/2020 11:06 AM	ELF File	493 KB
NVM_Flasher_STM32F072RB.hex	10/16/2020 11:06 AM	HEX File	51 KB
NVM_Flasher_STM32F072RB.map	10/16/2020 11:06 AM	MAP File	280 KB



Nucleo setup

- Connect NUCLEO-F072RB to PC
- Board will be mounted as mass storage device on windows browser:



- Drag and drop [NVM_Flasher_STM32F072RB.bin](#) in NODE_F072RB
→ LD1 of Nucleo is blinking during FW copy and then it is continuous green

or

- Use [STLinkUtility](#) to flash [NVM_Flasher_STM32F072RB.hex](#) in Nucleo board

PART 2:

Connecting Nucleo for STUSB4761 Flashing





Nucleo F072RB connections

Nucleo black-button:

- Reset stm32 firmware

Nucleo blue-button (PC13 available on Nucleo CN7 – pin23):

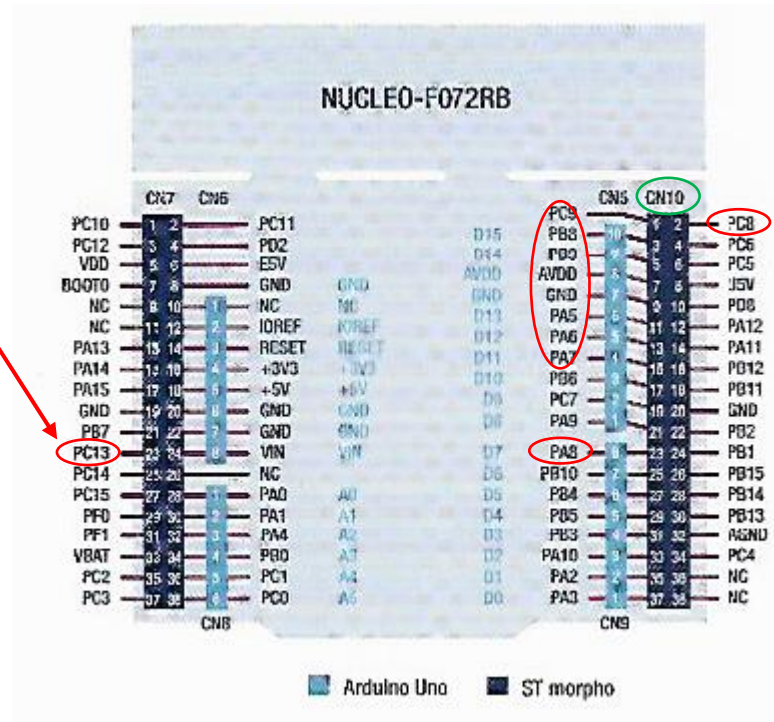
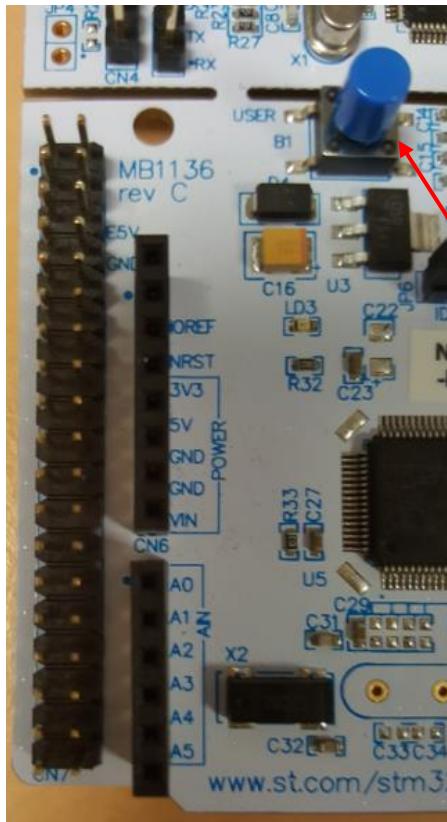
- Press: reset to default condition
- Release: start new flashing

STM32F072 pin	CN10 pin number (Nucleo)	Target board	Function
PC8	1	LED1	Orange LED: ON when STUSB4761 is not supplied (default condition) or action on going
PC9	2	LED2	Red LED: ON if NVM flashing fails
PB8	3	I2C_SCL	To be connected to stusb4761 SCL pin (pull up to 3V3 mandatory)
PB9	5	I2C_SDA	To be connected to stusb4761 SDA pin (pull up to 3V3 or stusb4761 Vreg_2v7 mandatory)
GND	9	GND	To be connected to stusb4761 GND
<i>AVDD</i>	<i>7</i>	<i>3V3</i>	<i>3V3 for I2C pull up if no pull up on stusb4761 board</i>
PA5	11	Nucleo LD2 Green LED	Green LED: ON when flashing done and OK
PA6	13	VCONN	Supply stusb4761 during NVM flashing
PA7	15	PWMout for Buzzer	Buzzer ON if flashing fails
PA8	23	Board connection detection	Active Low. Mechanical board insertion detection

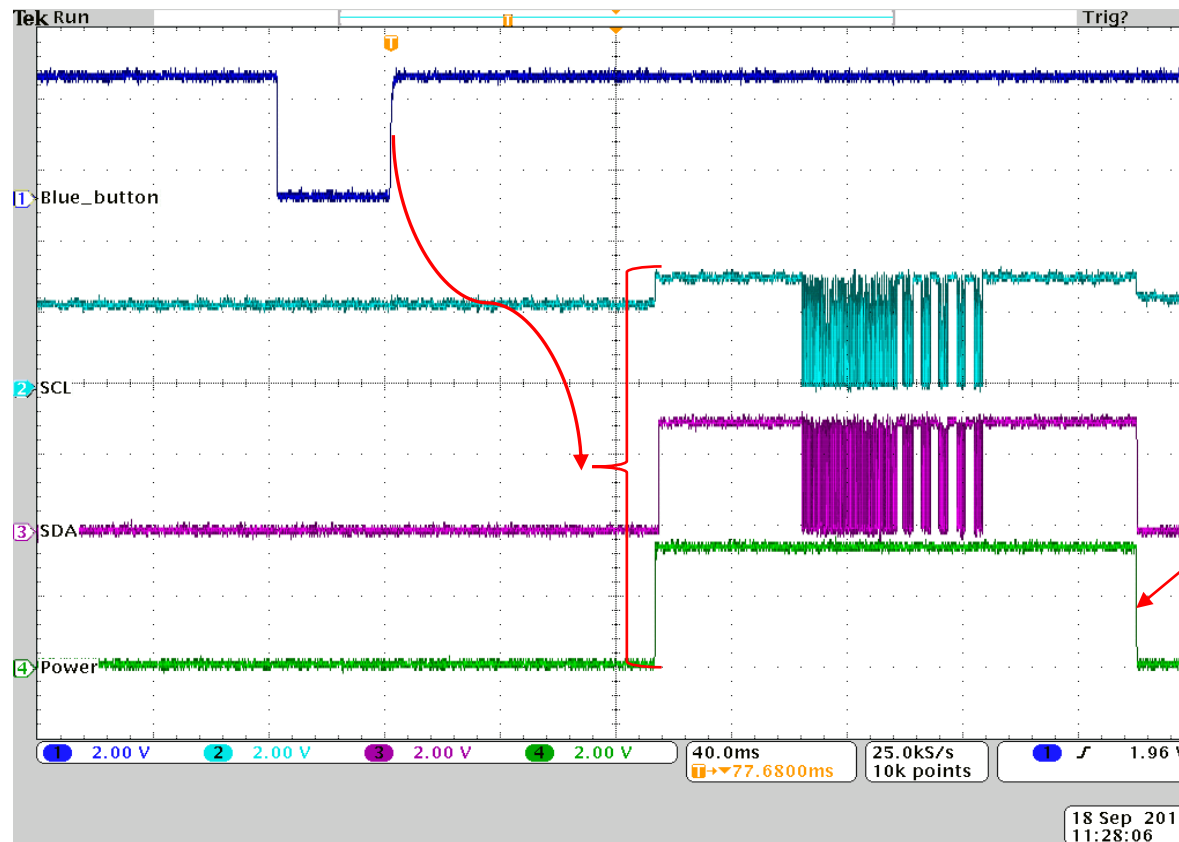


Nucleo F072RB connections location

- See below physical connection locations on Nucleo: red circles



- Rising edge of blue_button, enables power for STUSB4761 and I2C sequence starts 50ms after power up

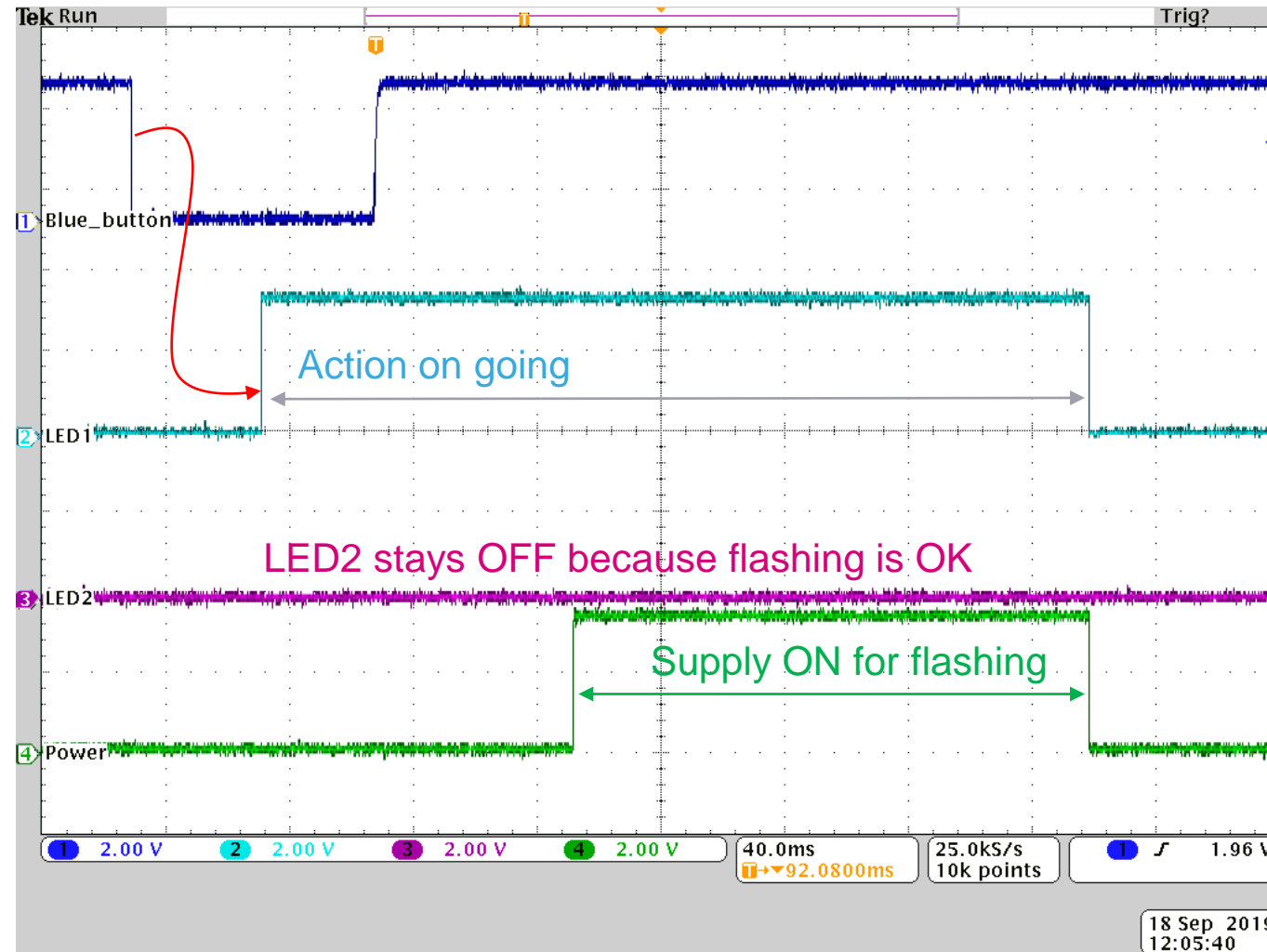


Flashing ends: supply is stopped



LED1 & LED2 behavior when flashing OK

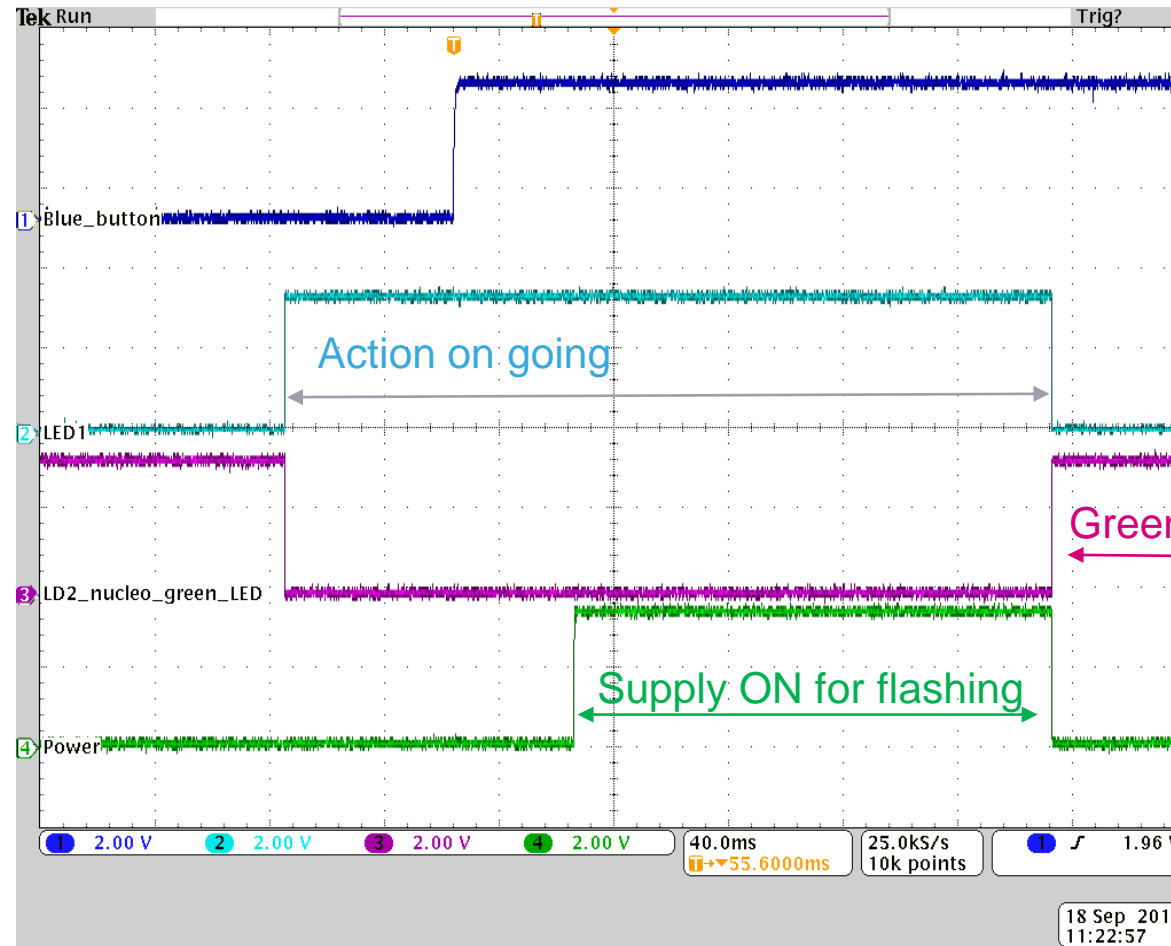
- LED1 is ON during flashing sequence. As flashing goes well LED2 stays OFF after flashing ends





LED1 & LED2 behavior when flashing OK

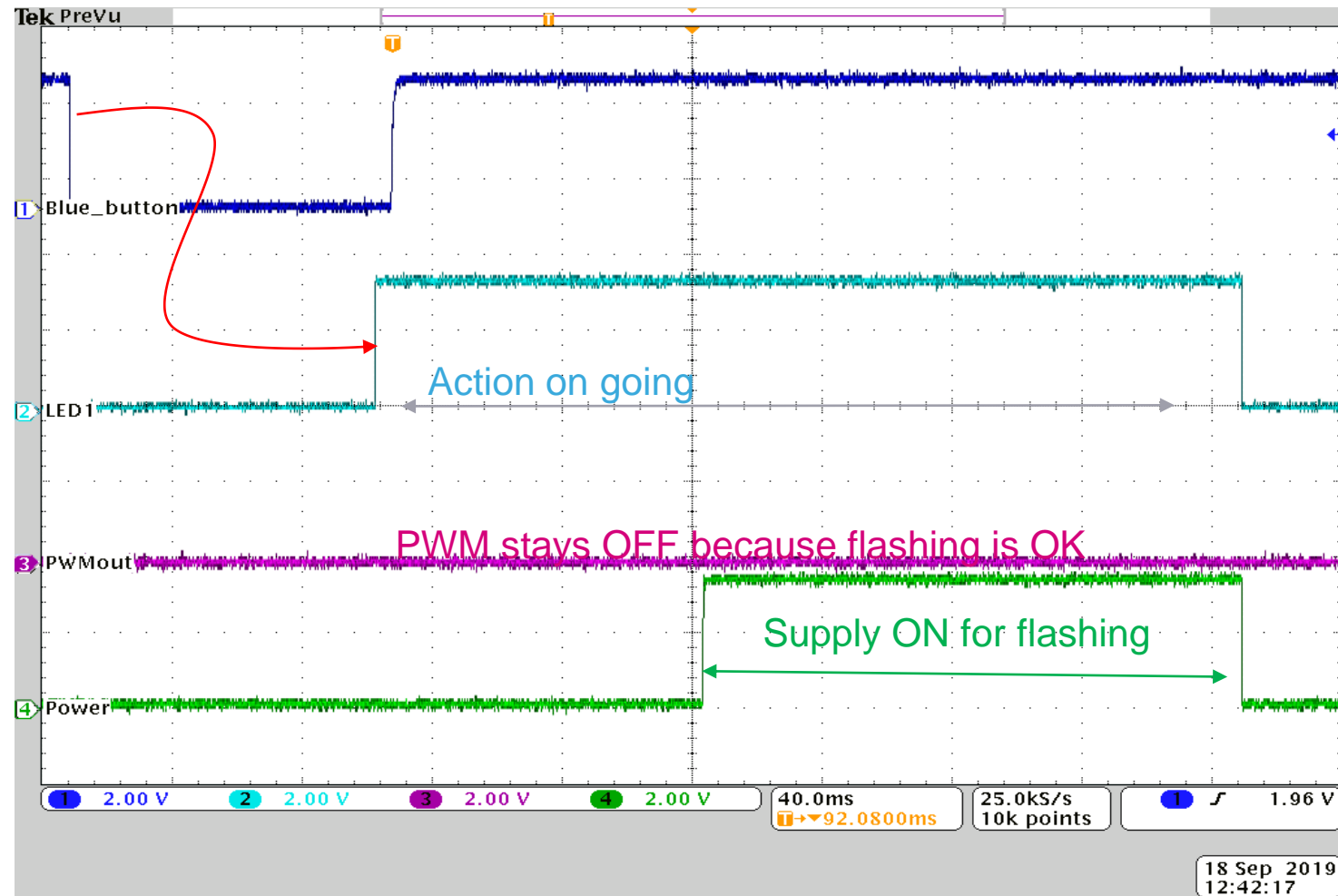
- LED1 is ON during flashing sequence, as flashing goes well Green led is ON after flashing ends





LED1 & PWM behavior when flashing OK

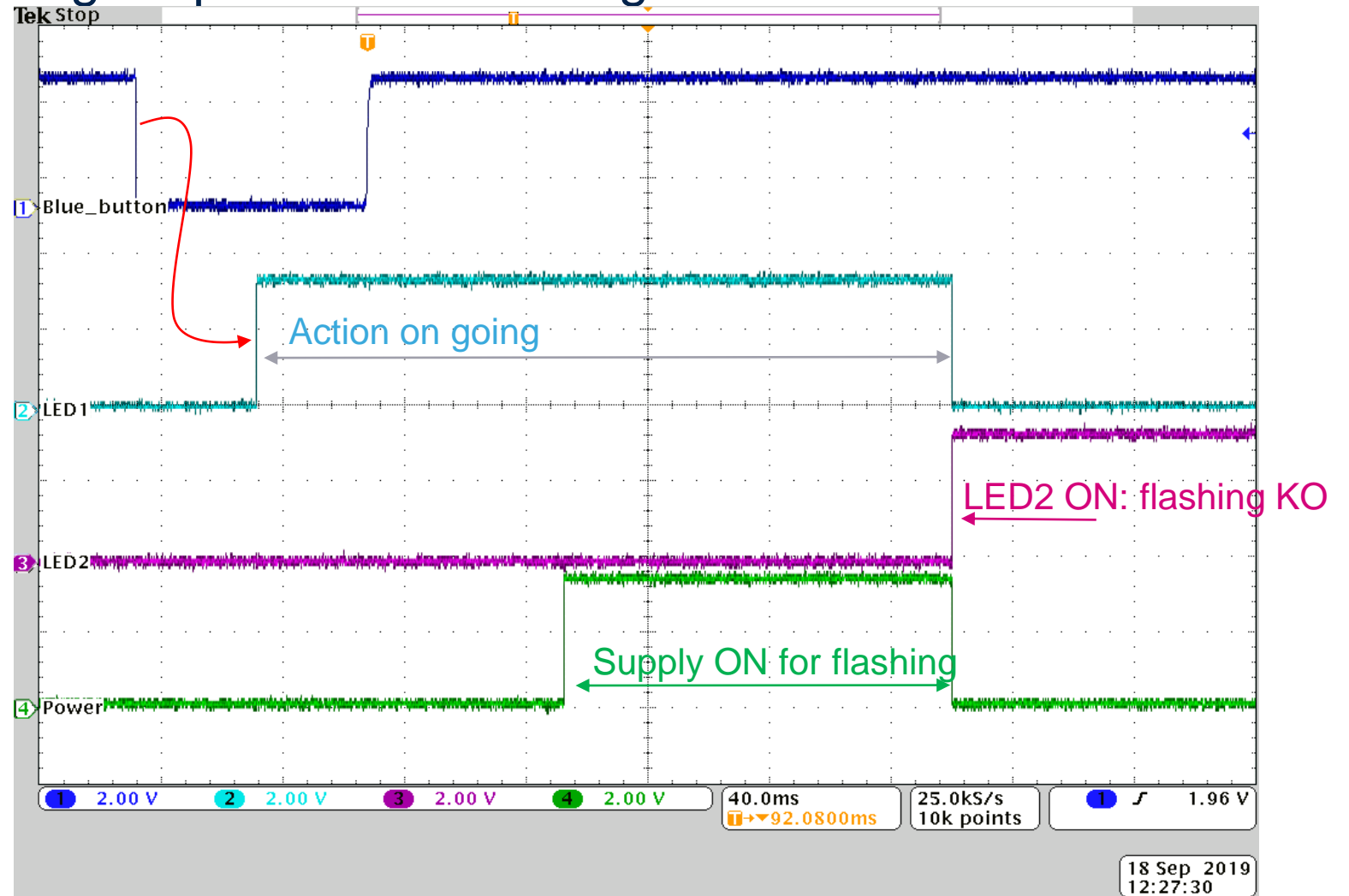
- LED1 is ON during flashing sequence. As flashing goes well PWM stays OFF after flashing ends





LED1 & LED2 behavior when flashing fails

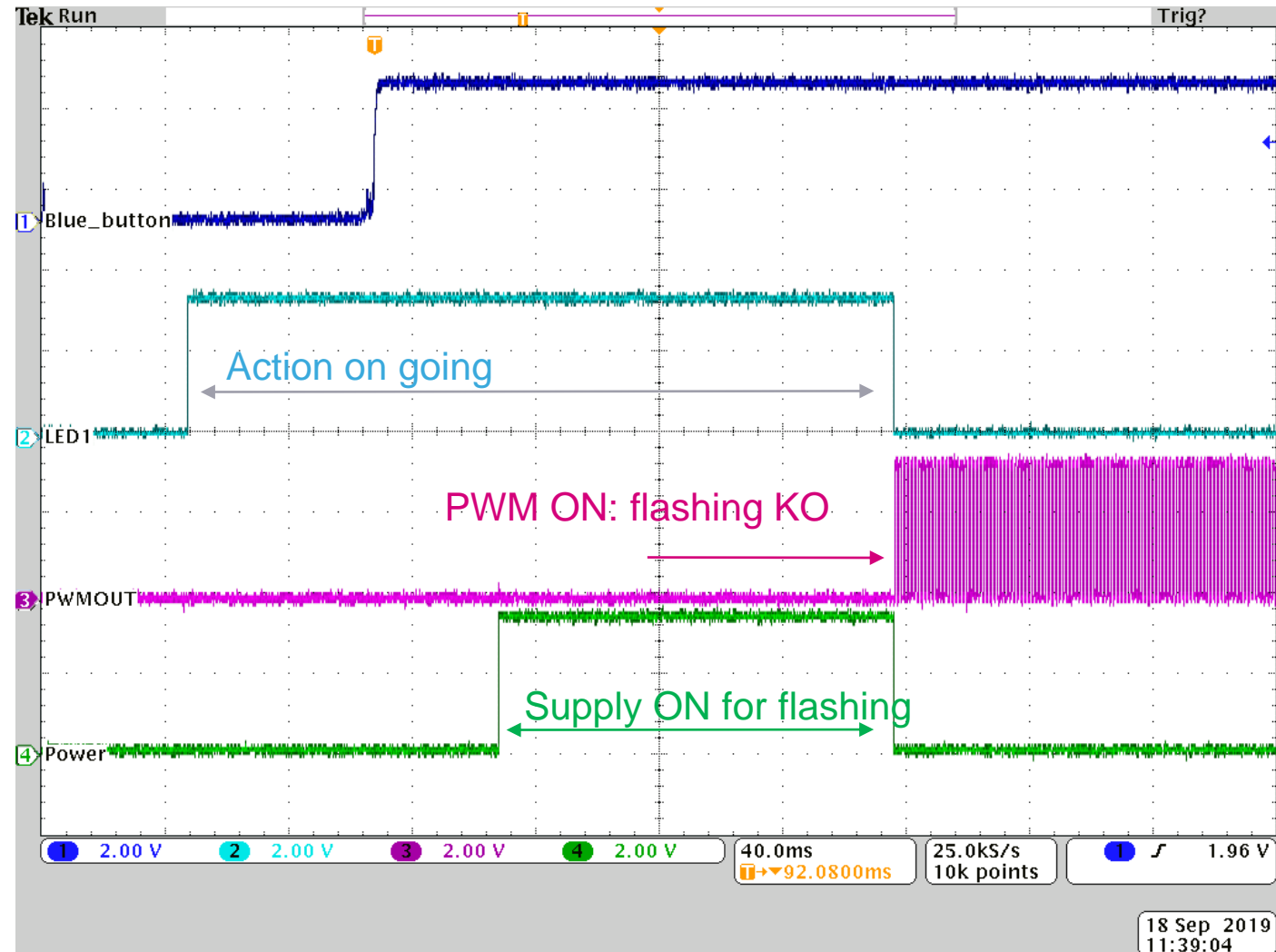
- LED1 is ON during flashing sequence. As flashing fails LED2 is ON after flashing ends





LED1 & PWM behavior when flashing fails

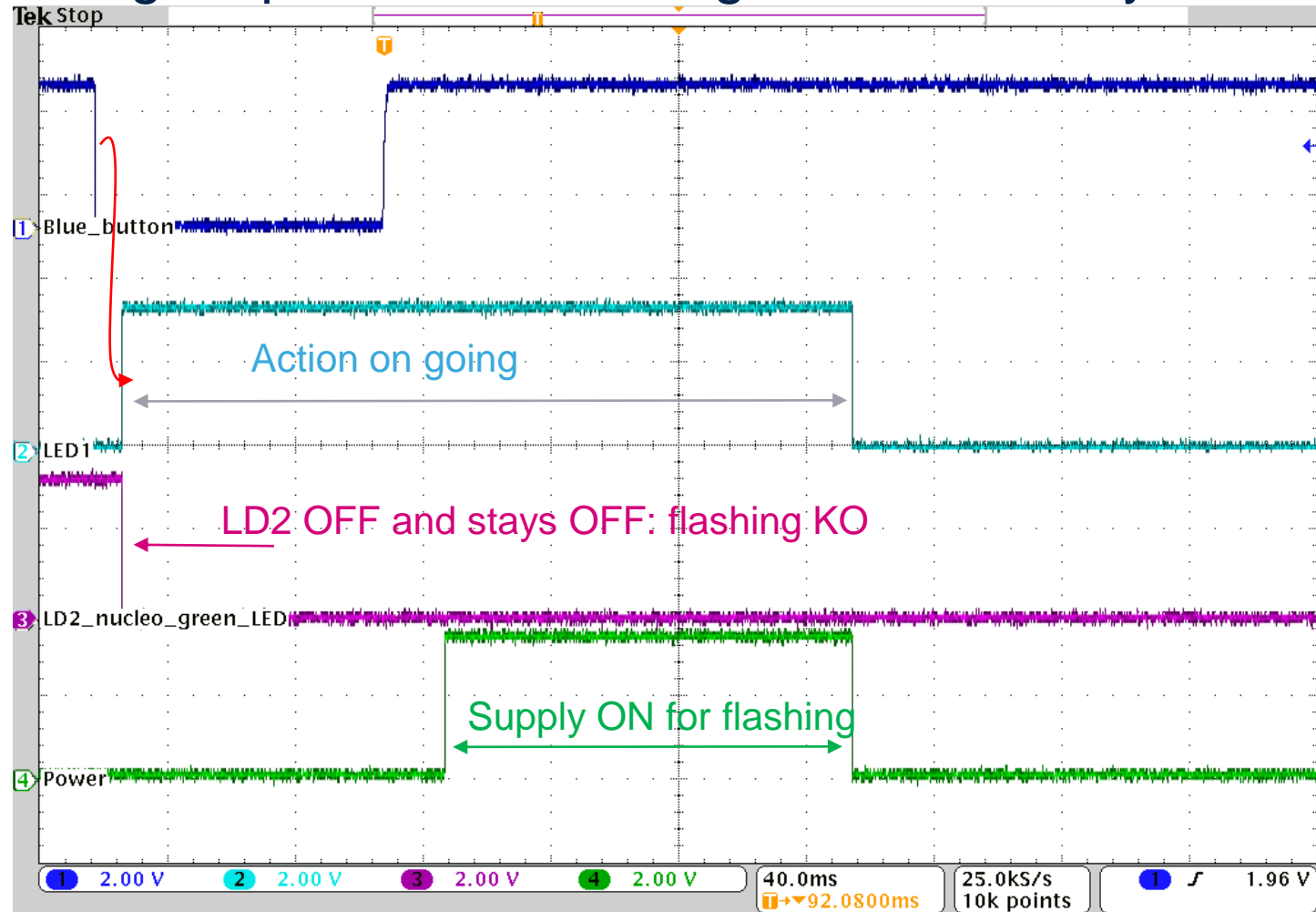
- LED1 is ON during flashing sequence. As flashing fails well PWM is switched ON after flashing ends



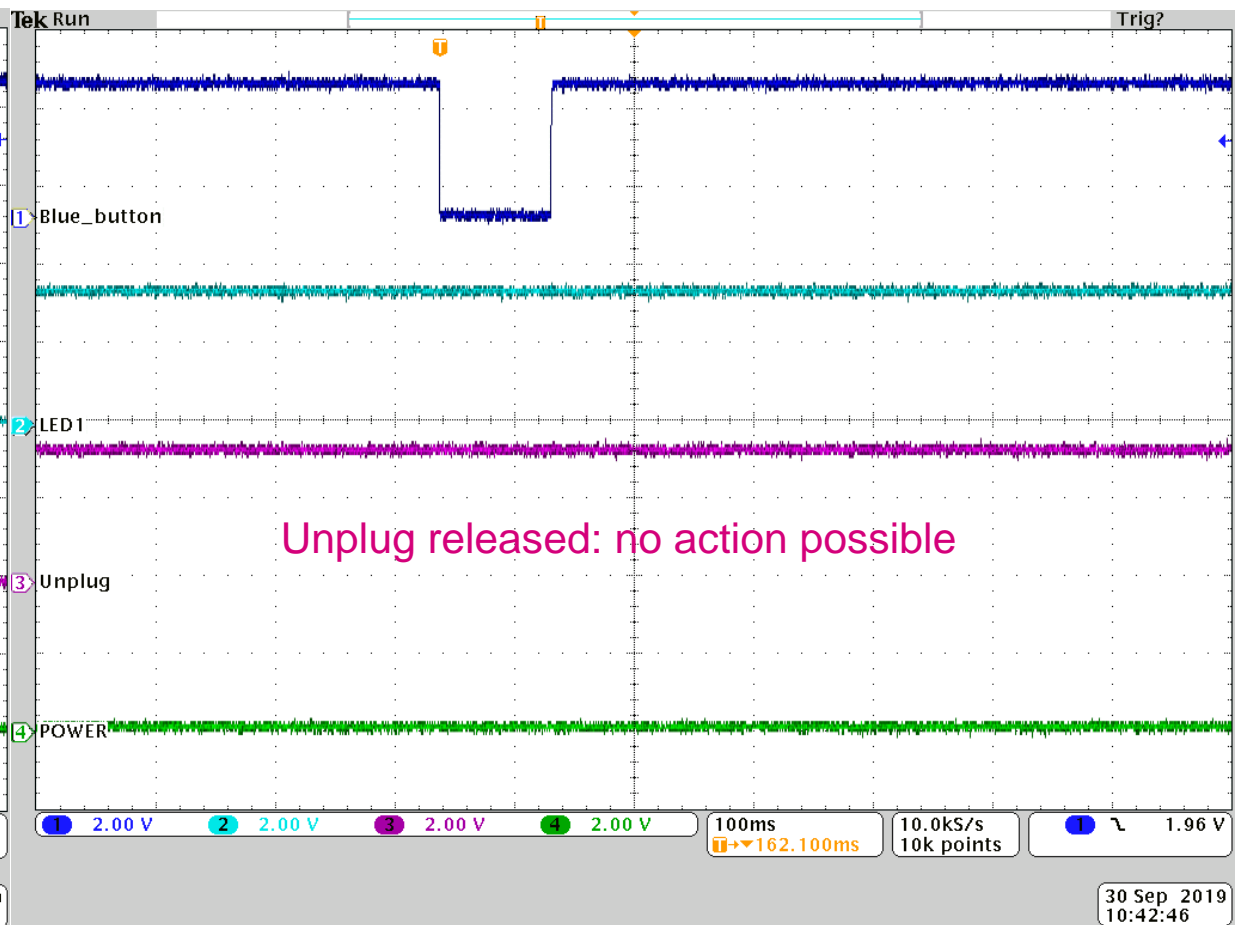
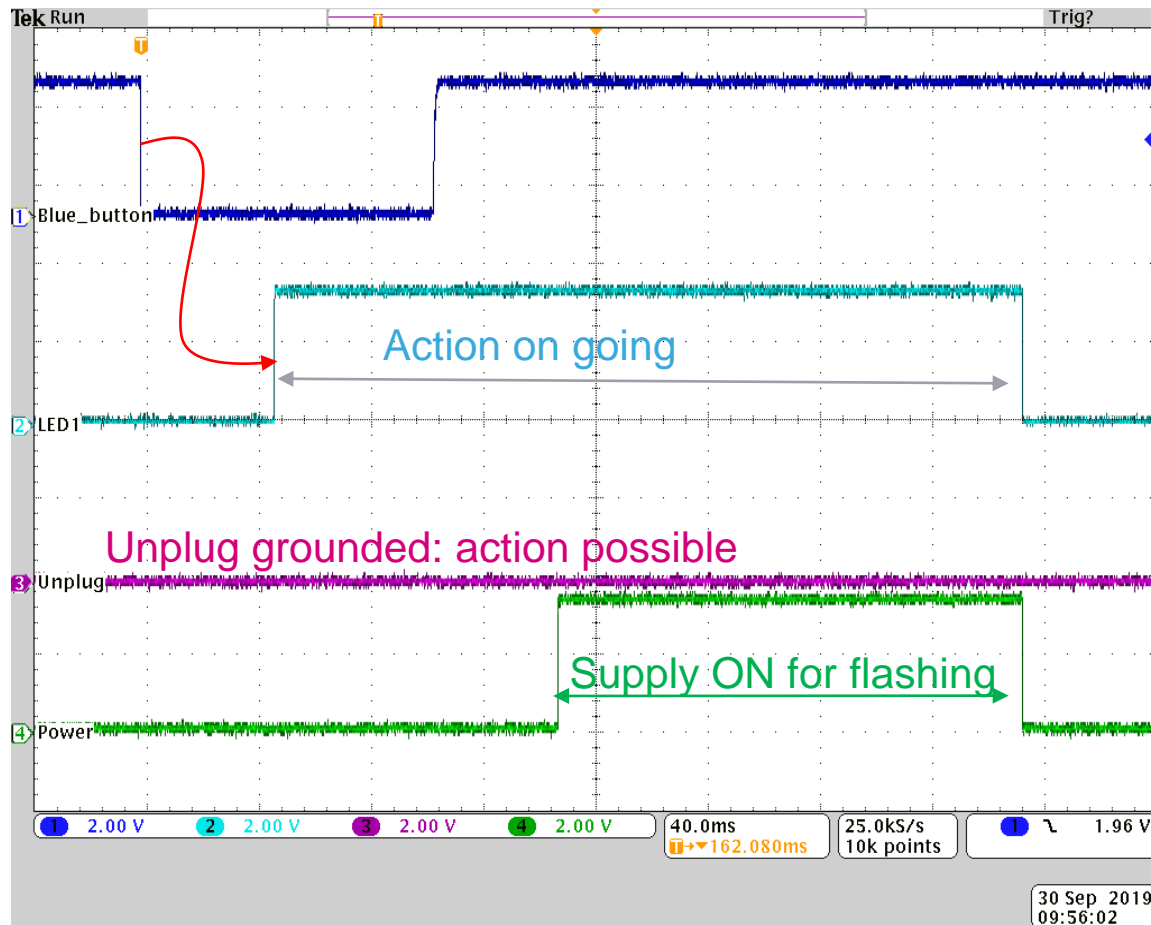


LED1 & LED2 behavior when flashing fails

- LED1 is ON during flashing sequence. As flashing fails LED2 stays OFF after flashing ends



- Behavior difference when Unplug pin is grounded or released while Blue button is pressed



LEDs behavior when Unplug set

- When Unplug in is released, Green LED and LED2 go to OFF. LED1 back to ON

