USB-6000 Test Strategy

Roman Iutsis

Structure

1. Simulation
2. Hardware
3. Software
4. Strategy scheme

Simulation

- Engineers during the development of PCB usually use Software simulation. Usually, it needs to understand the maximal current or signal form
- Some simulation SW is able to simulate MCU for Firmware. For test it is safer to run must-pass tests.
- It possible to implement in container and run tests when open a PR for new Firmware

Simulation: Test Example

1. Spec for counter input: max frequency is 5 MHz -> Expected that it works with all frequency under this
    1.1. Run simulation with connected impulse generator to ‘PFI 0’
    1.2. Run impulse generator with task:
        1.2.1. from 1 Hz to 10 kHz with step 1 kHz
        1.2.2. from 10 kHz to 100 kHz with step 30 kHz
        1.2.3. from 100 kHz to 1 MHz with step 300 kHz
        1.2.4. from 1 MHz to 5 MHz with step 1 MHz
    1.3. For each step wait except step 0 wait 500 ms. For step 0 wait 1s. After each step reset counter
    1.4. Calculate number of impulses for each step and compare date for output.

Notes:

- Steps has this resolution for minimize time and cover most relevant ranges
- Maybe It is a good idea to set up some accuracy for each step

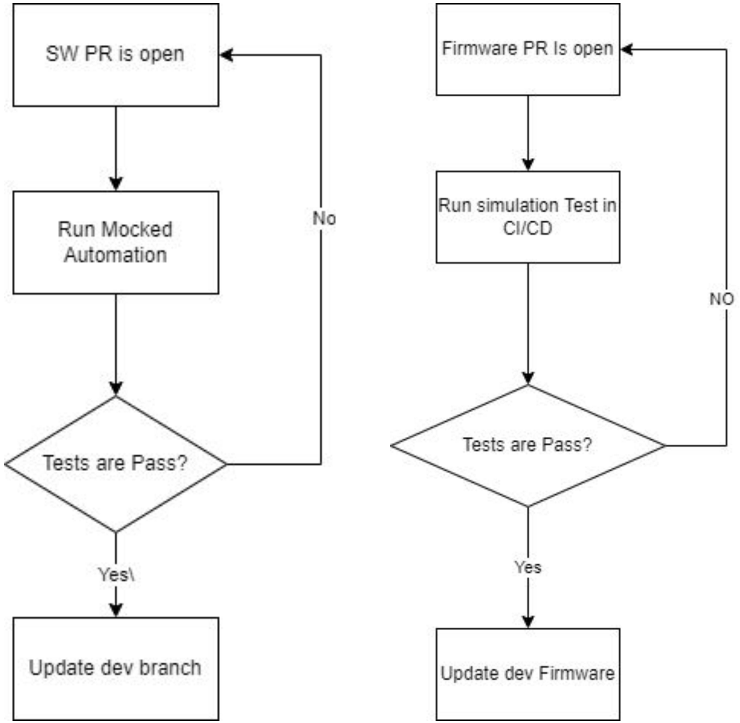
Hardware

- For test hardware possible to use special benches and firmware
- According user guide USB-6000 has 4 Digital IO(P0-P3), 8 Analog inputs(AI0 -AI7), 1 counter input, 1 digital trigger
- Two possible solutions for test:
    - 1) using builtin IOs. For example, possible to link all digital IO with P0 with other digital IO and send 2 IOs in all pairs (one from P0 second from another input). After this possible to understand which inputs are working. For analog inputs it possible to link using digital voltage divider.
    - 2) using outside signals. We should generate signals from the bench
- For communicate with device
- Control IO is possible by official library. For the bench possible to write special framework

Software

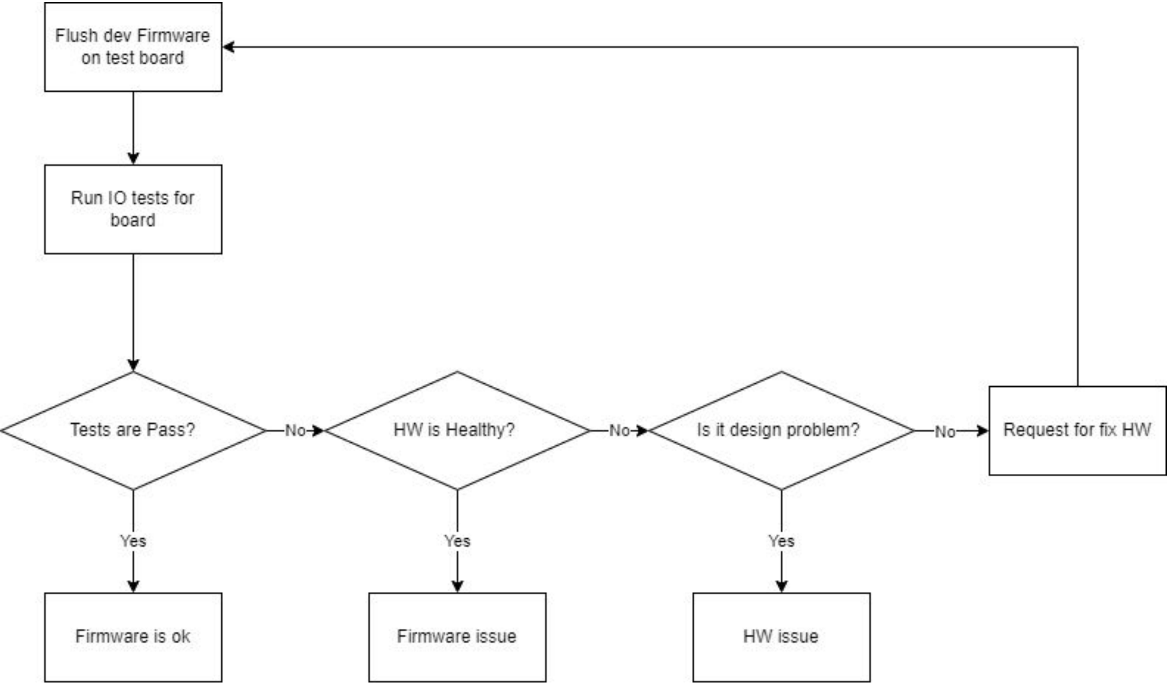
- Device has integration with LabVIEW, LabWindow and Measurement Studio. But only one Software DAQExpress
- For test we can use simple test:
    - Can SW detect device when it plugin?
    - Can we set IO for input-output?
    - Does signal form shows correct?
-

Strategy



Tests for FW and SW looks the same and some part of these releases are possible to cover by automation. After tests are passed, it possible to go for HW tests

Strategy



After SW and FW tests are done it is possible to run some end-to-end tests. It covers HW tests also and it is better to do it at the end of the test cycle due to HW change much rarely than SW and FW