# *TEST PROCEDURES FOR SOFTWARE DEFINED RADIO PLATFORMS*

an effort to standardize test procedures for actual SDRs

Fraunhofer
IIS

Felix Auer, Christopher Laske

# *TEST PROCEDURES FOR SDR PLATFORMS*

## CONTENTS

- Motivation
- Test Procedure Specifications
- Implementation of Test Cases
- Results
- Observations
- Summary and Outlook

# Motivation

- available: International Test Operations Procedure (ITOP)
  - compiled 1993/1995 by authorities from USA, France, Germany for analog and digital transceivers (ITOP 6.2.242 and ITOP 6.2.246)
  - targeted test items: complete (tactical) radios for a fixed waveform
  - not well suited to test SDR platforms *independent* of a waveform

→ **long dispute with vendor about how to test**
(in a recent case where the ITOP was referenced in the specifications)

→ desirable:

- standardized test procedure specification for characterizing and comparing actual (and future) SDRs independent of a waveform
- unambiguous parameter definitions and test procedures

# Idea

- take the ITOP as starting point
- modify the test cases to adapt to the interfaces of actual SDR-platforms
- remove test cases which are no longer important
- add test cases which are relevant but not yet addressed
- verify the test procedures by implementing them

- distribute the new specification for commenting
- standardize the new specification

work started at *Fraunhofer IIS* together with *Wehrtechnische Dienststelle für Informationstechnologie und Elektronik (WTD 81)* in Greding

***contributions welcome***

# Goals

- main topic: RF parameters (like ITOP)
- suitable for actual SDR platforms with digital interfaces
- useful for
  - assessing the performance independent of waveform
  - comparing platforms
  - identifying the best SDR for one's needs
  - determining suitability for new or future waveforms
- providing unambiguous parameters and procedures
- universally accepted

# Parameters

**Transmitter**

- Output Power/Power Consumption, Usable Bandwidth, Mismatched Loads, Frequency Resolution, Continuous Transmission Time
- Frequency Accuracy, Wideband Noise, Spurious
- Compression, Harmonics, Intermod, EVM
- Cosite Intermod, Output Power Control
- TX Hop Time, RX-to-TX Delay, Retransmission Delay

**Receiver**

- Gain Factor, Usable Bandwidth, Sensitivity, max SINAD, Dynamic Range, Input Impedance
- LO/Clock Emissions, Residual Spurious/Noise
- in-band Intermod
- out-of-band Intermod, Selectivity, Blocking, Input Protection, Spurious Response
- RX Hop Time, TX-to-RX Delay

# Examples

- Receiver In-Band Intermodulation
  - covered in ITOP
  - parameter and measurement method widely accepted
- Receiver Hop Time
  - not covered in ITOP
  - example for timing parameter (Transmitter Hop Time, RX-to-TX Delay, Retransmission Delay)
  - no widely accepted generally usable measurement method

# Test Procedures

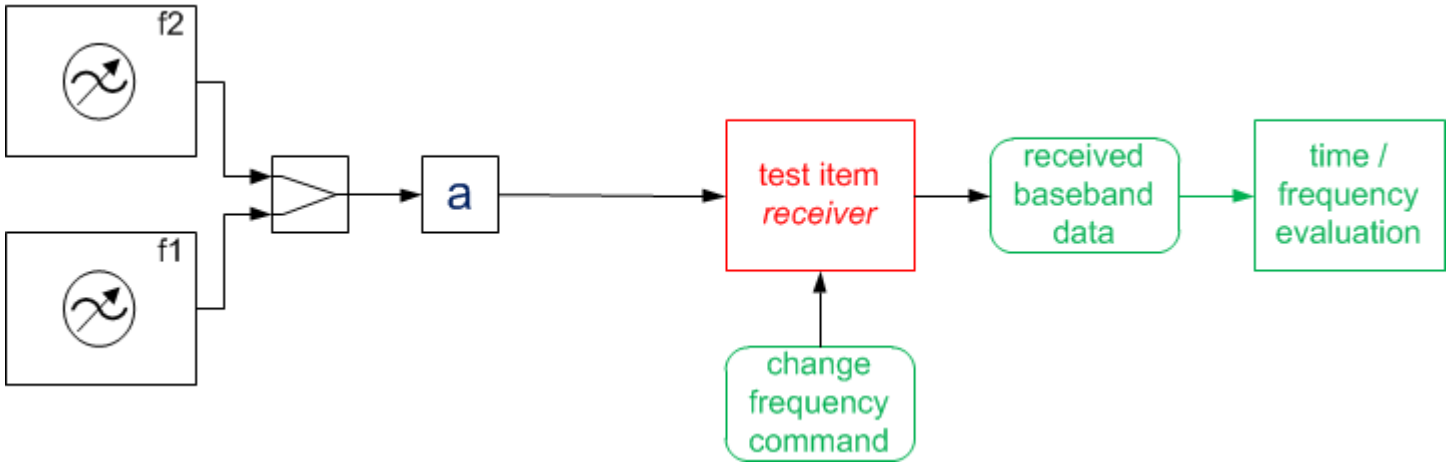
- Receiver In-Band Intermodulation
  - mainly according to ITOP
  - variable two-tone distance to characterize bandwidth behavior
  - adjusted for a digital (baseband) data interface

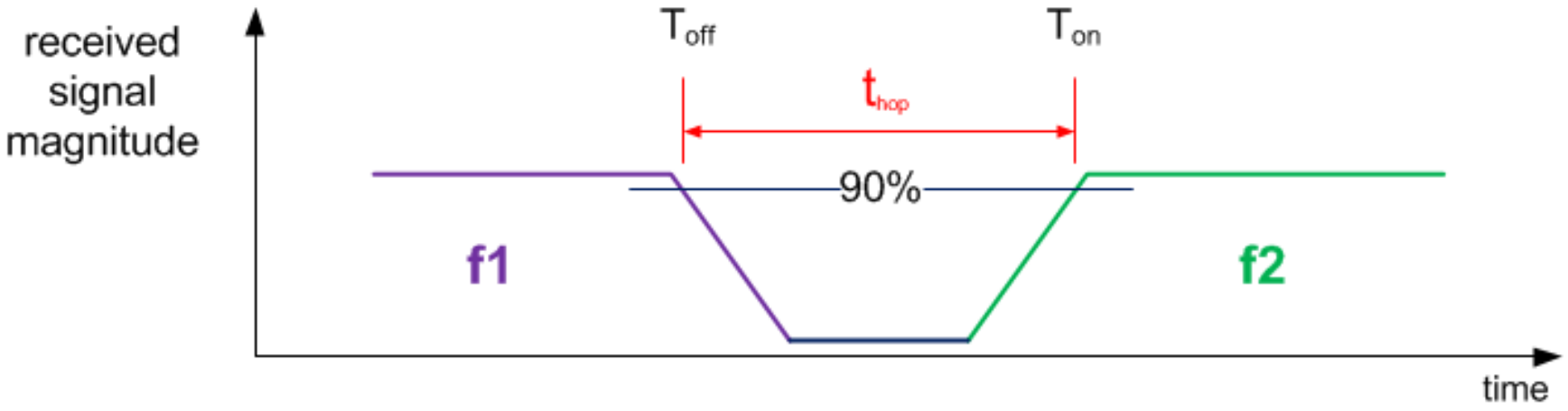
- Receiver Hop-Time

  *How fast is the receiver when hopping between frequencies $f_1$ and $f_2$?*

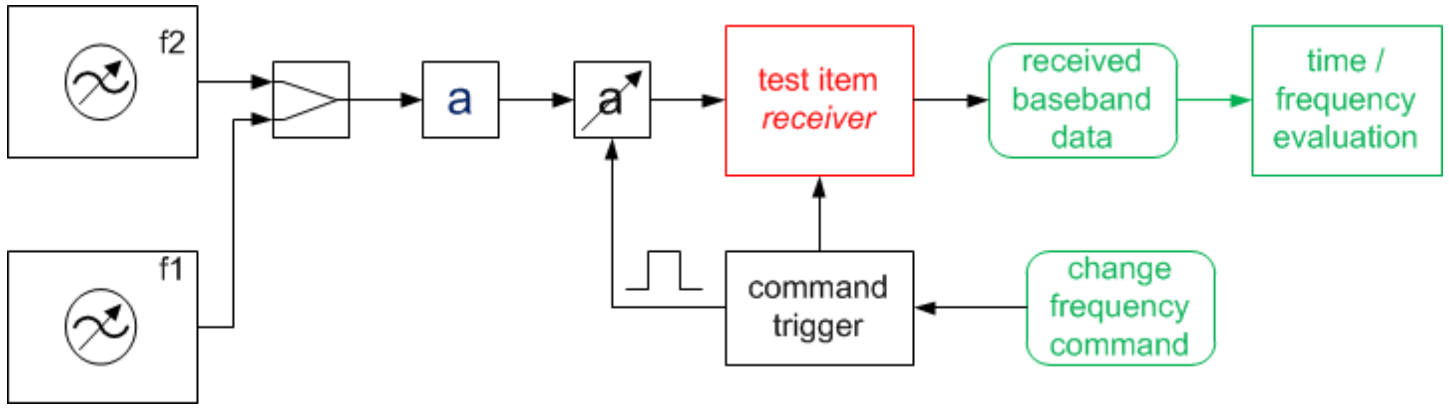
  - Definition:
    time duration between received signal level from $f_1$ down to 90% and from $f_2$ up at 90% of steady state level
  - Command Delay (“change frequency immediately”):
    time duration between command present at test item interface and observable effect (here: signal level down to 90% steady state)

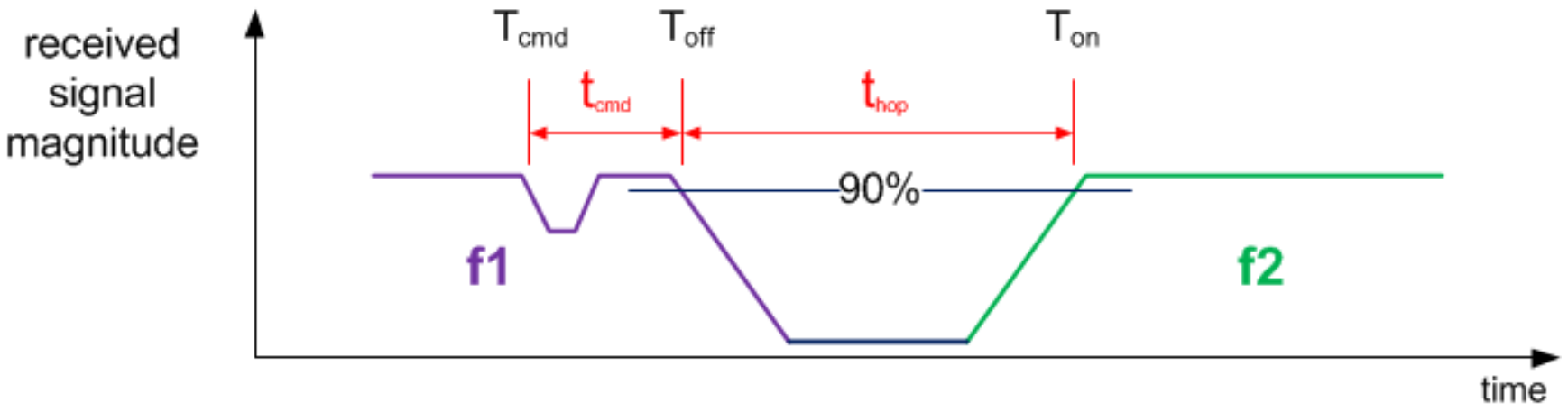
# Test Procedure – RX Hop Time



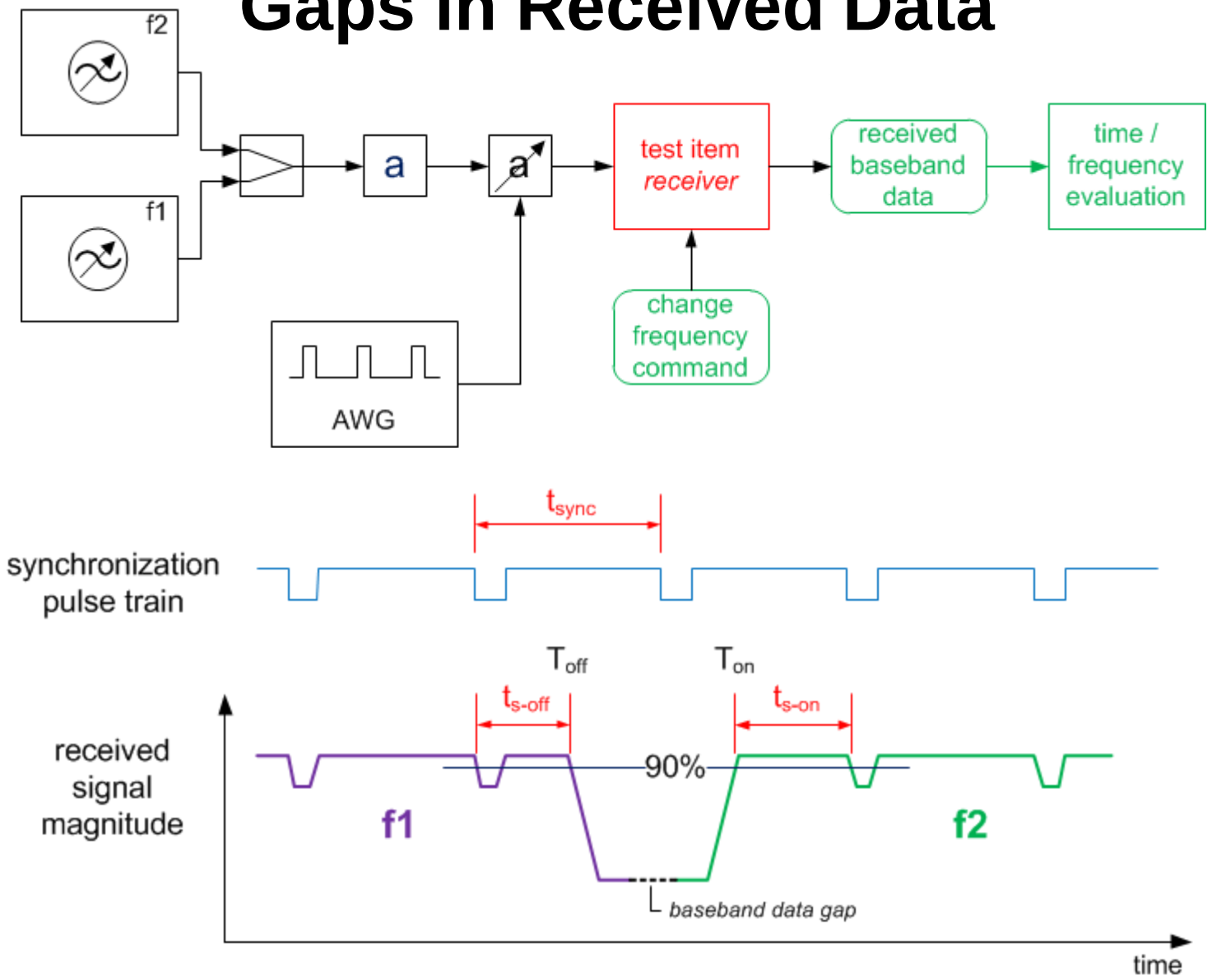



# Test Procedure – Command Delay



# Gaps in Received Data



# Implementation
## Set-up and DUT

- Measurement Set-Ups
  - Common lab equipment
  - Plug and Play components

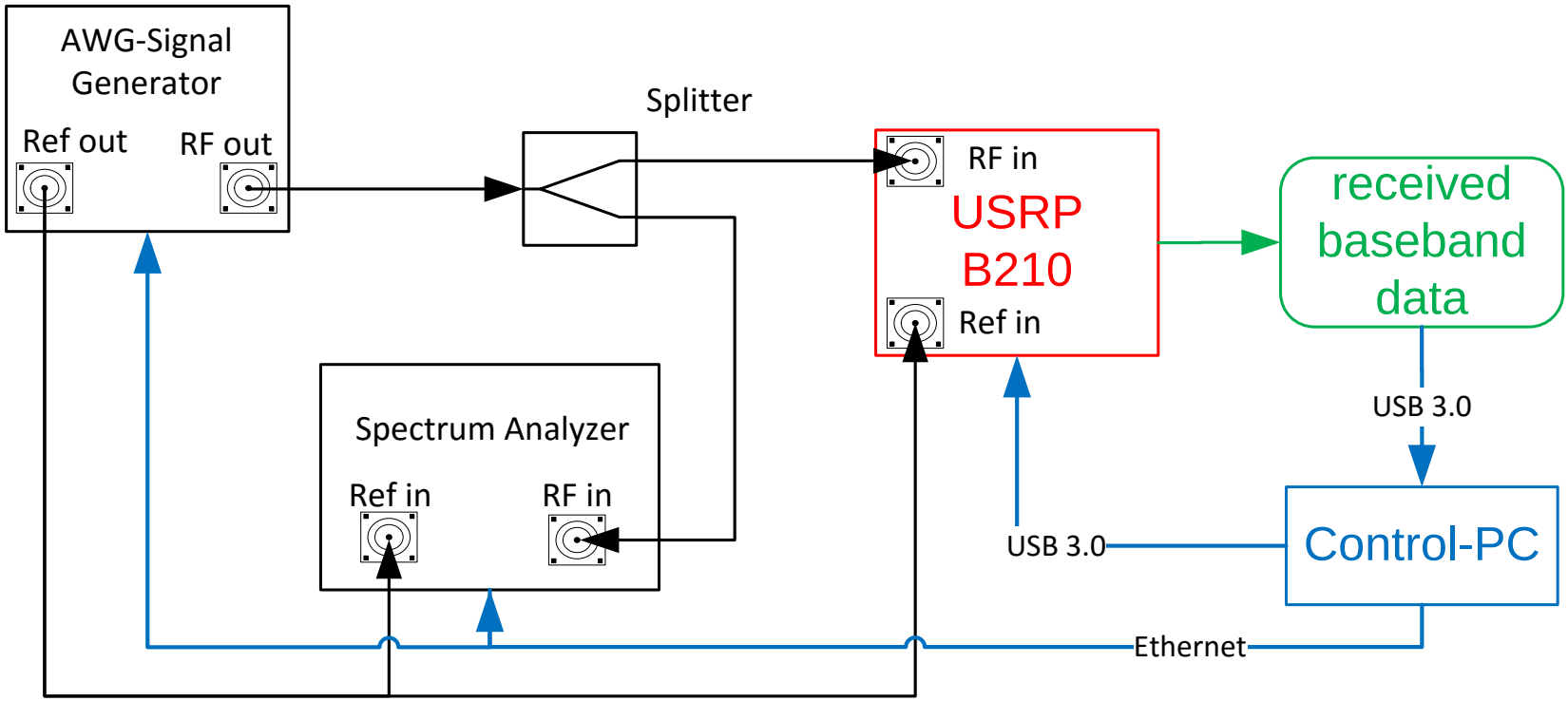
- DUT: Ettus USRP™ SDR
  - USRP B210 Bus Series SDR
  - Commercial SDR, based on the RF-Chip AD9361
  - Widely used in Universities and research Labs
  - Open Source and stand alone API

# Implementation
## Software Platform and Automation

- Automation and Software Platform
  - Python 2.7 programming language
  - Virtual Instrument Software Architecture (VISA)
  - PyVISA and PyVISA-py
  - Network and GPIB interfaces
  - C++ to communicate with the USRP

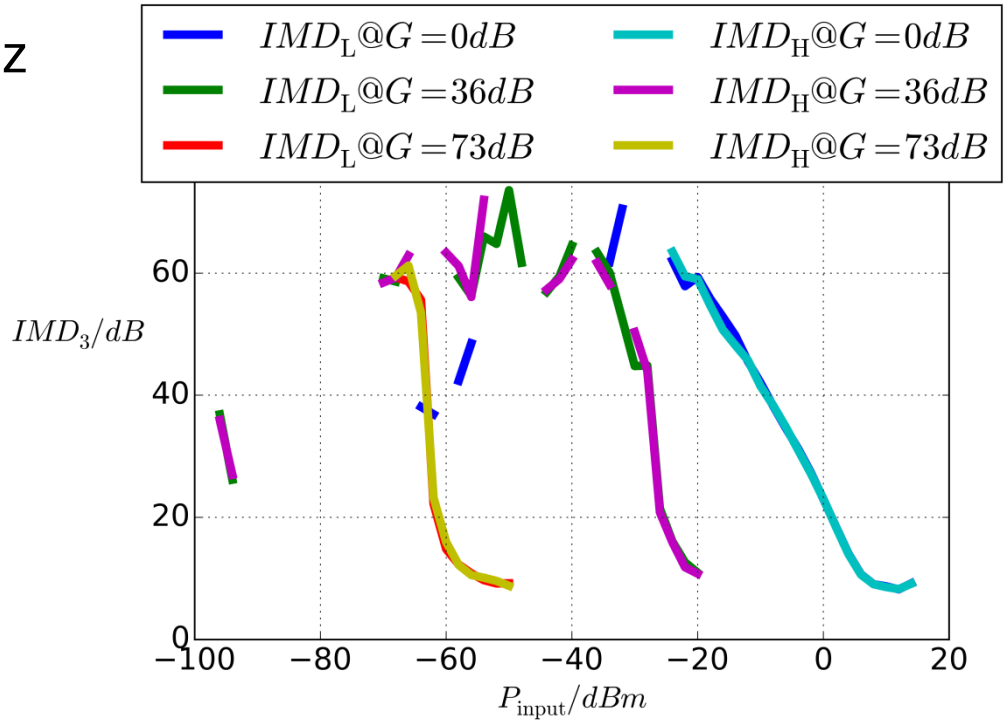
# Measurement Results
## Intermodulation Test

- Two-Tone Set-up

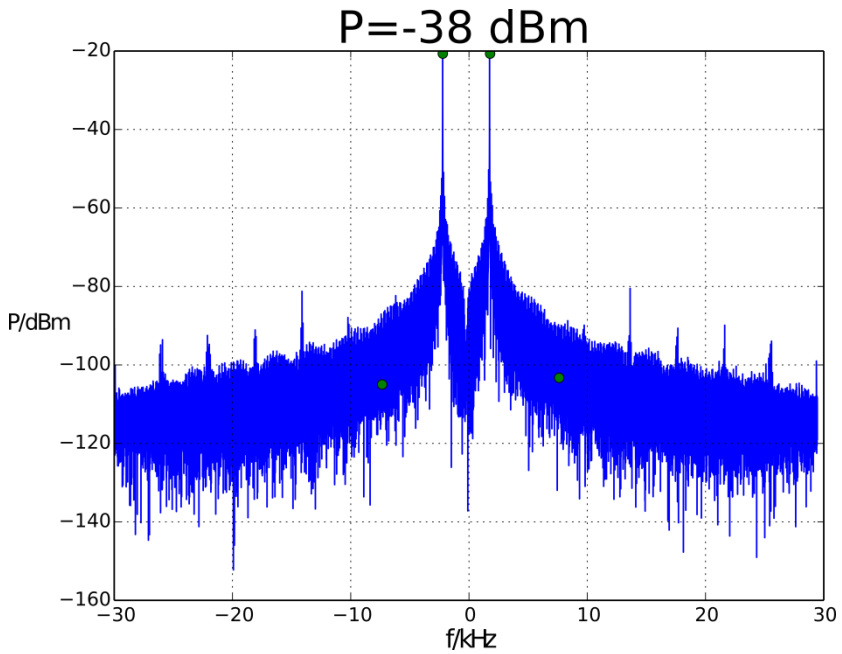


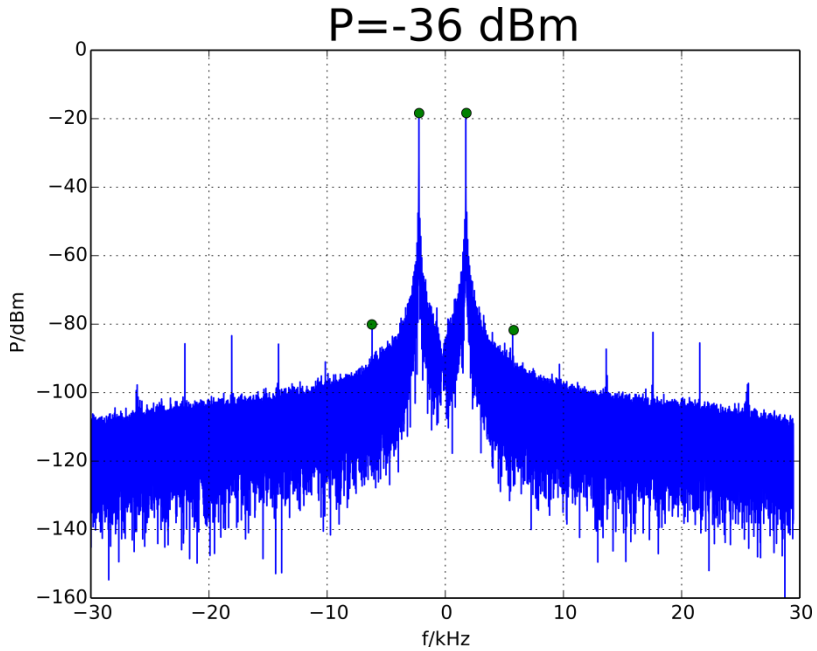
# Measurement Results
## Intermodulation Test

- Bandwidth 8 kHz
- IQ-Signal at 2 GHz +- 4 kHz
- Sampling 60 kHz
- 80 kSa
- f_sigGen = 2 GHz
- f_usrp = 2 GHz



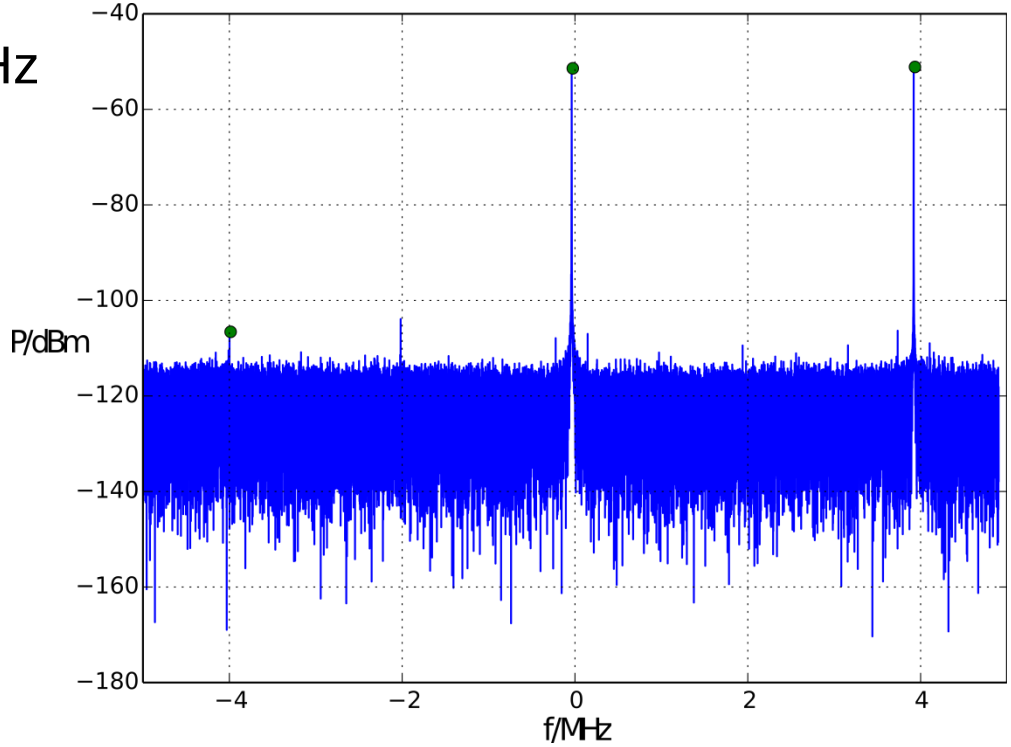
# Measurement Results
## Intermodulation Test



# Measurement Results
## Intermodulation Test

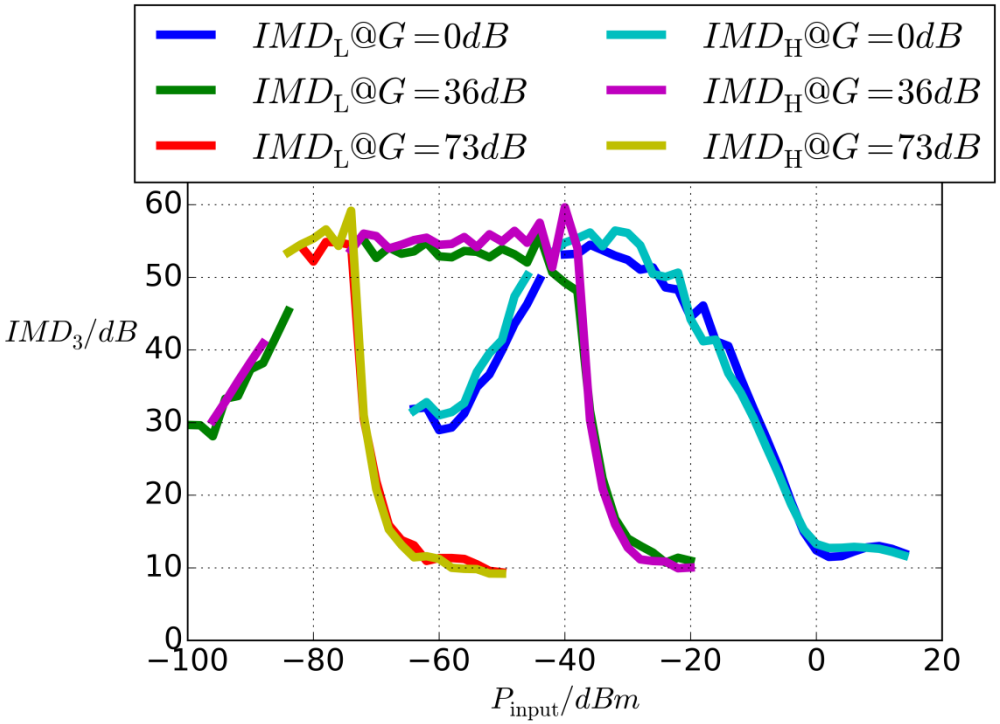
- Bandwidth 8 MHz
- IQ-Signal at 2 GHz +- 4 MHz
- f_sig1 = 2.002 GHz
- f_sig2 = 1.998 GHz
- Sampling 10 MHz
- 80 kSa



# Measurement Results
## Intermodulation Test

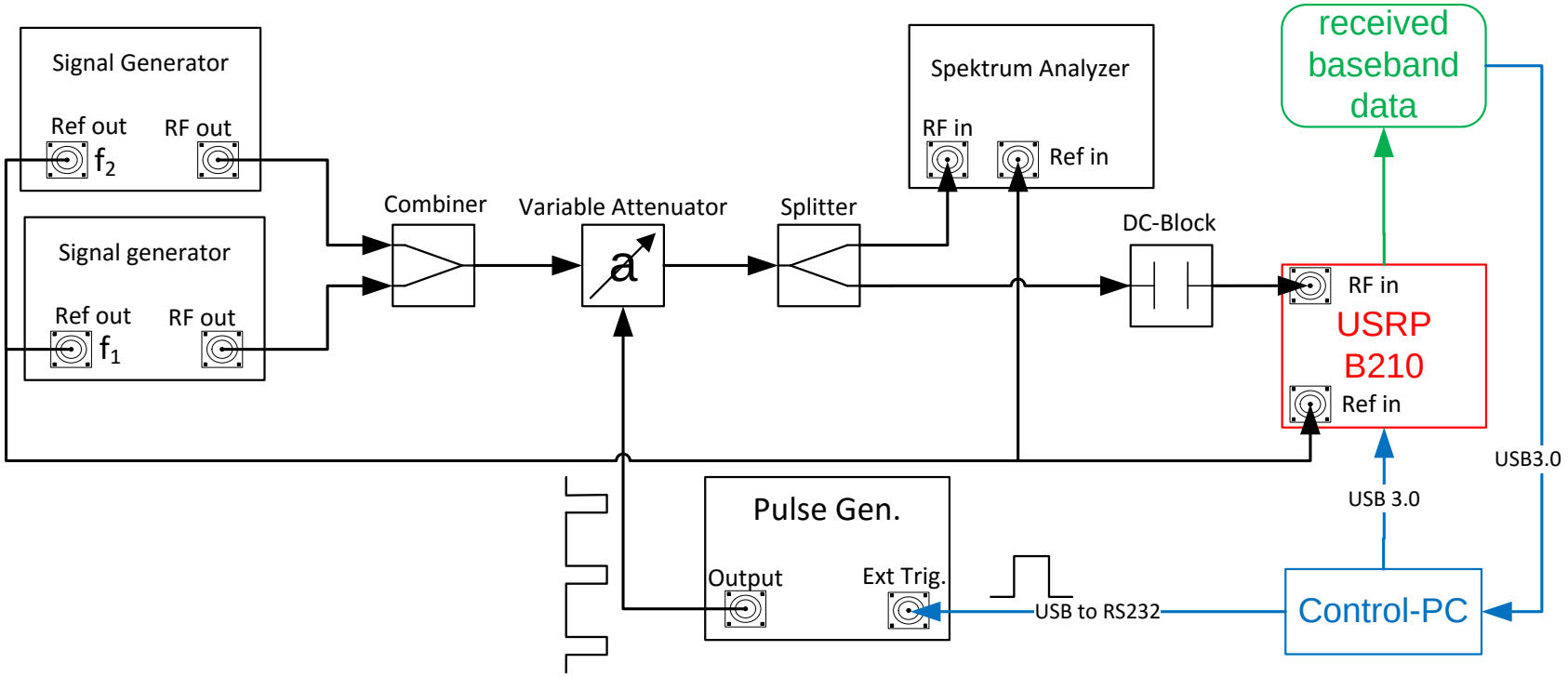
- Bandwidth 8 MHz
- Signal at 2 GHz +- 4 MHz

- Still missing values
- Gain depending dynamic range



# Measurement Results
## Receiver Hop-Time

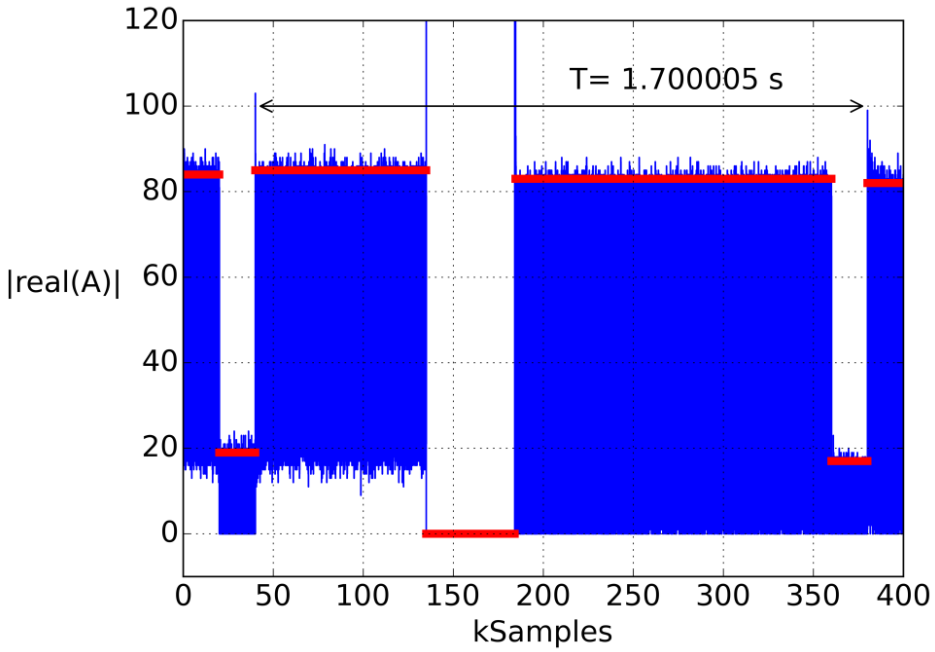
- Rx Hop-Time Set-Up



# Measurement Results
## Receiver Hop-Time

- Receive Frequency change
- Pulse train synchronization

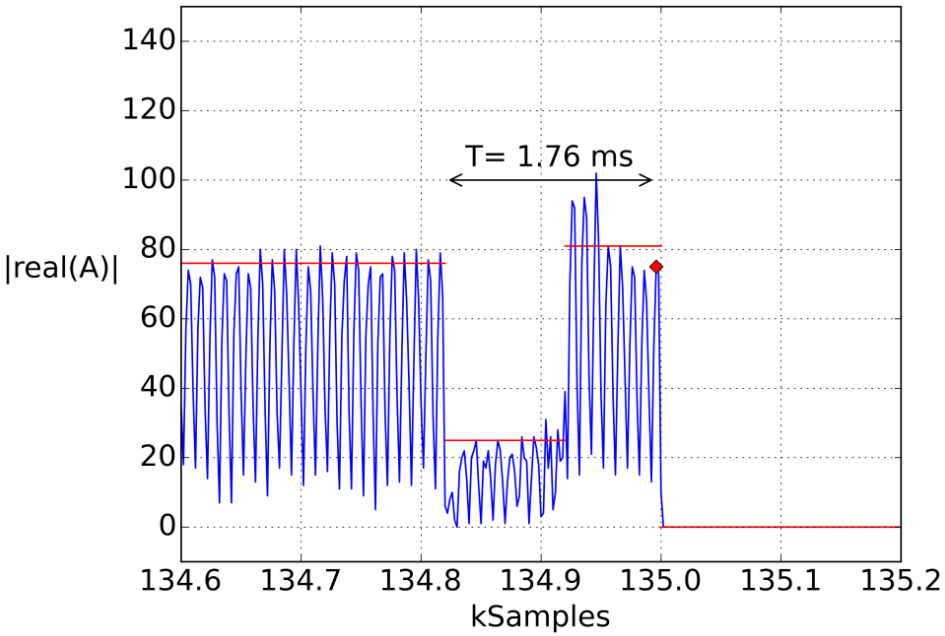
- Hard switching
- Overshooting
- Signal is not decreasing



# Measurement Results
## Receiver Hop-Time

- Command time
- Triggered pulse
- Up to 1.85 ms

- Overshoot minimizing
- Hard switching
- Processing Delay in PC and Serial connection



# Observations and Drawbacks

- Limited external support using the C++ and Python libraries
  - Problem with directly transmitting the IQ-Data
  - Workaround using FTP
- USRP has control limitations
  - Limits in the default FPGA Software
  - RF Chip AD9361 only partly configurable
- USRP suffers from DC offset
  - DC must be avoided for measurements as a signal here will disappear

# Used Equipment and Components

## Two-Tone and Rx Hop-Time Measurement

- Signal generator
  Keysight MXG 5182B
- Spectrum Analyzer
  Rhode & Schwarz FSP 7
- Splitter
  Mini-Circuits ZFSC-2-10G+
- Signal Generator
  Marconi Signalgenerator 2024
- Arbitrary waveform generator
  Keysight AWG 33250A
- Combiner/Splitter
  Mini-Circuits ZNP2D2-50
  Mini-Circuits ZX10-2-252-S+
- Variable Attenuator
  Mini-Circits ZX73-2500-S+
- DC-Block
  Suhner 1100.19A

# Summary and Outlook

**Summary**

- basis provided for an updated version of a test procedures standard
- procedures partly implemented for verification
- simple and proven method to synchronize different time-domain signals

**Outlook**

- first draft of document available
- ***contributions and comments welcome***
- automation and evaluation scripts (Python) will be made available

- goal: international standard, standardization body open

# References


- International Test Procedures (ITOP) 6-2-242 Analog Communication Transmitter and Receiver Test Procedures, 13.10.1993
- International Test Procedures (ITOP) 6-2-246, Digital Communication Transmitter and Receiver Test Procedures, 12.10.1995
- Test Operation Procedures - Software Defined Radio Platforms, IIS *DRAFT Wireless Innovation Forum Website*


- Ettus USRP B210: http://www.ettus.com/product/details/UB210-KIT
- Python: http://www.python.org/

- whom to contact:
  - Christopher Laske, Fraunhofer IIS: Christopher.Laske@iis.fraunhofer.de
  - Johannes Holzer, WTD81: JohannesHolzer@bundeswehr.org