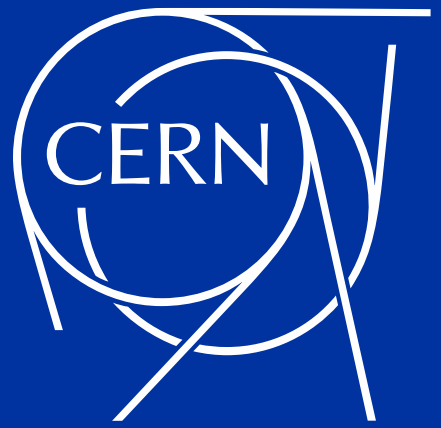


Algorithm to control the Dual Channel FMC High Voltage Supply EDA-04456

J. Emery, M. Saccani, W. Vigano’
08.06.2022

Algorithm to control the Dual Channel FMC High Voltage Supply EDA-04456

System description

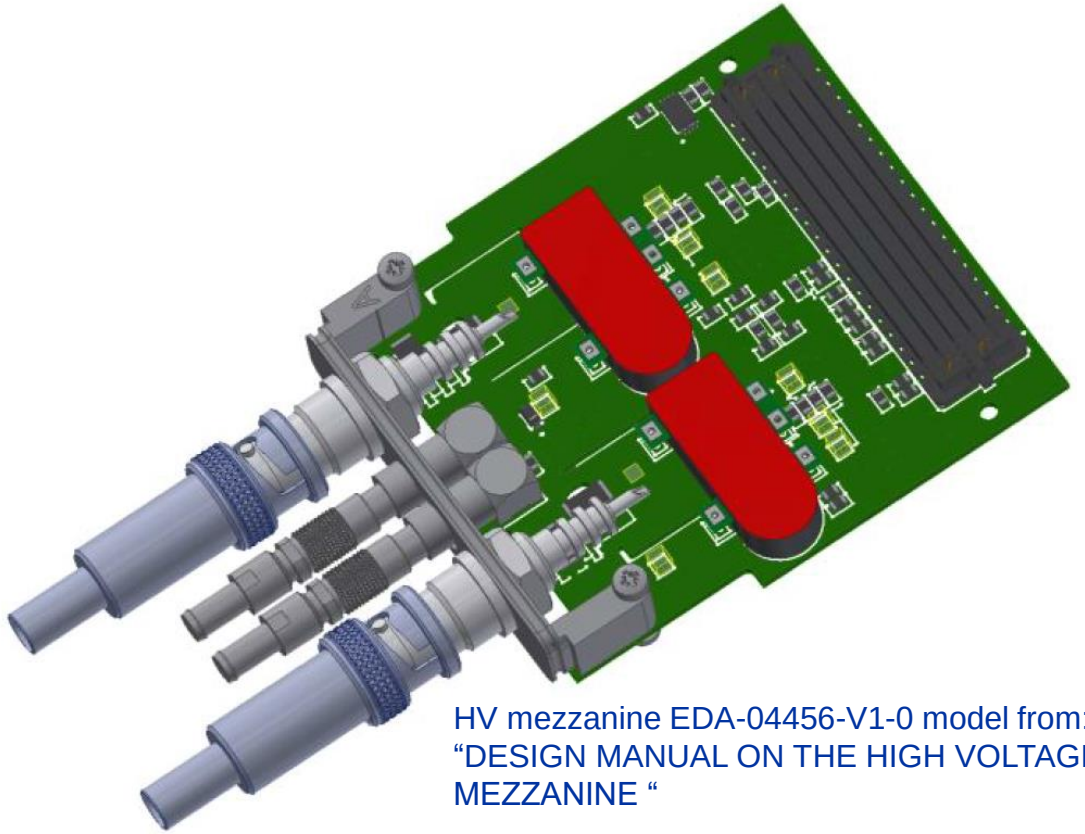
Problem definition

Algorithm specification

Modeling & simulation

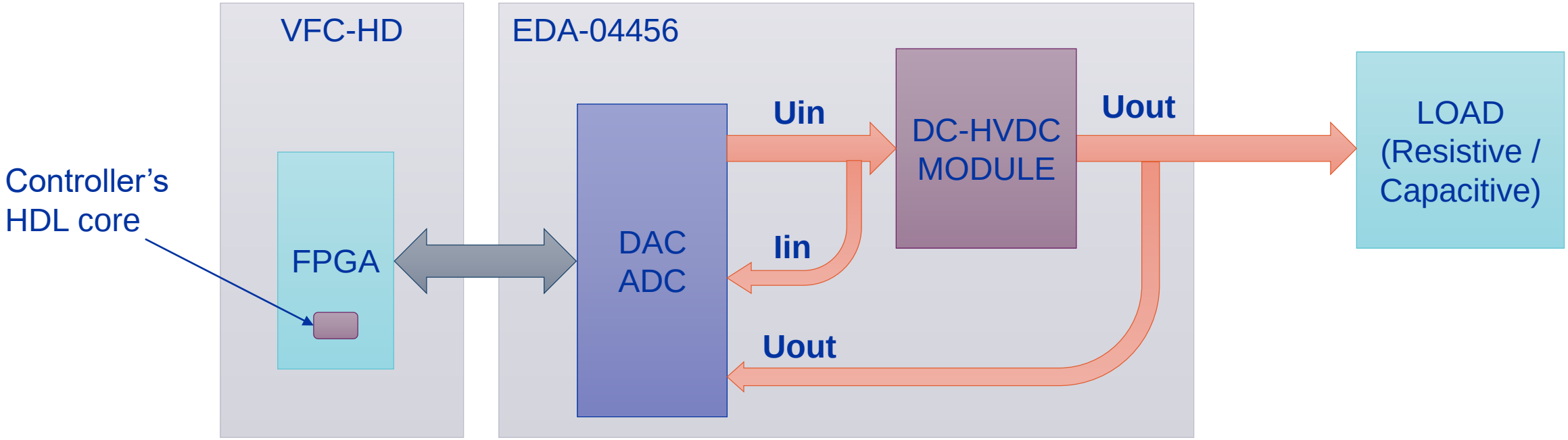
HDL implementation

Verification strategy

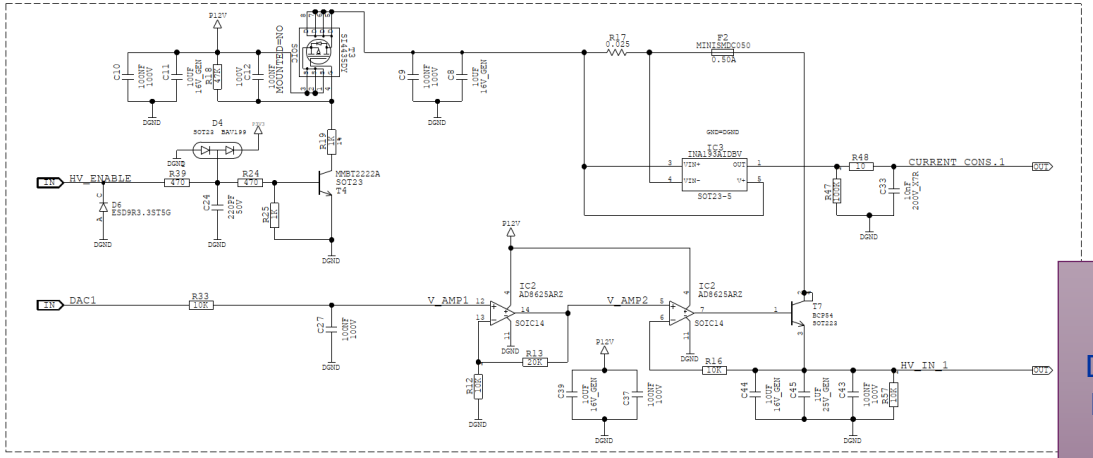


HV mezzanine EDA-04456-V1-0 model from:
“DESIGN MANUAL ON THE HIGH VOLTAGE FMC MEZZANINE “

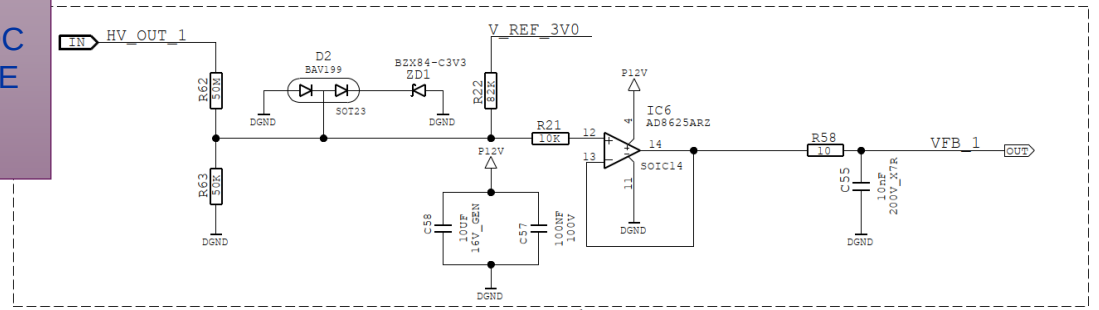
System architecture and controller’s location



Electronics interfaces from the controller's viewpoint



DC-HVDC MODULE



HV VOLTAGE FEEDBACK CIRCUIT 1

Controller input

3) **CURRENT CONS.1**: Shunt voltage to sense current Iin

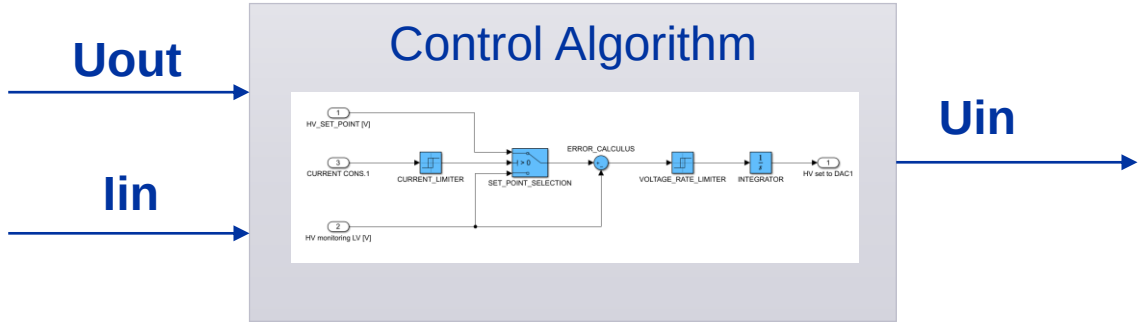
4) **VFB_1**: Voltage output measure Uout

Controller outputs:

1) HV ENABLE: ON/OFF switch to the bipolar
=> PWM modulation to set collector voltage of T7

2) **DAC1** : Analog voltage setting the emitter voltage Uin

During HW prototype validation (without feedback),
W. Vigano' found that the PWM can be set to a fixed duty cycle
=> PWM control out of the controller

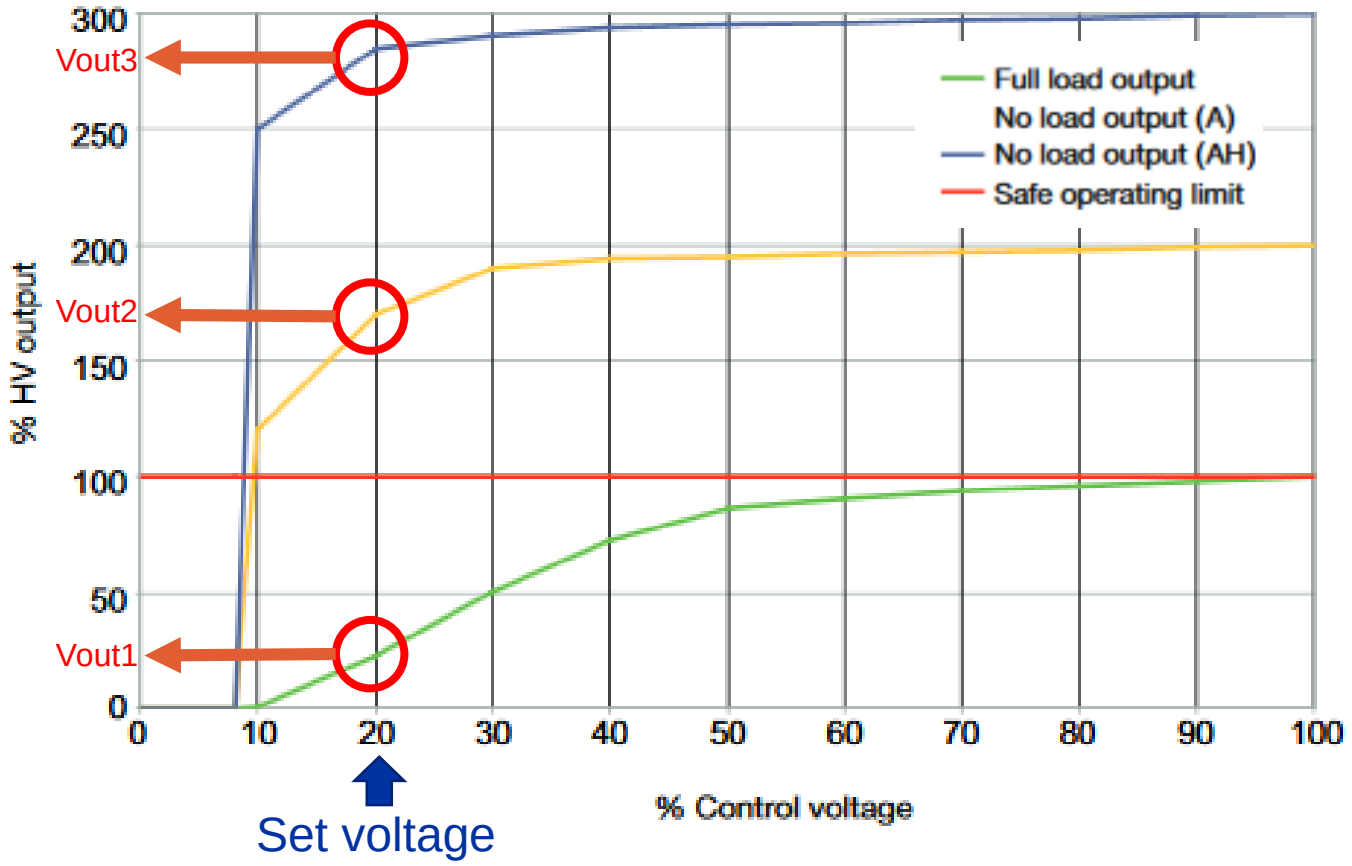


Problem definition: DC-HVDC MODULE

1. HV output vs ctrl voltage
It is load dependant. i.e., output current dependant
=> requires a voltage control based on feedback

2. Safe operating limit easily reachable
When operating with low impedance/capacitive load.
=> requires a voltage limit for a given module

3. Output current limit
=> requires a monitoring of the current
=> if max current is reached, voltage should be limited while delivering the max current (e.g., capacitive load)

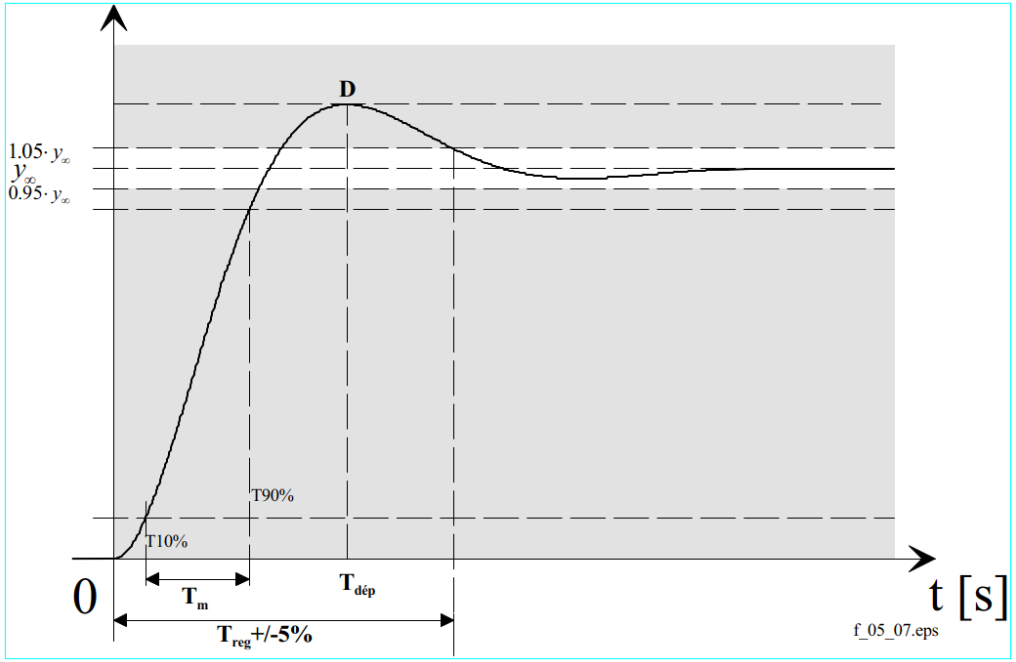


Model Number	Output Voltage	Output Current	Input Voltage	Input Current, No Load	Input Current, Full Load	Ripple
			1 Watt AG Models			
AG20P-5	0 to +2000V	0.5mA	5V	<300mA	<500mA	<0.3%
AG30N-12	0 to -3000V	0.33mA	12V	<100mA	<185mA	<0.3%
AG30N-5	0 to -3000V	0.33mA	5V	<300mA	<500mA	<0.3%
AG30N-5T	0 to -3000V	0.33mA	5V	<300mA	<500mA	<0.3%
AG30P-12	0 to +3000V	0.33mA	12V	<100mA	<185mA	<0.3%

Algorithm spec

- **Voltage feedback (HV module and load independent)**
- **Voltage limiter**
- **Current limiter**
- **Voltage rate selection**
- **Feedback loop period**
- **Coefficients editable**

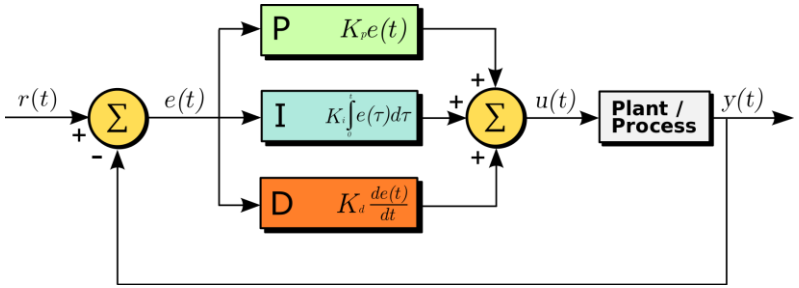
Settling time :
Treg should between 1s and 10s
Ripple :
as low as possible to limit induced current in detectors
=> correction (the FB) could be turned off when HV ok!
Loop period :
< 100ms (determined by HigRes ADC conv. time)



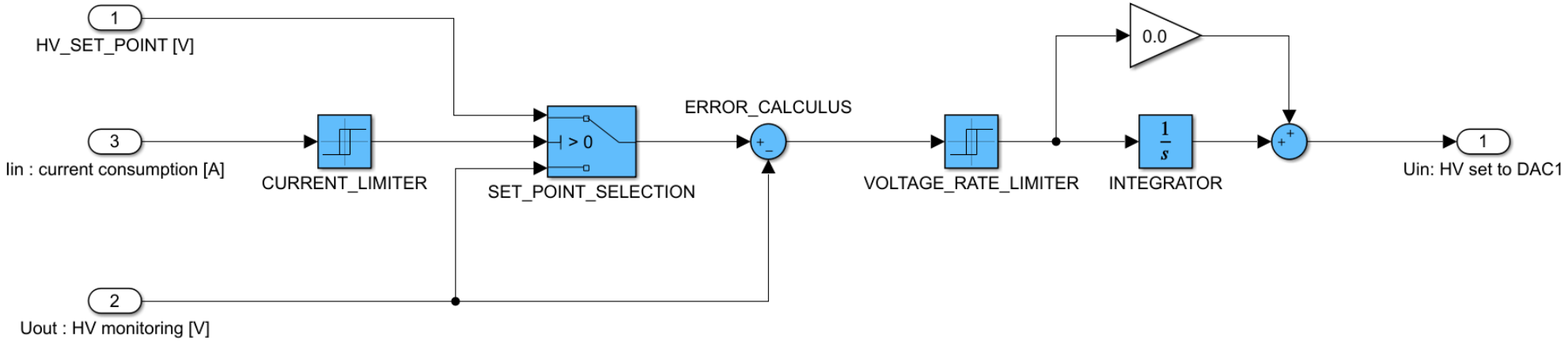
Courtesy of Prof. M. Etique, HEIG-VD, 2008

Control algorithm basic concept

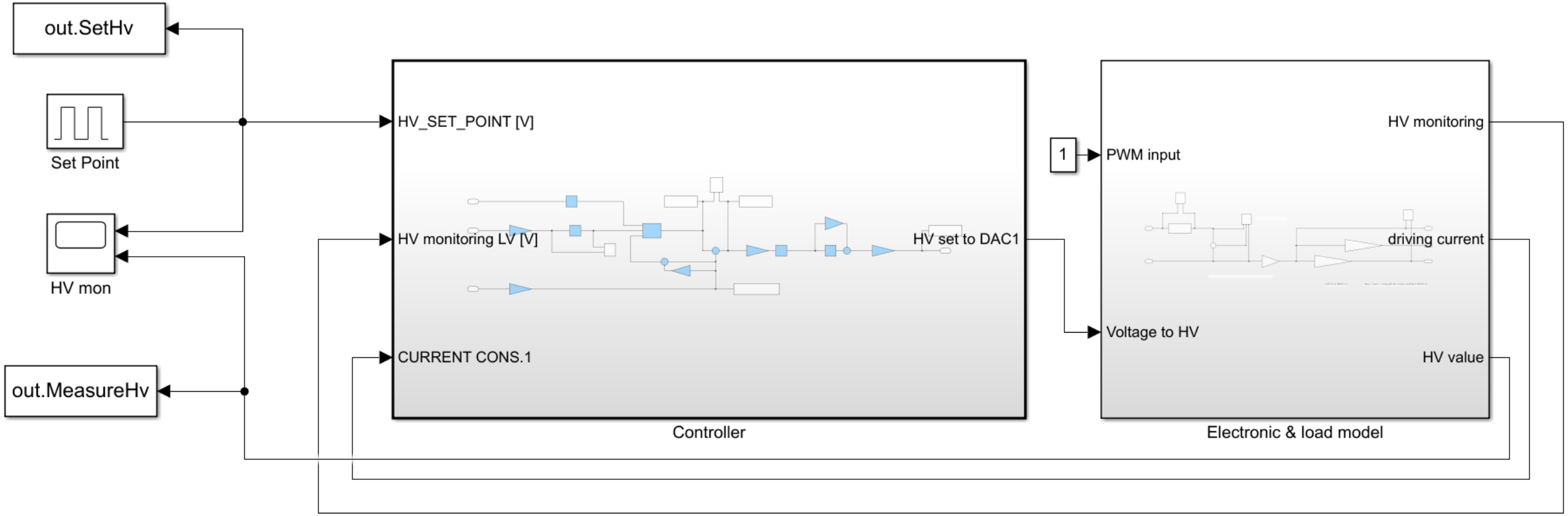
- A digital integrator accumulate error to set the HV module voltage input (limited steady state error)
- A proportional term can be set to improve performance if required (! stability vs performance !)
- The error limiter defines the maximum voltage rate
- Current limiter switch the voltage set point to current voltage (minus a defined portion)



PID controller structure (Wikipedia)

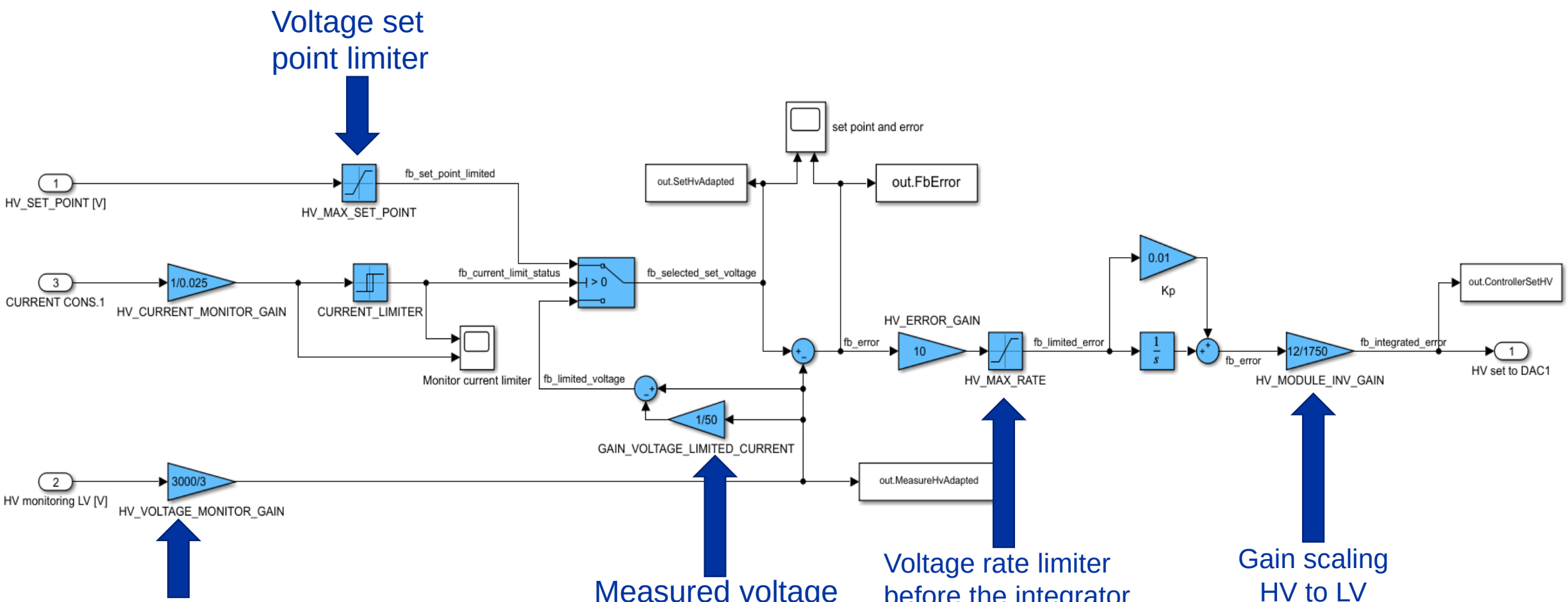


Algorithm modeling & simulation



MatLab/Simulink modeling of the controller and the load

Controller modeling



Control algorithm: simulation examples

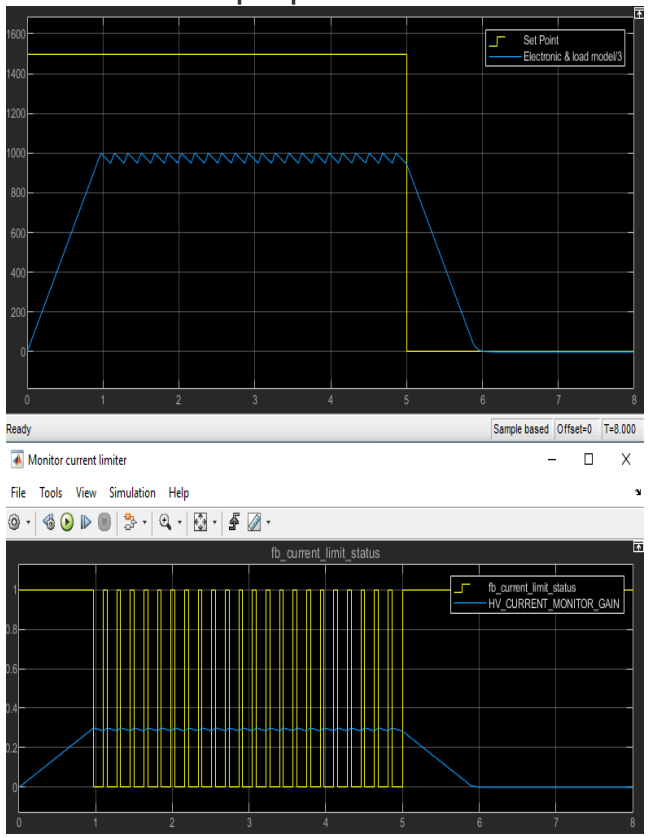
Set point 1500V
HV error rate limit : 1000V/s
load below max current

Set point: 1500V
HV error rate limit: 500V/s
load is above limit
current limits: thresholds (with hysteresis) 0.3V and 0.28V

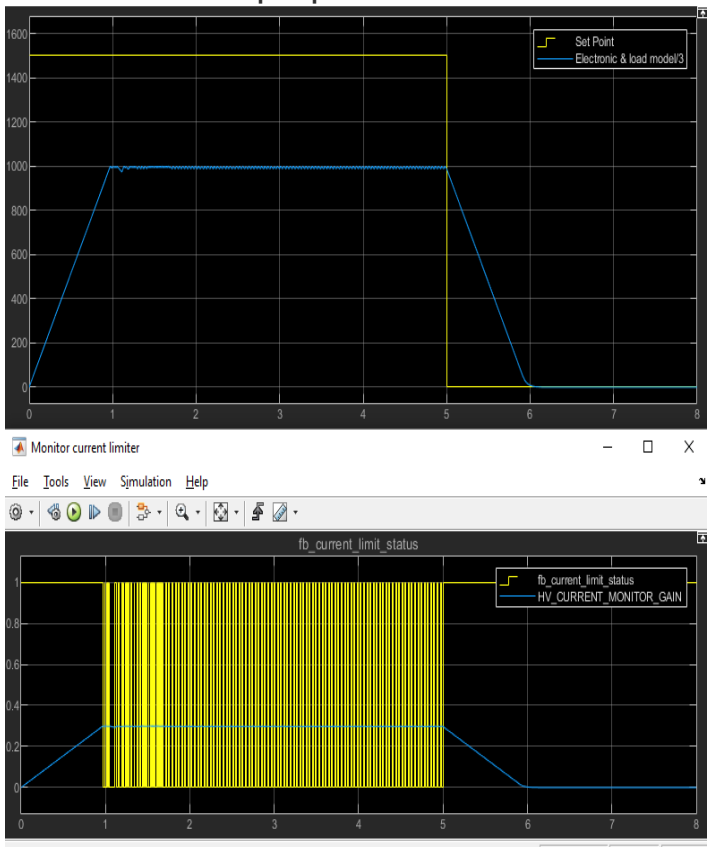
Vout [V]

Iin [A]

Without proportional term



With proportional term



VHDL implementation

Follows the Simulink modeling

Fixed point arithmetic set to 32 bits (16, 16) via IEEE library ‘ieee_proposed’ (for Quartus synthesis) but can be increased via generic param. Resolution internal calculation: 15.3 uV / uA

Components reuse from BWS project
- input scaling to sfixed
- proportional Integral (PI) core
- comparator for current monitoring

Internal feedback coefficients in registers (settable on the fly)

Feedback loop update externally triggered (user can vary the feedback period if needed)

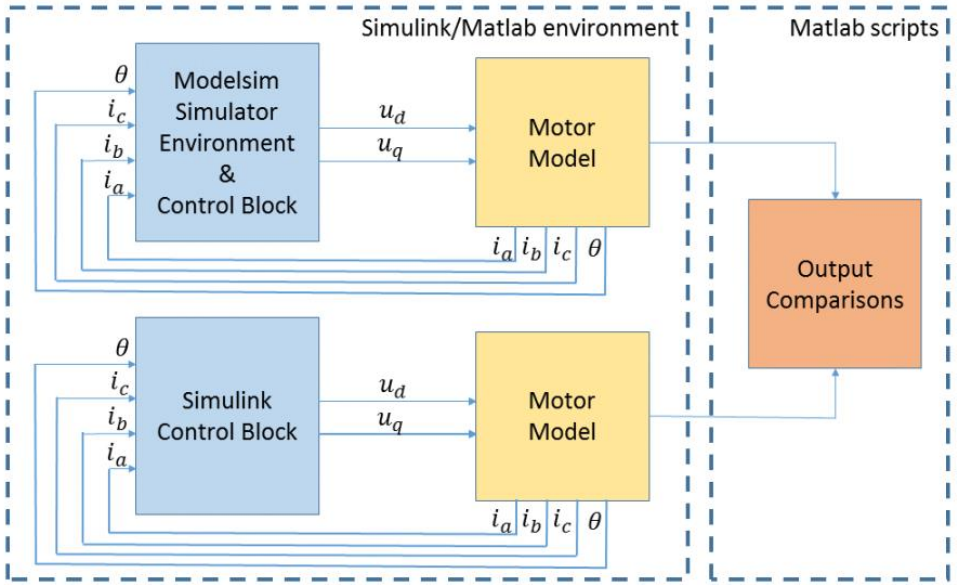
Internal states outputted to monitor via registers or connected to memory (monitoring of fast signals)

```
entity vfc_hd_fmc_hv_feedback_algo_top is
    generic(
        TOP_FIXPT_SIZE_HIGH                                 : integer   :=  16;
        TOP_FIXPT_SIZE_LOW                                  : integer   := -16;
        SETTING_HV_VOLTAGE_MONITOR_GAIN                     : real      := 4.0*1750.0;      -- 0.000
        SETTING_HV_CURRENT_MONITOR_GAIN                     : real      := 5.0;             -- curre
        FEEDBACK_LOOP_PERIOD_IN_S                           : real      := 0.1              -- scali
    );
    port(
        -- Clock & Reset --
        clk_i                                               : in std_logic;
        rst_i                                               : in std_logic;
        -- Inputs --
        fb_loop_enabled_i                                   : in    std_logic;  -- when 0, set all o
        fb_loop_calculate_and_update_output_i               : in    std_logic;  -- pulse (1 clk_i le

        -- settings --
        setting_hv_set_point_i                              : in std_logic_vector(31 downto 0); -- s
        setting_hv_max_set_point_i                          : in std_logic_vector(31 downto 0); -- s
        setting_fb_loop_hv_error_gain_i                     : in std_logic_vector(31 downto 0); -- s
        setting_fb_loop_hv_max_rate_i                       : in std_logic_vector(31 downto 0); -- s
        setting_fb_loop_hv_module_inv_gain_i                : in std_logic_vector(31 downto 0); -- s
        setting_fb_loop_hv_current_limit_hyst_high_i        : in std_logic_vector(31 downto 0); -- s
        setting_fb_loop_hv_current_limit_hyst_low_i         : in std_logic_vector(31 downto 0); -- s
        setting_fb_loop_hv_gain_to_hv_monitor_when_in_cur_limt_i : in std_logic_vector(31 downto 0); -- s

        -- signals from intenal calculations --
        internal_state_hv_voltage_set_point                 : out sensor_processed_type;
        internal_state_hv_voltage_in_volts                  : out sensor_processed_type;
        internal_state_hv_current_in_amps                   : out sensor_processed_type;
        internal_state_fb_limited_voltage                   : out sensor_processed_type;
        internal_state_fb_limited_error                     : out sensor_processed_type;
        internal_state_fb_set_point_limited                 : out sensor_processed_type;
        internal_state_fb_selected_set_voltage              : out sensor_processed_type;
        internal_state_fb_error                             : out sensor_processed_type;
        internal_state_fb_integrated_error                  : out sensor_processed_type;
        internal_state_fb_current_limiter_status            : out std_logic;

        -- analog world inputs --
        hv_voltage_monitor_data_i                           : in    std_logic_vector(23 downto 0);
        hv_voltage_monitor_ready_i                          : in    std_logic;
        hv_current_monitor_data_i                           : in    std_logic_vector(11 downto 0);
        hv_current_monitor_ready_i                          : in    std_logic;
        -- analog world outputs --
        hv_enable_pwm_ctrl_o                                : out   std_logic;
        hv_lv_set_point_data_o                              : out   std_logic_vector(15 downto 0);
        hv_lv_set_point_ready_o                             : out   std_logic
    );
end vfc_hd_fmc_hv_feedback_algo_top;
```

Verification strategy

1. Individual cores have simple unit tests (but could be improved for automation)
2. Overall functional test bench in place to visualize use case
3. Next steps:

- setup co-simulation within Simulink as for BWS feedbacks (2017)
- integrate code to the Board Support Package (M. Saccani) and test in the lab
- tune feedback coefficients with various load and define a range of '<u>robust</u>' values



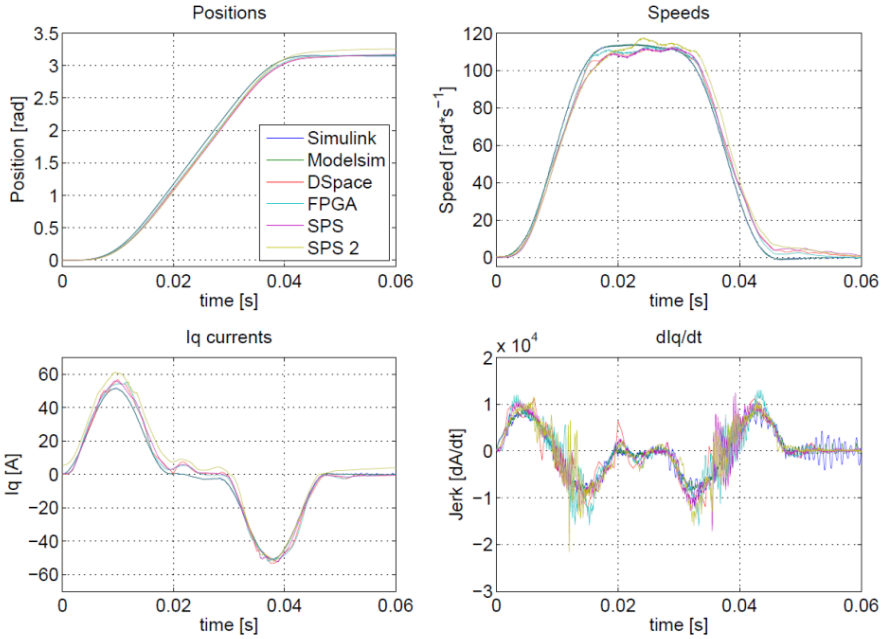
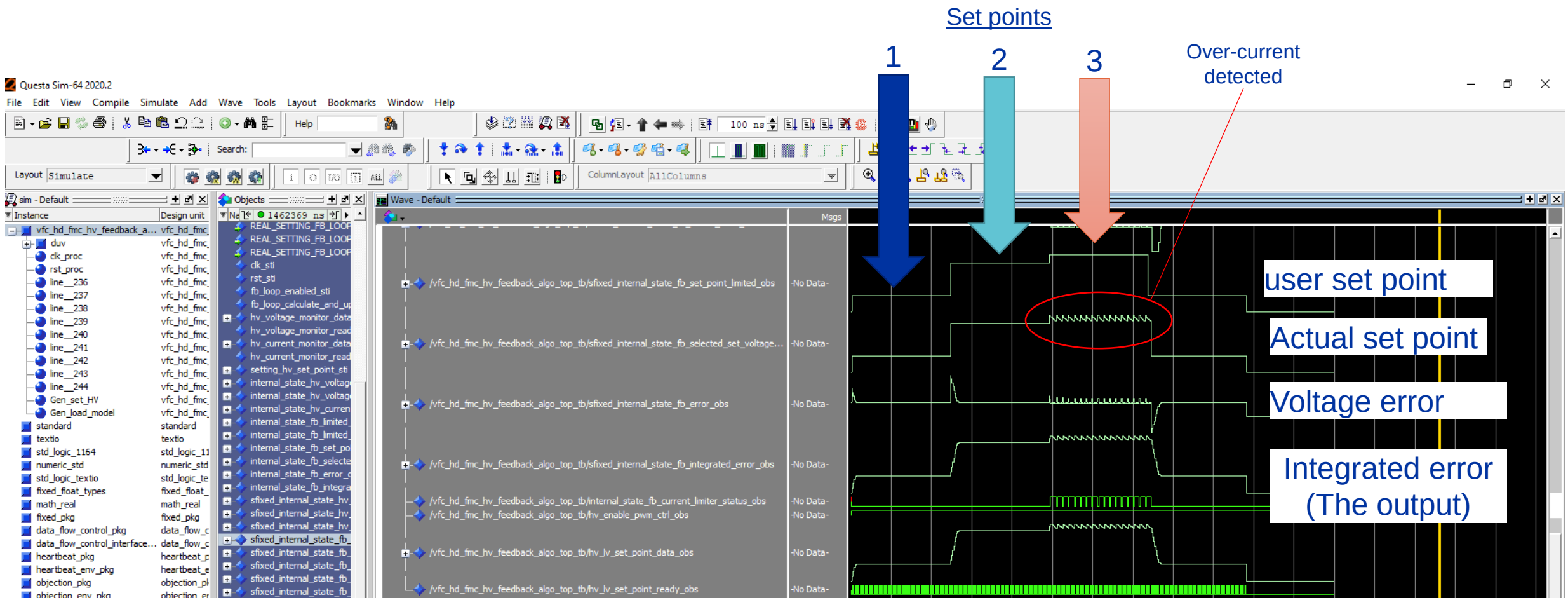


Fig. 16: Motion patterns at different design stages

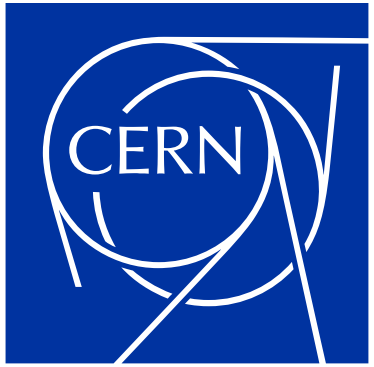
<u>Reference:</u>
"Design and Validation Methodology of the Control System for a Particle Beam Size Measurement Instrument at the CERN Laboratory", American Control Conference (2017)

‘Functional’ Simulation: visual example



Summary

- A digital controller for the HV mezzanine has been designed
- Voltage feedback mode with current limiter
- MatLab/Simulink modeling shows promising behavior
- HDL implementation with fixed point arithmetic
- User will have access to internal coefficients
- Feedback loop period is user actionable, allowing to start-stop
- **Next steps:**
  - Co-simulation MatLab/Simulink
  - Offset suppression on the measured inputs
  - Integration algorithm into the Board Support Package of M. Saccani
  - Test and optimise feedback on the physical prototype
  - Tune feedback coefficients with various load and define a range of ‘robust’ values
  - Validate and report performance
  - investigate into corrector action at settled voltage (reduce controller induced ripple)



home.cern

Functional simulation Generics

```
entity vfc_hd_fmc_hv_feedback_algo_top_tb is
    generic(
        -- SIMULATION RELATED GENERICS ------------------------------------------------------------------------------
        OUTFOLDER                                       : STRING    := "";
        FILE_PREFIX                                     : STRING    := "";

        NBR_CLK_PER_FB_LOOP                             : natural   := 100;
        NBR_CLK_PER_LOAD_UPDATE                         : natural   := 20;

        DCDC_MODULE_TIME_CST_IN_LOAD_UPDATE             : real      := 0.1;             -- 10 loop period to reach the value
        DCDC_MODULE_GAIN_WITH_DRIVE_CIRCUIT             : real      := 7.5/2.5*2000.0/7.5;  --
        DCDC_MODULE_LOAD_RESISTIVE                      : real      := 1500.0*22.0;     -- 22mA at 1500.0V
        HV_MON_VOLTAGE_LSB_IN_V                         : real      := 0.000310440;
        HV_MON_CURRENT_LSB_IN_I                         : real      := 0.0000763;

        -- HV => 310440 #nV/lsb => value are in 2's complement, but seems the ADC is driven only with positive voltage with an offset.
        -- current   offset=2000 #uA   conv_fact=7630 #uA/lsb

        -- ALGO CORE RELATED GENERICS -------------------------------------------------------------------------------
        TOP_FIXPT_SIZE_HIGH                             : integer   :=  16;
        TOP_FIXPT_SIZE_LOW                              : integer   := -16;

        FEEDBACK_LOOP_PERIOD_IN_S                       : real      := 0.1;
        SETTING_HV_VOLTAGE_MONITOR_GAIN                 : real      := 3.0*1750.0;
        SETTING_HV_CURRENT_MONITOR_GAIN                 : real      := 0.3;

        REAL_SETTING_HV_MAX_SET_POINT                   : real      := 1750.0;
        REAL_SETTING_FB_LOOP_HV_ERROR_GAIN              : real      := 10.0;
        REAL_SETTING_FB_LOOP_HV_MAX_RATE                : real      := 2000.0;
        REAL_SETTING_FB_LOOP_HV_MODULE_INV_GAIN         : real      := 1.0/3000.0;
        REAL_SETTING_FB_LOOP_HV_CURRENT_LIMIT_HYST_HIGH : real      := 0.050;
        REAL_SETTING_FB_LOOP_HV_CURRENT_LIMIT_HYST_LOW  : real      := 0.048;
        REAL_SETTING_FB_LOOP_HV_GAIN_WHEN_IN_CURR_LIMIT : real      := 1.0/50.0 -- remove 2 % of the HV to give a new setpoint!
    );
end vfc_hd_fmc_hv_feedback_algo_top_tb;
```