

LLRF Library.

Martin Killenberg

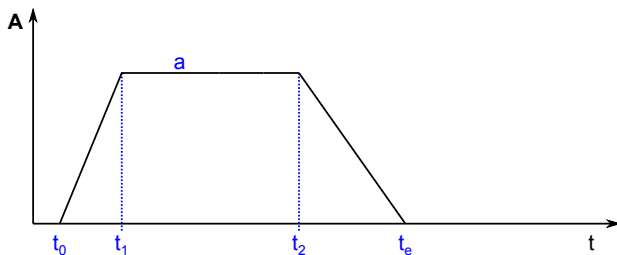


11th June 2015

MSK Collaboration Workshow 2015, Warsaw, Poland

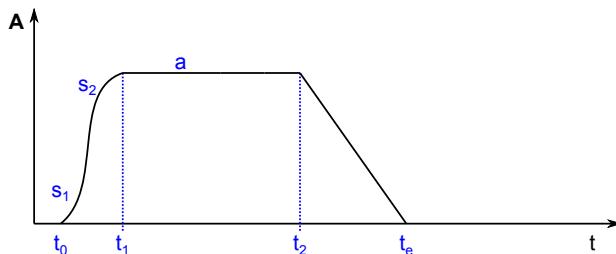
What should be improved?
Which features do we need?

- CW operation
- Multiple colliders, one library
- Improve usability (C++ interface)
- Improve maintainability
- Improve performance



Starting point

- Simple shape
- 5 parameters

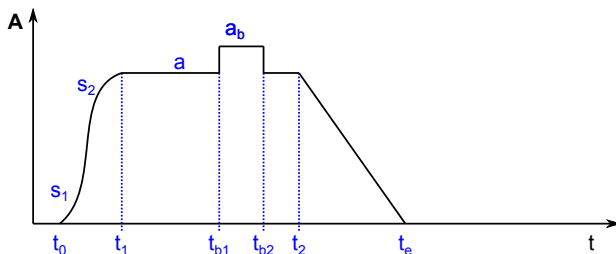


Starting point

- Simple shape
- 5 parameters

Smooth rising edge

- 2 new parameters
- + 1 loop
- + 1 if-statement



Starting point

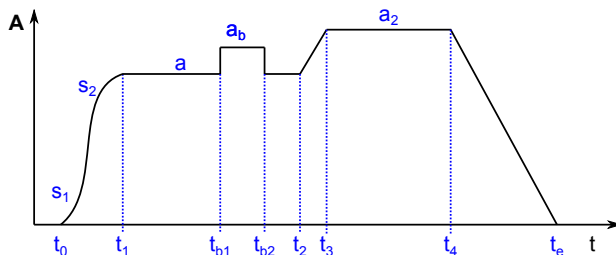
- Simple shape
- 5 parameters

The bump

- 3 new parameters
- + 1 loop
- + 1 if-statement

Smooth rising edge

- 2 new parameters
- + 1 loop
- + 1 if-statement



Starting point

- Simple shape
- 5 parameters

Smooth rising edge

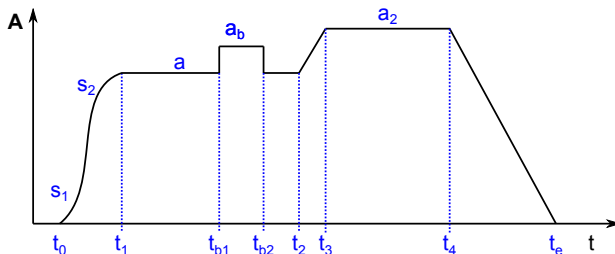
- 2 new parameters
- + 1 loop
- + 1 if-statement

The bump

- 3 new parameters
- + 1 loop
- + 1 if-statement

Flash 2: second flat top

- 3 new parameters
- + 1 loop
- + 1 if-statement



Starting point

- Simple shape
- 5 parameters

Smooth rising edge

- 2 new parameters
- + 1 loop
- + 1 if-statement

The bump

- 3 new parameters
- + 1 loop
- + 1 if-statement

Flash 2: second flat top

- 3 new parameters
- + 1 loop
- + 1 if-statement

CW operation

- ?? new parameters
- + ?? loops
- + ?? if-statements

XFEL

- ?? new parameters
- + ?? loops
- + ?? if-statements

This solution does not scale!

Sequence of different types

- Linear slope
- Flat top
- Smooth slope
- ...

Advantages

- Configurable number of sections
- Easy to add new types
- Performance optimisation
(Flat top: do not recalculate, you know that it's always the same value)
- ...

Disadvantages

- Library is a little bit more abstract
- Plugin mechanism for expandability

- Does it work with this scheme?
- One library for pulsed and CW?
- Normal operation = 1 short flat top?
- Ramp up of CW machine