Impulse

When calculating impulse, the first step is to break the signal into two signals which only contain positive or negative values.

Note that this script makes use of the Current_Signal function within Evaluate Expression.

```
! Set all frames for all three components to DATA NOT FOUND for any frame
!   ABOVE zero

Evaluate_Expression
/EXPRESSION= ( CURRENT_SIGNAL < 0 ) / ( CURRENT_SIGNAL < 0 ) *
CURRENT_SIGNAL
/SIGNAL_TYPES=LINK_MODEL_BASED
/SIGNAL_FOLDER=ORIGINAL
/SIGNAL_NAMES=RGRF+LGRF
! /RESULT_TYPES=DERIVED
/RESULT_FOLDERS=IMPULSE
/RESULT_NAME=_Neg
/APPLY_AS_SUFFIX_TO_SIGNAL_NAME=TRUE
;

! Set all frames for all three components to DATA NOT FOUND for any frame
!   BELOW zero

Evaluate_Expression
/EXPRESSION=( CURRENT_SIGNAL > 0 ) / ( CURRENT_SIGNAL > 0 ) * CURRENT_SIGNAL
/SIGNAL_TYPES=LINK_MODEL_BASED
/SIGNAL_FOLDER=ORIGINAL
/SIGNAL_NAMES=RGRF+LGRF
! /RESULT_TYPES=DERIVED
/RESULT_FOLDERS=IMPULSE
/RESULT_NAME=_Pos
/APPLY_AS_SUFFIX_TO_SIGNAL_NAME=TRUE
;

! Integrate over stance

Metric_Integrate
/RESULT_METRIC_FOLDER=IMPULSE
/RESULT_METRIC_NAME=
/APPLY_AS_SUFFIX_TO_SIGNAL_NAME=TRUE
/SIGNAL_TYPES=DERIVED+DERIVED
/SIGNAL_FOLDER=IMPULSE+IMPULSE
/SIGNAL_NAMES=RGRF_Neg+RGRF_Pos
/COMPONENT_SEQUENCE=ALL
/EVENT_SEQUENCE=RON+ROFF
```

```
/EXCLUDE_EVENTS=
/SEQUENCE_PERCENT_START=
/SEQUENCE_PERCENT_END=
! /GENERATE_MEAN_AND_STDDEV=TRUE
! /APPEND_TO_EXISTING_VALUES=FALSE
;

Metric_Integrate
/RESULT_METRIC_FOLDER=IMPULSE
/RESULT_METRIC_NAME=
/APPLY_AS_SUFFIX_TO_SIGNAL_NAME=TRUE
/SIGNAL_TYPES=DERIVED+DERIVED
/SIGNAL_FOLDER=IMPULSE+IMPULSE
/SIGNAL_NAMES=LGRF_Neg+LGRF_Pos
/COMPONENT_SEQUENCE=ALL
/EVENT_SEQUENCE=LON+LOFF
/EXCLUDE_EVENTS=
/SEQUENCE_PERCENT_START=
/SEQUENCE_PERCENT_END=
! /GENERATE_MEAN_AND_STDDEV=TRUE
! /APPEND_TO_EXISTING_VALUES=FALSE
;
```