# phasecalc

January 12, 2017

**Abstract**

Given an ephemeris, calculate a phase column for an event table.

## 1 Instruments/Modes

| Instrument | Mode |
|---|---|
| all | all modes with time information |

## 2 Use

| | |
|---|---|
| pipeline processing | yes (?) |
| interactive analysis | yes |

## 3 Description

Given an ephemeris and a list of event tables, **phasecalc** calculates the phase of each event in each of the tables. The calculated values are stored in a column named PHASE. If the column exists it is overwritten.

**phasecalc** relies on the event table having a TIME column and the two attributes MJDREF and TIMEZERO.

- MJDREF is the Modified Julian Day from which times are calculated.
- TIMEZERO is the clock correction at the start of the observation in seconds.
- **phasecalc** also uses the TIMESYS keyword to determine the Time System (UTC, TT, TAI) used in event table. If the TIMESYS keyword is not UTC, the Reference Epoch time of the ephemeris is converted to the appropriate Time System. In the absence of this keyword TT (XMM Time) is assumed.

The time of event $n$ is then:

$$t(n) = \mathtt{MJDREF} * 86400 + \mathtt{TIMEZERO} + \mathtt{TIME}[n] \quad (1)$$

where `TIME`[$n$] is the value stored in the `TIME` column.

An ephemeris is specified in terms of four quantities:

- reference epoch $t_0$; This is UTC.
- reference phase $\phi_0$;
- frequency $\nu$ in Hz;
- frequency time derivative $\dot{\nu}$ in Hz/s.

The phase of event $n$ is defined as:

$$\phi(n) = \phi_0 + \{t(n) - t_0\} \times \nu + \frac{1}{2}\{t(n) - t_0\}^2 \times \dot{\nu}\,. \tag{2}$$

### 3.1 Examples

-
  ```
  phasecalc --tables=events.ds:EVENTS --frequency=29.8309503578 --frequencydot=-374213.62e-15
  --epoch=2001-04-01T00:00:00
  ```

  Calculate the phase of each event in `events.ds:EVENTS` for the given ephemeris.

## 4 Parameters

This section documents the parameters recognized by this task (if any).

| **Parameter** | Mand | Type | Default | Constraints |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| **tables** | yes | T | set.ds:EVENTS | |

List of tables to operate on.

| | | | | |
|---|---|---|---|---|
| **frequency** | no | r | | |

Frequency [Hz].

| | | | | |
|---|---|---|---|---|
| **frequencydot** | no | r | | |

Frequency dot [Hz/s].

| | | | | |
|---|---|---|---|---|
| **epoch** | no | w | | |

Epoch of the phase in UTC

| | | | | |
|---|---|---|---|---|
| **phase** | no | r | | |

Phase at the epoch.

## 5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

**TimezeroDoesNotExist** *(error)*
The table does not contain the attribute TIMEZERO. The absoulte time of the events cannot be calculated.

**MjdrefDoesNotExist** *(error)*
The table does not contain the attribute MJDREF. The absoulte time of the events cannot be calculated.

**TimeColumnDoesNotExist** *(error)*
The table does not contain the column TIME. The absoulte time of the events cannot be calculated.

**PhaseColumnExists** *(error)*
The PHASE exists, and clobber is not enabled.

**SubProcessError** *(warning)*
Move on to the following table.
*corrective action:* An error was found while processing one of the inout tables. The message indicates what the table and error ware.

## 6 Input Files

1. An event dataset that must contain:
   - a real64 column TIME;
   - the attribute MJDREF;
   - the attribute TIMEZERO.

## 7 Output Files

1. The input data set, modified by the addition of a real64 column PHASE.

## 8 Algorithm

```
read ephemeris
open table
if( cannot calculate absolute event time ) error
call selectlib with table and phase formula
```

# 9 Comments

**phasecalc** is meant to be used in conjunction with **evselect** or **xmmselect** to perform second order filtering.

**phasecalc** is not a general tool, in that it has a very specific knowledge of how to calculate the absoulte time of an event based on the XMM-Newton conventions. On the other hand, this affords a paramter interface focussed on the astronomical data. For a more generic tool to perform arbitrary math on binary tables see **tabcalc** ().

# 10 Future developments

# References