



Navigation and Ancillary Information Facility

Time Conversion and Time Formats

September 2009



Topics

Navigation and Ancillary Information Facility

- **Time Systems and Kernels**
- **Converting Time Strings**
 - Calendar, Day-of-Year (DOY) and Julian Date strings to Ephemeris Time
 - Spacecraft Clock string to Ephemeris Time
 - Spacecraft Clock string to Encoded Spacecraft Clock (aka “Ticks”)
- **Converting Numeric Times - 1**
 - Ephemeris Time to Calendar, DOY and Julian Date strings
- **Use of Format Picture**
- **Converting Numeric Times - 2**
 - Ephemeris Time to Spacecraft Clock String
 - Encoded Spacecraft Clock to Spacecraft Clock String
 - Ephemeris Time to Local Solar Time String
- **Conversions Between Uniform Time Systems**
- **Customizing the Time System**
- **Pictorial Layout of the Time System**



Time Systems and Kernels

Navigation and Ancillary Information Facility

- **Three time systems are commonly used in SPICE for inputs and outputs in SPICE application programs:**
 - Coordinated Universal Time (UTC)
 - Spacecraft Clock (SCLK)
 - Ephemeris Time (ET, also referred to as Barycentric Dynamical Time, TDB)
- **Two time systems are commonly used in SPICE for kernel data look-ups or for computations**
 - Ephemeris Time (ET, TDB)
 - » Used in SPK, frames, PCK
 - » Used in high-level geometry interfaces
 - Encoded Spacecraft Clock (Ticks)
 - » Used in the intermediate level CK interfaces
- **The SPICE Leapseconds Kernel and Spacecraft Clock Kernel capture the relationships between**
 - UTC and ET (Leapseconds Kernel)
 - UTC or ET and SCLK (SCLK Kernel)
 - **Caution: the long-term future relationships between these systems cannot be accurately predicted**



Converting Time Strings

Navigation and Ancillary Information Facility

Examples for FORTRAN, C, IDL and MATLAB are shown on this page only.
Examples on subsequent pages are shown only in FORTRAN.

- **UTC, TDB, or TDT (TT) String to Ephemeris Time**

- STR2ET (*string*, ET)
- str2et_c (*string*, &et)
- cspice_str2et, *string*, et
- et = cspice_str2et (*string*)

Converts any string in a format recognized by SPICE, excepting SCLK

Requires Leapseconds Kernel (LSK)

- **Spacecraft Clock String to Ephemeris Time**

- SCS2E (*scid*, *string*, ET)
- scs2e_c (*scid*, *string*, &et)
- cspice_scs2e, *scid*, *string*, et
- et = cspice_scs2e (*scid*, *string*)

Requires Spacecraft Clock Kernel (SCLK)

Normally requires Leapseconds Kernel (LSK) as well, to handle a very small (~2 msec.) difference between TDB and TT

- **Spacecraft Clock String to Encoded Spacecraft Clock (used in the mid-level interfaces of the C-kernel system)**

- SCENCD (*scid*, *string*, SCLKDP)
- scencd_c (*scid*, *string*, &sclkdp)
- cspice_scencd, *scid*, *string*, sclkdp
- sclkdp = cspice_scencd (*scid*, *string*)

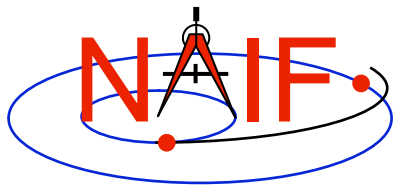
Requires Spacecraft Clock Kernel (SCLK)



Converting Numeric Times - 1

Navigation and Ancillary Information Facility

- **Ephemeris Time to Calendar, DOY or Julian Date String**
 - **TIMOUT** (*et*, *fmtpic*, **STRING**)
 - » The input *fmtpic* is a specification giving the user great flexibility in setting the appearance of the output time string, and the time system used (UTC, TDB, TDT).
 - YYYY Mon DD, HR:MN:SC.### ::UTC
 - 1999 Jan 12, 23:28:28.289
 - YYYY Mon DD, AP:MN:SC.### ampm ::UTC-8 (PST)
 - 1999 Jan 12, 03:28:28.289 p.m. (PST)
 - See the header for the TIMOUT module
 - The module TPICTR may be useful in constructing a format picture specification from a sample string
 - » Requires Leapseconds Kernel
 - **ET2UTC** (*et*, *format*, *prec*, **UTCSTR**)
 - » The *format* input specifies calendar, DOY, or Julian Date format for **UTCSTR**
 - » Requires Leapseconds Kernel
 - » Note: TIMOUT is a more capable routine
 - **ETCAL** (*et*, **STRING**)
 - » **STRING**, fixed format ephemeris calendar time string
 - » No Leapseconds Kernel is required.



Use of Format Picture

Navigation and Ancillary Information Facility

Example Time Strings and the Corresponding Format Pictures

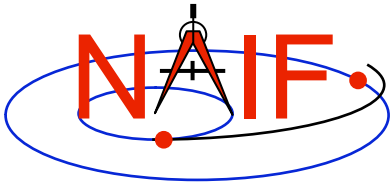
Time String	Format Picture Used (<i>fmtpic</i>)
1999-283T12:28:29 (UTC)	YYYY-DOYTHR:MN:SC (UTC)
1999-283T12:29:33 (TDB)	YYYY-DOYTHR:MN:SC (TDB) ::TDB
Wed Nov 03, 19:29:05 1999	Wkd Mon DD, HR:MN:SC YYYY
465 B.C. Jan 12 03:15:23 p.m.	YYYY ERA Mon DD AP:MN:SC ampm
1987-11-03T04:29:58	YYYY-MM-DDTHR:MN:SC
04:28:55 A.M. June 12, 1982	AP:MN:SC AMPM Month DD, YYYY
Thursday November 04, 1999	Weekday Month DD, YYYY
DEC 31, 15:59:60.12 1998 (PST)	MON DD, HR:MN:SC YYYY (PST)::UTC-8
2450297.1994 JDUTC	JULIAND.#### JDUTC



Converting Numeric Times - 2

Navigation and Ancillary Information Facility

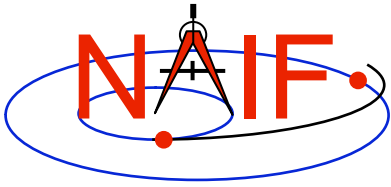
- **Ephemeris Time to Spacecraft Clock String**
 - SCE2S (*scid*, *et*, SCLKCH)
 - » Requires both LSK and SCLK
- **Encoded Spacecraft Clock to Spacecraft Clock String**
 - SCDECD (*scid*, *sclkdp*, SCLKCH)
 - » Requires SCLK
 - » SCLK string examples:
 - 1/1487147147.203 (Cassini, MGS)
 - 1/05812:00:001 (Voyager 1 and 2)
 - When literal clock strings are used as arguments in modules they must be contained in quotes:
 - Single quotes for Fortran, IDL, Matlab
 - Double quotes for C
- **Ephemeris Time to Local Solar Time String**
 - ET2LST(*et*, *body*, *long*, *type*, HR, MN, SC, TIME, AMPM)
 - » Requires SPK, PCK



Conversion Between Uniform Time Systems

Navigation and Ancillary Information Facility

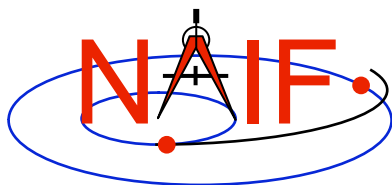
- Conversion between numeric representations of TAI, TDB(ET), TDT, JDTDB(JED), JDTDT
 - Return value = UNITIM (*epoch*, *insys*, *outsys*)



Customizing the Time System

Navigation and Ancillary Information Facility

- **Defaults**
 - Two digit year (a bad idea but supported):1969-2068
 - Time System: UTC
 - Calendar: Gregorian
- **Adjustments**
 - The one hundred year interval to which two digit years belong may be set. For example 1980-2079
 - Time Systems: UTC, TDB, TT (Terrestrial Time)
 - Calendar: Gregorian, Julian, or Mixed.
- **See the TIMDEF module header and *Time Required Reading* (time.req) for details**



Principle Time System Interfaces

Navigation and Ancillary Information Facility

