TCSng Commands

This is the most comprehensive list to date of the TCSng requests and commands.

Requests

```
ALL Bulk information
    Args: N/A
    Returns: [MOT] [RA] [DEC] [HA] [UT] [ALT] [AZ] [SECZ] [Epoch]
        MOT = (see MOTION)
        RA(Right Assention) = HH:MM:SS.ss
        DEC(Declination) = +DD:MM:SS.ss
        HA(Hour Angle) =  HH:MM:SS
        UT(Universal Time) = HH:MM:SS
        ALT(Elevation) = XXX.xx
        AZ(Azmouth) = XXX.xx
        SECZ(airmass) = XX.xx
        EPOCH = EEEE.e
```

```
AZ  Azimuth
    Args: N/A
    Returns: [ddd.dd]
```

```
BEAM    Chop/Nod info for chopping secondary
    Args: N/A
    Returns: ???
```

```
CORRECTIONS  String describing what corrections and rates are
enabled/disabled
    Args: N/A
    Returns: MPNARFp+tob
        M=Proper Motion
        P=Precession
        N=Nutation
        A=Aberration
        R=Refraction
        F=Flexure
        p=Parallax
        +=pointing model used...  can change to a,b,c,d?
        t=Sidereal
        o=Object
        b=Bias
    If disabled, character will be replaced by "_"
```

```
DATE    date based on UT
    Args: N/A
    Returns: [MM/DD/YYYY]
```

```
DEC Declination
    Args: N/A
    Returns [ddmmss.ss]
```

```
DISABLE Output state
    Args: N/A
    Returns: 1 for disabled, 0 for enabled
```

```
DISEPOCH Current Epoch
    Args: N/A
    Returns: XXXX.x
```

```
EL  Elevation
    Args: N/A
    Returns: [ddd.dd]
```

```
EQ  Equinox
    Args: N/A
    Returns: XXXXX.x
```

```
HA  Hour Angle
    Args: N/A
    Returns: [dd:mm:ss]
```

```
JD  Julian Date
    Args: N/A
    Returns: [JJJJJJJ.j]
```

```
      FLEXFILE Retrieve contents of current flex file (Pointing Correction)
being used   Added by Scott 12/2012
               Args: N/A
               Returns: [KEY1 Val1\n] [KEY2 VAL2\n] [KEY3 VAL3\n] ...
```

```
      LIMITPROF Horizon limit profile as a function of azimuth      Added by
Scott 12/2012
                     The first (and last) value given is the horizon limit
at 0 degrees azimuth
                     each subsequent value is a horizon limit at increments
of 15 degrees of azimuth
                     so the 2nd value is the limit at 15 degrees azimuth,
the third is the horizon limit
                     at 30 degrees azimuth etc.
               Args: N/A
               Returns [hh] [hh] [hh] [hh] ... (25 in all)
```

```
LIMIT   limit status bits (ACCORDING TO JAVA GUI)
    Args: N/A
    Returns: this returns an 8 bit integer whose bits represent the
following
```

```
        bit0(LSB) =     RA
        bit1    =    DEC
        bit2    =    Derotator
        bit3    =    Hardware Horizon
        bit4    =    Software Horizon
        bit5    =    Focus Upper
        bit6    =    Focus Lower
        a 1 indicates limit active, 0 indicates limit not active
```

```
LIMIT   limit status bits  (ACCORDING TO ERIC CHRISTIENSEN)
    Args: N/A
    Returns: this returns an 8 bit integer whose bits represent the
following
        bit0(LSB) =     RA/HA limit
        bit1    =    DEC limit
        bit2    =    derot.
        bit3    =    hor. hard limit
        bit4    =    hor. soft limit
        bit5    =    focus lo limit?
        bit6    =    focus hi limit?
        a 1 indicates limit active, 0 indicates limit not active

      LIMITINHIBIT   limit inhibit status
                     args N/A
                     Returns: 0 if limits are active and 1 if limits are
inactive
MOTION  Motion status bits
    Args: N/A
    Returns: this returns an 8 bit integer whose bits represent the
following
        bit0(LSB) =     RA/AZ
        bit1    =    DEC/EL
        bit2    =    FOC
        bit3    =    DOME
        bit4-8    =    undefined
        a 1 indicates axis in motion, 0 indicates no motion
```

```
PAD String describing hardware paddle button states
    Args: N/A
    Returns: any combination of the following charachers
        N = North
        S = South
        E = East
        W = West
        D = Drift (if not present, assume Guide)
    A characher is present if switch active, otherwise switch inactive
```

```
PADDRIFT Hardware paddle drift rate arcsec/sec
    Args: N/A
    Returns: -XXXXXXXXX.xxx
```

```
PADGUIDE Hardware paddle guide rate arcsec/sec
    Args: N/A
    Returns: -XXXXXXXXX.xxx
```

```
RA  Right Assention
    Args: N/A
    Returns: [ddmmss.ss]
```

```
SECZ    Air Mass
    Args: N/A
    Returns: [XXXXX.XX]
```

```
ST  Sidereal Time
    Args: N/A
    Returns: [hh:mm:ss]
```

```
TIME    Universal Time
    Args: N/A
    Returns: [HH:MM:SS]
```

```
VERIFY  Verify that an RA/DEC position is within telescope limits
    Args: [RA] [DEC] [Epoch]
        RA=HH:MM:SS.ss
        DEC=DD:MM:SS.ss
        Epoch=EEEE.eeee
    Returns: 1 if coordinate is within limits
```

```
XALL    "Extended ALL"
    Args: N/A
    Returns: [FOC] [DOME] [IIS] [PA] [UTD] [JD]
        FOC(focus pos) = +XXXXXX
        DOME(Dome Position) = -180 (always this value...)
        IIS(???) = -224.4 (always this value...)
        PA(Paralactic Angle) = -145.7 (always this value...)
        UTD(UT Date) = MM/DD/YYYY
        JD(Julian Date) = XXXXXXX.x
```

```
XDEC    "Extended DEC"
    Args: N/A
    Returns: [COM] [NEXT] [REF] [OFF] [WOB] [DIFF] [BIAS] [GUIDE] [DRIFT]
        COM(Commanded Position) = +HH:MM:SS.ss
        NEXT(Next Position) = +HH:MM:SS.ss
        REF(Reference Position) = +HH:MM:SS.ss
        OFF(Offset Position) = +HH:MM:SS.ss
        WOB(Wobble) = +HH:MM:SS.ss
        DIFF(Difference) = +XXXXXXXXX.xxx
        BIAS(Bias Rate) = +XXXXXXXXX.xxx
        GUIDE(Guide Rate) = +XXXXXXXXX.xxx
        DRIFT(Drift Rate) = +XXXXXXXXX.xxx
```

```
XRA "Extended RA"
    Args: N/A
    Returns: [COM] [NEXT] [REF] [OFF] [WOB] [DIFF] [BIAS] [GUIDE] [DRIFT]
        COM(Commanded Position) = HH:MM:SS.ss
        NEXT(Next Position) = HH:MM:SS.ss
        REF(Reference Position) = HH:MM:SS.ss
        OFF(Offset Position) = +HH:MM:SS.ss
        WOB(Wobble) = +HH:MM:SS.ss
        DIFF(Difference) = +XXXXXXXXX.xxx
        BIAS(Bias Rate) = +XXXXXXXXX.xxx
        GUIDE(Guide Rate) = +XXXXXXXXX.xxx
        DRIFT(Drift Rate) = +XXXXXXXXX.xxx
```

undefined Request

```
TEST1   ??? >> probably useless but is currently in the command set
    Args: ???
    Returns: ???
```

```
INDEX   ???
    Args: ???
    Returns: ???
```

```
PP  ???
    Args: ???
    Returns: ???
```

```
SRVFRQ  ???
    Args: ???
    Returns: ???
```

Dome Request

```
DOME    Dome control information
    Args: "PARAM" returns dome setup parameters, all other strings return
control info
    Returns:
        Parameters: [CPD] [SD] [W] [SDW] [NU] [RHO] [PHI] [LOOK] [HOLD]
            CPD(Counts Per Degree) = XXX.xxxxxxx
            SD(Stow Degrees)= XXX.xxxxxxx
            W(Dome Width) = XXX.xxxxxxx
            SDW(Stow Dome Width)= XXX.xxxxxxx
            NU = XXX.xxxxxxx
            RHO = XXX.xxxxxxx
            PHI = XXX.xxxxxxx
            LOOK(Lookahead) = XX
            HOLD(Hold Dome) = XX
        Control Info: [DEL] [MOD] [INIT] [TELAZ] [AZ] [HOME]
```

```
            DEL(Delta Position) = +XXX.XXXXXXX
            MOD(Mode) = XX
            INIT(Initialized) = XX
            TELAZ(Telescope Azmouth) =  +XXX.XXXXXXX
            AZ(Dome Azmouth) = +XXX.XXXXXXX
            HOME(Home Position) = +XXX.XXXXXXX
```

Focus Request

```
FOCSPEED Focus Speed
    Args: N/A
    Returns: ["FAST" or "SLOW"]
```

```
FOCUS   Focus position
    Args: N/A
    Returns: +XXXXX
```

Commands

```
ABERRATE  Aberration corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
BIAS    Bias enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
BIASDEC DEC bias rate in arcseconds/second
    Args: [XXXXX.XX]
    Returns: "OK" or "FAILED"
```

```
BIASRA  RA biasrate in arcsseconds/second
    Args: [XXXXX.XX]
    Returns: "OK" or "FAILED"
```

```
CANCEL  Cancel current move
    Args: N/A
    Returns:  "OK" or "FAILED"
```

```
CLEARDIFF  Clear RA and DEC difference value
    Args: N/A
    Returns: N/A
```

```
DISABLE disable motion output
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
DISEPOCH Set Epoch
    Args: XXXX.x
    Returns: "OK" or "FAILED"
```

```
DECLARE Initialize current position
    Args: "INITNEXT" to initialize "NEXT" position as current position
          "INITCOM" to initialize "COMMANDED" position as current position
    Returns: "OK" or "FAILED"
```

```
ELAZ    Move to position in Elevation and Azmouth
    Args: [EE.EE] [AAA.AA]
    Returns: "OK" or "FAILED"
```

```
ENABLE  enable motion output
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
FLEX    Flexure corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
LIMIT   Limit override >> USE WITH EXTREME CAUTION!!!!!
    Args: "INHIBIT" will override limits, all other strings will enable
limits
    Returns: "OK" or "FAILED"
```

```
MOVNEXT Move to NEXT position
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
MOVOFF  move to OFFSET position
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
MOVRADEC  Move to RA-DEC position
    Args: RA DEC EPOCH RAPM DECPM
        RA = HH:MM:SS.ss
        DEC = +DD:MM:SS.ss
        EPOCH = EEEE.eeee
        RAPM(RA Proper Motion) = XXXXX.xxx
        DECPM(DEC Proper Motion) = XXXXX.xxx
    Returns: "OK" or "FAILED"
```

```
MOVSTOW Move to stow position
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
MOVWOB  MOVWOB beam
    Args: ???
    Returns: ???
```

```
NEXTPOS Set NEXT position
    Args: RA DEC EPOCH RAPM DECPM
        RA = HH:MM:SS.ss
        DEC = +DD:MM:SS.ss
        EPOCH = EEEE.eeee
        RAPM(RA Proper Motion) = XXXXX.xxx
        DECPM(DEC Proper Motion) = XXXXX.xxx
    Returns: "OK" or "FAILED"
```

```
NUTAT   Nutation corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
PAD Software paddle command [Direction] [rate]  or PAD XX for termination
                    This function is not recommended for use near the
horizon limits!
    Args: [Direction] [rate]
        DIRECTION = NORTH, SOUTH, EAST, WEST, NE, NW, SE, SW
        RATE = XXXXXXX.xx (arcsecs/sec)
        any string not described in DIRECTION will terminate paddle
                       this terminate string must be sent at the end of each
movement
                       when the button is released.
    Returns: "OK" or "FAILED"
```

```
PADDLE  Paddle enable/disable
    Args: "ON" = enable, any other string will disable
    Returns: "OK" or "FAILED"
```

```
PADDRIFT  Paddle Drift rate in arcseconds/second
    Args: [XXXXX.XX]
    Returns: "OK" or "FAILED"
```

```
PADGUIDE  Paddle Guide rate in arcseconds/second
    Args: [XXXXX.XX]
    Returns:  "OK" or "FAILED"
```

```
PARALLAX  Parallax corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
PARAM   ???
    Args: ???
    Returns: ???
```

```
PRECES  Precession corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
PROPMO  Proper motion corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
REFPOS  REFerence POSition (tod)
    Args: ???
    Returns: ???
```

```
REFRAC  Refraction corrections enable/disable
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"
```

```
STEPDEC Move Declination XXXXX.XX arcseconds
    Args: [XXXXX.XX]
    Returns:  "OK" or "FAILED"
```

```
STEPRA  Move Right Assention XXXXX.XX arcseconds
    Args: [XXXXX.XX]
    Returns:  "OK" or "FAILED"
```

```
TRACK   Enable/Disable sidereal tracking
    Args: "ON" = enable, any other string will disable
    Returns:  "OK" or "FAILED"

WOBBLE  WOBBLE -HH:MM:SS.ss -DD:MM:SS.ss
    Args: ???
    Returns: ???
```

Catalogs Command

```
ABELL   ABELL Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
FK5 FK5 Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
IC  IC Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
NGC NGC Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
OKESTONE  Okestone Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED
```

```
PPM PPM Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
SAO SAO Catalog Object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
YBSC    YSBC Catalog Object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
ZWICKY  ZWICKY Catalog object XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

Planets Command

```
MERCURY Track Mercury
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
VENUS   Track Venus
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
MARS    Track Mars
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
JUPITER Track Jupiter
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
SATURN  Track Saturn
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
URANUS  Track Uranus
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
NEPTUNE Track Neptune
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
PLUTO   Track Pluto
    Args: N/A
```

```
    Returns: "OK" or "FAILED"
```

```
MOON    Track Moon
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
SUN Track Sun
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
DOME    Command dome control
    Args: This command takes one argument at a time from the following
        AUTO    Autodome enable
            Args: ON = autodome on, any other = autodome off
            Returns:   "OK" or "FAILED"
```

```
        INIT    Initialize dome
            Args: N/A
            Returns:   "OK" or "FAILED"
```

```
        STOW    Stow dome
            Args: N/A
            Returns:   "OK" or "FAILED"
```

```
        LOOKAHEAD  Lookahead enable
            Args: positive nonozero number=enable, any other = disable
            Returns:   "OK" or "FAILED"
```

```
        PARAM    Set Dome Parameters
            Args: [CPD] [SD] [W] [SDW] [NU] [RHO] [PHI] [LOOK] [HOLD]
                CPD(Counts Per Degree) = XXX.xxxxxxx
                SD(Stow Degrees)= XXX.xxxxxxx
                W(Dome Width) = XXX.xxxxxxx
                SDW(Stow Dome Width)= XXX.xxxxxxx
                NU = XXX.xxxxxxx
                RHO = XXX.xxxxxxx
                PHI = XXX.xxxxxxx
                LOOK(Lookahead) = XX
                HOLD(Hold Dome) = XX
                Returns:   "OK" or "FAILED"
```

```
        PADDLE    Control Paddle buttons
            Args: RIGHT = move right, LEFT = move left, any other= stop
            Returns:   "OK" or "FAILED"
```

```
FOCUS   move to absolute focus value XXXXXXX
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
RELFOCUS  relative move focus value XXXXXXX
```

```
    Args: XXXXXXX
    Returns: "OK" or "FAILED"
```

```
FOCZERO Zero current focus position
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
FOCSTOP focus paddle stop
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
FOCUP   focus paddle up
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
FOCDN   focus paddle down
    Args: N/A
    Returns: "OK" or "FAILED"
```

```
FOCSPEED  Set focus speed
    Args: "FAST" sets to fast, all other strings set speed slow
    Returns: "OK" or "FAILED"
```

PEC Request

```
PECSTAT Current PEC operation status
    Args: N/A
    Returns: [PEC_Condition] [PEC_Count] [PEC_Index] [PEC_Mode]
```

```
PECPROG Current PEC programming status
    Args: N/A
    Returns: [Percent_Done] [PEC_Correction]
```

Command

```
PECFILE Attempt to create a PEC file.
    Args: ???
    Returns: ???
```

```
PEC Turn on PEC
    Args: "ON" = enable, any other string will disable
    Returns: "OK" or "FAILED"
```

Servo Request

```
CON ??? >> SERVO STUFF...  NOT FOR NORMAL USE
    Args: ???
    Returns: ???
```

```
SAMDATA ??? >> SERVO STUFF...  NOT FOR NORMAL USE
    Args: ???
    Returns: ???

SAMDONE ??? >> SERVO STUFF...  NOT FOR NORMAL USE
    Args: ???
    Returns: ???
```

```
SERVO   ??? >> SERVO STUFF...  NOT FOR NORMAL USE
    Args: ???
    Returns: ???
```

Command

```
WCON    axis, gd gp gi dmax vmax groot  >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
SERVO   ???    >>  AXIS SERVO SAMPLING...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
GD  axis, value >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
GP  axis, value >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
GPI axis, value >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
DMAX    axis, value >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
VMAX    axis, value >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
PERMAX  axis, value >> SERVO CONST...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
SAMPLE  axis, interval, total samples  >>  AXIS SERVO SAMPLING...  DO NOT
MODIFY
    Args: ???
```

```
    Returns: ???
```

```
DUMPSAM axis   >>  AXIS SERVO SAMPLING...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
SAMSTART  ???    >>  AXIS SERVO SAMPLING...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
SAMABORT  ???    >>  AXIS SERVO SAMPLING...  DO NOT MODIFY
    Args: ???
    Returns: ???
```

```
SYSSAVE ???
    Args: ???
    Returns: ???
```

```
SYSKILL Kills TCS process after disabling stopping all telescope motion.
    Args: N/A
    Returns: OK or Failure
      SYSRESET Restarts TCS process after disbling the telescope       Added
by Scott 12/2012
                  Telescope should be stowed before doing this.
               Args: [TIME] [DD/MM/YY] [HH:MM:SS]
               If no arguments are used the TCS simply restarts
               as if you had restarted the TCS computer. If for some reason
               the time is bad on the TCS computer you can add the TIME
               argument followed by the date and time in UT and TCS will
               use this and not the computer time for astrometry.
               Returns OK or Failure
```