

```

pro read_swath_l1_hsb, pattern, numfp, numline, $
    tai, lat, lon, rad, solzen, $
    scanang=scanang, $
    sun_glint_distance=sun_glint_distance, $
    satheight=satheight

filename = findfile ( pattern, count = cnt )
if cnt ne 1 then begin
    print, 'findfile did not return exactly one file ', cnt, pattern
    if cnt eq 0 then return
endif

print, 'read_swath_l1_hsb: ', filename[0]

FAIL = -1
SWATH_L1HSB = 'L1B_HSB'

fid=EOS_SW_OPEN(filename[0], /READ)
if fid eq FAIL then return

SWid=EOS_SW_ATTACH(fid, SWATH_L1HSB)
if SWid eq FAIL then return

ret = EOS_SW_READFIELD(Swid,"Time",tai,EDGE=edge)
sz=size(tai)
numfp=sz[1]
numline=sz[2]

ret = EOS_SW_READFIELD(Swid,"Latitude",lat)
ret = EOS_SW_READFIELD(Swid,"Longitude",lon)
ret = EOS_SW_READFIELD(Swid,"solzen",solzen)
ret = EOS_SW_READFIELD(Swid,"brightness_temp",rad)

if arg_present(scanang) then begin
    print, "reading the scan angle"
    ret = EOS_SW_READFIELD(Swid,"scanang",scanang)
endif

if arg_present(sun_glint_distance) then begin
    print, "reading the scan angle"
    ret = EOS_SW_READFIELD(Swid,"sun_glint_distance",sun_glint_distance)
endif

if arg_present(satheight) then begin
    print, "reading the satellite height"
    ret = EOS_SW_READFIELD(Swid,"satheight",satheight)
endif

```

```
print, "done reading - detaching and closing the file"
ret = EOS_SW_DETACH(SWid)
ret = EOS_SW_CLOSE(fid)

end
```