

Description of the High Altitude MMIC Sounding Radiometer (HAMSR)

Level 2 data format

4/20/2015

Overview

The High Altitude MMIC Sounding Radiometer (HAMSR) is a 25 channel cross-track scanning microwave sounder with channels near the 60 and 118 GHz oxygen lines and the 183 GHz water vapor line. A detailed description of the instrument and a characterization of its performance are found in Brown et al. 2011. The Level2 product contains time-ordered and geo-located brightness temperatures for the Earth scan for each of the 25 HAMSR channels along with retrieved products, including temperature and water vapor profiles, precipitable water vapor and integrated cloud liquid water and derived reflectivity. There are three flags of note, a land flag, sea ice flag and retrieval quality flag which are described below. The retrieved products are only valid when the land flag and sea ice flag are equal to zero. The HAMSR Level2 data files are in netCDF format.

Level2 Contents

Level 2 Product

The contents of the Level 2 files are shown in the following table. The variables in the netCDF file are also fully attributed and self describing. The cross track dimension is 42 and the along track dimension varies from flight to flight.

Variable Name	Variable Description	Variable Dimensions
HAMSR time	seconds since 2000-01-01 00:00:00.0	along track
pixel latitude	Latitude for each HAMSR pixel [-90:90]	cross track x along track
pixel longitude	Longitude for each HAMSR pixel [-180:180]	cross track x along track
altitude	Aircraft altitude from GPS in meters	along track
Re-sampled brightness temperature	Calibrated Brightness Temperature for the Earth scene resampled to a uniform posting. Default value is -1.	channel x cross track x along track
pixel Earth incidence angle	Earth incidence angle for each HAMSR pixel [0:89.9]	cross track x along track
aircraft latitude	Aircraft Latitude [-90:90]	along track
aircraft longitude	Aircraft Longitude [-180:180]	along track
aircraft roll	Aircraft Roll [-180:180]	along track

aircraft pitch	Aircraft Pitch [-180:180]	along track
aircraft heading	Aircraft Heading [-180:180]	along track
Land flag	0 – ocean >0- not ocean (retrievals currently not valid over land)	cross track x along track
Sea Ice Flag derived from NCEP	0 – no sea ice 1- sea ice present (retrievals not valid)	along track
Ancillary surface temperature	Surface temperature from NCEP (K)	along track
Ancillary surface elevation	Surface elevation from NCEP (m)	along track
Ancillary surface pressure	Surface pressure from NCEP (mb)	along track
Ancillary surface wind speed	Surface wind speed from NCEP (m/s)	along track
HAMSR precipitable water vapor –regression algorithm	Integrated water vapor in cm	cross track x along track
HAMSR cloud liquid water – regression algorithm	Integrated cloud liquid water in mm	cross track x along track
HAMSR Air Temperature Profile	Vertical air temperature from HAMSR at 33 levels [K]	cross track x along track x vertical
HAMSR Absolute Humidity Profile	Vertical Absolute Humidity from HAMSR at 33 levels [g/m ³]	cross track x along track x vertical
HAMSR Cloud Liquid Water Profile	Vertical cloud liquid water density from HAMSR at 33 levels [g/m ³]	cross track x along track x vertical
HAMSR Relative Humidity Profile	Vertical relative humidity from HAMSR at 33 levels [%]	cross track x along track x vertical
HAMSR Potential Temperature Profile	Vertical potential temperature derived from HAMSR profiles at 33 levels [K]	cross track x along track x vertical
HAMSR Equivalent Potential Temperature Profile	Vertical equivalent potential temperature derived from HAMSR profiles at 33 levels [K]	cross track x along track x vertical
HAMSR Lifting Condensation Level	Lifting condensation level derived from HAMSR profiles [mb]	cross track x along track
HAMSR Level of Free Convection	Level of free convection derived from HAMSR profiles [mb]	cross track x along track
HAMSR precipitable water vapor from profile	Integrated water vapor derived from HAMSR absolute humidity profile in cm	cross track x along track
HAMSR cloud liquid water from profile	Integrated cloud liquid water derived from HAMSR cloud water profile in cm	cross track x along track
HAMSR air temperature at the surface	Air temperature at surface retrieved from HAMSR [K]	cross track x along track
HAMSR relative humidity at the surface	Relative humidity at surface retrieved from HAMSR [%]	cross track x along track
HAMSR absolute humidity at the surface	Absolute humidity at surface retrieved from HAMSR [K]	cross track x along track
HAMSR air temperature at the flight altitude	Air temperature at flight altitude retrieved from HAMSR [K]	cross track x along track

HAMSRS relative humidity at the flight altitude	Relative humidity at flight altitude retrieved from HAMSRS [%]	cross track x along track
HAMSRS absolute humidity at the flight altitude	Absolute humidity at flight altitude retrieved from HAMSRS [K]	cross track x along track
HAMSRS Profile Retrieval Quality Flag	0-good convergence and low residual error 1-converged with higher residual error (use with caution) 2-did not converge (use not recommended)	cross track x along track
HAMSRS Profile Pressure Levels	Pressure at each of the 33 levels for the HAMSRS vertical profiles [mb]	33 levels
HAMSRS Height of Pressure Levels	Height at each of the 33 pressure levels for the HAMSRS vertical thermodynamic profiles [m]	33 levels
HAMSRS derived radar reflectivity profile	X-band reflectivity derived from HAMSRS TBs at 33 levels [dBZ]	cross track x along track x vertical
HAMSRS Height of Reflectivity Profile Levels	Height at each of the 15 levels where reflectivity is retrieved from HAMSRS [m]	33 levels

HAMSR Level 2 NetCDF Header Dump

```
netcdf HAMSR_L2_20121105T105445_20121106T123042_v01.nc {
  dimensions:
    channel = 25;
    cross_track = 42;
    along_track = 18866;
    HAMSR_levels = 33;
    HAMSR_dBz_levels = 33;
  variables:
    double time(along_track=18866);
      time:units = "seconds since 2000-01-01 00:00:00.0";
      time:comment = "seconds since 2000-01-01 00:00:00.0";
      time:scale_factor = 1.0; // double
      time:long_name = "Measurement time";
      time:standard_name = "time";

    int lat(along_track=18866, cross_track=42);
      lat:units = "degrees_north";
      lat:comment = "Pixel Latitude [-90:90]";
      lat:scale_factor = 0.001; // double
      lat:long_name = "Pixel Latitude";
      lat:standard_name = "latitude";

    int lon(along_track=18866, cross_track=42);
      lon:units = "degrees_east";
      lon:comment = "Pixel Longitude [-180:180]";
      lon:scale_factor = 0.001; // double
      lon:long_name = "Pixel Longitude";
      lon:standard_name = "longitude";

    int altitude(along_track=18866);
      altitude:units = "m";
      altitude:comment = "Aircraft altitude from GPS in meters";
      altitude:scale_factor = 0.1; // double
      altitude:long_name = "Altitude";
      altitude:standard_name = "altitude";
      altitude:coordinates = "time AClat AClon";

    int TB(along_track=18866, cross_track=42, channel=25);
      TB:units = "K";
      TB:comment = "Calibrated Brightness Temperature";
      TB:scale_factor = 0.001; // double
      TB:long_name = "Brightness Temperature";
      TB:standard_name = "brightness_temperature";

    short inc(along_track=18866, cross_track=42);
      inc:units = "degrees";
      inc:comment = "Pixel Incidence Angle [-180:180]";
      inc:scale_factor = 0.01; // double
      inc:long_name = "Pixel Incidence Angle";
      inc:coordinates = "time lat lon";

    int AClat(along_track=18866);
      AClat:units = "degrees_north";
      AClat:comment = "Airplane Latitude [-90:90]";
      AClat:scale_factor = 0.001; // double
      AClat:long_name = "Airplane Latitude";
```

```

int AClon(along_track=18866);
    AClon:units = "degrees_east";
    AClon:comment = "Airplane Longitude [-180:180]";
    AClon:scale_factor = 0.001; // double
    AClon:long_name = "Airplane Longitude";

short ACroll(along_track=18866);
    ACroll:units = "degrees";
    ACroll:comment = "Airplane Roll [-180:180]";
    ACroll:scale_factor = 0.01; // double
    ACroll:long_name = "Airplane Roll";
    ACroll:standard_name = "platform_roll_angle";
    ACroll:coordinates = "time AClat AClon";

short ACpitch(along_track=18866);
    ACpitch:units = "degrees";
    ACpitch:comment = "Airplane Pitch [-180:180]";
    ACpitch:scale_factor = 0.01; // double
    ACpitch:long_name = "Airplane Pitch";
    ACpitch:standard_name = "platform_pitch_angle";
    ACpitch:coordinates = "time AClat AClon";

short ACheading(along_track=18866);
    ACheading:units = "degrees";
    ACheading:comment = "Airplane Heading [-180:180]";
    ACheading:scale_factor = 0.01; // double
    ACheading:long_name = "Airplane Heading";
    ACheading:standard_name = "platform_yaw_angle";
    ACheading:coordinates = "time AClat AClon";

short PWV(along_track=18866, cross_track=42);
    PWV:units = "cm";
    PWV:comment = "HAMSR Precipitable Water Vapor - Regression Algorithm";
    PWV:scale_factor = 0.001; // double
    PWV:long_name = "HAMSR Precipitable Water Vapor from Regression Algorithm";
    PWV:coordinates = "time lat lon";

short CLW(along_track=18866, cross_track=42);
    CLW:units = "mm";
    CLW:comment = "HAMSR Integrated Cloud Liquid Water - Regression Algorithm";
    CLW:scale_factor = 1.0E-4; // double
    CLW:long_name = "HAMSR Integrated Cloud Liquid Water from Regression Algorithm";
    CLW:coordinates = "time lat lon";

short sea_ice_flag(along_track=18866);
    sea_ice_flag:comment = "Sea Ice Flag (0-no ice, 1-ice)";
    sea_ice_flag:scale_factor = 1.0; // double
    sea_ice_flag:long_name = "Sea Ice Flag";
    sea_ice_flag:flag_values = 0S, 1S; // short
    sea_ice_flag:flag_meanings = "0_No_Ice 1_Ice";

short anc_sfc_ht(along_track=18866);
    anc_sfc_ht:units = "m";
    anc_sfc_ht:comment = "Ancillary Surface Height";
    anc_sfc_ht:scale_factor = 1.0; // double
    anc_sfc_ht:long_name = "Ancillary Surface Height";

```

```

short anc_Psfc(along_track=18866);
    anc_Psfc:units = "mb";
    anc_Psfc:comment = "Ancillary Surface Pressure";
    anc_Psfc:scale_factor = 0.1; // double
    anc_Psfc:long_name = "Ancillary Surface Pressure";

short land_flag(along_track=18866, cross_track=42);
    land_flag:comment = "Land flag (0-ocean, >0 not ocean)";
    land_flag:scale_factor = 1.0; // double
    land_flag:long_name = "Land flag";
    land_flag:coordinates = "time lat lon";

short anc_ws(along_track=18866);
    anc_ws:units = "m/s";
    anc_ws:comment = "Ancillary 10-m Wind Speed";
    anc_ws:scale_factor = 0.01; // double
    anc_ws:long_name = "Ancillary 10-m Wind Speed";

short ham_airT(HAMSR_levels=33, along_track=18866);
    ham_airT:units = "K";
    ham_airT:comment = "HAMSR Vertical Air Temperature";
    ham_airT:scale_factor = 0.1; // double
    ham_airT:long_name = "HAMSR Vertical Air Temperature";

short ham_airQ(HAMSR_levels=33, along_track=18866);
    ham_airQ:units = "g/m^3";
    ham_airQ:comment = "HAMSR Vertical Absolute Humidity";
    ham_airQ:scale_factor = 0.001; // double
    ham_airQ:long_name = "HAMSR Vertical Absolute Humidity";

short ham_airL(HAMSR_levels=33, along_track=18866);
    ham_airL:units = "g/m^3";
    ham_airL:comment = "HAMSR Vertical Cloud Liquid Water";
    ham_airL:scale_factor = 1.0E-4; // double
    ham_airL:long_name = "HAMSR Vertical Cloud Liquid Water";

short ham_airRH(HAMSR_levels=33, along_track=18866);
    ham_airRH:units = "%";
    ham_airRH:comment = "HAMSR Vertical Relative Humidity";
    ham_airRH:scale_factor = 0.01; // double
    ham_airRH:long_name = "HAMSR Vertical Relative Humidity";

int ham_airPT(HAMSR_levels=33, along_track=18866);
    ham_airPT:units = "K";
    ham_airPT:comment = "HAMSR Potential Temperature";
    ham_airPT:scale_factor = 0.1; // double
    ham_airPT:long_name = "HAMSR Potential Temperature";

int ham_airEPT(HAMSR_levels=33, along_track=18866);
    ham_airEPT:units = "K";
    ham_airEPT:comment = "HAMSR Equivalent Potential Temperature";
    ham_airEPT:scale_factor = 0.1; // double
    ham_airEPT:long_name = "HAMSR Equivalent Potential Temperature";

short ham_LCL(along_track=18866);
    ham_LCL:units = "mb";
    ham_LCL:comment = "HAMSR Lifting Condensation Level";
    ham_LCL:scale_factor = 0.1; // double

```

```

    ham_LCL:long_name = "HAMSR Lifting Condensation Level";

short ham_LFC(along_track=18866);
    ham_LFC:units = "mb";
    ham_LFC:comment = "HAMSR Level of Free Convection";
    ham_LFC:scale_factor = 0.1; // double
    ham_LFC:long_name = "HAMSR Level of Free Convection";

short ham_prof_PWV(along_track=18866);
    ham_prof_PWV:units = "cm";
    ham_prof_PWV:comment = "HAMSR Precipitable Water Vapor from Retrieved Profile";
    ham_prof_PWV:scale_factor = 0.001; // double
    ham_prof_PWV:long_name = "HAMSR Precipitable Water Vapor from Retrieved Profile";

short ham_prof_CLW(along_track=18866);
    ham_prof_CLW:units = "mm";
    ham_prof_CLW:comment = "HAMSR Integrated Cloud Liquid Water from Retrieved Profile";
    ham_prof_CLW:scale_factor = 1.0E-4; // double
    ham_prof_CLW:long_name = "HAMSR Integrated Cloud Liquid Water from Retrieved
Profile";

short ham_sfc_airT(along_track=18866);
    ham_sfc_airT:units = "K";
    ham_sfc_airT:comment = "HAMSR Air Temperature at Surface";
    ham_sfc_airT:scale_factor = 0.1; // double
    ham_sfc_airT:long_name = "HAMSR Air Temperature at Surface";

short ham_sfc_airQ(along_track=18866);
    ham_sfc_airQ:units = "g/m^3";
    ham_sfc_airQ:comment = "HAMSR Absolute Humidity at Surface";
    ham_sfc_airQ:scale_factor = 0.001; // double
    ham_sfc_airQ:long_name = "HAMSR Absolute Humidity at Surface";

short ham_sfc_airRH(along_track=18866);
    ham_sfc_airRH:units = "%";
    ham_sfc_airRH:comment = "HAMSR Relative Humidity at Surface";
    ham_sfc_airRH:scale_factor = 0.01; // double
    ham_sfc_airRH:long_name = "HAMSR Relative Humidity at Surface";

short ham_FltLvl_airT(along_track=18866);
    ham_FltLvl_airT:units = "K";
    ham_FltLvl_airT:comment = "HAMSR Air Temperature at Flight Level";
    ham_FltLvl_airT:scale_factor = 0.1; // double
    ham_FltLvl_airT:long_name = "HAMSR Air Temperature at Flight Level";

short ham_FltLvl_airQ(along_track=18866);
    ham_FltLvl_airQ:units = "g/m^3";
    ham_FltLvl_airQ:comment = "HAMSR Absolute Humidity at Flight Level";
    ham_FltLvl_airQ:scale_factor = 0.001; // double
    ham_FltLvl_airQ:long_name = "HAMSR Absolute Humidity at Flight Level";

short ham_FltLvl_airRH(along_track=18866);
    ham_FltLvl_airRH:units = "%";
    ham_FltLvl_airRH:comment = "HAMSR Relative Humidity at Flight Level";
    ham_FltLvl_airRH:scale_factor = 0.01; // double
    ham_FltLvl_airRH:long_name = "HAMSR Relative Humidity at Flight Level";

short ham_ret_qual_flag(along_track=18866);

```

```

        ham_ret_qual_flag:comment = "HAMSR Profile Retrieval Quality Flag [0-good convergence
and low RMS, 1-converged with higher RMS;use with caution, 2-did not converge;use not recommended";
        ham_ret_qual_flag:scale_factor = 1.0; // double
        ham_ret_qual_flag:long_name = "HAMSR Profile Retrieval Quality Flag";
        ham_ret_qual_flag:flag_values = 0S, 1S, 2S; // short
        ham_ret_qual_flag:flag_meanings = "0_good_convergence_and_low_RMS
1_converged_with_higher_RMS_use_with_caution 2_did_not_converge_use_not_recommended";

short ham_pres_levels(HAMSR_levels=33);
    ham_pres_levels:units = "mb";
    ham_pres_levels:comment = "HAMSR Profile Pressure Levels";
    ham_pres_levels:scale_factor = 0.1; // double
    ham_pres_levels:long_name = "HAMSR Profile Pressure Levels";

short ham_airHt(HAMSR_levels=33, along_track=18866);
    ham_airHt:units = "m";
    ham_airHt:comment = "HAMSR Height of Pressure Levels";
    ham_airHt:scale_factor = 1.0; // double
    ham_airHt:long_name = "HAMSR Height of Pressure Levels";

short ham_dBz(HAMSR_dBz_levels=33, along_track=18866, cross_track=42);
    ham_dBz:units = "dBz";
    ham_dBz:comment = "HAMSR Reflectivity Profile";
    ham_dBz:scale_factor = 0.01; // double
    ham_dBz:long_name = "HAMSR Derived Reflectivity Profile";
    ham_dBz:coordinates = "time lat lon ham_dBz_heights";

short ham_dBz_heights(HAMSR_dBz_levels=33);
    ham_dBz_heights:units = "m";
    ham_dBz_heights:comment = "HAMSR Height of Reflectivity Profile";
    ham_dBz_heights:scale_factor = 1.0; // double
    ham_dBz_heights:long_name = "HAMSR Height of Reflectivity Levels";
    ham_dBz_heights:standard_name = "height";

// global attributes:
:Conventions = "CF-1.6";
}

```

References

Brown, S. T.; Lambrigtsen, B.; Denning, R. F.; Gaier, T.; Kangaslahti, P.; Lim, B. H.; Tanabe, J. M.; Tanner, A. B.; , "The High-Altitude MMIC Sounding Radiometer for the Global Hawk Unmanned Aerial Vehicle: Instrument Description and Performance," *IEEE Transactions on Geoscience and Remote Sensing*,; doi: 10.1109/TGRS.2011.2125973

Acknowledgments:

Copyright 2013 California Institute of Technology. Government sponsorship acknowledged.