Recommendation P.372 Calculation Engine and Associated Utility Programs

This document is a brief description of P372.DLL, ITURNoise.exe, and MakeP732figs.py programs. Please consult the source of ITURNoise() to see more details of the interfacing to the P372.DLL. (See https://github.com/ITU-R-Study-Group-3/ITU-R-HF). This document is provided as an overview of these programs and an initial descripting of their use.

P372.DLL

Functions within P372.DLL can be accessed from either the _cdecl or __stdcall calling conventions.

    __cdecl Calling convention

        AllocateNoiseMemory

        AtmosphericNoise_LT

        FreeNoiseMemory

        InitializeNoise

        MakeNoise

        Noise

        P372CompileTime

        P372Version

        ReadFamDud

    __stdcall Calling convention

        __AllocateNoiseMemory@4

        __AtmosphericNoise_LT@36

        __FreeNoiseMemory@4

        __InitializeNoise@4

        __MakeNoise@52

        __Noise@32

        __P372CompileTime@0

        __P372Version@0

        __ReadFamDud@12

The input parameters to these subroutines are in the following forms.

```
int AllocateNoiseMemory(struct NoiseParams *noiseP);

int FreeNoiseMemory(struct NoiseParams *noiseP);

int Noise(struct NoiseParams *noiseP, int hour, double lng, double lat, double
frequency);
```

```
int ReadFamDud(struct NoiseParams *noiseP, const char *DataFilePath, int
month);
```

```
void InitializeNoise(struct NoiseParams *noiseP);
```

```
char const * P372CompileTime();
```

```
char const * P372Version();
```

```
void AtmosphericNoise_LT(struct NoiseParams* noiseP, struct FamStats* FamS,
int lrxmt, double lng, double lat, double frequency);
```

```
int MakeNoise(int month, int hour, double lat, double lng, double freq, double
mmnoise, char* datafilepath, double* out, int pntflag);
```

For additional details please see noise.h in the ITU-R-HF repo.

ITURNoise.exe

The program ITURNoise.exe is a utility program that has two modes:

Mode 1 which returns the combined total noise, total noise upper decile,

and total noise lower decile., and the constituent parts of the total noise:

galactic, atmospheric, and man-made noise calculation or

Mode 2 generates the data necessary to create Recommendation P.372-14 Figures 13 through 36: a), b), and c)

Mode 1 – Noise Calculation for a Single Location

To return the noise parameters for a single location, Mode 1 above, 7 command line arguments are required. Below is an example of running ITURNoise.exe to return noise for a single location

```
C:\>ITURNoise 1 14 1.0 40.0 165.0 0 "G:\User\Data\" 0
```

Argument 1 is an integer that indicates the month (1 to 12) of interest. In the example above the desired month is January.

Argument 2 is an integer that indicates the hour (1 to 24) UTC at the receive point. In most cases this will require the calculation of the time UTC from the longitude at that receive point. In the example above the receive point is 11 hours (165° E/(15°/hr) ahead of UTC so to determine the noise for the 12th local time hour the 1st hour UTC is chosen.

Argument 3 is a float that indicates the latitude (degrees). In the example above the latitude desired is 40° N.

Argument 4 is a float that indicates the longitude (degrees). In the example above the longitude desired is 165° E

Argument 5 is a float that indicates the man-made noise which can either be categorical (0-5) or value of man-made noise in dB (represented by a negative number). In the example above the value 1.0 indicates that the noise category residential. The category codes appear below.

```
code   Noise Category
0.0    City
1.0    Residential
2.0    Rural
3.0    Quiet Rural
4.0    Noisy
5.0    Quiet
```

If argument 5 is give as a negative number, the man-made noise will be set to that value where the galactic and atmospheric noise calculations are not performed. This second mode is of little value for this stand-alone program, ITURNoise.exe, running, P533.dll. Overriding the P372 calculation is included here and is consistent with some analyses desired in ITURHFProp which also uses the P372.dll.

Argument 6 indicates the location of the required CCIR data files. The data file path must be given as a string enclosed in double quotes without trailing back slash. In the example above the location of the required CCIR data files are on drive C: in the directory Data, "C:\Data"

Argument 7 allows the program to provide return data from the P.372 calculation is several formats. The table below gibes the integer print flag codes.

```
code          Print Mode
No argument   Prints detailed output to stdout
0             No output
1             Prints detailed outpuit to stderr
2             Prints detailed output to file "MakeNoiseOut.txt"
3             Prints detailed header, column description, and CSV
              output to stdout
4             Prints CSV output to stdout
```

Example ITURNoise() Output

```
*********************************************************
      ITU-R Study Group 3: Radiowave Propagation
*********************************************************
      Analysis: 1/7/20 - 12:47:04
      P372 Version:      14.2
      P372 Compile Time: Wed Jul  1 11:00:38 2020
*********************************************************

      JANUARY  : 14 (UTC)
      40.0000 (deg lat) 165.0000 (deg long)
      1.000 (MHz)

      [FaA]  Noise Component (Atmospheric): 60.733
      [DuA]  Upper Decile    (Atmospheric): 10.601
      [DlA]  Upper Decile    (Atmospheric): 8.278
      [FaM]  Noise Component    (Man-Made): 76.800
      [DuM]  Upper Decile       (Man-Made): 11.000
      [DlM]  Lower Decile       (Man-Made): 6.700
      [FaG]  Noise Component    (Galactic): 52.000
      [DuG]  Upper Decile       (Galactic): 2.000
      [DlG]  Lower Decile       (Galactic): 2.000
      [FamT] Noise                 (Total): 76.987
      [DuT]  Upper Decile          (Total): 10.940
      [DlT]  Lower Decile          (Total): 6.574


*********************************************************
```

Mode 2 – Generate Atmospheric Noise Figure Data Output

There is one command line argument required for Mode 2, the data file path in quotes without the trailing back slash (Please see the description of Mode1 Argument 6 above). The following example would generate the figure data in the root directory.

```
C:\>ITURNoise "G:\User\Data\"
```

Mode 2 creates the following directory structure in the current directory or directory where it is run.

```
.\ ----- P372_figures --|-- a ----- csv
                        |
                        |-- b ----- csv
                        |
                        |-- c ----- csv
```

There are 72 figure data files that are created, 24 per category (a), b), and c)). The naming convention for the output files follow x_ymzh.csv, where x is the figure data type (a, b, or c), y is the month (1, 4, 7, or 10), and h is the local time hour (0, 4, 8, 12, 16, or 20).

MakeP372figs.py

MakeP372figs.py was developed for Python 3.7.7 and is dependent on the Python libraries os, glob, numpy, cartopy, matplotlib, math, and pandas. At present MakeP372figs.py only runs on Windows.

Generation of the Recommendation P.372-14 Figure files

The data files, and P372_figures directory structure, are required as input for the Python program MakeP372figs.py. When the program MakeP372figs.py is run from a directory, that immediately contains the directory structure P372_figures above, the following output directories will be created

```
.\ ----- P372_figures --|-- a --|-- svg
                        |       |-- png
                        |       |-- pdf
                        |
                        |-- b --|-- svg
                        |       |-- png
                        |       |-- pdf
                        |
                        |-- c --|-- svg
                                |-- png
                                |-- pdf
```

MakeP372figs.py writes each figure in three common formats: scalable vector graphics (SVG), portable networks graphics (PNG), and portable document format (PDF). The later two are produced at 300 dpi.

Note that the P.372-14 13-36 a) Figures are labeled relative season in the northern hemisphere. The figures generated from MakeP372figs.py are now labeled with the central month of the relative northern hemisphere season.

N. Hemisphere Season (P.372-14 Figures)	Months
Winter	DEC-JAN-FEB
Spring	MAR-APR-MAY
Summer	JUN-JUL-AUG
Autumn	SEP-OCT-NOV

Specifically, the a) figure data generated by ITUNoise() in Mode 2 are January, April, July, and October. Also note the a) Figures created by MakeP372figs.py have longitude range from 180 degrees west longitude to 180 degrees east longitude instead of the current a) Figure range from 60 degrees west longitude to 60 degrees east longitude. The b) and c) Figures are very similar to the published figures in P.372-14 with the exception that the new c) Figures no longer include V_dm, "Expected value of median deviation of average voltage."

behm/july 2020