**ACC to CSV converter  
Version 0.0  
16th February 2018  
Oisin Creaner**

This program extracts data from ACC files and adds it to a CSV file which includes header and structured data elements.

**Outline**

The system opens the raw data file, reads in the floating-point data for one covariance matrix at a time and converts it to an array of complex numbers. The program then iterates over the array and creates data rows consisting of subband, RCU information and the complex covariance coefficient for the given pair of RCUs. These are then written to a CSV file. The draft version includes hard-coded elements for demonstration purposes

**Design Diagram**

**Write Header**

**Telescope File**

**CSV Output File**

**Check Input**

**For Each Subband**

**Read raw data**

**Convert to Array**

**Array: Covariance matrix**

**For Each RCU (i)**

**For Each RCU (j)**

**Convert Array [i,j] element to string**

**Append element to covariance matrix**

**String: Covariance matrix**

**Write string to file**

**Set variables**

Figure 1: Outline of the design of the conversion from ACC file to CSV

**Operation**

1. Variables to control the program are set
   1. Input filename (*Currently hard coded for testing purposes*)
   2. Output filename (replace .dat extension of input file with .csv)
   3. RCU Count (*n*: *Hard Coded to 192*)
      1. Count (size of the covariance matrix: 2*n*2)
      2. Count string (format string for reading using struct.unpack method)
   4. Number of Subbands (*N*: *Hard coded to 512*)
   5. Float Size (*f*: *Hard coded to 8 – probably possible to do this automatically*)
2. Checks the inputs:
   1. Output file name different from input
   2. Input file of the correct size (2 *Nf n*2)
3. Opens the input and output files
4. Loops over each subband
   1. Reads the covariance matrix for a given subband from the input file and stores it in an array
      1. Converts the data type of that array to complex
   2. Creates a string to output the covariance matrix to text
   3. Loops over each RCU (i)
      1. Loops over each RCU (j) against which a covariance has been calculated
         1. Creates a string containing the subband number, RCU(i), RCU(j) and the complex covariance coefficient between i and j
         2. Appends this string to the output string
   4. Writes the output string to the Output file
5. Closes the files

**Sample Output**

**Subband,RCU(i),RCU(j),Covariance**

**1,0,0,(13329498112+0j)**

**1,0,1,(4605345792-4394118j)**

**1,0,2,(1124073472+1775524j)**

**1,0,3,(2499805184-88966j)**

**1,0,4,(-3196059648-1525678j)**

**…**

**34,20,180,(1806+1769j)**

**34,20,181,(-1133+308j)**

**34,20,182,(-660+939j)**

**34,20,183,(-973-640j)**

**…**

**512,191,187,(-2813-1055j)**

**512,191,188,(88-641j)**

**512,191,189,(497+120j)**

**512,191,190,(190-1699j)**

**512,191,191,(798519+0j)**