# Development of software algorithm for compression of Ultrasonic C-Scan data files
# (for Non-Destructive Testing Applications)

Under the guidance of:


Mr S. K. Lalwani, Scientific Officer (H)

Data Acquisition and Processing Systems Section (DA&PS),

Electronics Division (ED)

Bhabha Atomic Research Centre, Mumbai.



**Presented By:**
**Amitabh Yadav**
**B.Tech. Electronics Engineering**
**University of Petroleum and Energy Studies**


**June 3, 2016 to July 26, 2016**

# Bhabha Atomic Research Centre

- The Bhabha Atomic Research Centre (BARC) is India's premier nuclear research facility based in Trombay, Mumbai.

- Multi-disciplinary research centre with extensive infrastructure for advanced research and development covering the entire spectrum of nuclear science, engineering and related areas.

- Department of Atomic Energy (DAE), Government of India.

- The Government of India created the Atomic Energy Establishment, Trombay (AEET) on 3 January 1954.

- After Homi J. Bhabha's death in 1966, the centre was renamed as the Bhabha Atomic Research Centre on 22 January 1967.

- BARC has consistently maintained that the reactors are used for this purpose only – Apsara, Kamini, Purnima (I, II and III), Zerlina, Cyrus and Dhruva.

# Electronics Division (ED), BARC

Location: Modular Labs, BARC.
The main Sub-divisions of Electronics and Instrumentation Group are:

**Electronics Division**
**Accelerator Control Division**

**Control Instrumentation Division**

**Reactor Control Division**

**Reactor Control System Design Section**

**Advanced Technology Systems Section**

**Electronics & Instrumentation Systems Division etc.**

Electronics Division Head:
**Dr. Debashis Das**

DA&PS Section Head:
**Mr. S. K. Lalwani**

## **Project :** Development of Ultrasonic-Data Compression Algorithm

- Large Amount of Data is generated during Ultrasonic imaging in medical, industrial and non-destructive testing (NDT) applications.
- Consequently, it is desirable to use data compression techniques to reduce data and to facilitate the analysis and remote access of ultrasonic information.
- Ultrasonic Data Compression not only solves the problem of data storage but also facilitates the ease of data transmission.
- In this project, a study of various Ultrasonic Data Compression Algorithms, primarily, lossless algorithms, is presented.
- Keywords: Data compression, ASCII, non-destructive testing, information theory and coding, wavelet transforms and ASIC.
- Software/Programming: Matlab and C++

# Introduction

- *Data with redundancy can be compressed. Data without redundancy cannot be compressed.*
- In ASCII, each character is assigned an 8-bit code (code is of 7 bits, 8th bit is parity). A fixed size code is important because it makes it easy for software applications to handle characters of text.
- *ASCII code is redundant because it assigns to each character, common or rare, the same number of bits.*
- Removing redundancy can be done by assigning variable size codes to the characters, with short codes assigned to the common characters, and long size codes assigned to the rare ones.
- Information Theory and Coding can be utilized to removed redundancies while bearing no loss to the data.

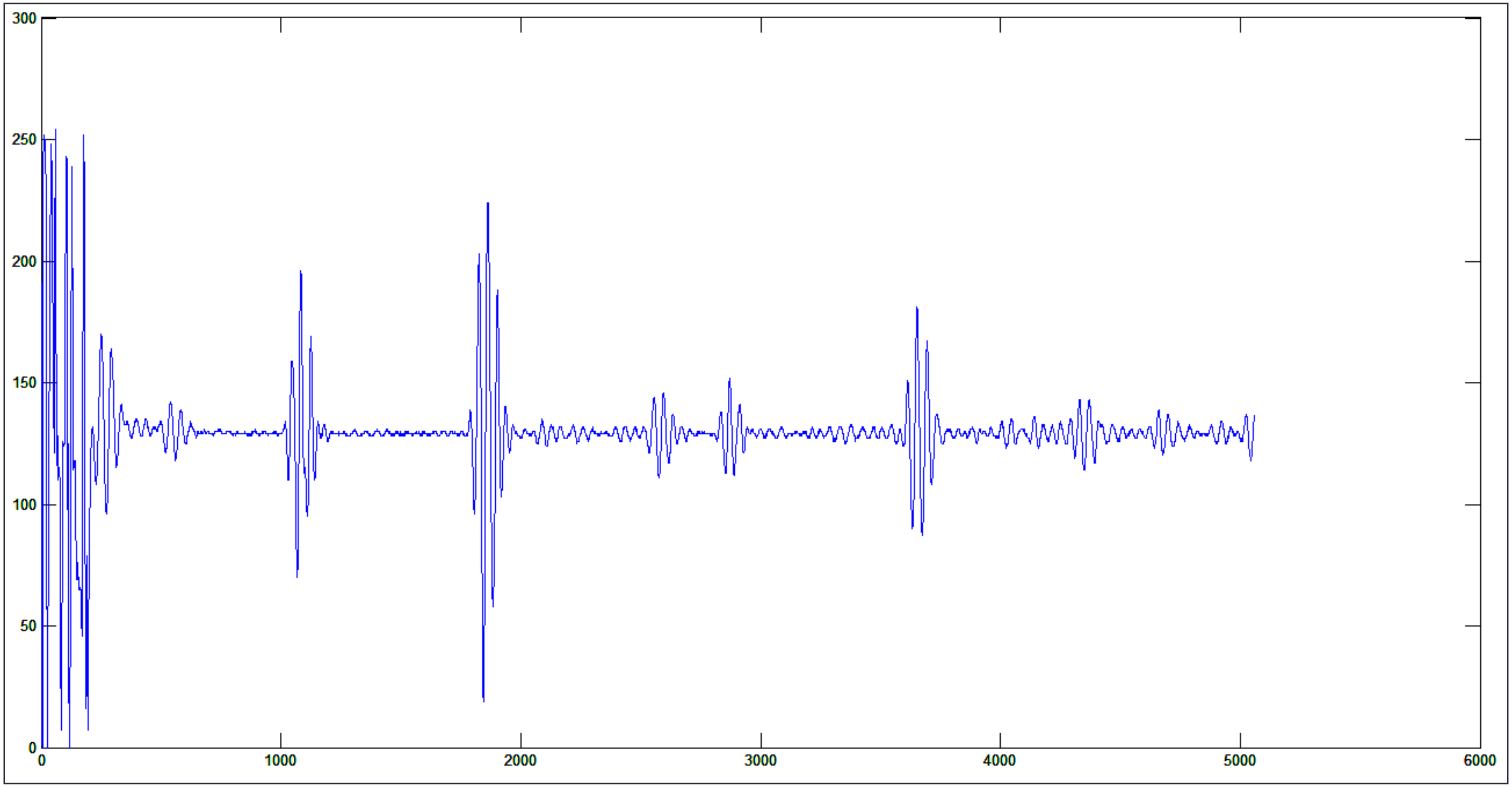
# Pulse-Echo Method for NDT Ultrasonic Testing of Metals

- Testing on a single point on the surface of metal, generates a data file containing ultrasonic echo data on that point which contains number of samples identified by the quantity ***MSPS (millions of samples per second).*** A single run on the point generates an A-Scan.
- An ***A-Scan*** data contains information of the received echo at the receiving transducer after the ultrasonic pulse is scattered from the internal defect.

- This signal is then passed on to amplifier and then digitizer for sampling. The digitized data are then stored in the ASCII format. The generated file is thus an ASCII file identified by the extension, ***.asc***.

- An A-Scan is the basic form of data obtained from NDT. Practically, on a piece of metal, the NDT is performed linearly in one dimension or in two dimensions.
- In 1 dimension: B-Scan (.bsn)
- In 2 dimension: C-Scan (.csn)

# Ultrasonic Data




A-Scan Data (2MHz/50MSPS) (Source: ED, BARC)

# Data Compression

- ***Regardless of the method used to compress, the effectiveness of the compression depends on the amount of redundancy in the data.***
- ***There is no such thing as universal, efficient data compression algorithm.***
- ***Compression can be LOSSY or LOSSLESS.***

Certain compression methods are **lossy**. They achieve better compression by losing some information. Such a method makes sense, when compressing movies, images or sound.
e.g. DCT, DFT, Wavelet Transform etc.

In contrast, text files, specially files containing computer programs, may become worthless if even a single bit is missing and therefore, they must be compressed only by **lossless** algorithms.
e.g. Huffman Coding, Arithmetic Coding

$$Compression\ Ratio = \frac{Size\ of\ Output\ File}{Size\ of\ Input\ File}$$

# Data Compression

- Run-Length Encoding is a scheme to reduce data redundancy.
- Data compression may be viewed as a branch of information theory in which the primary objective is to minimize the amount of data to be transmitted.
- Lossless data compression algorithms usually exploit statistical redundancy to represent data without losing any information, so that the process is reversible.
- Modern Lossless Compression Algorithms:

***Lempel–Ziv (LZ) compression***

***Burrows–Wheeler transform*** etc.

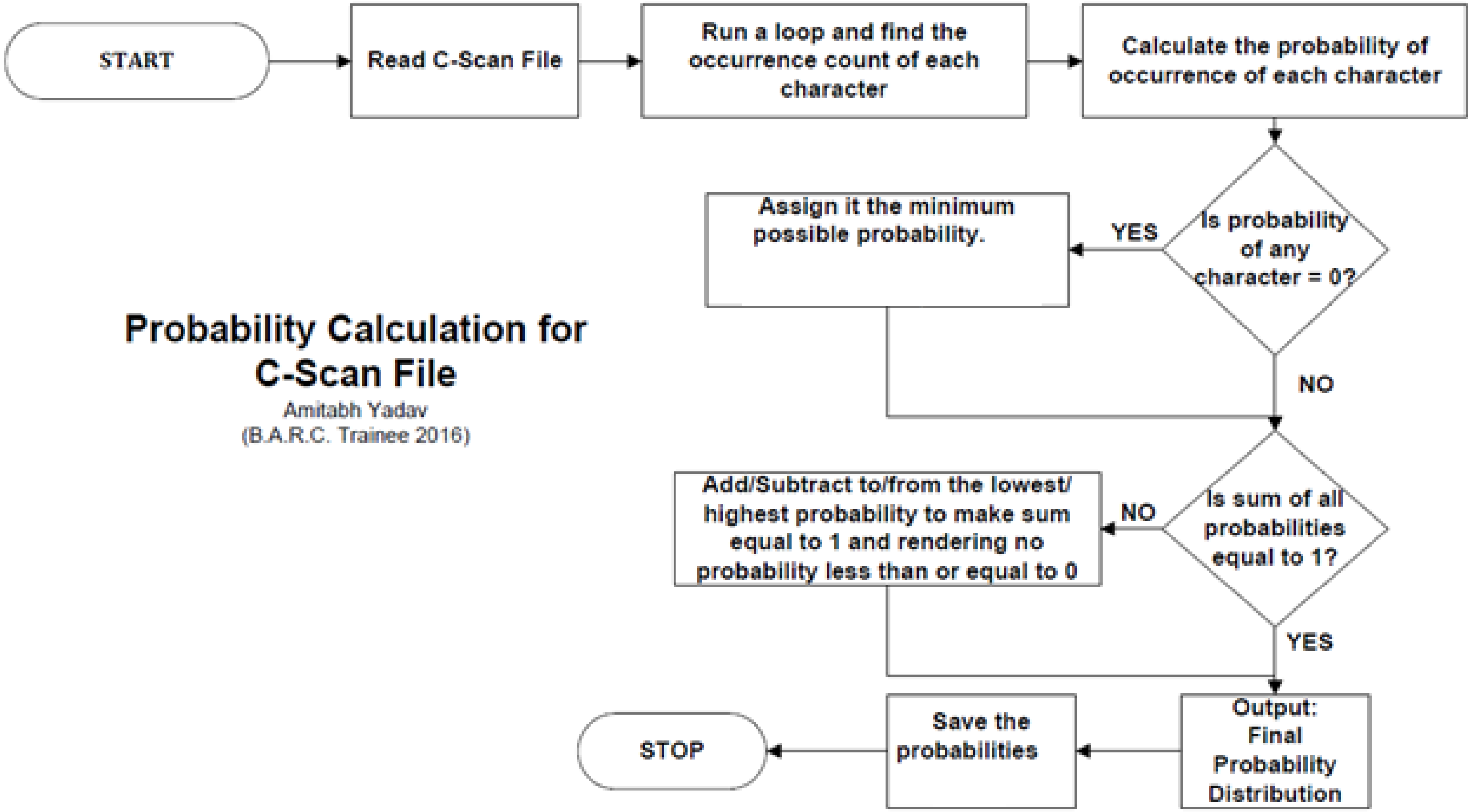
# Huffman Coding

- Huffman's algorithm, expressed graphically, takes as input a list of nonnegative weights {w(1), ... ,w(n) } and constructs a full binary tree [a binary tree is full if every node has either zero or two children] whose leaves are labeled with the weights.
- When the Huffman algorithm is used to construct a code, the weights represent the probabilities associated with the source letters.
- Removing redundancy is done by assigning variable size codes to the characters, with short codes assigned to the common characters, and long size codes assigned to the rare ones. This is precisely how Huffman Coding works.

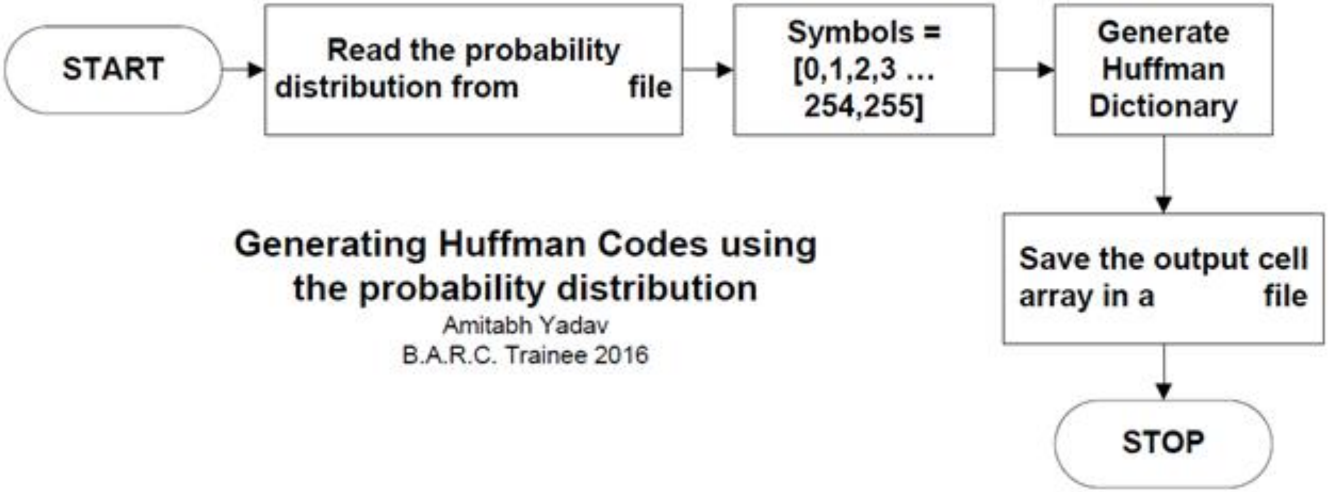
- ***Further, the algorithm is guaranteed to produce an optimal (minimum redundancy) code [Huffman 1952]***

# Compression Algorithm

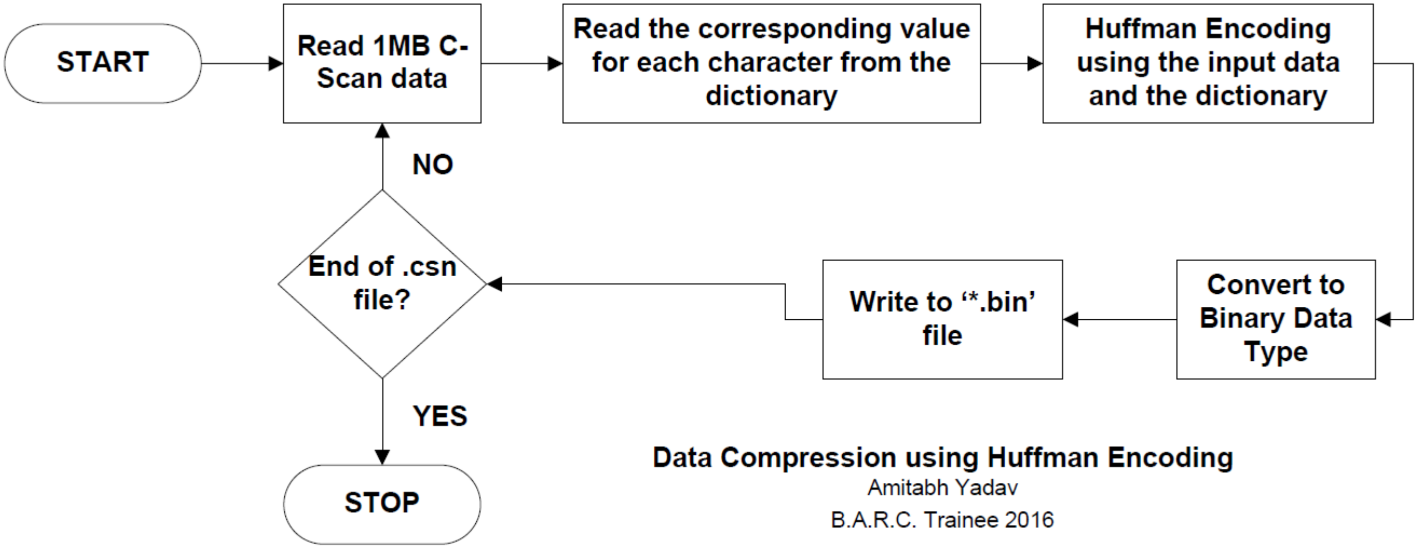
We have the probability distribution, using these we will generate variable size codes to the individual characters sequentially. The set of these variable size codes is called a ‘Dictionary’. To generate a dictionary we need an array of possible characters (or symbols) and another array with corresponding value of probability.



# Compression Algorithm



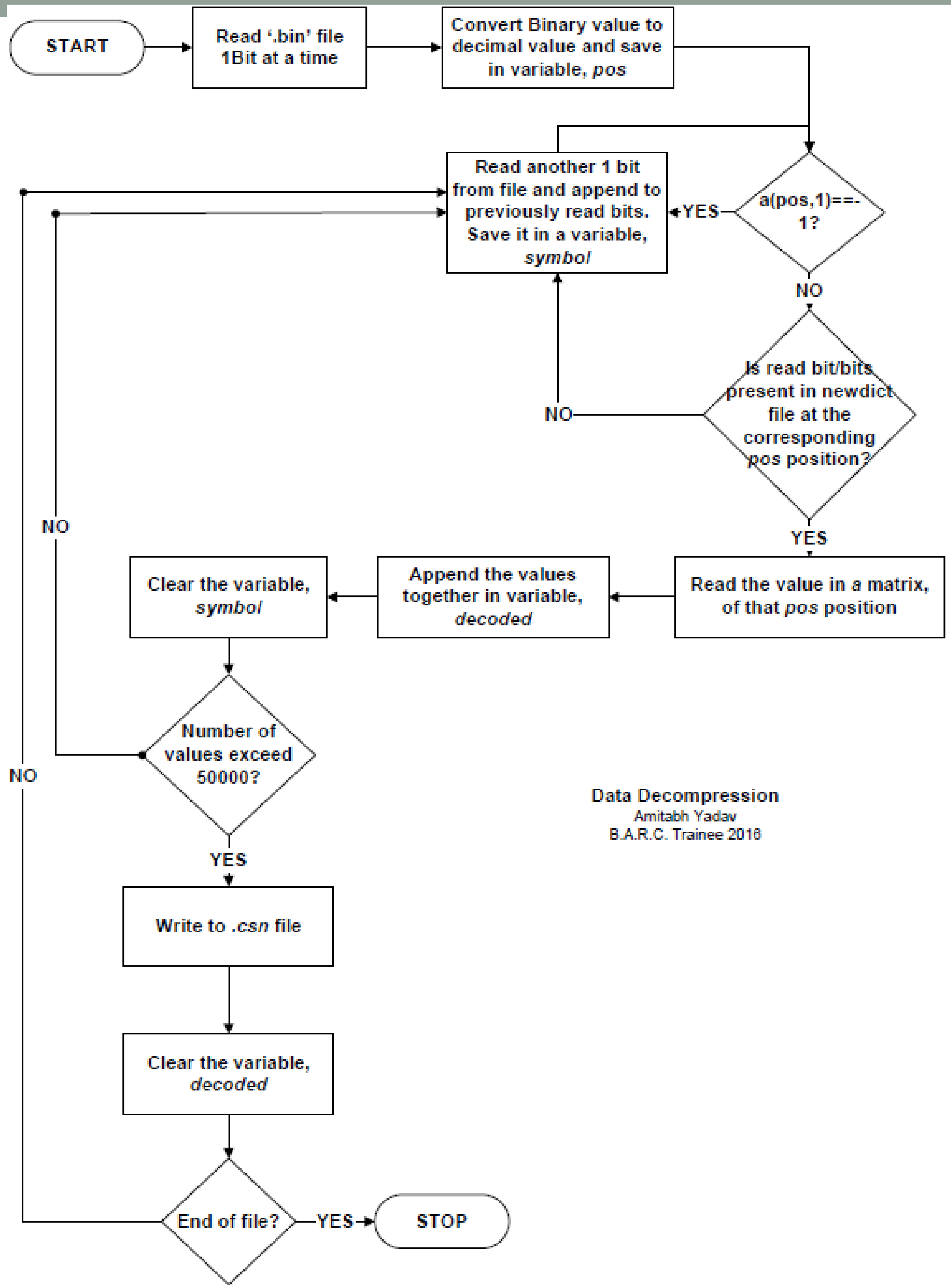

Generating Huffman Codes using the probability distribution
Amitabh Yadav
B.A.R.C. Trainee 2016



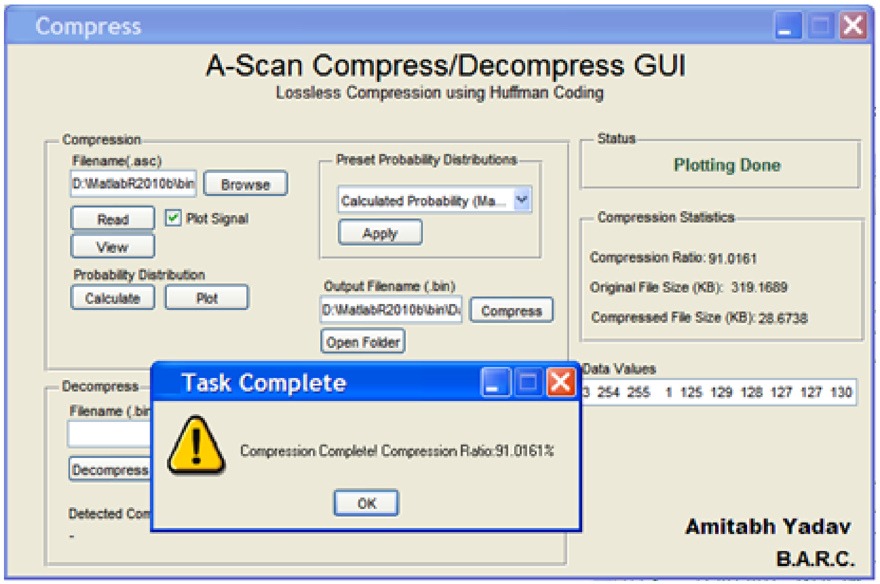

Data Compression using Huffman Encoding
Amitabh Yadav
B.A.R.C. Trainee 2016

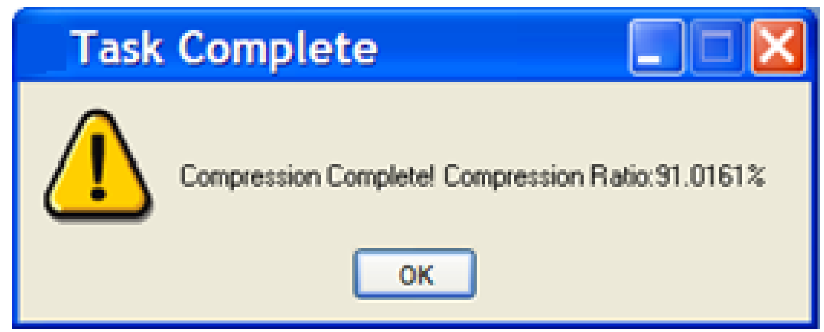
# De-Compression Algorithm

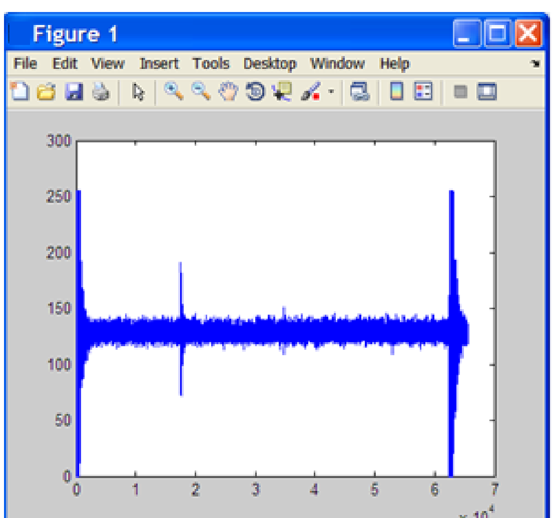
From the dictionary created during the compression, read all the variable size codes assigned to the symbols and save it in a file sequentially.



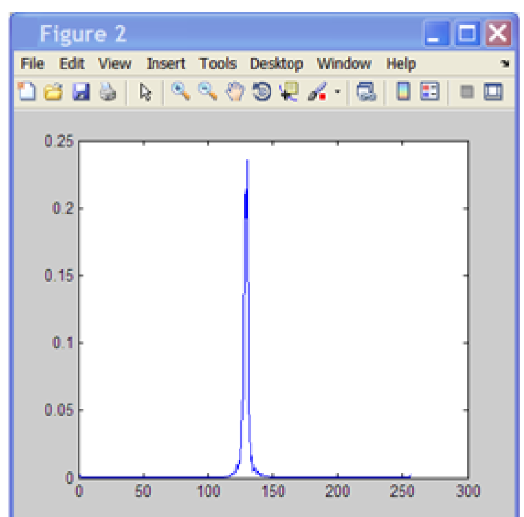
# Results : A-Scan Compression




A-Scan Datafile




Calculated Probability Distribution

Input File: .asc
Output File: .bin

# Results : C-Scan Compression

- C-Scan file size: 16MB
- Compressed File Size: 4.06MB (CR = 75.37%)

- This algorithm was tested for various probability distributions including equal probabilities (dictionary 1), rough approximation (dictionary 2) and calculated probabilities and approximation (dictionary 3).

- For dictionary 1, compression was ~1%. For Dictionary2, the compression achieved is ~11.2% and finally, for Dictionary3, the compression achieved is ~75.37%.

- Case Study: For a C-Scan file of size 16928KB (16.5MB), the compressed file size achieved was 4168KB. **The time elapsed was ~456sec (7.5mins)**.

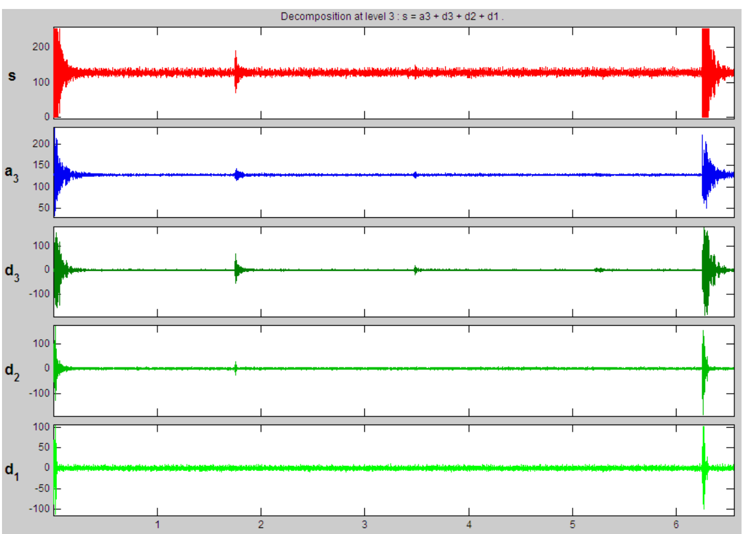
- **Data Decompression was achieved correctly but with drastic <u>performance reduction</u>.** Execution took long to actually decompress the data! Attempts to reduce time taken were performed to improve the performance.

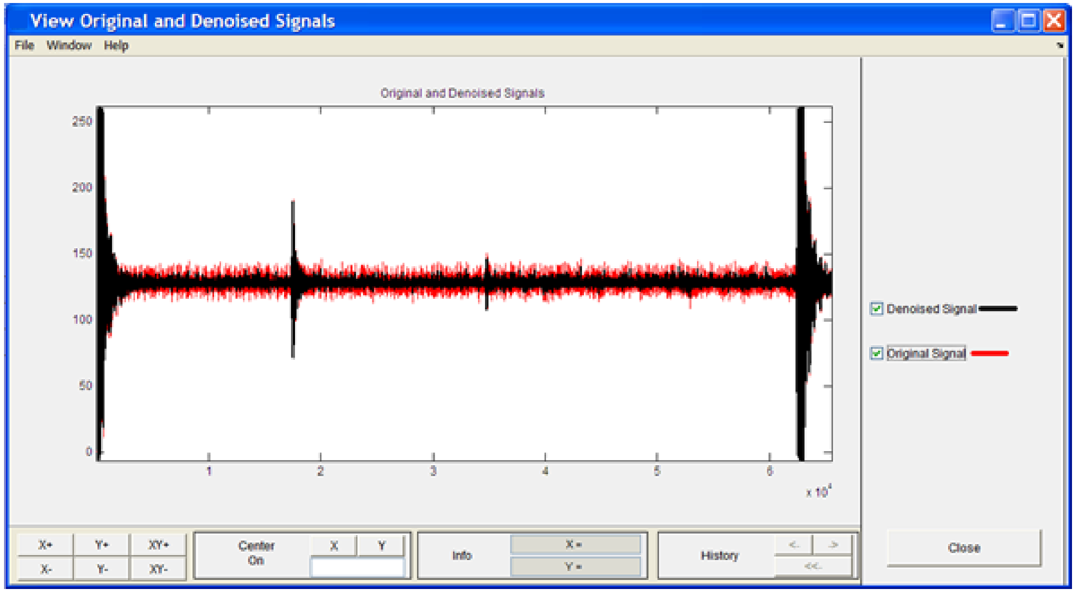
- Permission to produce image was not granted by DA&PS, ED, BARC.

# Results : Other Analysis

Lossy Compression Methods were studies and tested during the course of the internship, but rendered inefficient/complicated, including the following:

1. Discrete Cosine Transform (DCT), also used in JPEG.
2. JPEG Compression
3. Discrete Fourier Transform
4. Wavelet Analysis and Transform (Compression and Noise Reduction)



# References


- “Ultrasonic Data Compression via Parameter Estimation”, Guilherme Cardoso and Jafar Saniie; IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 52, No. 2, February 2005 (313-325)
- “Compression of Ultrasonic RF Data” A. Pesavento, V. Burow, H. Ermert; IEEE 1997 Ultrasonics Symposium Proceedings (1996) pp. 1471-1474
- “Multidimensional Representation of Ultrasonic Data Processed by Reconfigurable Ultrasonic System-on-Chip Using OpenCL High-Level Synthesis”, Spenser Gilliland, Clementine Boulet, Thomas Gonnot and Jafar Saniie; 2014 IEEE International Ultrasonics Symposium Proceedings pp.1936-1939
- “3D Ultrasonic Signal Compression Algorithms for High Signal Fidelity”, Pramod Govindan, Thomas Gonnot, Spenser Gilliland and Jafar Saniie, IEEE 2013 pp.1263-1266
- “Model-based estimation of ultrasonic echoes”, Ramazan Demirli, Student Member, IEEE, and Jafar Saniie, Senior Member, IEEE; IEEE Transactions on Ultrasonics Ferroelectrics and Frequency Control · June 2001
- “Ultrasonic Signal Compression Using Wavelet Packet Decomposition and Adaptive Thresholding”, Erdal Oruklu, Namitha Jayakumar and Jafar Saniie; 2008 IEEE International Ultrasonics Symposium Proceedings  pp.171-175
- [LZ77] Ziv J., Lempel A., "A Universal Algorithm for Sequential DataCompression", IEEE Transactions on Information Theory", Vol. 23, No. 3,pp. 337-343
- “A guide to data compression methods” by David Salomon. Springer.
- “Fractal and Wavelet Image compression Techniques” by Stephen Welstead.

# Thank You