International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991
Vol. 4, No. 4
November 2015

www.ijerst.com

Email: editorijerst@gmail.com or editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 4, No. 4, November 2015

Research Paper

REALIZATION OF INPUT TEST DATA VOLUME REDUCTION USING XMATCHPRO TECHNIQUE

Ch Bharath Venkatesh[1]* and V Srinivasa Rao[2]

*Corresponding Author: Ch Bharath Venkatesh ✉ bharath.venkatesh304@gmail.com

Test data compression methods reduce the input test data volume by allowing compressed tests to be stored on a tester. Test data is data which has been specifically identified for use in tests, typically of a computer program.The compression architecture is based around a block of CAM to realize the dictionary.Input data is driven as a size of 32 bits, and the same output will appear as input using a dictionary technique. First the data is given to the comparator and the same stored in the memory, defined by the address, and then now the comparator performs the comparison between the given data and the locations in the memory. These outputs from the comparator is driven to Content Address Memory (CAM), which sends the data; depending whether it is one or zero from the comparator. If one, the cam doesn't specify the output data, instead of that the stored address location will be driven to the output, with an indication of match hit one which indicates the matching of data. If zero, no data matches so the mismatched data will be seen at the output of compression module with a match hit of zero. Since the values of compression module is applied to the decompression module as input which operates with same clock, reset, match hit signal.

Keywords: Comparator, Cam, Compression, Decompression

INTRODUCTION

Nowadays, Single Event Upsets (SEUs) altering digital circuits are becoming a bigger concern for memory applications. Previously there are more an error-detection methods for difference-set cyclic codes with majority logic decoding. Majority logic decodable codes are suitable for memory applications due to their capability to correct a large number of errors. However, they require a large decoding time that impacts memory performance. The proposed fault-detection method significantly reduces memory access time when there is no error in the data read. The technique uses the content addressable memory, which makes the area overhead minimal and keeps the extra power consumption low. Dictionary based schemes copy repetitive or redundant data into a lookup table (such as CAM)

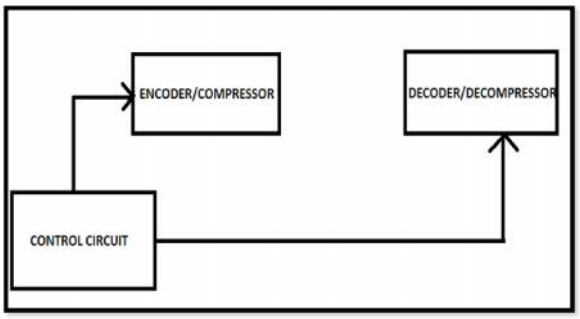
[1] M.Tech Student, Department of E.C.E., Chirala Engineering College, Ramapuram Beach Rd, Chirala, Andhra Pradesh 523157, India.

[2] Assistant Professor, Department of E.C.E., Chirala Engineering College, Ramapuram Beach Rd, Chirala, Andhra Pradesh 523157, India.

and output the dictionary address as a code to replace the data. The compression architecture is based around a block of CAM to realize the dictionary. This is necessary since the search operation must be done in parallel in all the entries in the dictionary to allow high and data-independent throughput. There are many applications that could benefit from economical large-capacity CAM's. Uses for large-capacity associative memories include "smart" databases, memory for Prolog computers, and artificial neural networks. Megabit look-up tables could make hardware dictionary processors commercially feasible, or they could increase resolution and system throughput for motion detection and image compression.

Lossless compression and decompression is the technique, which is used to perform the loss less data compression and decompression operations, which indeed used in the communication systems, to perform the low power data transmitting and receiving. First the clock is given which is used for transmitting and receiving the data within a single clock of positive edge. Input data is driven as a size of 32 bits, and the same output will appear as input using a dictionary technique. First the data is given to the comparator and the same stored in the memory, defined by the address, and then now the comparator performs the comparison between the given data and the locations in the memory. If the output from the comparator is one, indicates the matching content in the message, otherwise zero which indicates the no matching in the message. These outputs from the comparator is driven to Content Address Memory (CAM), which sends the data; depending whether it is one or zero from the comparator.

Figure 1: Model Daigram



If one, the cam doesn't specify the output data, instead of that the stored address location will be driven to the output, with an indication of match hit one which indicates the matching of data. If zero, no data matches so the mismatched data will be seen at the output of compression module with a match hit of zero. Since the values of compression module is applied to the decompression module as input which operates with same clock, reset, match hit signal. Whenever the match hit is high, the decompression module takes out the data, from the memory depending upon the address out pin from the compression module. These operations work for the entire date ,which shows the data in, compressed data, decompression output, which can analyzed by observing the input and the final output.

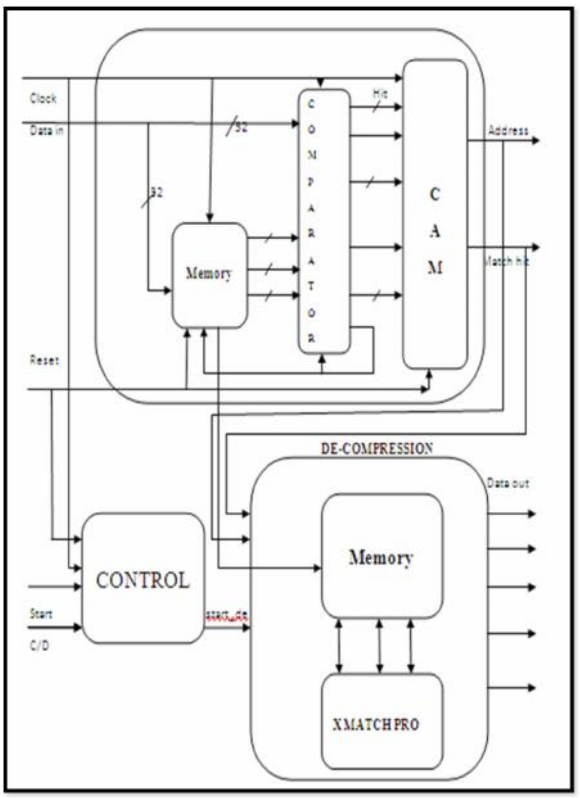
The number of bits in a CAM word is usually large, with existing implementations ranging from 36 to 144 bits. A typical CAM employs a table size ranging between a few hundred entries to 32 K entries, corresponding to an address space ranging from 7 bits to 15 bits. The length of the CAM varies with three possible values of 16, 32 or 64 tuples trading complexity for compression. The no. of tuples present in the dictionary has an important effect on compression.

LOSSLESS COMPRESSION

With the increase in silicon densities, it is becoming feasible for multiple compression systems to be implemented in parallel onto a single chip. A 32-BITsystem with distributed memory architecture is based on having multiple data compression and decompression engines working independently on different data at the same time. This data is stored in memory distributed to each processor. The objective of the project is to design a lossless parallel data compression system which operates in high-speed to achieve high compression rate. By using Parallel architecture of compressors, the data compression rates are significantly improved. Also inherent scalability of parallel architecture is possible. The main parts of the system are the two Xmatchpro based data compressors in parallel and the control blocks providing control signals for the Data compressors, allowing appropriate control of the routing of data into the system.

Each Data compressor can process four bytes of data into and from a block of data every clock cycle. The data entering the system needs to be clocked in at a rate of 4 n bytes every clock cycle, where n is the number of compressors in the system. This is to ensure that adequate data is present for all compressors to process rather than being in an idle state. To achieve higher compression rates using 32 bit compression/ decompression architecture with least increase in latency. Dictionary Methods try to replace a symbol or group of symbols by a dictionary location code. Some dictionary-based techniques use simple uniform binary codes to process the information supplied. Both software and hardware based dictionary models achieve good throughput and competitive compression.

Figure 2: Circuit Daigram for Cam Based Method for Reducing Memory Access Time



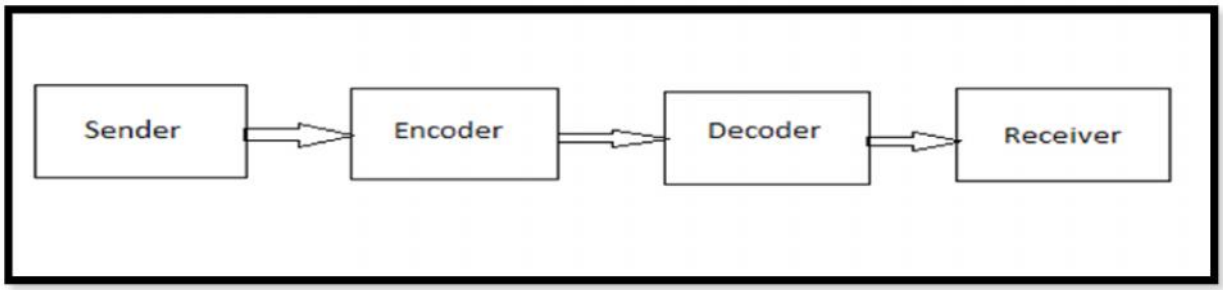
COMPRESSION TECHNIQUES

At present there is an insatiable demand for ever-greater bandwidth in communication networks and forever-greater storage capacity in computer system. This led to the need for an efficient compression technique. The compression is the process that is required either to reduce the volume of information to be transmitted text, fax and images or reduce the bandwidth that is required for its transmission speech, audio and video. The compression technique is first applied to the source information prior to its transmission.

FUNCTIONS OF LOSSLESS COMPRESSION

A sender can compress data before transmitting it and a receiver can decompress the data after

Figure 3: Basic Communication System



receiving it, thus effectively increasing the data rate of the communication channel. Lossless data compression is the process of encoding a body of data into a smaller body of data that can at a later time be uniquely decoded back to the original data.

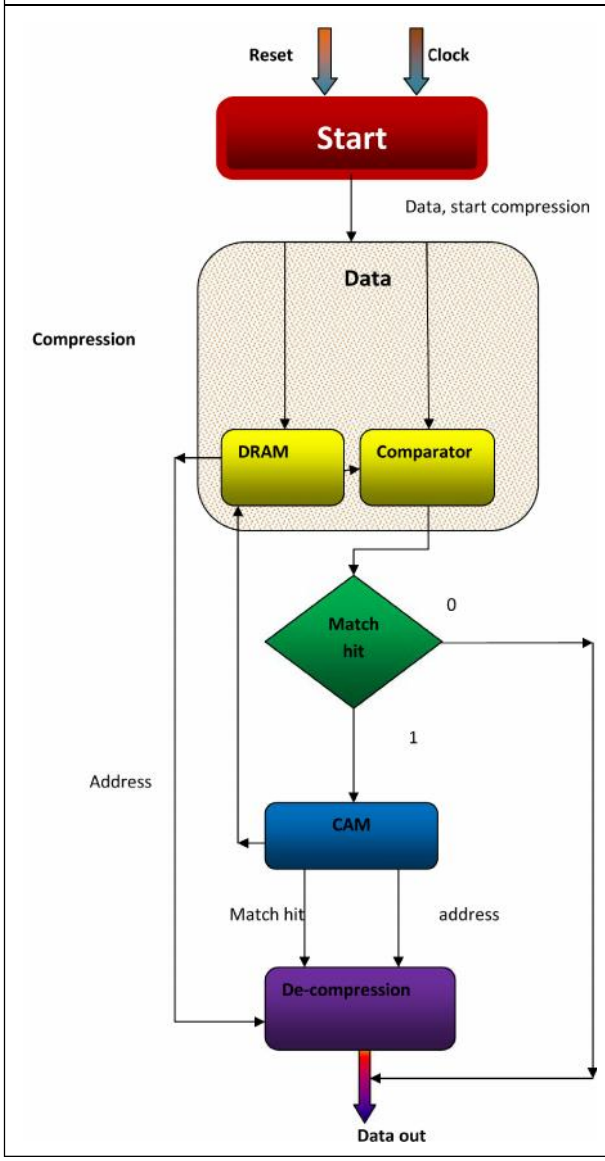
Lossless compression removes redundant information from the data while they are being transmitted or before they are stored in memory, and lossless decompression reintroduces the redundant information to recover fully the original data. In the same way, the data is compressed before it is stored and decompressed when it is retrieved, thus increasing the effective capacity of the storage device.

ALGORITHM FOR CAM

XMATCHPRO Algorithm

The Lossless Parallel Data Compression system designed uses the XMatchPro Algorithm. The XMatchPro algorithm uses a fixed-width dictionary of previously seen data and attempts to match the current data element with a match in the dictionary. It works by taking a 4-byte word and trying to match or partially match this word with past data. This past data is stored in a dictionary, which is constructed from a content addressable memory.As each entry is 4 bytes wide, several types of matches are possible. If all the bytes do

Figure 4: Flow Chart for Cam Based Method for Reducing Memory Access Time



not match with any data present in the dictionary they are transmitted with an additional miss bit. If all the bytes are matched then the match location and match type is coded and transmitted, this match is then moved to the front of the dictionary.

The dictionary is maintained using a move to front strategy whereby a new tuple is placed at the front of the dictionary while the rest move down one position. When the dictionary becomes full the tuple placed in the last position is discarded leaving space for a new one.With the increase in silicon densities, it is becoming feasible for multiple XMatchPros to be implemented in parallel onto a single chip. A parallel system with distributed memory architecture is based on having multiple data compression and decompression engines working independently on different data at the same time.

COMPRESSOR AND DECOMPRESSOR

Memory

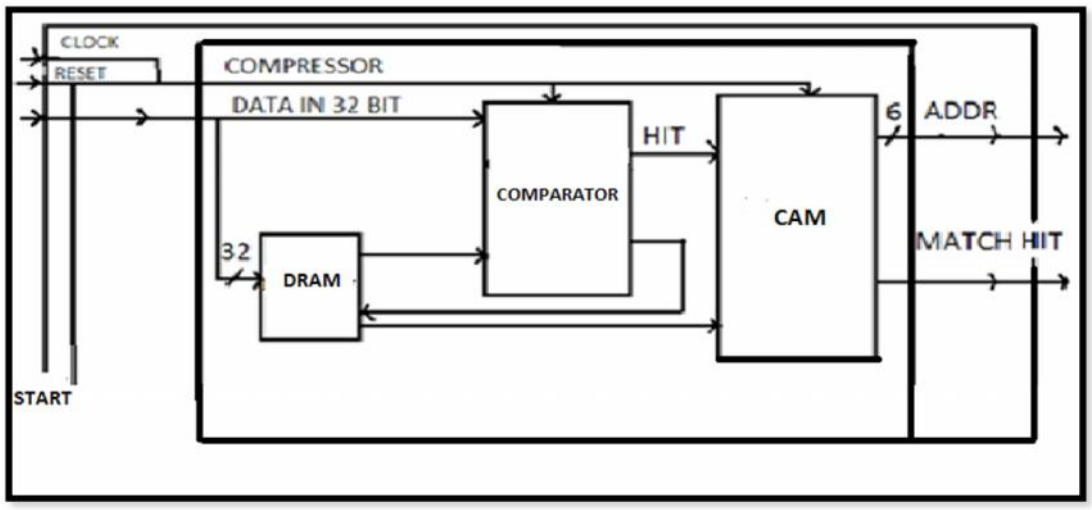
Memory circuits can largely be seperated into two major groups: dyanamic memories that store data for use in a computer system (such as the RAM in a PC); and static memories that store information that defines the operating state of a digital system. Dynamic memory circuits for computer systems have become very specialized, and they will be covered in a later lab. This exercise will present memory circuits that are used to store information about the operating state of a digital system.

DRAM

- Invented by IBM researcher (1T) but first marketed by Intel (3T)
- Store binary data as charge on capacitance
- In contrast to the SRAM, no constraints exist on the device size ratio

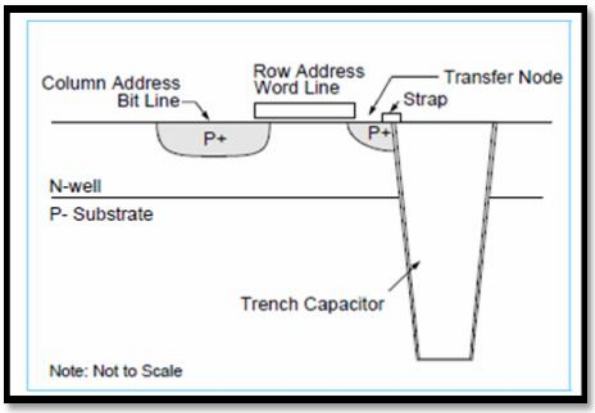
Dynamic Random Access Memory (DRAM) Integrated Circuits (ICs) have existed for more than twenty-five years. DRAMs evolved from the earliest 1-kilobit (Kb) generation to the recent 1-gigabit (Gb) generation through advances in both semiconductor process and circuit design technology. Tremendous advances in process technology have featuresize, permitting ever

Figure 5: Compressor for 32 Bits



higher levels of integration. These increases in integration havebeen accompanied by major improvements in component yield to ensure that overallprocess solutions remain cost-effective and competitive. Technology improvements, however, are not limited to semiconductor processing. Many of the advances in process technology have been accompanied or enabled by advances in circuit design technology. In most cases, advances in one have enabled advances in the other. Dynamic Random Access Memory (DRAM) is the most common kind of Random Access Memory (RAM) for personal computers and work stations. The network of electrically-charged points in which a computer stores quickly accessible data in the form of 0s and 1s is called memory. Random access means that the PC processor can access any part of the memory directly rather than having to proceed sequentially from some starting place.

Figure 6: DRAM



DRAM is dynamic in that, unlike static RAM (SRAM), it needs to have its storage cells refreshed or given a new electronic charge every few milliseconds. Static RAM does not need refreshing because it operates on the principle of moving current that is switched in one of two directions rather than a storage cell that holds a charge in place. DRAM: In 1T cell read process is destructive thus stored data must be regenerated every time they are read.

RESULTS

Figure 7: Rtl Schematic for Compressor

Figure 8: 32-Bit Compression Output Waveform

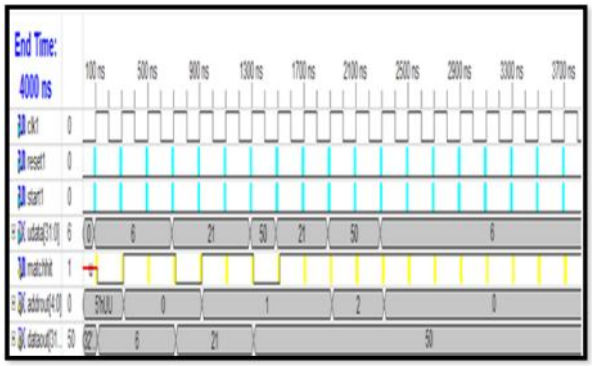


Figure 9: The RTL Schematic for De-compression Using Xilinx

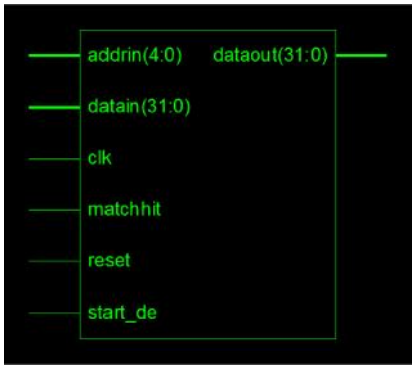


Figure 10: Data Decompression Output Waveform

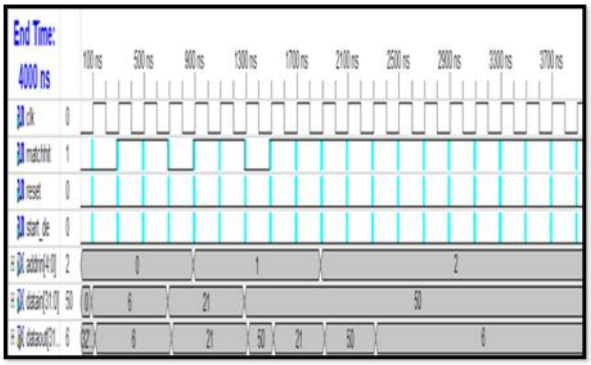
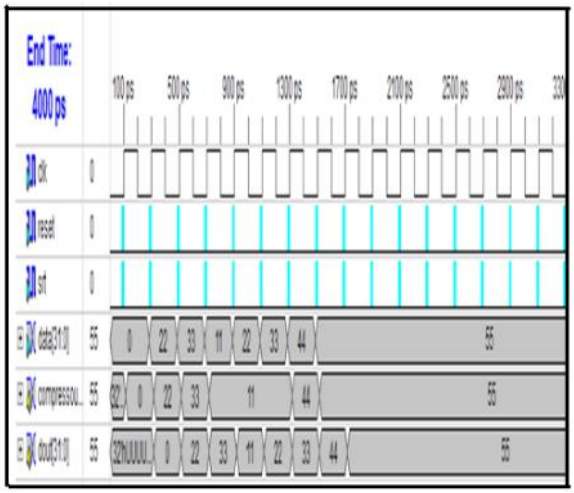


Figure 11: Compression / Decompression Output



CONCLUSION

In this paper it is presented a simple method to implement lossless data compression system which operates at high-speed to achieve high compression rate. By using architecture of compressors, the data compression rates are significantly improved and also inherent scalability of architecture is possible. The algorithm "XMATCHPRO" used in this project is efficient at compressing and the flexibility provided by using this technology is of great interest, since the chip can be adapted to the requirements of a particular application easily. The decompression (decoding) of the architecture is also made to obtain the original data. In future there may be a chance to obtain a better decoder.In future there is a chance to develop an architecture which should transmit the data effectively with less cost. CAM's are vast used in lookup table functions, network routers and cache controllers. Since basic lookups are performed over all the stored memory information there is a high power dissipation. In future there may be a chance to obtain a better decoder.In future there is a chance to develop asip with content addressable memory as inbuilt memory for utilising memory efficiently. The HDL is developed based on the VHDL language. The RTL is simulated and synthesized in the XILINX ISE.

REFERENCES

1. Balakrishnan K J and Touba N A (2006), "Improving Linear Test Data Compression", *IEEE Trans. Very Large Scale Integr. Syst.*, Vol. 14, No. 11, pp. 1227-1237.
2. Bayraktaroglu I and Orailoglu A (2001), "Test Volume and Application Time Reduction Through Scan Chain Concealment", in Proc. Des. Autom. Conf., pp. 151-155.
3. Forney G D (1973), "The Viterbi Algorithm", *Proc. IEEE*, Vol. 61, No. 3, pp. 268-278.
4. Koenemann B (1991), "LFSR-Coded Test Patterns for Scan Designs", in Proc. Eur. Test Conf., pp. 237-242.
5. Krishna C V, Jas A and Touba N A (2002), "Reducing Test Data Volume Using LFSR Reseeding with Seed Compression", in Proc. IEEE Int. Test Conf., October, pp. 321-330.
6. Rajski J, Tyszer J, Kassab M and Mukherjee N (2004), "Embedded Deterministic Test",

IEEE Trans. Comput.-Aided Des. Integr. Circuit Syst., Vol. 23, No. 5, pp. 776-792.

7. Touba N A (2006), "Survey of Test Vector Compression Techniques", *IEEE Des. Test Comput.*, Vol. 23, No. 4, pp. 294-303.
8. Wang Z, Fang H and Chakrabarty K (2009), "Deviation-Based LFSR Reseeding for Test-Data Compression", *IEEE Trans. Comput.-Aided Des. Integr. Circuit Syst.*, Vol. 28, No. 2, pp. 259-271.

International Journal of Engineering Research and Science & Technology
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijlerst@gmail.com or editor@ijerst.com
Website: www.ijerst.com