Lower Memory Utilization for Digital Data Acquisition and Transmission Systems.

M.C. Cheng, B.Eng. and Z. Abdul Halim, M.Sc.

University Sains Malaysia, School of Electrical and Electronic Engineering
14300, Nibong Tebal, SPS, Penang, Malaysia.

ABSTRACT

In most digital data acquisition systems, a digital sampler with high sampling rate may be employed to record signals before being analyzed by an embedded processor or stored into a memory buffer. Due to high sampling rate, the signals are recorded as redundant bits of ‘1’s or ‘0’s, which inadvertently increases the memory storage utilization. The data representation technique and memory cell design steps proposed in this paper are aimed at providing compression to the redundant binary bits. Specific purpose applications where the technique can be employed are digital logic analyzers, universal remote controls, and data loggers.

(Keywords: digital sampling, data acquisition, memory, universal remote, logic analyzer, compression, sampling rate, digital data storage)

INTRODUCTION

Digital signal sampling is a technique where a serial line of digital signals is continuously recorded by a sampler. Such applications can be found in digital data acquisition systems, data loggers, and digital transmission systems where the receiver has no proper information of the clock synchronization of the transmitter, and therefore, high sampling rates have to be employed to preserve the accurate representation of the original digital signal’s waveform. The proposed technique is a method to capture sampled data into memory systems without compromising the accurate representation of the digital signal during the reconstruction process.

Conventional techniques used by a data acquisition system directly capture the sampled logic values into the memory cells as a series of long logic representation of the signal [1],[2]. However, by dividing the memory array into a cluster of cells with a proper designed value of its depth, and applying the data representation technique to compress its data, the array of memory being occupied for digital storage can be minimized.

PROPOSED TECHNIQUE

a. Data representation of the Sampled Digital Data

Suppose a binary waveform is sampled at a rate which is high enough for the waveform reconstruction to produce an approximate identical binary waveform: a string of binary numbers are recorded.

The proposed technique of data representation for the string of binary numbers would be in the format of $x(y)$, where;

x = logic representation in logic 1 or logic 0.

y = the number (in decimal value) of sampling period (T_{SSR}) recorded for a specific binary number before the next transition to another binary number.

T_{SSR} = Uniform time period between the interval of signal sampling or signal reconstruction process. (unit: s). For example, data representation for the reconstructed signal in Figure 1 is 0(3), 1(2), 0(3), 1(3), ... , 1(2).

b. Memory Cell Architecture

Most memory array organization has an evenly distributed number of columns (also known as width, represented by W number of bits) and number of rows (also known as depth, represented by D number of bits) in a W x D matrix. Designing the architecture of the memory to fit the sampled binary representation, the memory array may consist clusters of *information cell* as shown in Figure 2(a).

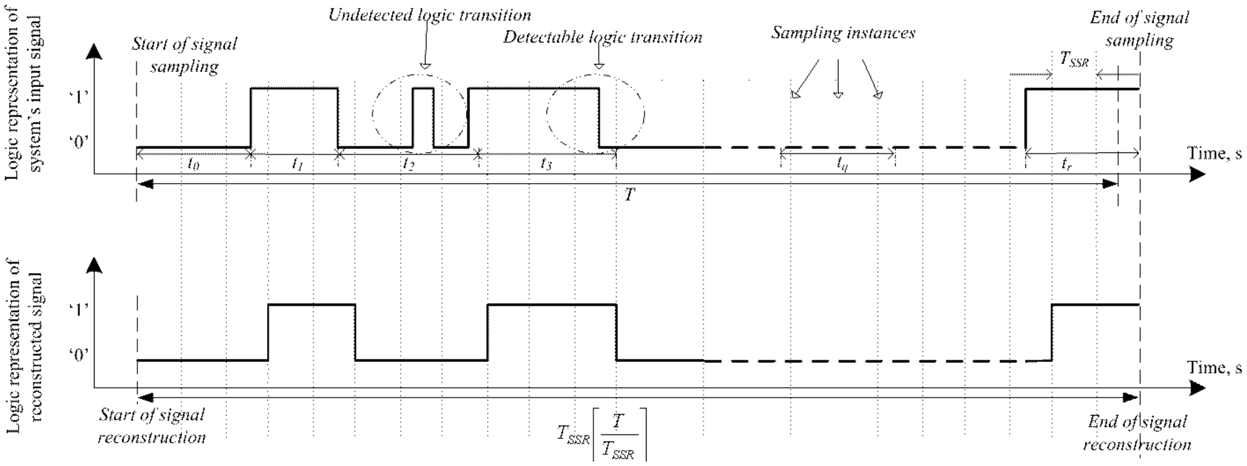

Figure 1: Logic Representation of an Input Signal and Corresponding Reconstructed Signal for a Binary Waveform.

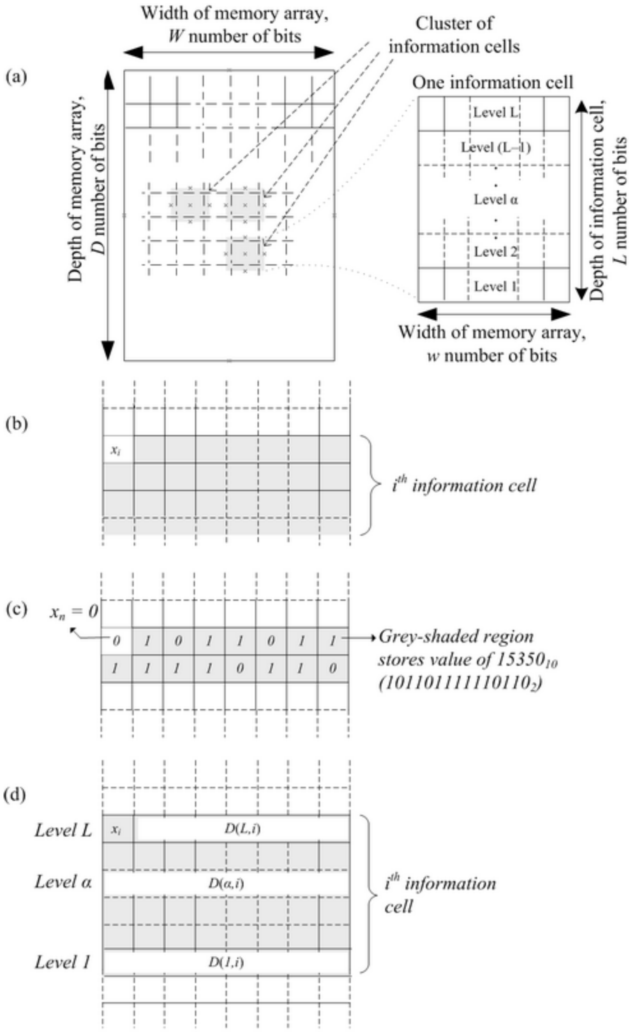

Figure 2: (a) Memory Array of *W*x*L* Matrix and its Cell Architecture. (b) Structure of the i^{th} Information Cell. (c) Example of Data 0(15350) being Stored into one Information Cell with *L*=2. (d) Memory Content of the Cell in Level-*α* , Represented in *D*(*α*,*i*) Format.

Each information cell can be used to fit information regarding the binary signal's waveform.

Each information cell is defined by a matrix of *w* x *L*, where;

L = finite integer representing the level of the information cell used in the array or equivalent to the depth of the cell represented in number of the bits. *L* = 1, 2, 3, …

w = finite integer representing the width of the information cell specified in number of bits. *w* = 1, 2 , 3, …

It is most convenient for digital system designers to choose integer *w* as equals to the width of a memory unit used in a system to store the sampled data, mathematically,

$$w = W \qquad \textbf{(1)}$$

Equation (1) will be used for the rest of the text. Typical values of *W* found in most memory unit for embedded systems are 8, 9, and 16.

c. Data Representation Technique

The proposed technique place major emphasis in reducing the memory utilization. For a string of binaries with data representation of:

$$x_0(y_0), x_1(y_1), x_2(y_2),....x_i(y_i),......, x_n(y_n) \qquad \textbf{(2)}$$

where, n = finite integer where the total number of information cells being used is equivalent to ($n+1$).

The most significant bit of level L for the i^{th} information cell of matrix array w x L in Figure 2(b) is used to store the logic level of bit x_i . While remaining bits in the memory array will be directly used to store the value y_i in either big-endian or little-endian format. The rest of the examples are based on big-endian format. The maximum value of y_i (ie. max(y_i)) that can be stored in the grey-shaded area is calculated as in (3).

$$\max(y_i) = 2^{wL-1} \qquad \textbf{(3)}$$

The example for w = 8, L = 2 and $x_i(y_i)$ = $0(15350_{10})$ is also shown in Figure 2(c). In this example only 2 bytes of the memory array has been used. However by conventional method of storing, a long string of 15350 bits of binary 0's would have been used.

d. Capturing and Retrieving Data from Information Cell

Data can be stored (in signal sampling process) or retrieved (in signal reconstruction process) from the cluster of information cells fitted in the memory array. These processes can be performed by a general purpose microcontroller or by a specifically designed embedded processor which is capable to perform the two algorithmic flow in Figure 3.

CONSTRAINT OF PROPOSED TECHNIQUE

From the earlier algorithmic flow chart of Figure 3(a), every transition of information cell for storage will occur whenever the input logic has changed or the boundary condition in (3) has been reached. If the logic levels of the binary waveform fluctuate frequently so that the time interval between a valid logic transition is less than the value of T_{SSR}/wL, the transition to the next information cell would have occurred more frequently. Hence, the memory utilization would have been higher by using the proposed technique than the conventional method of direct storage. This section is devoted to mathematically derive the constraint of the proposed technique as compared to conventional technique.

Referring back to Figure 1, assume an arbitrary waveform is recorded for a finite T length of time (unit: s), the total number of memory bits required to capture the binary waveform using conventional method of direct storage would be equivalent to the total number of sample counts conducted for the time interval of T. The total number of sample counts can be mathematically expressed as function $g(T,T_{SSR})$ given in (4).

$$g(T, T_{SSR}) = \lceil T / T_{SSR} \rceil \qquad \textbf{(4)}$$

where $\lceil z \rceil$ denotes the ceiling function [3], [4] of the positive real number z. Refer to Appendix for further elaborations.

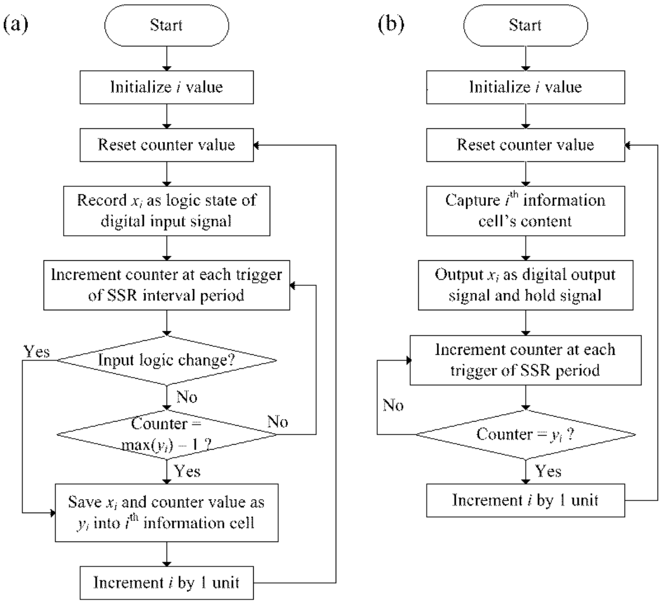

Figure 3(a): Algorithmic Flow for Capturing Digital Input Data to be Fitted into the Information Cell Iteratively **(b)** Algorithmic Flow for Retrieving Data from Information Cell and Reconstruction of the Desired Output's Signal.

Within the range of T, the number of information cells $k(t_q)$, that is required to capture a signal with a length of time t_q, can be computed with the equation in (5).

$$k(t_q) = \left\lceil \frac{g(t_q, T_{SSR})}{\max(y_i)} \right\rceil \qquad \textbf{(5)}$$

where t_q = time interval between the *detectable logic transition* of the logic level, and detectable logic transition is defined as change of logic level that can be detected by the digital signal sampler. (unit: s)

$q = 0, 1, 2, \ldots \ (q \le i)$

The number of bits used for storage is represented by function $h(w,L,t_q,T_{SSR})$ is equivalent to the number of bits inside each information cell times the total number of information cells being used, where the parameter L can be found with the method of later section. Mathematical expression is given in (6).

$$h(w, L, t_q, T_{SSR}) = wL\sum_{q=0}^{r} k(t_q) \qquad \textbf{(6)}$$

The proposed method can be said to utilize lower memory space than direct storage when the condition in (7) can be fulfilled by choosing appropriate value of L or smaller value of T_{SSR}.

$$h(w,L,t_q,T_{SSR}) < g(T,T_{SSR}) \qquad \textbf{(7)}$$

DESIGN OF INFORMATION CELL

The following sections are mathematical analysis to guide designers choosing the optimal value of the parameter L and manually encoding the binary values into each level of information cell for a known binary waveform.

a. Design of Parameter L for a Periodic Binary Waveform

In certain digital transmission systems (e.g. infrared remote control systems), a repetitive message signal can be sent by the system. Series of a repetitive message signal constructs a series of periodic signals being sent. The transmission process can be achieved by retrieving data from a known cluster of information cells to the controller of the transmitter. The value of L has to be designed and optimized to provide minimum size of memory array.

Defining an arbitrarily periodic signal with time period $T_{message}$ (unit: s), and substituting $T = T_{message}$ into (4), the values of $h(w,L,t_q,T_{SSR})$ with respect to variation of L can be calculated using (6) and tabulated or graphically plotted as $h(w,L,t_q,T_{SSR})$ vs. L. If the *exact value* of T_{SSR} is not specified in design specifications, the value T_{SSR} can be gauged at any reasonable range (lower value of T_{SSR} yields higher accuracy of signal reconstruction) to yield minimum value of $h(w,L,t_q,T_{SSR})$ at any values of L, or else this step can be skipped. The value L is chosen as the value that yields the minimum value of $h(w,L,t_q,T_{SSR})$.

b. Design of Parameter L for an Random Binary Waveform

Random binary waveform can be found in application where the system is used to sample unpredicted pattern of binary waveform (e.g. of application is a digital logic analyzer).

For the contemporary time, the proper design method of parameter L for an random binary waveform has not yet been fully been developed. However, it is wise to apply $L = 1$ as the binary waveform might fluctuate frequently, implicating in higher usage of memory space.

c. Formulae for Encoding

A known input logic for any time interval t_q can be mathematically encoded into the information cells. This can be done by calculating the each row's (w x 1) decimal representation of the memory matrix inside the information cells. An example of such application can be found in a remote control system, where the binary waveform of a message signal can be directly encoded into the memory array.

The information cell representing the logic level of x_i with an time interval of t_q can be fitted into one information cell if the condition (8) is met.

$$\max(y_i) > g(t_q, T_{SSR}) \qquad \textbf{(8)}$$

Referring to Figure 2(d), the level *α's* binary content of the i^{th} information cell can be represented in decimal form with the function represented by $D(\alpha,i)$. If condition (8) is met, the value of $D(\alpha,i)$ can be calculated easily with (9).

$$D(\alpha, i) = \left\lfloor \frac{g(t_q, T_{SSR})}{2^{w(\alpha-1)}} \right\rfloor \bmod 2^w \qquad \textbf{(9)}$$

where $\lfloor z \rfloor$ denotes the floor function [3], [4] of the real number *z*. Refer to Appendix for further elaborations and proofs.

However if condition (8) is not met, more than one information cells will be used and the y_i section of the initial $(k(t_q)-1)$ number of cells will be entirely filled with strings of binary '1's or mathematically encoded as the decimal values calculated from the equations in (10);

$$D(\alpha, i+s) = 2^w - 1 \qquad \text{for } \alpha = 1, 2, 3, \ldots, (L-1)$$

$$D(\alpha, i+s) = 2^{(w-1)} - 1 \qquad \text{for } \alpha = L \qquad \textbf{(10)}$$

where s = {0, 1, 2, ……, $(k(t_q)-2)$}.

The last information cell is to be encoded with binaries calculated using (11).

$$D(\alpha, \beta) = \left\lfloor \frac{g(t_q', T_{SSR})}{2^{w(\alpha-1)}} \right\rfloor \bmod 2^w \qquad \textbf{(11)}$$

where $t_q' = t_q - (k(t_q)-1) \cdot \max(y_i) \cdot T_{SSR}$ and $\beta = i + k(t_q) - 1$. Refer to Appendix for further proof.

For information cells of *L* = 1, or for *α* = 1, equations in (9) and (11) can be simplified into (12).

$$D(1, \beta') = g(t_q'', T_{SSR}) \bmod 2^w \qquad \textbf{(12)}$$

where the corresponding sets used for substitution are (β', t_q'') = (i , t_q) and (β', t_q'') = (β , t_q') .

DESIGN OF INFORMATION CELL USING A HARDWARE EXAMPLE OF A UNIVERSAL REMOTE CONTROL SYSTEM

An application example of a universal remote control system is chosen because it could demonstrate how the proposed technique could accommodate a multi-protocol embedded system, where message signals generated from the controller of infrared transmitter might not be generated by key logics, but a direct readout of data from the memory unit [5]. This eliminates the need for various hardware-built key logics for unknown protocols that might not be available to the universal remote control designer [6].

a. Design of Parameter L for Philips RC5 Protocol

The Philips RC5 protocol is available in [7]. A typical remote control key's message is a periodic binary waveform to be transmitted via the infrared transmitter after going through a modulator. Figure 4 is a typical message of 14 bits per message constructed using Philips RC5 remote control's protocol.

A high signal reconstruction rate would require greater memory usage, while a low signal reconstruction rate would result in poor timing accuracy of reconstructed waveform. Specifying a minimum of 97.5% accuracy relative to the minimum signal length of 899µs of the RC5 protocol, the maximum value of T_{SSR} equals to 22.475µs. Choosing several values for T_{SSR} , the $h(w,L,t_q,T_{SSR})$ function can be calculated by applying equation in (6) to the problem in Figure 4.

Results in Table I are calculated from;

$$h(8, L, t_q, T_{SSR}) = 8L\left(5\left\lceil \frac{\left\lceil \frac{899\mu s}{T_{SSR}} \right\rceil}{2^{8L-1}} \right\rceil + 11\left\lceil \frac{\left\lceil \frac{1789\mu s}{T_{SSR}} \right\rceil}{2^{8L-1}} \right\rceil + \left\lceil \frac{\left\lceil \frac{89727\mu s}{T_{SSR}} \right\rceil}{2^{8L-1}} \right\rceil\right)$$

To meet minimum requirement and least memory array to obtain best accuracy, the value of *L* = 2 at T_{SSR} = 5µs should be chosen. And for a higher precision waveform, *L* = 3 at T_{SSR} = 0.1µs can be selected by the designer.

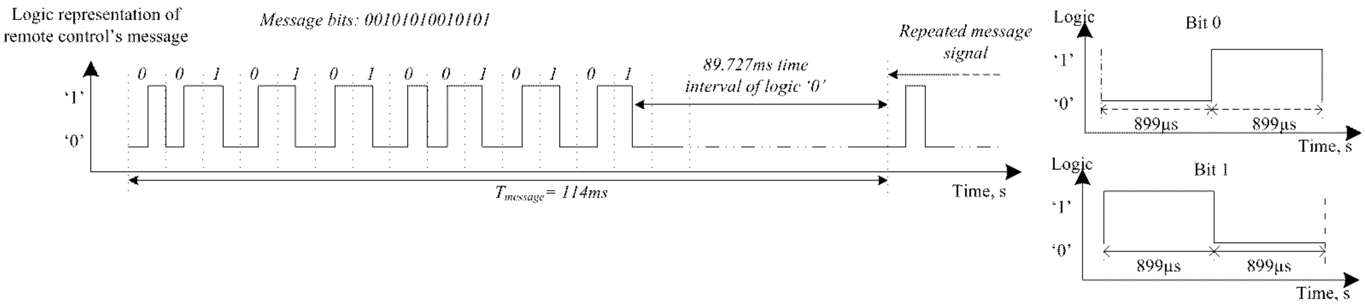

Figure 4: Logic Representation of a Typical Message Generated from Philips RC5 Remote Control.

It has to be noted that for higher values of L, the six sets of $h(8,L,t_q,T_{SSR})$ will merge to a asymptotic linear line. Employing a higher reconstruction rate will no longer bring significant improvement to accuracy, and depending on the applications, signal reconstruction frequency should be optimized with the power consumption of the particular digital system.

b. Efficiency Analysis

Choosing $L = 2$ at $T_{SSR} = 5\mu s$, only 34 bytes would have been used to store a message signal of 99.4438% timing waveform accuracy, relative to the smallest value of t_q (899 µs). However, using conventional methods, signal reconstruction at such accuracy for $T_{message} = 114ms$ would require 2850 bytes (approximately 84 times higher memory usage).

c. Translating Message Signal to Information Cells

Upon designing $L = 2$ and making decision that $T_{SSR} = 5\mu s$, the known waveform is ready to be encoded into the information cells by applying equation in (9), and the summarized result can be found in Table II.

FUTURE SCOPE AND CONCLUSION

The present proposed technique is undeniably capable to reduce redundancy of sampled logic values if the speed of the sampling clock applied is much higher than the speed of the input's signal variation.

TABLE I
NUMBER OF BYTES REQUIRED AS A FUNCTION OF L AND T_{SSR}

T_{SSR}	Accuracy	$h(8,L,t_q,T_{SSR})$					
		L=1	L=2	L=3	L=4	L=5	L=6
20	87.7753	416	272	408	544	680	816
10	98.8888	784	272	408	544	680	816
5	99.4438	1472	272	408	544	680	816
1	99.8888	7160	304	408	544	680	816
0.5	99.9444	14280	352	408	544	680	816
0.1	99.9999	71240	704	408	544	680	816

Unit for T_{SSR} is in μs, Accuracy is measured in percentage (%), unit for $h(8,L,t_q,T_{SSR})$ is in *number of bytes*.

TABLE II
ENCODED DATA FOR INFORMATION CELLS

q	t_q	Number of information cells	i	x_i	$D(2,i)$	$D(1,i)$
0	899	1	0	0	0	180
1	899	1	1	1	0	180
2	899	1	2	0	0	180
3	1789	1	3	1	1	102
4	1789	1	4	0	1	102
5	1789	1	5	1	1	102
6	1789	1	6	0	1	102
7	1789	1	7	1	1	102
8	1789	1	8	0	1	102
9	899	1	9	1	0	180
10	899	1	10	0	0	180
11	1789	1	11	1	1	102
12	1789	1	12	0	1	102
13	1789	1	13	1	1	102
14	1789	1	14	0	1	102
15	1789	1	15	1	1	102
16	89727	1	16	0	70	26

Unit for t_q is in μs. Encoded data in $D(2,i)$ and $D(1,i)$ are the decimal representation for the values in binary.

It is possible to eliminate this limitation issue. The suggested hint for improvement is to hybrid both conventional technique and the new technique, in other words, improving the algorithmic flow of Figure 3(a) and Figure 3(b).

APPENDIX

1) Ceiling function $\lceil z \rceil$:

Ceiling function for a real number *z* is equaled to the lowest value of integer greater than *z*. Optional computation formula for $z = A/B > 0$ can be given as;

$$\left\lceil A/B \right\rceil = 1 + (A - A \bmod B)/B$$

where mod is a function that can be described as; *A mod B* gives the remainder on division of *A* by *B*.

2) Floor function $\lfloor z \rfloor$:

Floor function for a real number *z* is equaled to the highest value of integer lower than *z*. Optional computation formula for $z = A/B > 0$ can be given as; $\left\lfloor A/B \right\rfloor = (A - A \bmod B)/B$

3) Proof of Equation (9):

For the information cell of level *α*, each unit of $D(\alpha,i)$ represents the maximum value than can be fitted into the level ($\alpha - 1$) row. Hence, the total number of samples counted for time interval of t_q that is represented by level *α* and above, is a multiple of $\left\lfloor \frac{g(tq, T_{SSR})}{2^{w(\alpha-1)}} \right\rfloor$.

Since, each row is capable to represent the value up to 2^{w-1}, therefore, the value of $D(\alpha,i)$ is given by the remainder of division on $\left\lfloor \frac{g(tq, T_{SSR})}{2^{w(\alpha-1)}} \right\rfloor$ by 2^{w}.

4) Proof of Equation (11):

Since the values of y_i for the earlier $(k(t_q) - 1)$ number of information cells has been fully loaded with the value of $\max(y_i)$, the remaining time section to be encoded into the last information cell using the equation in (9). This can be performed by substituting the variable t_q with t_q'. t_q' is the remaining time section that has not been encoded into the earlier information cells

REFERENCES

1. Schumacher, M. and LeCroy S.A. 1989. “Advantages of Long Memories”. 6th IEEE Instrumentation and Measurement Technology Conference, , Geneva, Switzerland. IMTC-89. pp 156-158.
2. Beenstra, K., K. Rangasayee, and A.L. Herrman. 2001. "Enhanced Embedded Logic Analyzer". United States Patent, Patent No.: US6247147B1.
3. Croft, H.T., K.J. Falconer, and R.K. Guy. 1991. *Unsolved Problems in Geometry*. Springer-Verlag: New York, NY.
4. Graham, R.L., D.E. Knuth, and O. Patashnik. 1994. “Integer Functions”. *Concrete Mathematics: A Foundation for Computer Science, 2nd ed.* Addison-Wesley: Reading, MA. pp. 67-101.
5. Van Dootingh. 1990. “ESPIRIT Home Systems Project”. *IEEE Transactions on Consumer Electronics*. 36(3):612- 618.
6. Sathyan, J. and A.R. Ramakrishnan. 2004. “A Unique Self-Contained Universal Remote Control”. *IEEE Transactions on Consumer Electronics*. 50(4):1151-1155.
7. Seerden, P. 2003. “Application Note: AN10210 Using the Philips 87LPC76x Microcontroller as a Remote Control Transmitter”. Philips Semiconductors. May 2003. http://www.nxp.com/acrobat_download/applicationnotes/AN10210_2.pdf

ABOUT THE AUTHORS

M.C. Cheng received his B.Eng. in Electronics Engineering from Universiti Sains Malaysia (USM) in April 2007. He received training in RTL development using FPGA from Intel, Penang in May 2006 and will be joining Intel as a System Validation Engineer. Mr. Cheng Mun Chun has completed an R&D project in developing a new embedded processor using FPGA targeting for the new universal remote control market, and his present focus is in optimizing memory utilization for data acquisition systems.

Z. Abdul Halim received her B.Sc. (Eng) and M.Sc. (Eng) in Electrical & Electronic Engineering in 1997 and 2000 from USM. In April 1997, she joined the Test Department of Applied Magnetic Malaysia Sdn. Bhd. as a product engineer. She has been a researcher and lecturer for USM, School of Electrical & Electronic Engineering since April 2004. Mdm. Zaini Abdul Halim's area of interest include digital circuit and hardware design using Microcontrollers, Microprocessor and FPGA. Her current activities have been focused on developing data acquisition system using FPGA.

SUGGESTED CITATION

Cheng, M.C. and Z.A. Halim. 2008. "Lower Memory Utilization for Digital Data Acquisition and Transmission Systems". *Pacific Journal of Science and Technology.* 9(2):363-370.

Pacific Journal of Science and Technology