

Dynamic Current-Mode Multi-Valued MOS Memory With Error Correction

Edward K.F. Lee and P. Glenn Gulak
VLSI Research Group

Department of Electrical Engineering, University of Toronto, Toronto, Canada M5S 1A4

Abstract

The design of a dynamic current-mode multiple-valued MOS memory with error correction used to increase the noise margins is presented in this paper. It is based on the current-copier and single-slope A/D techniques. The noise margin of the multiple-valued stored data is increased by storing and comparing the least significant bits of its binary representation during each refresh cycle.

1 Introduction

It is well known that multiple-valued logic (MVL) has potential advantages in both reduction of interconnection [1] and signal processing complexity [2]. Various different MVL circuit-implementation techniques have been investigated in the literature. However, logic elements alone are not sufficient to construct more complex systems such as a multi-valued computer. Therefore, voltage-mode multi-valued memory systems have been developed [3] [4] [5] to fulfill the requirements of multi-valued storage in such systems.

At the same time, submicron fabrication technologies and speed requirements are forcing designers to use reduced power supply voltages. Recently, current-mode CMOS MVL circuits were demonstrated to have the capability of operating at lower power supply voltages [6]. Furthermore, fast sensing speed for the stored value in the memory cell can be achieved using current-mode signals as demonstrated in [7].

In this paper, the design of a dynamic current-mode multi-valued random access memory is presented to satisfy the storage requirements for current-mode MVL circuits. To address this challenge, we first discuss the general approach to realize a dynamic multi-valued memory. Since the noise margin of a multi-valued memory reduces as the number of storage levels increases, an error correction technique will be introduced to improve the noise margins of the stored values. As an illustration of the techniques discussed, the design and implementation details of a current-mode multi-valued memory will be presented.

2 Generalized Dynamic Multi-Valued Memory

Analogous to the situation in a conventional binary DRAM, the stored values in a multi-valued memory system must be refreshed periodically and restored after the READ operation. Figure 1 shows a generalized

block diagram for the refresh or READ operation in a dynamic multi-valued memory system. It consists of three parts: a quantizer, a storage element and a logic block. When the storage element is to be refreshed, with switch S_1 closed (and S_2 open), the quantizer will quantize the value in the storage element and the logic block will determine the appropriate time to close the switch S_2 (and open S_1) so that the quantized value will be loaded back onto the storage element. Hence, the storage element is refreshed to the desired quantized level. It is important to prevent direct feedback between the output and input of the quantizer since the direct feedback may lead to unstable operation [8].

Various multi-valued memory designs result from the use of different types of quantization techniques, which, in term, ultimately determine the required time for refresh and implementation area. In general, a quantizer can be designed based on a combination of A/D and D/A converters. Table 1 shows different quantization techniques and their corresponding quantization time where N is the number of quantization levels.

A/D techniques	quantization time
Single-slope	$O(N)$
Successive approximation	$O(\log_2 N)$
Algorithmic	$O(\log_2 N)$
Flash	$O(1)$

Table 1: Different quantization techniques and their quantization time

In the literature, flash type quantization techniques have been applied to implement 4-valued RAMs and static latches [1] [5] [9] due to the fast quantization time which results. On the other hand, high resolution memories [3] [10] [11] usually utilize single-slope technique to reduce the implementation area requirement, and associated circuit complexity.

3 An Error-Correction Technique

Since the number of stored levels in a multi-valued memory is inherently large, it may have poor noise immunity and, hence, a large soft-error rate. To study the noise immunity situation, we propose the following definition for the noise margin to be associated

with multi-valued memories: as the allowable value of spurious variation or noise added to the stored value so that the composite value will not be quantized to an incorrect adjacent value. From this definition, the noise margin of a multi-valued memory is equal to almost one quantized level¹ as shown in Figure 2.

The noise margin can be increased with a simple error-correction technique [12], which requires storing and comparing one or more Least Significant Bits (LSBs) from the A/D conversion of the stored level. We consider the case with only one LSB as shown in Figure 3 in which a 1-bit latch is associated with each multiple-valued storage element. When an analogue level L_i is loaded into the memory, the corresponding LSB value (0) is also loaded to the 1-bit latch. During the period of quantization, the stored value is quantized by the A/D which generates a LSB. If the generated LSB and the value of the 1-bit latch are the same as in the case of no error, the stored value will be refreshed with a quantized value obtained by converting the digital output from the D/A converter directly to an analog value. When the generated LSB is different from the value of the 1-bit latch, an error has occurred. One can correct the error by subtracting (adding) the A/D output by 1 quantized level and refreshing the stored value with the result from the subtraction (addition). Alternatively, we could design a system that uses only addition rather than subtraction. Whether addition or subtraction is used depends on the expected direction of error accommodation and noise distribution. Addition or subtraction will each have different upper and lower noise margins as shown in Figure 3, but in either case the total noise margin is increased to two quantized levels. The region between the two dotted (dashed) lines represents the noise margin that results from subtracting (adding) the A/D output by 1 level. In general, if M LSB bits are used, the noise margin becomes 2^M quantized levels. However, the implementation area increases linearly as M increases. In more sophisticated systems, the number of latches can be reduced by representing a number of multi-valued memory cells with a block code.

4 Circuit Techniques

Based on the above techniques, a prototype dynamic current-mode multi-valued memory with a large number of stored levels ($N > 16$) has been designed. The single-slope technique was chosen to reduce the area requirement. A block diagram of the overall system is shown in Figure 4. The single-slope converter consists of a current comparator and a quantized-level generator (QLG) which produces all the quantized levels L_i monotonically as time increases. The output of the QLG also passes through a unit delay element which produces L_{i-1} at each time step. A 1-bit counter will be used to provide the LSB for error correction. When the output of the QLG goes higher than the stored value, the comparator will turn on and trigger the phase detector to generate a pulse. At the same time, the counter generates a LSB. A logic

¹The comparator used in the quantizer has finite gain and offset error that precisely define the actual noise margin.

unit will compare the LSB and the value of the latch. The memory will be refreshed with an error-corrected value using either the output directly from the QLG or the delayed output, as determined by the logic unit. The circuit will be ready for the next refresh cycle by resetting the QLG to the lowest quantized level. In the following sections, the circuit details for each of the building blocks are discussed.

4.1 Memory Cell

A current-mode storage element is one in which stored values are represented by discrete current levels. To implement a current-mode storage element a current copier [13] is used as illustrated in Figure 5. To store the current I , switches S_1 and S_2 are closed and the capacitor C will be charged to support a current equal to I . When S_1 and S_2 are opened, the transistor is capable of sinking a current I to the load with S_3 closed. Therefore, the current value I is related to the charge stored in the capacitor C and the transconductance of M_1 . The purpose of transistor M_2 is to increase the output resistance [14].

4.2 Quantized-Level Generator

The quantized-level generator required for the single slope A/D technique is realized using a discrete current mode integrator which consists of two current copiers as shown in Figure 6. When ϕ_1 is on, the reference current I_{ref} and the current stored $I(n-1)$ in the current copier 2 are added and loaded into the current copier 1. The resulting current $I(n)$ is loaded to current copier 2 while ϕ_2 is on. The current copiers are capable of sinking or sourcing the resulting current $I(n)$ to the loads during ϕ_3 , where

$$I(n) = I(n-1) + I_{ref} \quad n = 1, \dots, N$$

The quantized-level generator is reset when H is low.

4.3 Current Comparator

Figure 7 shows the schematic diagram of the current comparator. It consists of three current mirrors [6]. When I_1 is larger than I_2 , the voltage output V_o will go low. The required large voltage gain for a comparator is achieved by using cascode current mirrors, which have high output impedance. If higher gain is required, inverters can be cascade at the output of the comparator and the output signal will be buffered as well.

4.4 1-Bit Counter

Since the single slope A/D conversion technique is chosen for implementation, the least significant bits can be generated using a counter. Since only 1-LSB is used for error correction, a 1-bit counter will be used. The 1-bit counter consists of a D-flip flop and an inverter as shown in Figure 8. The counter will stop counting and the output gives the LSB when the input goes high.

4.5 Phase Detector

A phase detector, shown in Figure 9, is used to detect the rising edge of the current comparator output. The dynamic node D stores a logic 1 unless the comparator output, H and ϕ_1 go high. If the comparator output rises from low to high when ϕ_1 is low, M_1 and

Input signals		States
W	A	
0	X	Refresh
1	0	Load (analog current input)
1	1	Load (PWM input)

Table 2: States of the multi-valued memory shown in Figure 10

the inverter will latch the input to high. The output Z will go high when X goes high. Therefore the phase detector can detect the rising edge of the current comparator output after ϕ_1 goes low. The detector output will be a pulse with its width equal to the width of the signal X . The phase detector is reset by resetting the latch with H .

5 Implementation

Figure 10 shows the schematic diagram of the current mode multi-valued memory with the WRITE circuitry. A 1-bit counter is used to generate the LSB. When a value is written (as controlled by W) to the memory, the LSB bit is also loaded to the latch of the memory cell. The memory is controlled by a set of four clock signals - ϕ_1, ϕ_2, ϕ_3 and ϕ_4 . The signal H defines the time for each A/D conversion and the number of storage levels. When H and ϕ_1 goes high (refer to Figure 11), I_{ref} and the current stored in current copier 2 is added and loaded into current copier 1. The resulting current $I_r(n)$ is compared with the current output I_s from the storage element during ϕ_2 . At the same time, the value of the 1-bit counter will be compared with the value of the 1-bit latch using an exclusive-or gate which determines the value of X (either equal to ϕ_3 or ϕ_4). When $I_r(n)$ is larger than I_s , the comparator will go low and trigger the phase detector. If the value of the latch is the same as the value of the counter (no error), $I_r(n)$ will be loaded into current copier 2 and the phase detector will generate the refresh signal R at ϕ_4 . The storage element will be refreshed with $I_r(n)$. If the value of the latch is not equal to the value of the counter (an error), the phase detector will generate the refresh signal R at ϕ_3 and the current stored (equal to $I_r(n-1)$) in current copier 2 will be written into the storage element. The timing diagram for the case of an error is shown in Figure 12. The noise margins for this design will be defined by the region inside the two dotted lines as shown in Figure 3. As we can see, this circuit does not require an extra delay element. The delayed QLG output is achieved by changing the value of R from ϕ_4 to ϕ_3 .

The stored value can be loaded by applying a pulse-width-modulated (PWM) signal at terminal P or an external current input as controlled by the signal A . The states of the memory are summarized in Table 2. The timing diagrams for loading a stored value using these methods are shown in Figure 13 and Figure 14, respectively.

The circuit shown in Figure 10 is being fabricated in a $1.2\mu\text{m}$ double metal CMOS technology. The size of the memory cell and the refresh circuitry is $90\mu\text{m} \times 60\mu\text{m}$ (excluding the latch) and $850\mu\text{m} \times 220\mu\text{m}$, respectively.

6 Conclusion

The design of a current-mode dynamic multi-valued memory is presented which is based on current-copier and single-slope A/D techniques. An error-correction technique is used to increase the noise immunity of the stored value. The noise margins are increased by storing and comparing the LSBs of the stored values. With this error-correction technique, the period between refreshes can be increased allowing a system designer to multiplex many storage elements (current-copier cells) to one refresh circuit. Alternatively, the number of bits of precision can be increased for a given noise margin. Furthermore, this technique provides a design trade-off between noise margin and implementation area.

References

- [1] G. Epstein, G. Frieder, and D.C. Rine. "The Development Of Multi-Valued Logic As Related To Computer Science". *Computer*, 7:20-32, 1974.
- [2] M. Kameyama, T.T. Hanyu, and T. Higuchi. "Design And Implementation Of Quaternary NMOS Integrated Circuits For Pipeline Image Processing". *IEEE Journal of Solid State Circuits*, 12(1):20-27, 1987.
- [3] M. Horiguchi, et al, "An Experimental Large-Capacity Semiconductor File Memory Using 16-Levels/Cell Storage". *IEEE Journal of Solid State Circuits*, 23(1):27-33, Feb. 1988.
- [4] T. Furuyama, et al, "An Experimental 2-bit/Cell Storage DRAM for Macrocell or Memory-on-Logic Application". *IEEE Journal of Solid State Circuits*, 24(2):388-393, April 1989.
- [5] K.W. Current and M.E. Hurlston. "A Bi-Directional Current-Mode CMOS Multiple Valued Logic Memory Circuit". *Proceedings of the International Symposium on Multiple Valued Logic*, pages 196-202, May 1991.
- [6] D.A. Freitas and K.W. Current. "A Quaternary Logic Encoder-Decoder Circuit Design Using CMOS". *Proceedings of the International Symposium on Multiple Valued Logic*, pages 190-195, May 1983.
- [7] E. Seevinck. "Analog Interface Circuits for VLSI". In *Analogue IC design: the Current-Mode Approach*. Peter Peregrinus Ltd., 1990.
- [8] O. Ishizuka, Z. Tang, and H. Matsumoto. "On Design Of Multiple-Valued Static Random Access Memory". *Proceedings of the International Symposium on Multiple Valued Logic*, pages 11-17, May 1990.

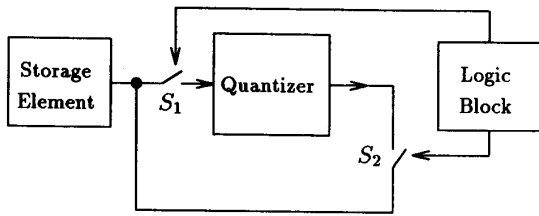


Figure 1: Generalized block diagram for dynamic multi-valued memory

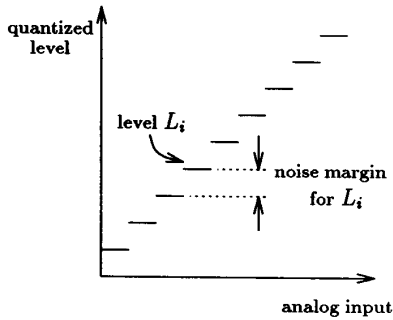


Figure 2: Noise margin for a multi-valued memory

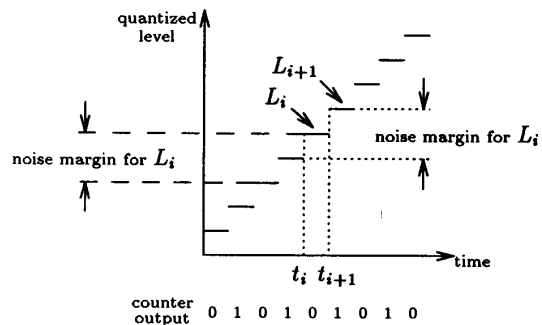


Figure 3: Noise margins with error correction

- [9] D.A. Rich. "A Survey of Multivalued Memories". *IEEE Transactions on Computers*, 35(2):99-106, Feb. 1986.
- [10] B. Hochet. "Multivalued MOS Memory for Variable-Synapse Neural Networks". *Electronics Letters*, 25(10):669-670, May 1989.
- [11] B. Hochet, et al, "Implementation of a Learning Kohonen Neuron Based on a New Multilevel Storage Technique". *IEEE Journal of Solid State Circuits*, 26(3):262-267, Mar. 1991.
- [12] Edward K.F. Lee and P.G. Gulak. "Error Correction Technique for Multi-Valued MOS Memory". *Electronics Letters*, 27(15):1321-1323, July 1991.
- [13] S.J. Daubert, D. Vallancourt, and Y.P. Tsiividis. "Current Copier Cells". *Electronics Letters*, 24(25):1560-1562, Dec. 1988.
- [14] E.A. Vittoz and G. Wegmann. "Dynamic Current Mirrors". In *Analogue IC Design: the Current-Mode Approach*. Peter Peregrinus Ltd., 1990.

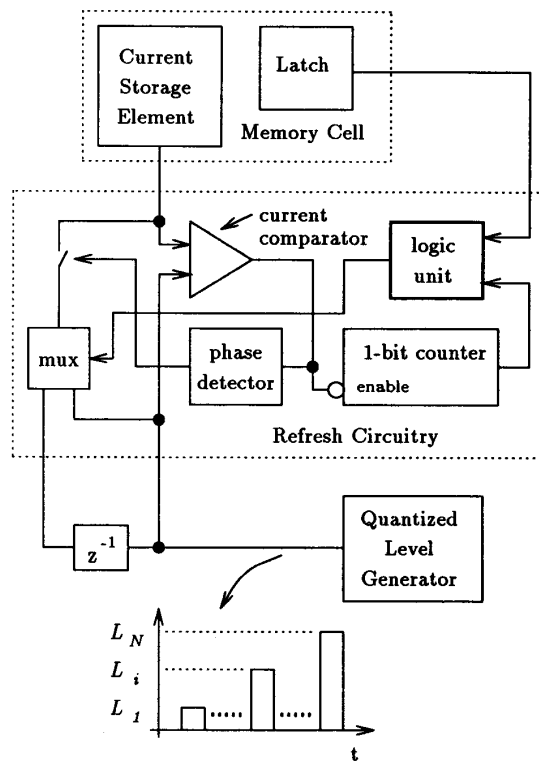


Figure 4: Block diagram of the current-mode multi-valued memory with error correction

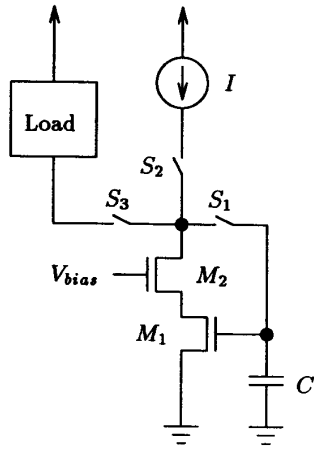


Figure 5: Schematic diagram of the storage element (current-copier cell)

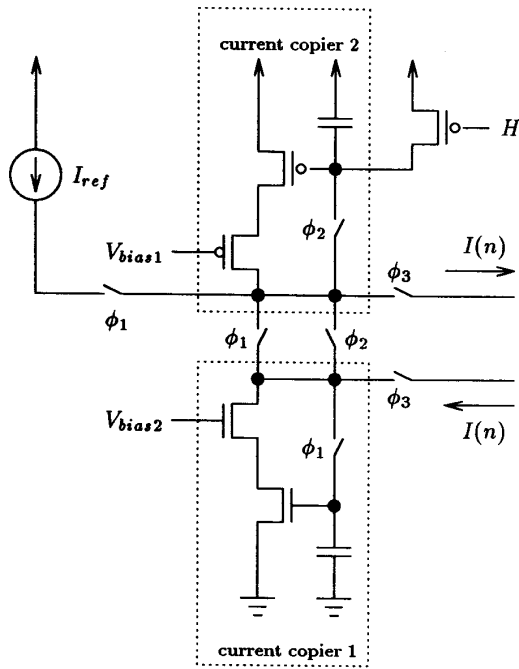


Figure 6: Schematic diagram of the quantized-level generator

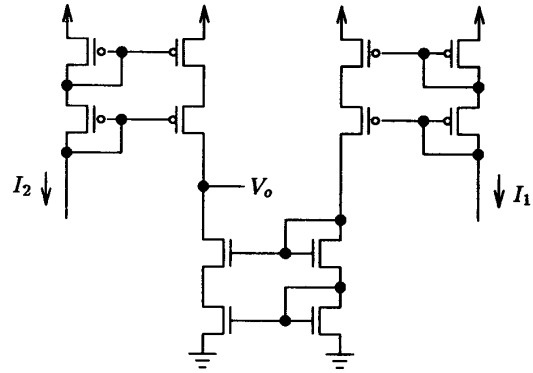


Figure 7: Schematic diagram of the current comparator

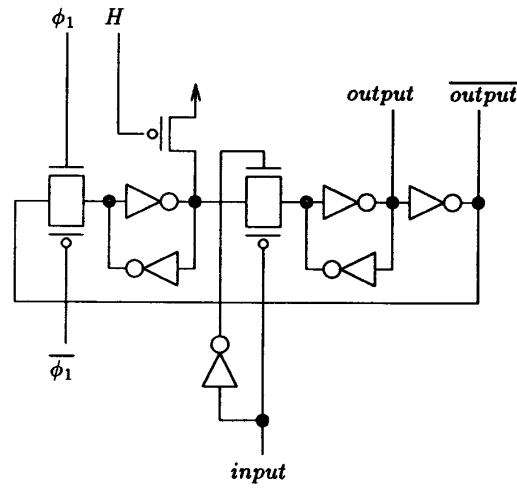


Figure 8: Schematic diagram of the 1-bit counter

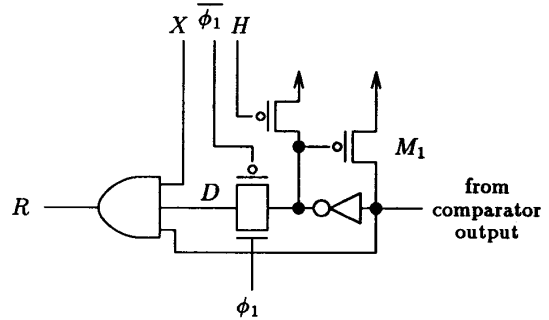


Figure 9: Schematic diagram of the phase detector

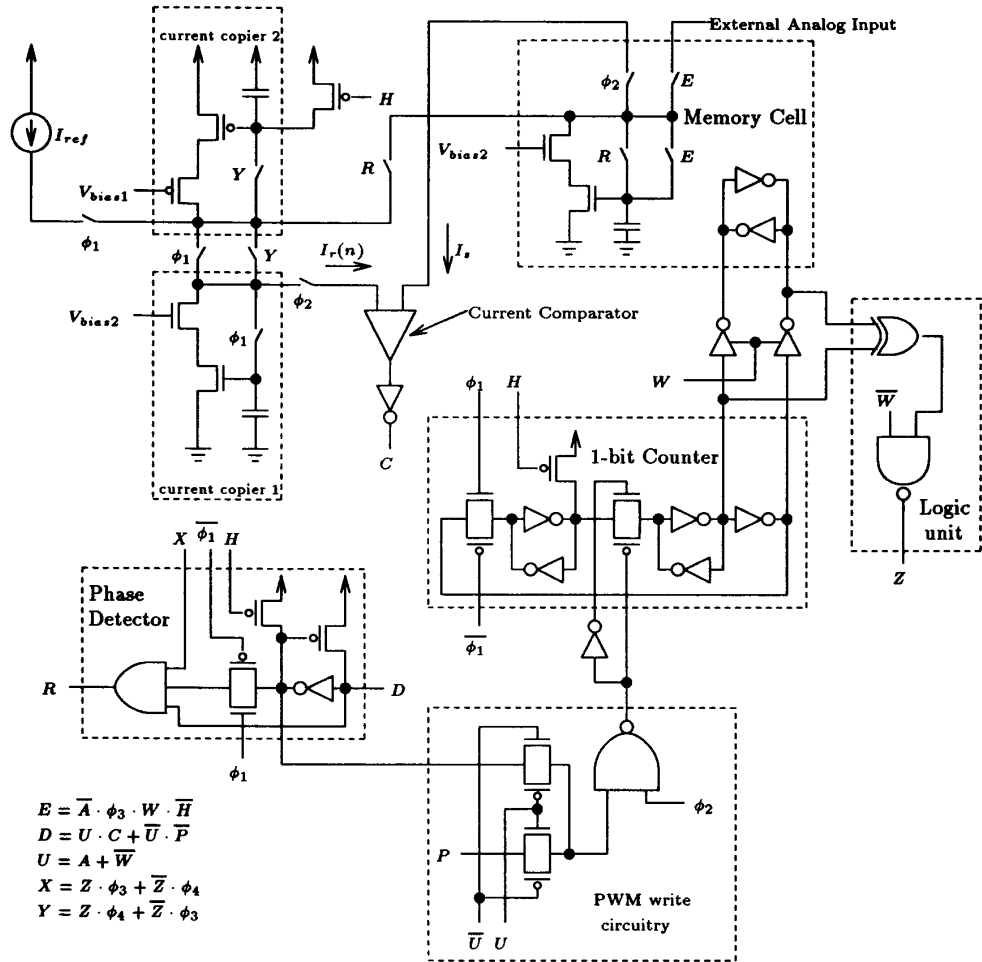


Figure 10: Schematic diagram of the current-mode multi-valued memory with error correction

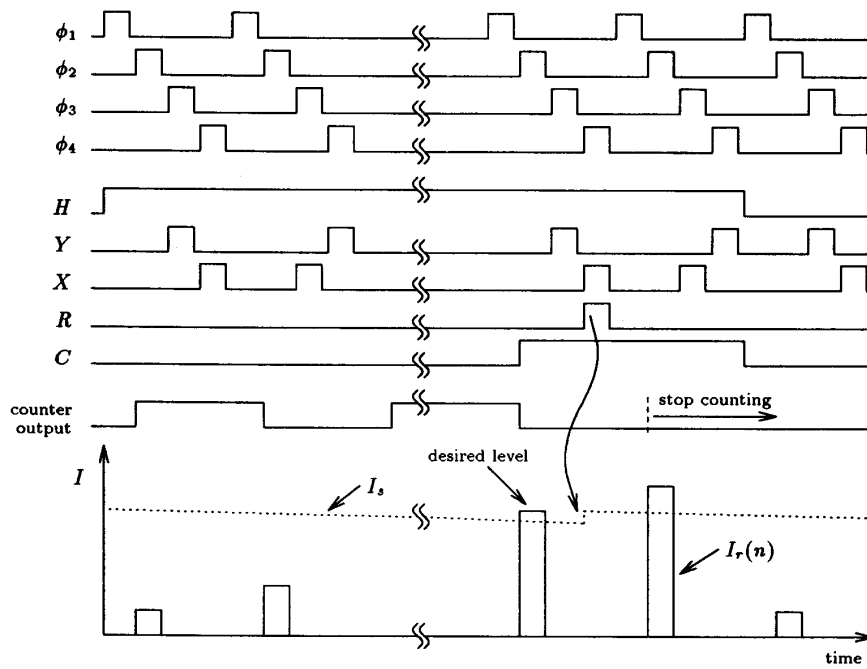


Figure 11: Timing diagram for the refresh operation with the case of no error and latch value preset to 0 (I_s is a in fact discrete signal - only the envelope is illustrated)

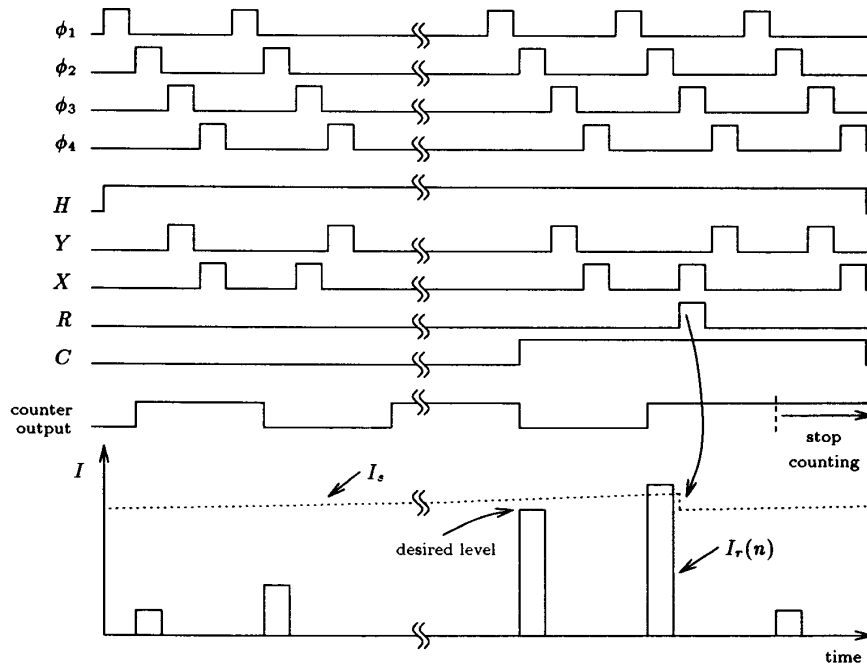


Figure 12: Timing diagram for the refresh operation with the case of an error and latch value preset to 0

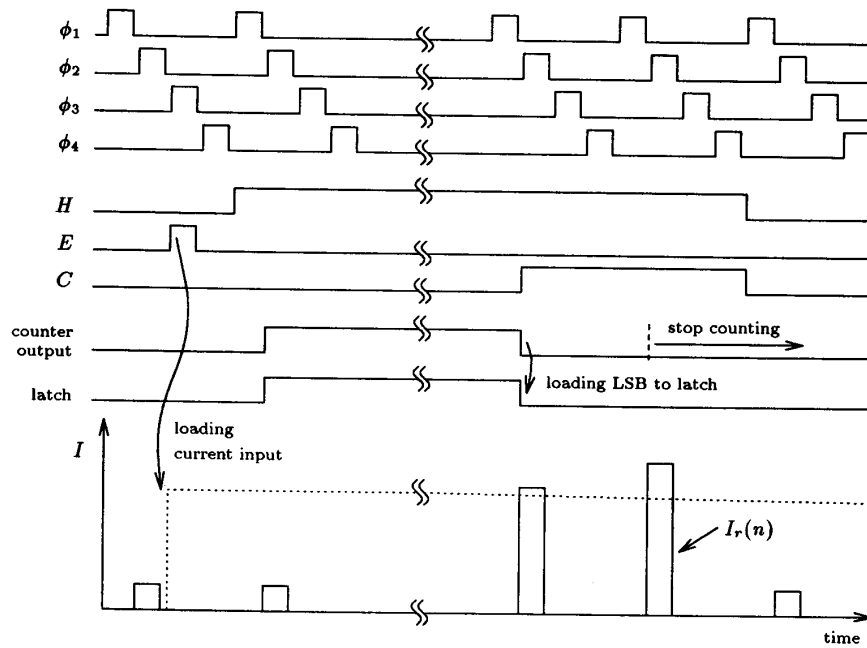


Figure 13: Timing diagram for loading a stored value from an external current input ($W = 1$ and $A = 0$)

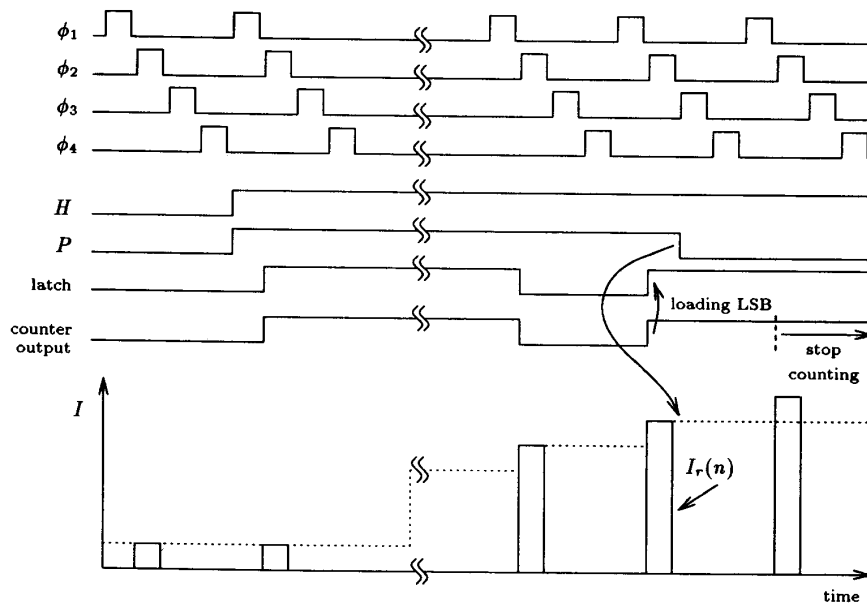


Figure 14: Timing diagram for loading a stored value via pulse-width- modulation ($W = 1$ and $A = 1$)