Latching Scheme in Multiple-Valued Recharged Logic

Omid Mirmotahari, Yngvar Berg

Abstract—In this paper we present several different proposals for implementing a multiple-valued (MV) semi-floating-gate (SFG) latches. This paper aims to illustrate the advantages and disadvantages of each approach. Measurements from a fabricated chip at a 0.6µm CUP process is included for verifying the detailed arguments.

Index Terms—Floating-gate, Multiple-Valued Logic, Latch.

I. INTRODUCTION

Multiple-valued logic has in the last few decades been proposed as a possible alternative to binary logic. Whereas binary logic is limited to only two states, "true" and "false", multiple-valued logic (MVL) replaces these with finite or infinite number of values. An MVL system is defined as a system operating on a higher radix than two [1]. A radix-n set has n elements, {0, 1, ..., n-1}. The feasibility of MVL depends on the availability of devices constructed for MVL operations [2]. The devices should be able to switch between the different logical levels, and preferably be less complex than their binary counterparts. We prefer voltage-mode due to its independent signal response.

Multiple-input floating-gate (FG) transistors can be used to simplify the design of multiple-valued logic [3]. The initial charge on the floating-gates may vary significantly and therefore impose substantial inaccuracy unless we apply some form of initialization. Some work on floating-gate reset strategies has been presented [4]. By recharging the semi-floating-gates (SFG), we not only avoid the problems linked to programming or initializing of the floating-gates, but we convert the non-volatile floating-gates to semi-floating-gates. Thus the control of the actual floating-gate charges in terms of predictable long-term charge restoration becomes easier. The SFG is not influenced by a random floating-gate charge distortion, due to a periodic or frequent charge restoration or reset. A local recharge transistor temporarily connecting the output to the floating-gate of a gate accomplishes the recharge of the SFGs. The potential improvement of recharge MVL is briefly discussed in [5].

Multiple-valued logic requires some kind of memory element. A latch is usually a dynamic memory element, and can be used to construct static memories. A multiple-input multiple-valued latch has been proposed [6], based on basic multiple-valued recharge logic structures [5].

In section V we present the basic latching scheme proposed in [6], and in section VI we propose different D-latch structures (one of them using the latch in section V). We then pinpoint the strengths and weaknesses of the different designs. Measurements are achieved from a chip fabricated at AMS 0.6µm (CUP) process.

II. THE MULTIPLE-VALUED INVERTER

Multiple-valued circuits must be able to distinguish between different logic levels. We can make use of the floating-gate, as in neuromorphic engineering [7], to assign different input-signals different weights. With the weighted sum and the threshold operation we are able to compute both arithmetical and logical functions. And, in order to have an output that can toggle different logical voltage levels, a circuit that can output a linear signal with respect to the input, more precisely satisfying $V_{out} = V_{dd} - V_{in}$, is preferred. A binary floating-gate inverter is modified to become multiple-valued, as shown in Fig. 1 [8]. The multiple-valued inverter is distinguished from the binary FG inverter, by capacitive coupling the output to the floating-gate, thus by adding a feedback capacitor, Cf. By adjusting the Cf /Ci ratio, we can control the gain. To obtain a linear output, which fulfills $V_{out} = V_{dd} - V_{in}$, the gain must be equal to 1, i.e. unity-gain.

Although parasitic capacitances, i.e. Cgd, are small, they could affect the gain. The relationship of the parasitic...
capacitance $C_{gd}$ to the gain depends on the size of the transistors and capacitors. If minimum sized transistors and large capacitors are used, the $C_{gd}$ would be quite small relative to $C_i$. Under the opposite scenario, the $C_{gd}$ would become so large, relative to $C_i$, that it might assume the role of $C_f$. Although this indicates that it might be possible to omit the $C_f$, we know from the non-linear characterization of the $C_{gd}$ that it is not a recommended approach.

![Diagram of a floating-gate binary inverter modified to become a multiple-valued floating-gate inverter by adding a capacitor, $C_f$, between the output and the floating-gate.](image)

Fig. 1. The floating-gate binary is modified to become a multiple-valued floating-gate inverter by adding a capacitor, $C_f$, between the output and the floating-gate.

Fig. 2 shows the measurement of a single-input multiple-valued inverter, and its respective gain appears in Fig. 3. They both clearly demonstrate the non-linear properties of the output signal near the rails. This non-linearity is due to the saturation range of the transistors.

To calculate the number of levels (radix) a few practical matters one must be considered, and those are as follows:

$$\text{Linear Voltage Range} = V_{satp} - V_{satn}$$ (1)

where $V_{satp}$ and $V_{satn}$ are the voltages where the pMOS and nMOS transistors enter saturation, respectively. Note that the floating-gate is always $V_{dd}/2$. Using simple formulas for the two saturation voltages:

$$V_{satn} = V_{gs} - V_{tn} = V_{dd}/2 - V_t$$ (2)

Which also gives $V_{satp} = V_{dd}/2 + V_t$. Furthermore the the calculation of the linear voltage range becomes:

$$\text{Linear Voltage Range} = V_{dd}/2 + V_t - (V_{dd}/2 - V_t) = 2V_t$$ (3)

Now that we have in place the linear voltage range and its dependency on the supply voltage, we look at how to divide the voltage range into logical levels. Somewhere in a multiple-valued design a binary inverter may be needed, e.g. in a converter. Therefore the smallest voltage, $V_{trig}$, which is large enough to make a binary inverter switch, sets the highest boundary for how many logical levels we can denote for the linear voltage range. We can say that the maximum number of logical levels (radix) is given by

$$\text{max (radix)} = \frac{\text{The voltage range which fulful a unity-gain (2Vt)}}{\text{The voltage needed for a binary inverter to switch (Vtrig)}}$$
Fig. 2. Measurement of a single-input multiple-valued inverter. The capacitor values are $C_i = 10 \, \text{fF}$ and $C_f \approx 9 \, \text{fF}$, while the transistor sizes are $1.0/1.2\mu$ for nMOS and $2.5/1.2\mu$ for pMOS.

The $V_{\text{trig}}$ can be expressed as $V_{\text{dd}}/A$, where $A$ is the stage gain. Therefore the maximum radix can be computed with

$$\max(\text{radix}) = \frac{2V_{\text{t}}}{V_{\text{dd}}} \times A$$

Throughout this paper we primarily use a radix-8 input signal with a 2.0 V supply voltage.

III. RECHARGING THE FLOATING-GATE

All FG circuits need to be initialized, either once initially or frequently. The once and for all initialization is synonymous with programming. By recharging the SFG frequently we avoid problems with any leakage currents and random or undesired disturbance of the floating-gate charges. The reset or recharge scheme presented in this paper can be used to overcome some of the problems associated with floating-gate circuit design. The recharge condition is different than the reset in clocked-Neuron-MOS logic proposed by Kotami et.al. [4]. When resetting or recharging a gate the inputs are recharged simultaneously and not set to a reference voltage, normally $V_{\text{ss}}$ or $V_{\text{dd}}$. During recharging the gates are short-circuited, and the output and the semi-floating-gate of a logic gate is forced to $V_{\text{dd}}/2$. The clocked-Neuron-MOS logic was initially proposed for binary logic gates and for threshold gates [9]. The recharge scheme is similar to biasing of single-ended auto-zeroing comparators which have been used in high-speed flash AD converters. The main purpose of the recharge scheme is to initialize or recharge the semi-floating-gates to an equilibrium state, which can be utilized to yield fast binary and multiple-valued signal...
processing. In addition, we may reduce the effect of mismatches, especially transistor mismatches, and power supply noise. The recharge scheme provides a simple, fast and accurate recharge to the equilibrium state for all gates, regardless of logical depth.

Fig. 3. The gain of a single-input multiple-valued inverter. The supply voltage is 2.0V. The plot clearly shows the non-linear characterization near the rails.

IV. SYNCHRONIZATION AND LATCHING

In the previous section the basic MVL recharge inverters where presented. Different circuit designs have been presented in [10]–[12]. As more circuits are developed, a synchronization mechanism is needed. The synchronization is quite a natural discussion here in this paper, due to the fact that these MVL recharge circuits operate with clock signal for recharging. When synchronization is to be implemented in designs, it necessitates control of the timing of signal propagations. Sometimes signals should be delayed or stored for a small amount of time. A straightforward approach would be by exploiting the C2MOS concept. A design that combines MVL with C2MOS is proposed in [13]. This latch can be modified to serve as a multiple-valued latch by adding a recharge transistor. The recharge signal and the two clock signals make the circuit rather complex in terms of clock signaling. In the next section a less complex clocking strategy is demonstrated and a MVL latch is constructed without the conventionally clocking strategies.
V. A NOVEL LATCHING SCHEME

First we briefly present the latching concept with binary recharge circuits, and then show how the same concept can be applied to the multiple-valued recharge circuits. Two cascaded binary recharge inverters are shown in Fig. 4. Both inverters operate the recharge switch at a $\phi$ clock period; hence, when $\phi$ is “high” the inputs, semi-floating-gates and outputs are all $V_{dd}/2$. The signal propagation through the design is illustrated in Fig 5.

The output, out1, inverts the input, In. The output, out2, inverts out1, thus outputting the same signal as the input, In. Not surprisingly, this design works like a buffer. A circuit as simple as a buffer is used to construct a latch by applying the inverted recharge signal to the second inverter. Fig. 6 shows the same design, but notice the replaced recharge signal at the second inverter. The signal propagation is as indicated in Fig. 7, and we emphasize that the latching is actually performed by the second inverter. When the first inverter is recharging, the second inverter is evaluating.

Fig. 4. The design shows a buffer implementation, intended as an illustration for the latch description provided in the text.

Fig. 5. A graphical description of the signal propagation through the design in Fig. 4.
Fig. 6. A proposed use of the recharge signal to construct a binary latch with only two binary recharge inverters.

We cannot determine the first evaluation period of the second inverter (latch) since it is actually evaluating a previous change at output out1. When φ is “on”, the first inverter is recharging and thus its output, out1, is Vdd/2. The recharge period is followed by the evaluation period, when φ is “off”. The input In is changed from Vdd/2 to Vdd. The voltage change ΔV = Vdd/2 causes the output out1 to become Vdd/2 - ΔV = Vss. The second inverter is recharging during this period, while the input voltage is Vss. Notice that the output out2 is Vdd/2. When the first inverter switches over to recharging, there will be a positive voltage change at the output out1, from Vss to Vdd/2. Simultaneously the same voltage step (+Vdd/2) is being evaluated by the second inverter, and thus the output out2 is Vdd/2 - (+Vdd/2) = 0 (Vss). Looking closer we find that output out2 behaves as output out1, differentiated only by a phase-shift. Thus the second inverter has latched the output-signal out1.

Fig. 7. Signal propagation through the latch design in Fig. 6.

The multiple-valued recharge latch can be constructed along the same lines, replacing the binary inverters with the multiple-valued recharge inverters. The new latch design is shown in Fig 8.
Fig. 8. The multiple-valued latch constructed in the same manner as in Fig. 6, but only with multiple-valued inverters instead of binary inverters.

The signal propagation closely resembles the one described for the binary inverters. The difference here is that $\Delta V$ is not necessarily $V_{dd}/2$, but a voltage change that is significant enough to toggle between the different logical levels, i.e. a voltage step. The generalized signal propagation for the multiple-valued recharge latch is demonstrated. The first inverter receives a voltage change, $\Delta V$, at the input, $I_n$, after a recharge period. This voltage change causes the output $out_1$, which was initially $V_{dd}/2$, to become $V_{dd}/2 - \Delta V$. Since the second inverter is recharging, the output $out_2$ becomes $V_{dd}/2$. What we have so far is thus:

\[
\begin{align*}
I_n &= V_{dd}/2 + \Delta V \\
out_1 &= V_{dd}/2 - \Delta V \\
out_2 &= V_{dd}/2 \\
\varphi &= V_{ss} \\
\bar{\varphi} &= V_{dd}
\end{align*}
\]

The situation where the first inverter is recharging, thus causing $out_1$ to become $V_{dd}/2$. The effect at $out_1$ is a voltage change, $+\Delta V$, which results in a voltage change, $-\Delta V$, at $out_2$, so that it becomes $V_{dd}/2 - \Delta V$. As the measurements in Fig. 9 demonstrate, the output of the second inverter, $out_2$, is similar to the input, $I_n$, inverted and differentiated only by a phase-shift. This phase shift is not considered to be a error or a problem, but merely a question of synchronization, can draw parallels to pipeline systems. In the third plot of Fig. 9, the output is manipulated, thus phase-shifted and inverted, so that it becomes easier to see the output in relation to the input.

We believe it is important to stress that the latching is a direct result of the differential use of the recharge clock signals, and hence every recharge inverter, either binary or multiple-valued, can potentially be a latch (with a few exceptions, e.g. the output of the latch may be logically inverted. It may therefore at worst need an additional inverter, while on the other hand removes an inverter). Therefore there is great motivation for finding both the best recharge switches and signals. Conventionally transmission gate is used, but another possible design could be created relying on an nMOS and a pMOS switch alternation as shown in Fig. 10. The strategy is to assert an nMOS switch at the inverters intended to operate on $\varphi$ and a pMOS switch at those intended to operate on $\bar{\varphi}$. When the same recharge signal is applied to all inverters, those with nMOS are recharging while those with pMOS are evaluating.

When it comes to recharge clocking methodologies there are several factors one must be aware of. Below we discuss different clocking schemes and pinpoint their advantages and disadvantages. The first approach is to use the single-phase clocking scheme. This scheme suffers from the well-known phenomenon, clock skew. For our circuits we could encounter situations when both inverters are recharging, or evaluating. We therefore find the single-phase clocking scheme only appropriate for the design shown in Fig. 10, since only one clock signal is required for latching (assuming that the paths to both recharge switches are approximately equivalent). The measurements in Fig. 11 verifies that the design in Fig. 10 with the single-phase clocking scheme work satisfactorily.
Fig 9. Measurements of the circuit in Fig. 8. In the lowest plot the output is manipulated, thus phase-shifted and inverted, so that it is easier to see the latched signal in relation to the input.

Another aspect of using the design in Fig. 10 and the single-phase clocking scheme is the overlap in the on-times of the switches. This overlap tends to attenuate the voltage change at the input and is also indicated in the measurements.

A common replacement for the single-phase clocking scheme is the non-overlapping clocking scheme. Non-overlapping means that only one of the clock signals is high at any given time, and there is a sufficient time slot between the clock edges. The time slot would consume a potential clock skew, and moreover an attenuation of the input voltage change would not be present with the use of this clocking scheme. A third possible clocking scheme could be to use a multiple-valued clock signal, more specifically a ternary signal. The ternary clock signal contains three levels, Vdd, Vdd/2 and Vss, as shown in Fig. 12.

Fig. 10. Another variation of the latch implementation. This design strategy makes it easier to separate the latches from the others, and furthermore the circuits are operating on the same recharge signal.
Fig. 11. Measurements of a single-phase clocking scheme used in the design in Fig. 10. The output indicates a small attenuation of the input-voltage change.

The ternary clocking scheme is based on the non-overlapping scheme and a proposal for its respective clock generator is presented in Fig. 13. Using the design shown in Fig. 10 with the ternary clocking scheme would work satisfactorily (with a few exceptions). When φ1 is high the first inverter has to recharge, and since the first inverter has an nMOS switch the ternary clock signal should be high. When φ2 is high the second inverter is to recharge, which would mean that the ternary clock signal ought to be low, due to the pMOS switch. Thus the ternary clock signal has to be high whenever φ1 is high and low whenever φ2 is high. At the time slot the ternary clock signal can be neither high nor low. A possible representation of the time slot can be Vdd /2. And since both the nMOS and pMOS switches are not completely turned “on” or “off”, a small short-circuit current between the output and the semi-floating-gate exists. This would tend to attenuate the voltage change at the inputs. To avoid this short-circuit current at Vdd /2, or at least to diminish it, one can decrease the supply voltage. A clock supply voltage of 1.8 V would give 0.9 V at the time slot, which is almost the same as the threshold voltage of the switches. A simulation of the design in Fig. 10 with the ternary clocking scheme is presented in Fig. 14.

Although these circuits are simple, they can be used to construct more powerful and complex circuits. This concept can furthermore use the advantages of the capacitive coupled inputs. As Fig. 15 demonstrates, we can easily utilize the multiple-valued recharge latch to handle multiple inputs. In the next section we demonstrate how the innovative latching scheme can be used to incorporate latches without any additional elements or circuits.
Fig. 12. The ternary clocking scheme (in the bottom of the figure) is constructed based upon the two non-overlapping clocking schemes.

Fig. 13. A proposal for a multiple-valued clock generator. The signals are as follows; $A = \phi_1 \cdot \phi_2'$ and $B = \phi_1' \cdot \phi_2$ and recharge $= \phi_1' \cdot \phi_2'$. 
Fig 14: Simulation result of the circuit in Fig. 10 with the ternary clock signal.

VI. D-LATCH AND LARGER MEMORY BLOCKS

A storage cell can be constructed by using, among other things, latches and flip-flops. In the next section the multiple-valued D-latch, which employs the innovative latching scheme, is presented, and its strengths and weaknesses are pinpointed.

In the binary context, one way to construct a D-latch is to cascade two inverters with a feedback path. We could adopt this means of constructing a D-latch by replacing the binary inverters with the multiple-valued recharge inverters. In the previous sections the outputs of the dynamic latches were operating on a φ recharge period. It might be desirable to design D-latches (static memories) whose output(s) follows the same recharge period as the dynamic latches. A paired multiple-valued inverter instead of only one multiple-valued inverter could therefore design a possible implementation of the D-latch. The first and the third inverter operate the recharge on φ and the second and the fourth operate on φ’. The output Q’ would thus have an 180° phase-shift compared to the input, DATA, and the output Q would have a 360° phase-shift (i.e. delayed one period). As mentioned earlier the latching is actually performed by the inverter that operates on the complemented recharge signal (in this case the φ’ signal). It is therefore reasonable to believe that a simpler design is possible. Taking a closer look, we find the first inverter only useful for complementing the input, DATA. By removing this part, the second inverter (which has φ as recharge) would be latching in the DATA. In addition, we can remove the fourth inverter, since it would only cause a malfunction when the STORE signal is low. What we have left is a simplified D-latch, shown in Fig. 16. The smaller area consumption is balanced by the loss of the possibility to extract Q’. It is worth noticing that this implementation enables the stored signal to be extracted out in both a φ and a φ’ recharge period, which may turn out to be a quite powerful ability when used in synchronization.

Assuming ideal multiple-valued inverters, this D-latch would be storing the value of DATA. A small mismatch between the input capacitances and the feedback capacitance would force the stored signal to either the rails or Vdd/2. If the |gain| > 1 the stored signal would eventually become either Vdd or Vss, while if the |gain| < 1 it becomes Vdd/2. Although the gain mismatch is small (and it would take many iterations before the stored signal’s logical level is shifted), some kind of refreshing...
mechanism, either globally or locally, is needed. One can either incorporate the refresh mechanism into the latch, or build up a refresh circuit structure. One other way to construct a simple short-time memory is to just leave the signal floating on a capacitance, for example on an input to a multiple-valued inverter. The leakage time would decide how long it could be valid.

A multiple-valued D-latch design that uses the multiple-valued inverters and also includes a refresh mechanism is shown in Fig. 17. By converting the multiple-valued signal to binary and back again, a kind of refresh of the multiple-valued signal is performed, since it is being generated in each iteration. The design uses the MVBC and BMVC [10], [11]. The output of the BMVC is logically inverted, due to the fact that the BMVC consists of one inverter. A multiple-valued recharge inverter has therefore been added at the output of the BMVC, so that the stored signal becomes logically correct. This design occupies more area than the other proposed D-latches. By applying the innovative latching concept to D-latch design in Fig. 17, we can get rid of the multiple-valued inverter added at the output of the BMVC. The new design is presented in Fig. 18. Note that the BMVC operates on a \( \phi' \).
recharge period. The BMVC would actually be latching in the signal from the MVBC. Since latching does not invert the input-signal, there is no need for the multiple-valued inverter at the output of the BMVC.

Fig. 17. A proposal for a multiple-valued recharge D-latch, implemented with the BMVC and the MVBC designs.

Fig. 18. This design avoids the additional multiple-valued inverter at the output of the BMVC in Fig. 17, by applying the φ’ recharge signal to the BMVC, and hence latching the signals sent by the MVBC.
When it comes to large memory chips, e.g. RAM, where there are numerous D-latches, neither of the D-latch designs proposed would be convenient. It might be wiser to convert the multiple-valued signal to binary and store them in the well-developed binary memories. In this way only one MVBC and one BMVC would be necessary, together with a multiplexor and an array of binary memory cells.

VII. BENCHMARKS AND DISCUSSION

Although the structures presented in this paper work very good in theory and also when tested separately, it is when they are assembled into a complete system that their real performance can be proven. We elaborate on several aspects of their potential and some sources of malfunction.

A. Stability and precision

Throughout this paper we justify the use of frequent recharge as a method to initialize the floating-gates. Due to the junction contact at the floating-gate, a small leakage would naturally exist. The leakage current determines the lowest operation frequency of the recharge clock, i.e. its stability. The leakage is measured to be approximately 0.06 V/sec.

The precision of the presented structures is determined by the evaluation error, which could exist at each inverter. The evaluation error can be used to determine the largest logical depth and how many cascaded multiple-valued recharge inverters one can use without needing to add a refresh mechanism. However, we did not find any dependencies or trends of the evaluation error as a function of the supply voltage. The structures as we see it, mainly consist of three parts, e.g. transistors (nMOS, pMOS), capacitors (Ci and Cf) and the recharge switch. Therefore each of these parts has individually some sources of malfunction, and they can be dealt with as conventional procedures.

![Fig. 19. The current plot for both a binary and a multiple-valued inverter.](image)
B. Power dissipation

We have chosen to present both the static and the dynamic dissipations. A single-input multiple-valued inverter with the same transistor sizes and output load capacitance as a binary inverter is used for comparison. The static power consumption is presented in Fig. 19.

When it comes to dynamic power dissipations we find two interesting aspects; (i) the peak and (ii) the average, where both apply in 10% – 90% of the transition phase. We have deliberately used a worst-case transition situation for the multiple-valued inverters. Fig. 19 presents the average power dissipation in a transition of both the multiple-valued and the binary inverter. Interestingly, the figure illustrates that the multiple-valued inverter consumes more power that the binary ones for a supply voltage higher than 3.3 V. Looking at the peak power, presented in Fig. 19, we find that the multiple-valued inverter has a significantly lower peak value, independent of the supply voltage. Based on this data we can see the following trend developing: the multiple-valued inverters may have higher average power dissipation, but they also have a lower peak. Therefore, we can claim that the variation in the power dissipation is significantly smaller for the recharge multiple-valued circuits than for binaries. We believe that the small variation would make the multiple-valued recharge circuits attractive for mix-mode. One can question the sensitivity for the power-supply noise. As presented the multiple-valued recharge circuits do not generate glitches (to the same amount as binary circuits) at the power-supply, due to their low variation in the power dissipation. We could therefore say that the multiple-valued recharge circuits are not very sensitive for high frequency power-supply noise. On the other hand, the low frequency power-supply noise may lead to a minor error. The error would be noticeable if there is a change at the supply voltage for two succeeding evaluation periods. For the latch a direct result would be an attenuation of the input voltage.

C. Area Consumption

We believe that these multiple-valued structures use less area than its binary counterpart. There are different abstraction levels to see this discussion in. One of the first disadvantages is the size of the capacitors, they actually consume more area than all the other part of the structures together. As long as we can argue for the total size of these structure this should not be of concern. The big advantages for these structures are their complexity. They perform more logic on less elements (i.e. transistor) and due to the fact that the signals carry multiple-values the routing and interconnections becomes less area consuming.

D. Conclusion

The structures have been experimentally fabricated in a CMOS 0.6µm CUP process in order to measure their performances on an actual ASIC. The essence of this paper lies in the latching scheme presented. Although it may not be a innovative approach to use φ and φ to construct a latch, it is shown how the concept can be used in multiple-valued recharge structures to create an innovative type of latch. The power of the latching scheme lies in the use of the otherwise wasted recharging period. Some aspects of the multiple-valued recharge structures have been elaborated. Although the achievable speed may not be as high as binary, it is worth emphasizing that the multiple-valued structure performs a higher number of functions in one period compared to binary.

REFERENCES


