Deglitching of 6-Bit, 3-NMOS Pair per Bit Conversion of a DAC under 90nm CMOS Technology Using Sample and Hold Circuitry

Mohammed Abdul Muqeet, Mohammed Jayeed, M.S. Sameera Begum,

1 King Khalid University-Abha, KSA, abdulmqt19@gmail.com,
2 Buraydah Colleges-Al Qassim, KSA, mhdzaid@gmail.com,
3 JNTUH, India, sameera_475@yahoo.com.

Abstract:
The efficiency of digital to analog converter (DAC) is limited due to static and dynamic errors occurred. Many applications urge for the elimination of glitch (the type of dynamic error), hence the need for glitch reduction circuitry. In this paper three architectures for DAC were designed namely Multiplying DAC, Back DAC and String DAC using a pair of three n-type metal oxide semiconductor (NMOS) transistors for 1-bit conversion under 90nm CMOS technology. Glitch refers to the transient activity in the output current during the major carry transition of input code. Sample and hold circuitry is used as a glitch reduction technique. Using this deglitching technique samples were held for 1-bit conversion under 90nm CMOS technology. Glitch refers to the transient activity in the output current during the major carry switching activity.

Keywords: voltage & current mode DAC, string DAC, Glitch reduction, sample & hold.

I. INTRODUCTION

In today’s world, the data or information to be processed is available in analog form, which contradicts the fact that the information acquired can be processed only in its digital counterpart. Hence, there is a vast need for precision converters to fulfill this need. In this paper, various DAC architectures were discussed and designed using NMOS transistors. An appropriate intended design can be obtained by scaling of MOS transistors to meet both speed and complexity. On the other hand, appropriate speed and minimized die area can also be achieved by using floating gate transistors. In [1], using 600nm CMOS technology, the floating gate transistors were visualized as a voltage-gated current source.

The well-known primary DAC is the binary-weighted DAC, which is the simplest but yet not the most efficient to manufacture in high-resolution applications. Another type of DAC referred to its building block structures is the R-2R ladder. This DAC can be designed in two modes, namely voltage mode (Multiplying DAC) and current mode (Back DAC). In voltage mode, the rungs of the ladder are switched between reference voltage and ground, and the output is obtained from the end of the ladder. This DAC is widely used in display driving system applications [2]. Transistor matching is difficult in this type of DAC, and this can be accurately matched by using a compensation circuit and Monte-Carlo simulation [3, 4]. In the current mode, the reference value is provided from the end of the ladder, and the rungs are stretched between output and ground [5, 6].

Another principle architecture used widely is the string DAC architecture. As the name itself illustrates, it uses a stack of resistors to accomplish conversion [7]. This type of architecture requires a large number of resistors, i.e., for N-bit DAC the number of resistors required is $2^N$ [8]. Modern DACs usually use a combination of the architectures where there is a need to compromise between precision and speed. Using dual architecture, (one for least significant bits and the other for most significant bits), is referred to as a segmented architecture [9]. On the other hand, most of the DAC’s integrated circuits require a combination of the architectures mentioned above due to difficulty in obtaining high speed, high precision, and low cost. Such DACs are referred to as hybrid DAC’s. In [10], slew-rate enhanced class AB output amplifier was used to achieve high speed in hybrid DAC.

Practical implementation of DAC is mainly affected by its non-ideal transfer characteristics. The ideal behaviour is mainly characterized by static and dynamic performance [11]. Static errors mainly affect the accuracy of a converter when it converts the static signals. On the other hand, dynamic errors are the ones that affect the speed, are mainly caused by switching fluctuations. In this paper, the most common dynamic error ‘glitch’ was studied. Many glitch reduction methods are available, some of which reduces the glitch area but indeed affects the performance of the system. The main source of a glitch is ‘major carry switching activity’.

In [12], the gray code method was incorporated for reducing glitch due to the fact that, it switches only 1-input in
a sequence of inputs. Another method is to use first-order low
pass filter [13, 14] to calibrate glitch error. Another commonly
used method is to use a capacitance compensation filter for
carbon nanotube FET (CNFET) in pseudo-segmented
structure also reduces glitch and power consumption [16]. In
[17], variable delay buffers were used for glitch reduction
using current steering DAC. In [18] hybrid wideband R2R
LSB segmentation with impedance attenuator was used for
glitch reduction. All the methods mentioned above with
compensating glitch error also affect the performance of the
device in case of gray code technique, whereas some methods
compensate the unwanted area in cases of designing an ideal
filter. In this paper, a sample and hold (S&H) circuitry was
used [19] to reduce the glitch error that suffice the need for
reducing glitch without affecting the performance of the
circuit.

II. DAC ARCHITECTURES

In this paper, the design of three architectures for a 6-bit
input sequence was accomplished with and without the glitch
reduction technique, and the simulation results were compared
to justify the efficiency of the design. With the basic R-2R
design, it is always possible to generate either a voltage output
or current output. Generation of voltage output requires an
additional output buffer whereas for output current no output
buffer is required. In this paper, all the circuits are designed to
generate an output current.

A. 6-BIT MDAC ARCHITECTURE

MDAC architecture is the most flexible building block
providing efficient design specifications. In this design, for
each bit conversion, 3-NMOS transistor pairs under 90nm
technology were used. Inconsistent with the traditional design
of an MDAC in R-2R configuration, the implemented design
for each bit operation consists of two rungs. The NMOS
transistors in both the rungs were designed to operate in the
triode region, whereas the third NMOS transistor operates in
either weak-inversion region or active region based on the
digital (LOW/HIGH) input provided for conversion expressed
by the equation (1).

For the cutoff region, \( V_{gs} > V_t \)
For the linear region, \( V_{gs} < V_{gs} - V_t \)
For saturation region, \( V_{gs} > V_{gs} - V_t \)  \( (1) \)

The transistors in the circuit are designed under 8:1 and
4:1 equivalency for the parameter W/L ratio to achieve
appropriate matching considerations which are shown in
Table-1 along with the design considerations. The
implemented circuit for MDAC is shown in Figure-1. MDAC
is a low noise DAC that uses various reference voltages at
every bit conversion. The output current \( I_{out} \) was measured
which was found to be proportional with the digital input code
\( (D_0-D_5) \) as shown in Figure-4(a).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>MDAC</th>
<th>Back DAC</th>
<th>String DAC</th>
<th>S&amp;H</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V_{ref} )</td>
<td>5V</td>
<td>5V</td>
<td>5V</td>
<td>-</td>
</tr>
<tr>
<td>( V_g )</td>
<td>7.5V</td>
<td>7.5V</td>
<td>7.5V</td>
<td>-</td>
</tr>
<tr>
<td>Varied ( V_{ref} )</td>
<td>5V (MSB) – 4.8V (LSB)</td>
<td>5V (MSB) – 4.8V (LSB)</td>
<td>5V (MSB) – 4.8V (LSB)</td>
<td>-</td>
</tr>
<tr>
<td>NMOS in linear Region</td>
<td>Q1,Q3, Q4,Q6, Q7, Q9,Q10,Q12, Q13, Q15,Q16,Q18</td>
<td>Q1,Q3, Q4,Q6, Q7, Q9,Q10,Q12, Q13, Q15,Q16,Q18</td>
<td>Q1 through Q12</td>
<td>-</td>
</tr>
<tr>
<td>NMOS in Cutoff/Saturation Region</td>
<td>Q2,Q5, Q8,Q8, Q11,Q14, Q17</td>
<td>Q2,Q5, Q8,Q8, Q11,Q14, Q17</td>
<td>Q13 through Q18</td>
<td>-</td>
</tr>
<tr>
<td>W/L ratio</td>
<td>8:1 (Linear) 4:1 (Cutoff/Saturation)</td>
<td>8:1 (Linear) 4:1 (Cutoff/Saturation)</td>
<td>4:1 10:1</td>
<td></td>
</tr>
<tr>
<td>( T_r )</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>20ms 10µs</td>
</tr>
<tr>
<td>( T_i )</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>10µs</td>
</tr>
<tr>
<td>( V_m )</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>10V</td>
</tr>
<tr>
<td>( C )</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1pF</td>
</tr>
</tbody>
</table>

B. 6-BIT BACK DAC ARCHITECTURE

In general, a back DAC finds its place in
instrumentation and digitally controlled calibration
applications. Both the back DAC and MDAC are analog
circuits with a small difference in the digital control logic. All the NMOSes used for bit conversion are designed under 90nm technology. In the designed circuit, 3 transistors were used per bit conversion. The NMOS transistors in both the rungs were designed to operate in the triode region, whereas the third NMOS transistor operates in either weak-inversion region or active region based on the digital (LOW/HIGH) input provided for conversion as expressed in equation (1). The transistors in the circuit are designed under 8:1 and 4:1 equivalency for the parameter W/L, ratio to achieve appropriate matching considerations, which is shown in Table-1 along with the design considerations. The implemented circuit for back DAC is shown in Figure-2. The design considerations are shown in Table-1. The output current \( I_{out} \) versus digital input code is shown in Figure-4(b).

\[
V_{in} \quad Q_3 \quad Q_6 \quad Q_9 \quad Q_{12} \quad Q_{15} \quad I_{out}(\mu A)
\]

Figure-2: 6-bit Back DAC

C. 6-BIT STRING DAC ARCHITECTURE

The string architecture also uses NMOS transistors stacked together in an orderly manner to achieve the design as shown in Figure-3. The transistors \( Q_1 - Q_{12} \) are designed to operate in triode region whereas the transistors \( Q_{13} - Q_{18} \) operates in either weak-inversion region or active region based on the digital (LOW/HIGH) input provided for conversion as expressed in equation (1). The transistors in the circuit are designed under 4:1 equivalency for the parameter W/L, ratio to achieve appropriate matching considerations, which is shown in Table-1 along with the design considerations.

String DAC’s are widely used due to their properties such as monotonic nature, smallest die area, and low-cost design. This type of architecture provides the lowest glitch area as compared to other two R-2R architectures mentioned in this paper.

\[
V_{in} \quad Q_{1} \quad Q_{2} \quad Q_{3} \quad Q_{4} \quad Q_{5} \quad Q_{6} \quad Q_{7} \quad Q_{8} \quad Q_{9} \quad Q_{10} \quad Q_{11} \quad Q_{12} \quad Q_{13} \quad Q_{14} \quad Q_{15} \quad Q_{16} \quad Q_{17} \quad I_{out}(\mu A)
\]

Figure-3: 6-bit string DAC

D. R2R DAC WITH S&H CIRCUITRY

The most common dynamic errors that affect the performance of a DAC are the glitch and settling time. Glitches are mainly experienced due to the switching activity from the incoming digital code, which causes a momentary surge in current. The nature of glitch is assumed as a step response consisting of glitch area and settling time. In practical DAC’s, the glitch is often experienced when switching the digital input code from 00001111 to 00010000 and 00011111 to 00100000. In this paper, Sample and Hold (S&H) circuit was used to reduce the glitch area. The obtained analog signal was sampled using an NMOS transistor \( Q_1 \) under 2µm technology with 10:1 W/L ratio.

A gate voltage was applied using a step input source with pulse time, \( T_p=20\text{ms} \) and rise and fall time of \( T_r=T_f=10\mu\text{s} \). A 1pF output capacitor was used to hold the output current as shown in Figure-4.

\[
\text{Figure-4: } I_{out} \text{ without glitch reduction technique. (a) MDAC, (b) Back DAC and (c) String DAC.}
\]

\[
\text{Figure-5: Sample and Hold Circuitry}
\]

\[
\text{From DAC } \quad Q_1 \quad C_{out} \quad 1pF \quad \text{V}_1
\]

International Journal of Electrical Engineering and Ethics- Volume 2 Issue 1, September 2019

ISSN: 2455-9771             http://www.ijeeejournal.org
Glitch area is a measure of the area under transition of the output of DAC. In this paper, the glitch area (A) was computed by partitioning the glitch duration, i.e., the time interval determining the start and end occurrences of a glitch into sub-intervals \( N \) as expressed in equation (2) for the specified glitch amplitude \( h_i \) at a specified interval \( w_i \).

\[
A = \sum_{i=1}^{N} h_i w_i
\]  

The net glitch area refers to the summation of individual glitch area. The net glitch was calculated without and with glitch reduction technique for all the three architectures mentioned in this paper. The output current \( I_{out} \) from glitch reduction circuitry versus digital input code is shown in figure-6.

![Graphs showing output current vs input code for different DACs](image)

Figure-6: \( I_{out} \) with glitch reduction technique. (a) MDAC, (b) Back DAC and (c) String DAC.

### Table-2: Net Glitch Area.

<table>
<thead>
<tr>
<th>DAC</th>
<th>Glitch occurrence</th>
<th>Net Glitch Area</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>W/O S&amp;H</td>
<td>With S&amp;H</td>
</tr>
<tr>
<td>MDAC</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>001111 → 010000, 011111 → 100000, 101111 → 110000</td>
<td>51nA</td>
</tr>
<tr>
<td>Back DAC</td>
<td>000011 → 000100, 000111 → 001000, 011111 → 100000, 101111 → 110000</td>
<td>655nA</td>
</tr>
<tr>
<td>String DAC</td>
<td>000111 → 001000, 001111 → 010000, 010111 → 011000, 101111 → 110000</td>
<td>15nA</td>
</tr>
</tbody>
</table>

### III. CONCLUSION

A novel deglitching method for R-2R DAC architectures was proposed. Experimental results validate the glitch reduction method of sample and hold circuit. Significant glitch reduction was observed in all the DAC architectures designed. It was observed experimentally that the net glitch area was reduced by 76.5% in MDAC, 41% in Back DAC and 3.4% in string DAC. Also, it was observed that the string DAC possess the least net glitch area without deglitching method.

### ACKNOWLEDGMENT

The authors would like to thank King Khalid University, Abha, KSA and Buraydah Colleges for supporting and encouraging this work.

### REFERENCES


