Low Voltage CMOS Circuits for Analog Decoders

Chris Winstead, Nhan Nguyen, Vincent C. Gaudet, Christian Schlegel

University of Alberta

Electrical and Computer Engineering Dept.

Generous funding and resources for this project have been provided by the Canadian Microelectronics Corporation
Analog computation for iterative decoders

- Elegant message format:
  - Probabilities encoded as electrical currents.
  - Significantly reduces routing congestion.

- Simple circuits: 15 transistors per operation:
  - Small node size $\rightarrow$ fully parallel decoding.
  - Efficient use of silicon space.

- Very low power:
  - Device currents typically $< 1\mu A$.
  - Total power 100 times less than digital circuits.

- No iterations: information exchange is continuous.
- High throughput is achieved through parallelism.
CMOS Voltage Requirements

- CMOS is often the preferred technology:
  - It is ubiquitous and inexpensive.
  - Simplifies integration with other system components.

- More advanced CMOS processes are sensitive to voltage. (Thin gate oxide destroyed by even small voltages).

- Current CMOS analog decoders require Vdd of ~1.2V.
  - Saturation assumption: all devices require $v_{ds} > 200$ mV.
  - Devices are “stacked” between Vdd and ground.
  - If Vdd less than ~1V, some devices are unsaturated.

- At right is shown a portion of a typical analog sum-product circuit. There are four voltage drops.
Low-voltage CMOS analog decoders

- Can operate with supply as low as .4V.
  - Based on a more accurate device model – not all transistors must be in saturation.
  - Reduce inaccuracy due to Early effect (proportional to $v_{ds}$).
  - Further reduce power consumption (proportional to $V_{dd}$).

- Add a few transistors per operation, but reduce message routing complexity.

CMOS Device Models

\[
\text{Subthreshold transistor model:} \quad I_D = I_0 \cdot \frac{W}{L} \cdot e^{\left(\frac{\kappa v_{gs}}{U_T}\right)} \left[1 - e^{\left(-\frac{\kappa v_{ds}}{U_T}\right)}\right]
\]

\[
\text{Saturated transistor model} \quad \text{with} \; v_{ds} >\sim 200\text{mV}
\]

\[
I_D \propto \exp(v_{gs})
\]

\[
I_D \propto \exp(v_{gs}) [1 - \exp(-v_{ds})] = \exp(v_{gs}) - \exp(v_{gd}) = I_f - I_r
\]

\[
\begin{align*}
\text{Non-saturated transistor model} \\
\end{align*}
\]
Basic Principles of Analog Decoders

- Decoders are derived from the sum-product algorithm on a code’s factor graph.
- A message is a set of currents \( \{I_{x_i}\}_{i=1}^{N} \) which represent a probability mass. Thus

\[
\sum_i I_{x_i} = \text{constant}
\]

- A common topology for analog decoding circuits is based on the Gilbert Multiplier (shown on next slide)
  - Two messages are input, one is output.
  - The Row Inputs are a vector of currents \( X \) which represent one message.
  - The Column Inputs are a vector of currents \( Y \).
- The diode connected transistors (M1 and M3) perform current-to-voltage conversion.
  - Note that the X inputs are referred to \( V_{ref} \) instead of ground.
  - The voltage at the drain of M2 is increased when \( V_{ref} > 0 \).
  - This ensures that transistor M2 is saturated.
- The source-connected transistors (indicated by multi-input boxes) produce products of all the (current) components of \( X \) and \( Y \).
- The connectivity box consists of wire connections which produce the sum of probabilities by shorting currents.
- Taken together, these operations implement the sum-product algorithm.
Input sets are $X$ and $Y$.

Connectivity

The "box" symbol represents an array of source-connected transistors:
Translinear Loops

- Because $I_D \propto \exp(v_{gs})$, in a closed loop of devices:
  \[ \sum_i v_{gs}^{(i)} = \sum_j v_{gs}^{(j)} \]
  \[ \Rightarrow \sum_i \ln I_D^{(i)} = \sum_i \ln I_D^{(j)} \]
  \[ \Rightarrow \prod_i I_D^{(i)} = \prod_j I_D^{(j)} \]

- Some important loops are examined below and at right. The term to be solved is indicated in red.

1. $I_{xj} \cdot I_{zik} = I_{xk} \cdot I_{zij}$
   \[ \Rightarrow I_{zik} = I_{xk} \cdot \frac{I_{zij}}{I_{xj}} \]

2. $I_D^i = I_{yi}$
   $Id_i = If - Ir$

3. $If \cdot I_{zij} = Ir \cdot I_{xj}$
   \[ \Rightarrow Ir = \frac{If \cdot I_{zij}}{I_{xj}} \]
Reducing the Voltage

- We will consider the “slice” of the Gilbert multiplier shown below:

- Under saturation assumptions,
  \[ I_{zij} = \frac{I_{xj} \cdot I_{yi}}{\sum_k I_{xk}}. \]

- If \( V_{ref} = 0 \), then M2 is not in saturation. We may use translinear loops to determine the correct output:

  \[
  Id_i = If - Ir = \sum_k I_{zik} \quad \Rightarrow \quad Iy_i = \frac{I_{zij}}{I_{xj}} \left( Iy_i + \sum_k I_{xk} \right)
  \]

  \[
  \Rightarrow \quad Iy_i - Iy_i \cdot \frac{I_{zij}}{I_{xj}} = \sum_k I_{xk} \cdot \frac{I_{zij}}{I_{xj}} \quad \Rightarrow \quad I_{zij} = \frac{Iy_i \cdot I_{xj}}{Iy_i + \sum_k I_{xk}}
  \]

- \( Iy_i \) appears in the denominator, destroying the normalization.
Correcting the Normalization

- To correct the normalization is simple: both the X and Y inputs are probability masses. Therefore

\[
\sum_k I_{x_k} = \text{constant} \\
\sum_i I_{y_i} = \text{constant}
\]

- Normalization is corrected when all terms from Y appear in the denominator.

- Let \( Y_i \) be the the set of column inputs Y, excluding input \( I_{y_i} \). If the column inputs from \( Y_i \) are provided as “dummy” row inputs in each column \( i \), then the outputs become:

\[
I_{z_{ij}} = \frac{I_{y_i} \cdot I_{x_j}}{\sum_l I_{y_l} + \sum_k I_{x_k}}
\]

- The denominator is now a constant.
  The resulting low-voltage analog decoder topology is shown on the next slide.

- The large denominator results in substantial attenuation of current’s at the circuit’s output. Renormalization is therefore required to prevent signals from converging to zero.
Low-Voltage Sum-Product Circuit

Input sets are X and Y. Current to voltage conversion:

Current to voltage conversion:

\[ I_x_1 \rightarrow x_1 \quad \ldots \quad I_x_N \rightarrow x_N \]
\[ I_y_1 \rightarrow y_1 \quad \ldots \quad I_y_M \rightarrow y_M \]

Connectivity

Renormalization

Outputs (currents directed toward ground)
Renormalization

- It is important for many reasons to control the average device current in an analog circuit.
- It is especially important that the currents do not converge to zero. The circuits need gain in order to achieve this.
- In the low-voltage circuit presented above, the gain is always less than one. This must be compensated by a renormalizing circuit.
- Analog decoders commonly use a circuit such as this one:

![Diagram of analog decoder circuit]

- Ordinary, M2 is assumed to be in saturation, and $n = m = 1$. In that case: $\sum_i O_{out_i} = I_u$.
- We can lower the supply voltage further if we do not require saturation for M2. In that case, translinear analysis reveals that:

$$\sum_i O_{out_i} = \frac{n \cdot I_u \cdot m \cdot \sum_i I_{in_i}}{n \cdot I_u + m \cdot \sum_k I_{in_k}}$$
Iterated Analysis

- We are interested in the sum of currents at the output of each stage in the circuit.
- Recall that a message is a bundle of currents which represent a probability mass. For any message $X$, we define a summary variable $k_X = \sum_i l_{x_i}$.
- This allows us to evaluate the circuit in terms of a single variable, $k_X$. The renormalization circuit has the transfer function:

$$ k_{out} = \frac{n \cdot m \cdot k}{n + m \cdot k} = f_n(k) $$

- We may now consider the evolution of $k$ as an iterated map in one dimension:

- Fixed points of $f_n$ occur at $k_0 = 0$ and $k_1 = n - \frac{n}{m}$.
- For an iterated function $f_n(k)$ in one dimension, a fixed point $k_0$ is stable iff $f'(k_0) \leq 1$.
- For the renormalizer function: $f'(0) = m$ and $f'(k_1) = \frac{1}{m}$.

- Therefore, if $m > 1$, $k$ is always pushed away from zero, and toward a fixed point $k_1$ greater than zero.

- It is interesting to note that, in the conventional design where $m = 1$, zero is a stable fixed point. Currents would therefore converge to zero in the conventional circuit under low-voltage conditions.
The Renormalizer Transfer Function

- The renormalization circuit works more effectively as $m$ increases.
- Increasing $m$ means increasing the size of the circuit and the amount of parasitic capacitance.
- Low currents in the decoder will make it operate slowly. Large device capacitance will also slow down the circuit. There is probably an optimal $m$ in terms of speed, but this has not yet been determined.
Example Decoder

- As an example, we consider the design of a decoder for a simple (7,4) Hamming code. (A decoder for an (8,4) Hamming code has also been designed and will be fabricated soon)

- The design has been simulated in SPICE, assuming parameters for a TSMC .18µm process.

- The circuit was found to work with Vdd as low as .4V.

Results for the decoder operating at a Vdd of .8V are shown below.
Observations from the Example Decoder

- Low-voltage analog decoders appear to be somewhat slower than the more common Gilbert-based decoders. This is consistent with a general tradeoff between speed and power consumption.

- For very large, parallel analog decoding networks, the speed of the interfaces should still be the limiting factor, rather than the speed of the decoder itself.

- Curiously, the low-voltage analog circuits appear to have a much stronger memory effect than Gilbert-based circuits. Simulations have revealed that it is necessary to erase memory of a previous codeword before attempting to decode a new one. This is accomplished using a well-known reset circuit:

  ![Reset Circuit Diagram]

  - The NMOS transistors act as digital switches. When the ‘rst’ signal occurs, all probabilities in a message are shorted together. This forces a uniform probability mass on every message, and initializes the network in preparation for the next cycle.
Conclusions

- CMOS analog decoders have been introduced.
  - Conventional (Gilbert-based) circuits require Vdd > 1V.

- Translinear analysis has been used to explore the effect of
  - Eliminating reference voltages.
  - Reducing the supply voltage.

- By adding dummy transistors to the low-voltage Gilbert circuit,
  - The circuit’s function is corrected.
  - Attenuation of currents results.

- A renormalization circuit
  - Restores current levels in the network.
  - Requires more careful design when Vdd is low.

- The resulting circuits have been demonstrated using SPICE simulations. A simple proof-of-concept decoder chip is to be fabricated in the near future, using a TSMC .18μm process.

- Some differences and tradeoffs have been noted between conventional Gilbert-based decoding circuits and low-voltage decoding circuits. Plans for future research include a more detailed examination of the relationships between these topologies.