BIPOLAR VLSI - AN APPLICATION FOR A HIGH PERFORMANCE MICROCONTROLLER

Peter L. Jones*

This paper looks at the special merits of bipolar differential logic for high performance VLSI. A design route is developed using a high level hardware description language leading to a semicustom style of automatic layout. Preliminary assessment from simulation suggests that 100MHz operation of an eight-bit high-functionality microcontroller can be achieved using this approach.

1. Introduction

This paper describes a high-speed, high functionality 8-bit microcontroller design using a multi-level differential current mode logic (MDCML) cell library (Plessey, 1988), fabricated in the Plessey bipolar collector diffusion isolation process.

The following design objectives were set:

- To develop a high-speed bipolar 8-bit microcontroller design capable of clock speeds of 100MHz plus.
- The supply current should not exceed that in the commonly available 8-bit NMOS microcontrollers.
- The design methodology should be flexible enough to enable any comparable microcontroller or microprocessor instruction set and architecture to be developed.
- The initial design of microcontroller should be instruction set compatible with a commonly encountered commercial 8-bit microcontroller for which development systems and software were readily available.

To meet this final criterion, it was decided after market research that the Intel 8051 family (Intel, 1989) of microcontrollers be chosen for the basis of the design to be developed.

2. Multilevel Differential Current Mode Logic

The MDCML standard cell library has been developed using 3-level differential logic. An example of a 3-input AND gate is given in Figure 1. Complex logic functions can easily be created as single gate functions. Figure 2 shows a 4:1 multiplexer as a single gate function with one current source. For comparison, a 4:1 multiplexer implemented

* Department of Electrical Engineering, University of Manchester, Oxford Road, Manchester M13 9PL, UK

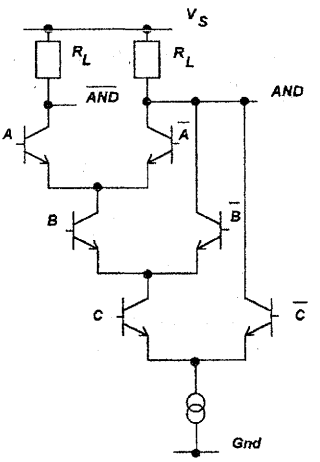

Fig. 1. MDCML 3-input AND gate.

in Plessey's standard CML technology (Plessey, 1985) requires four 3-input NOR gates and three inverters, with a total of 15 current sources.

A complexity aspect of MDCML circuits is that voltage level shifters and regenerators are required to transfer logic signals between different differential levels. Also, because of the differential drive, two tracks are required per signal. However, the

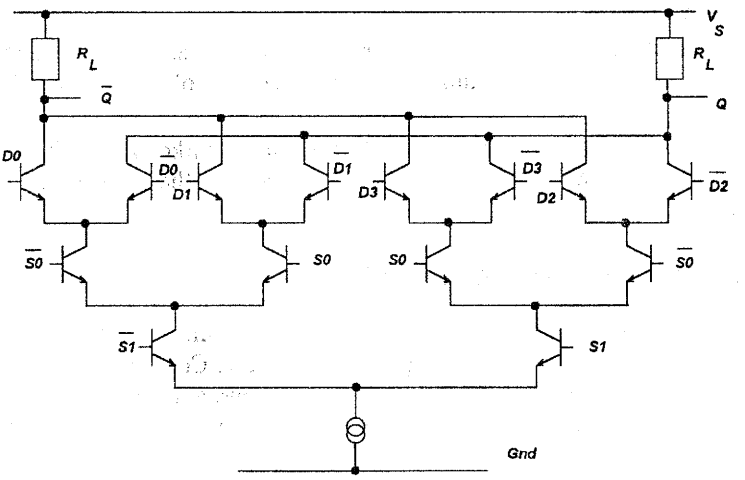

Fig. 2. MDCML 4: multiplexer.

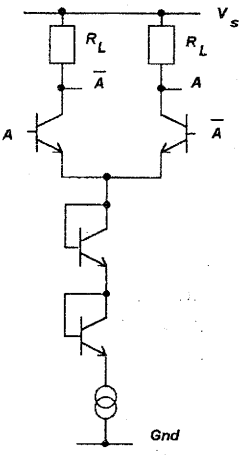

Figure 3: MDCML single differential pair, DBUF3.

availability of the true and complement of a signal means that complexity is reduced since logic inversion is achieved by simply swapping connections to the transistor pairs in the differential tree. Also the differential nature of the signal gives improved noise immunity and so a logic swing of only 160mV is permissible which in turn provides for a high speed of operation.

With careful circuit design, the extra area required for level shifters, level regenerators plus the two tracks per signal can be offset by the fact that inverters are not required and that complex logic functions are available with delays, current consumption and area comparable with that of a single gate. This MDCML circuit approach is particularly effective for multiplexer and latching circuits.

3. Delay Characterisation Of Standard Cell Structures

The circuit principles described in the previous section form the basis of the Plessey differential logic library used to design the MDCML microcontroller. The logic design was developed using the ELLA (Computer General) capture and simulation program. ELLA descriptions of the cell library of logic primitives were written with delays based on a relative time unit, T. Timing measurements have therefore been normalised to the propagation delay through a single differential pair, the cell circuit DBUF3, shown in figure 3. The propagation delay tpd for each library primitive is determined from the differential signal crossover points using the SPICE simulation program. Delays are then related to the T-unit as follows:

$$\frac{t_{PD}\ of\ primitive}{t_{PD}\ of\ DBUF3} = relative\ propagationdelay(T units)$$

The propagation delays for all the primitives in the Plessey differential logic cell library are calculated in terms of T-units and incorporated into the ELLA library files for the cells.

The use of the T-unit enables the rapid revision of the library to account for changes of SPICE parameters without having to re-simulate all circuits and cells. SPICE parameters have to be revised from time to time to account for process changes and for down-scaling. Re-simulation of the DBUF3 circuit using SPICE is therefore all that is required, all other delays being specified as multiples of this delay. This approach to ease maintenance of a complex circuit cell library has been applied over a period of four years during which there have been several process changes and one major shrink. The accuracy of the T-unit has been retained to within 10% across cells and between process changes and has proved adequate for all applications so far encountered. Currently the library comprises 39 logic primitives plus level shift variants and power options.

The calculation of T-unit delays was employed in the design of the MDCML microcontroller. The T-unit was measured for minimum feature size switching transistors of the Plessey FAB4 process with the speed/power options provided by gate current of 50μA, 100μA and 200μA. Table 1 shows the T-unit values measured for DBUF3 with three power options. The ratios of T-units for different power options are of course reflected in all primitives of the cell library, thus providing a useful flexibility to meet speed and power constraints in critical regions of a design by choosing appropriate gate current and load resistor combinations.

Table 1. T-unit measurements for MDCML simulations.

SPEED/POWER OPTION (gate current / R_L (ohms)	DBUF3 t_{PD} (T unit)
50μA/3.2K	0.811nS
100μA/1.6K	0.580nS
200μA/800	0.508nS

Applying the T-unit values shown in Table 1 to the MDCML 4:1 multiplexer circuit shown in Figure 2 gives the following. The gate level 3 D0,D1, D2 and D3 inputs to Q outputs have a delay of 1.39T defined in ELLA. Therefore for the speed power options of 50μA/3.2K, 100μA/1.6K and 200μA/800, propagation delays of 1.13nS, 0.81nS and 0.71nS, respectively, would be used. Similar calculations can be made for the multiplexer level 2 SO inputs to Q outputs, which have a delay of 1.76T, and level 1 S1 inputs to Q outputs, which have a delay of 2.29T.

4. Design Implementation

Every effort has been applied in the design of the microcontroller both in the low level circuit design and in the total system architecture to exploit the features of MDCML to provide high speed and area efficiency with low current consumption by implementing complex functions with multiplexer and register components which have similar power-delay characteristics to those of a single gate. The following aspects of the design are chosen to illustrate the matching of MDCML circuits to the microcontroller subsystem function requirements.

The 8-bit ALU is based on a Mead-Conway (Mead and Conway, 1980) design using 4-bit propagate, kill and result control signals generated from a PLA instruction decoder. The basic building block for the ALU is based on the following function:

$$G(A,B) = G0 \cdot \overline{A} \cdot \overline{B} + G1 \cdot A \cdot \overline{B} + G2 \cdot \overline{A} \cdot B + G3 \cdot A \cdot B$$

This is ideal for MDCML implementation, being just a 4:1 multiplexer generating the propagate (P), kill (K) and result outputs. The carry chain function is also just a single gate performing the logical operation:

$$K + P \cdot \overline{Cin}$$

This method of implementing the ALU is highly efficient in area and gives great flexibility for incorporating other instruction sets as different decoding or ALU functions can be generated by simply redefining the PLA. Simulations carried out on this circuit using ELLA show that for gate currents of 50uA, the worst case circuit delay is 11.9nS for addition, with total circuit current consumption being 5.6mA. The carry-chain implementation was found to be adequate for this design but the delay could be further improved if the carry-chain was replaced with a look-ahead carry function.

The full CPU structure is based on the Mead-Conway design with additional multiplexers to carry out shift-right operations, nibble swaps and to define the required input/output for the ALU. The Boolean processor is also built up from 4:1 multiplexers in MDCML with control signals generated by the PLA. The CPU includes a hardware multiplier and hardware divider both of which are built of MDCML full-adder circuits and D-type latches. To save on hardware the divider and multiplier are similarily designed with the accumulator data being simultaneously shifted in and resultant data being shifted back into the accumulator, thus requiring a total of 8 clock cycles to complete either the division or multiplication. There are also two configurable 16-bit timer/counters. All registers and timing logic are implemented in MDCML.

To further increase the speed of operation all instructions have been reduced to run in a single machine cycle (Note: 32% of the 8051 instructions take at least 2 cycles). This is achieved by compressing the internal timing and by introducing extra hardware where needed. This has meant that minor changes to the standard input/output format have to be made for the device to access external RAM and program memory but this is

justifiable when considering the gains to be obtained with the 100MHz clock speed of the device.

To meet the specification on quiescent supply current it has proved necessary to use differing speed/power options so that those sections of the design which do not require the maximum available gate speed can use a lower gate current. To a first approximation the logic uses 30,000 transistors, consumes 300mA of quiescent current and has a chip area requirement of 4mm square, (excluding routing). The final layout design must include sufficient space to accommodate a 256 x 8 bit RAM a 1k x 8 bit ROM and the PLA instruction decoder.

The design has been captured and simulated using ELLA to verify that the system is functionally correct and meets the desired performance specification. The ELLA model will be converted into a physical layout by mapping the standard cell logic onto a regular array of fixed geometry component cells with autorouted interconnect similar to a gate-array format. The larger customised blocks for the RAM, ROM and PLA will however be hand crafted and placed for optimum area utilisation and performance.

5. Conclusions

The results presented demonstrate the capability of applying MDCML to design a high-functionality 8-bit microcontroller with clock speed in excess of 100MHz. The design has been optimised to utilise the advantages that MDCML offers to increase speed and performance, Also, the use of a PLA provides the flexibility to incorporate any comparable instruction sets. The use of ELLA for capture and simulation coupled with the easily maintained logic primitive library has been shown to be a viable design methodology.

References

Plessey Semiconductors (1988): *Differential Logic Design Manual.*- Issue 1.0.

Intel (1989): *8-bit Embedded Controller Handbook.*

Plessey (1985): *ULA "DS" Design Manual.*- DS/100.

Computer General: *The ELLA User Manual and Language Reference Manual.*

Mead C.A. and Conway L.A. (1980): *Introduction to VLSI Systems.*- Addison-Wesley, pp.150-154.