EE 434

Final Projects

Fall 2006

Six projects have been identified. It will be our goal to have approximately an equal number of teams working on each project. You may work individually or in groups of 2. All designs are to be implemented in the AMI 0.5u CMOS process and must include a complete layout ready for fabrication along with post layout simulation results showing that your design meets the performance specifications. MOSIS is providing space for fabrication of all student designs for those that are interested. A commitment for testing of the silicon after fabrication is necessary, however, before formal submission for fabrication if you would like to have your circuit fabricated.. The fabrication dates are currently scheduled for January 8 and February 12. They usually hold that date firm. Normal processing time is around 8 weeks which means that silicon should be back for testing sometime in the month of March for the first run or in April for the second run. The die size you have to work with is 2.2mm x 2.2mm inclusive of the bonding pads. MOSIS terms this the “tiny chip”. This should provide more area than you will need for the design. You may consider putting more than one copy of your circuit in the space provided or putting some test structures in any excess space. It is a good idea to provide access to selected internal test points as well should you find it necessary to access an internal node if there is a problem with your circuit after fabrication. MOSIS will return 5 packaged parts. A standard pad frame is available and can be obtained from your TA.

The first project is related to current products of Maxim (the Dallas Semiconductor part of the company). The specifications have been modified to make them compatible with the background you have in this course. The second is a variant of a transceiver manufactured by several different companies.

Projects should be selected by Nov. 10. Your TA will keep a list of the projects with a goal to keep the number of groups working on each project about the same with projects selected on a first-come basis. If your first choice is not selected, please give a prioritized list of the projects to your TA.

Project 1 Digital Potentiometer/Amplifier/DAC

Design a multi-purpose digitally controlled analog building block. This structure can serve as a digital potentiometer, an inverting or noninverting amplifier and a DAC depending upon the state control inputs. A cell of the operational amplifier will be provided to you by your TA. Assume V_{DD}=2.5V and V_{SS}=-2.5V. The state control signals A_O and A_1 will identify one of four states of operation of this device. The operation control signals C_O, C_1, C_2 and C_3 are used to control the characteristics of the device in each of the four states.

When A_O and A_1 are high, the circuit is to perform independently as a digital potentiometer and an operational amplifier. The digital potentiometer should have 16 taps, each with a nominal impedance of 5K. When A_0 is high and A_1 is low, the circuit is to perform as a 4-bit DAC where the op amp is connected in a unity gain configuration to a tap on the potentiometer and the DAC output is determined by the control settings on the potentiometer. The DAC input, often termed "V_{REF}" should be connected to one end of the resistor string and the other end should be grounded. When A_0 is low and A_1 is high, the circuit is to perform as a programmable inverting finite gain amplifier. One end of the resistor string should go to the op amp output, the "wiper" to the "-" input and the other end of the resistor string to the input. Finally, when A_0 is low and A_1 is low, the circuit is to perform as a programmable noninverting finite gain amplifier.

The digital potentiometer is similar in principle to the Dallas Semiconductor DS 1666 (www.dalasemi.com) but with a reduced number of taps, with parallel rather than serial control of the tap position, and with a linear taper rather than an audio taper.

Project 2 2B:3B Transceiver Block

Serial channels are widely used for communicating between computers that may be a few feet apart of on the other side of the world. Invariably the data that is to be transmitted is parallel data so a parallel to serial conversion is needed to get the data ready for transmission and a serial to parallel conversion is needed to convert the data from serial data to parallel data at the receiver. Invariably the data is transmitted from a synchronous system on transitions of a clock and invariably the data at the received is synchronized relative to a clock at the receiver. Unfortunately the two clock frequencies may not be exactly the same and unfortunately it is generally considered impractical to transmit the clock signal to the output so the clock must be recovered from the serial data stream itself. This is often done with a phase-locked loop (PLL) at the receiver which contains an internal voltage controlled oscillator (VCO) that must be "locked" to the input data sequence. The "recovered" clock is simply the output of the VCO in the PLL. The PLL must obtain regular measurements of the phase difference between the VCO output and the data input to maintain lock. It is common in many applications to have periods of time where no data is available and during these intervals, long strings long serial strings of 0's or 1's must be transmitted. Unfortunately, it is difficult (actually impossible) for the PLL to maintain lock in the absence of transitions on the incoming data stream. To circumvent this problem, the parallel data is often coded prior to serial transmission to guarantee that there will be ample transitions in the transmitted data to recover the clock. Of course, the received data must be decoded at the output to recover the intended data sequence. 8B: 10B and 4B:5B coders are often used for this purpose. In an nB: (n+1)B coder, an n-bit word is converter to an n+1 bit word with a fixed mapping that will guarantee that the maximum number of consecutive 0's and 1's in the transmitted data stream is small (like 3 or 4) irrespective of the nature of the input data.

In many communication channels, data itself is arranged in packets in which a fixed number of bytes are put together sequentially to form a packet. A header is generally placed at the front of each packet. This header serves two purposes. One is to give information about where the packet is to go or where it comes from. The second is to allow for synchronization of the packet so that the bytes within the packet can be appropriately framed.

The design of transceivers which perform these functions is widely undertaken in industry but it is beyond the scope of this course. This project will focus on a part of a transceiver block. Details follow.

a) Devise a 2B-3B coding scheme that will guarantee at most 3 consecutive 0's or 1's for any input data sequence.

b) Design a circuit that will take an 8-bit wide parallel data sequence at 10K bytes/sec and serialize it using the 2B-3B coding scheme you devised in part a). You may assume that a 10KHz clock is present that is synchronous with the input data.

c) Design a receiver that will take the serial data string, decode it, and recreate an 8-bit wide data sequence at the output.

d) (extra credit) Design a "comma detect" circuit that will allow for proper framing of the received data. The "comma" should be a 12-bit code that can not represent any data sequence. The "comma" would be inserted in place of a byte in the transmitted data stream for synchronization and the receiver should frame the received data relative to the detected "comma" whenever a comma is detected.

Project 3 Transceiver block for radio controlled toys

A standard full function transmitter controller for radio controlled (RC) toys has seven controls:

- Forward
- Reverse
- Forward and Left
- Forward and Right
- Reverse and Left
- Reverse and Right
- Idle

The transmitter (TX) transmits a frame of data whenever a trigger is pressed on the controller. Each data frame consists of a

- Synchronization segment – indicates the start of new data
- Pulse segment – fixed no. of pulses for each function

  Forward: 16 pulses

  Reverse: 40 pulses

  Forward/Left: 28 pulses

Forward/Right: 34 pulses

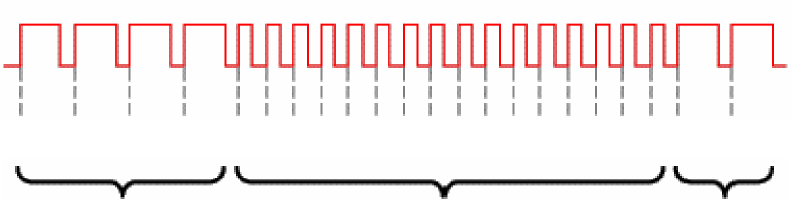
Reverse/Left: 52 pulses

Reverse/Right: 46 pulses

Idle: 22 pulses

- Repeat segment – tells the receiver (RX) to repeat the same function as long as the corresponding trigger is held.

Data burst for ‘forward’ function is shown below with timing.



Sync segment:

4 pulses each

150µs long with 50µs spacing between pulses

Pulse segment: Each pulse is 50µs long with 50µs spacing between pulses.

No. of pulses as given above for each function

Repeat segment:

2 pulses each

150µs long with 50µs spacing between pulses

The data stream is pulse modulated on to a RF carrier which is then transmitted as radio waves to the receiver. In this project, we are only concerned with the base band portions of the TX and RX that deal with the generation and detection of the bit patterns. So, the goal of this project is to:

Design the base band TX and RX blocks for the rf transceiver. The TX block should generate data bursts based on the trigger inputs for the six functions of the controller described above. The RX block should decode the information sent by the TX and generate appropriate Boolean signals to control the mechanical section of the toy.

Timing issues:

- You may assume a 10 kHz clock is present in the system (both in the TX and RX)
- Start of the frame has to be 1 ms after a trigger input is active
- The receiver has to assert a signal corresponding to the function it detects based on the no. of pulses it counts in the pulse segment. This signal has to be made active 2 ms after the end of the repeat segment is detected. Once made active this signal stays high until the start of a new frame is detected.1 ms after the start of a new frame is detected, this signal has to go low.

Other issues:

- You may include global reset for your system

- One of the seven trigger inputs is active all the time and no two trigger inputs can be active at the same time.

Project 4 Alphanumeric Display Driver

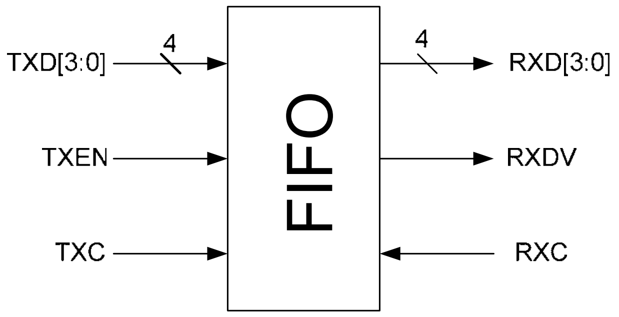
Design a circuit that will display the first and last name of you and your lab partner sequentially at a 1Hz rate on a 7-segment display. There are to be three control inputs. Controls A_0 and A_1 will provide the following operation: When A_0 is 1 and A_1 is 0, the circuit will repeatedly display one name. When A_0 is 0 and A_1 is 1, it will repeatedly display the other name. When both A_0 and A_1 are high, it will alternate between the two full names. Finally, when A_0 and A_1 are low, it will alternate between the first names of the two individuals. The third control input, Co, is to be used to specify either a common-cathode (C_0=0) or a common-anode (C_0=1)display.

Project 5 Class D audio amplifier

Design a Class D audio amplifier. The amplifier should accept stereo input signals with input amplitudes up to ±5V and should not be destroyed if the input signals were to increase up to ± 20V. There should be stereo outputs as well that can drive 8 Ω speakers. The gain should be digitally controlled in approximately 3dB increments with a maximum output power level commensurate with what can be achieved when the amplifier is biased with a single 3.5V supply. The supply voltage should be 3.5V. Class D amplifiers are receiving considerable attention recently and are digital devices that perform much as analog amplifiers but which ideally are very energy efficient. One company that sells high-end (at least expensive) Class D amplifiers can be found on the WEB site: http://www.spectronav.com. One discussion about Class D amplifiers can be found on http://www.ee.ucr.edu/~rlake/EE135/Class_D_amp_notes_AL.pdf Several pdf files will also be posted on the class web site that provide some additional details about Class D amplifiers.

Project 6 Asynchronous FIFO

The project will focus on the design of a FIFO that can be used for transmitting and receiving packets of data that are controlled by clock signals that are not at exactly the same frequency. The FIFO will be (4-bits) wide and will be able to handle packets that are of length up to n=10000 nibbles deep. The clock frequencies will be nominally 25MHz and will have a maximum offset of 600ppm. That is, they will both be in an interval 15 KHz wide and arithmetically centered at 25MHz. A block diagram of the FIFO is shown in the following figure. The pins TXC and RXC are the transmit and receive clocks respectively. TXEN is the transmit enable signal and RXDV is the receive data valid signal. When TXEN is taken high, data is clocked into the FIFO. TXEN is synchronous to the transmit clock TXC and changes on rising edges of TXC. The 4-bit input bus is TXF[3:0]. Data on this bus is valid when TXEN is high. A high value on RXDV indicates when the receive data is valid. Once RDXV foes high, the entire packet of n nibbles will be transmitted. RDXV is to be synchronous to TXR and will go high on a rising edge of TXR.



Additional details on this project will be provided within the next few days. Technical guidance on this project will come from Richard Thousand of Broadcom who has been directing digital design projects in industry for over 10 years. Richard works out of Ankeny so face-to-face interactions with him throughout this project should be possible.

.