Final Exam

- When: 8:00-10:00 am Wednesday, May 13th

- Location: here

- 1/3: midterm material
  - See lecture notes for midterm preparation
- 2/3: material since midterm

- 1 page of personal notes
- No electronic devices/books/other notes

Final Preparation

- Exam discussion on D2L
  - Post sample questions (and answers)
  - Some may appear on the exam
- Look to homework assignments and exams from last year (both the midterm and final) for the types of questions
  - Note that class coverage in previous years has been different

Pre-Midterm Material

- Basic gates
- Boolean algebra
- Digital circuits and circuit reduction
- Number representations (binary, hex)
- Arithmetic: addition and shifting
- Bit-wise operators
- Sequential logic (flip-flops)
- Components of microprocessors
- Memory behavior (input/output signals, buses, addressing)

Key Microprocessor Components

- General- versus special-purpose registers
- Instruction decoder
- Data memory (RAM)
- Program memory (EEPROM in our case)
- I/O modules
  - Digital input/output
  - Serial UART

Special-Purpose Registers

What does each do?

- Program counter
- Instruction register

Timer/Counters

- Prescalars
- Counters (hardware)
  - Timer0, timer2: 8-bit
  - Timer1: 16-bit
- Interrupts on timerX overflow

- Computing timerX count frequencies/periods
- Computing timerX interrupt frequencies/periods

Interrupts

- What are they?

- Interrupt service routines. Examples:
    - Pulse Width Modulation (PWM)
    - Producing digital signals of various frequencies (e.g., can introduce software counters, too)
    - Sensor control (sonar)

Input/Output Systems

- Polling vs interrupt-driven input/output
- Modes of communication:
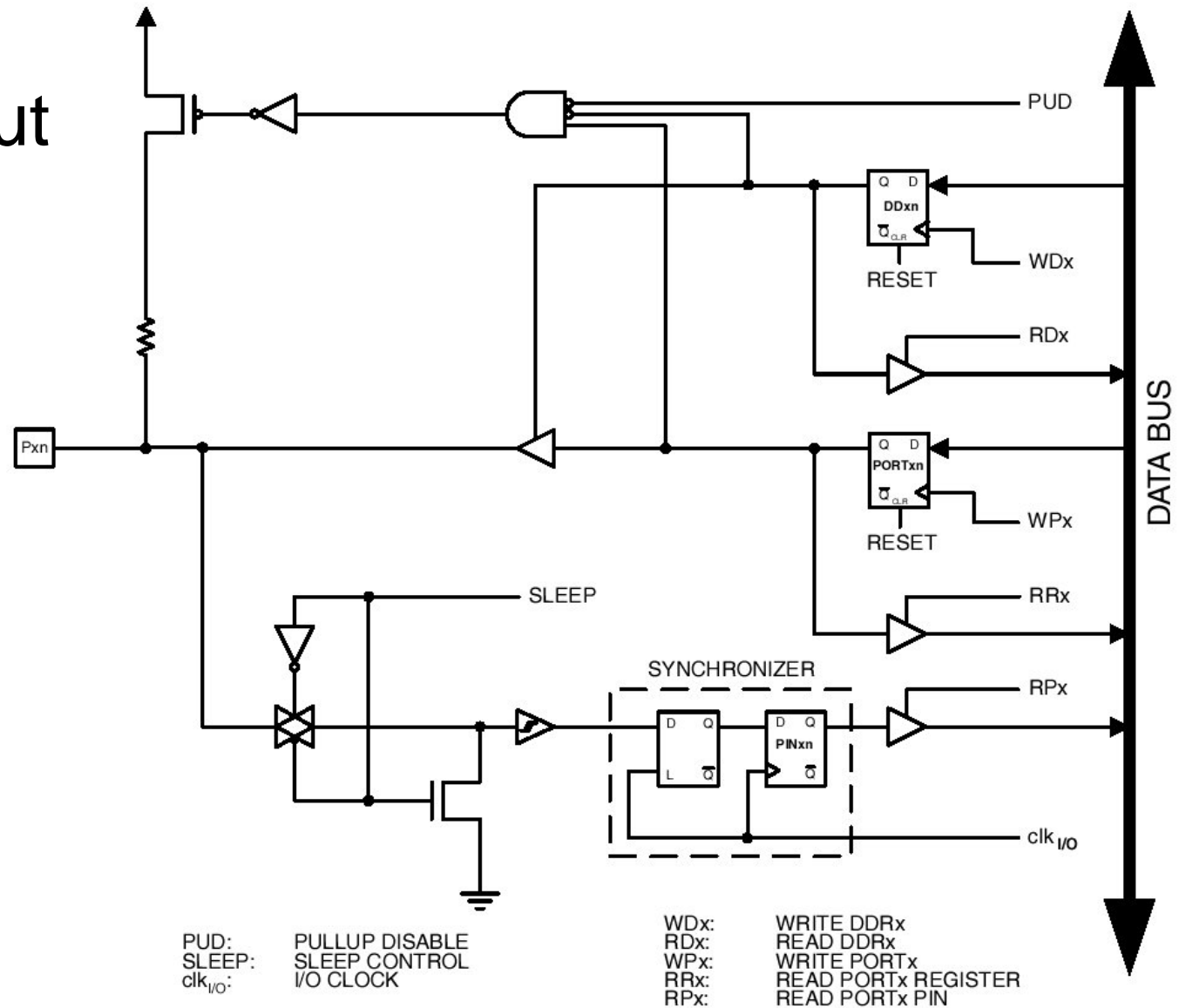 - Parallel, serial, analog, PWM

Serial Communication

- What is it?

- How does it work?
  - Start/stop

- Software implementation
  - getchar(), putchar()

Finite State Machines

- Definition
  - States
  - Events
  - Transition function
  - Outputs and output function
- State transition diagrams
- FSMs for control

Basics of Digital Port I/O

- Input/output selection
- Output value
- Input

PUD: PULLUP DISABLE
SLEEP: SLEEP CONTROL
clk$_{I/O}$:   I/O CLOCK

WDx: WRITE DDRx
RDx: READ DDRx
WPx: WRITE PORTx
RRx: READ PORTx REGISTER
RPx: READ PORTx PIN

C Code

- Be prepared to read (and possibly fix) simple C code

- Look to lecture discussions of code and your projects as you prepare

Analog Processing

- Digital to analog:
  - Resistive network