Final Exam Review - Part 1

Exam Format

Questions on the final exam may include any of the material covered in the lectures or labs. The style and difficulty of the questions will be similar to the questions in the lecture exercises, the assignments and the mid-term exam. You will have three hours to complete the exam.

The list of topics below starts with the material covered since the mid-term exam.

You should be able to:

Lectures 8 and 16: Framing, Protocol Layering and Encapsulation

- decide whether circuit- or packet-switching is most appropriate for a particular application (8)
- convert a bit stream to/from an HDLC frame including adding/removing flags and bit stuffing and de-stuffing (8)
- identify the names and functions provided by the lowest 3 layers of OSI 7-layer model (16)
- distinguish between MAC and PHY SDUs and PDUs (16)
- parse an 802.3 frame header and trailer (16)

Lecture 9: Duplexing and Modulation

- determine if a communication system is half- or full-duplex and whether it uses TDD or FDD
- for each of the following modulation techniques, give the number of bits per symbol, an equation for the time-domain waveform, a phasor or "constellation" diagram, and block diagrams for a modulator and demodulator:
  - ASK/OOK
  - BPSK
  - QPSK/QAM
  - M-ary QAM modulation
- explain why gray coding is used and assign binary codes to levels that results in a gray code
- compute the approximate bandwidth of a QAM-modulated signal based on the symbol period and the Nyquist excess bandwidth parameter α
- draw a block diagram of a quadrature down-converter and obtain the levels of the I and Q outputs from the input signal amplitude and phase
- identify an angle-modulated signal as FSK, MSK and/or GMSK based on the frequency deviation, bit rate and filtering of the modulating signal

Lecture 11: ARQ and Channel Codes

- compute Shanon capacity for a bandlimited AWGN channel
- explain effect of coding on the error rate for systems operating above and below Shanon capacity
- describe how ACK frames ensure error-free transmissions
- select an appropriate type of ARQ (from stop-and-wait, go-back-N and selective repeat) based on channel error rate and delay
- compute the value of an even or odd parity bit
- convert to/from (k,n) notation and code rate, number of data and parity bit for a block code
- compute Hamming distance between two code words
- do FEC correction by exhaustive search

Lecture 12: Polynomials in GF(2) and CRCs

- do computation using values (0/1) and operations (add/multiply) from GF(2)
- convert codewords to/from polynomials
- add, subtract, multiply and divide polynomials with coefficients from GF(2)
- compute CRCs
- draw schematic for circuits that perform multiplication and division of polynomials with coefficients from GF(2)
- use these circuits to implement a CRC generator/checker and a self-synchronizing scrambler/de-scrambler
- compute the undetected error probability for random errors and for a single error burst less than $n-k$ bits.

Lecture 13: FEC and Convolutional Codes

- compute the code rate R, constraint length (K), k, and n for a convolutional encoder
- compute the output of a convolutional encoder given the schematic, starting state, inputs and puncturing pattern
- compute coding gain
- explain the advantages of LDPC, Turbo and Reed-Solomon codes

Lecture 14: PN Sequences and Scramblers

- distinguish between noise, a random signal, a pseudo-random signal, a PN signal, a PRBS and a ML PRBS
- do computations relating the ML PRBS sequence period and the number of bits of state (flip-flops) in its LFSR generator
- select the most appropriate type of scrambler (additive vs multiplicative)
- draw the schematic for additive or self-synchronizing scrambler and de-scrambler

Lecture 15: Miscellaneous Topics

- compute result of processing of escape sequences embedded in a byte stream
- draw clock and data waveforms for a synchronous interface and identify setup and hold times.
- select appropriate flow-control methods to avoid under- and over-flow of synchronous interfaces

Lecture 17: Implementation

- explain why communication devices are increasingly being implemented using integrated circuit and digital techniques
- identify the functions of each block in a high-level diagram of a typical communication device
- select whether a particular DSP function would be more suitable for hardware or software implementation based on the sampling rate, algorithmic complexity, computational complexity, and other factors
- select the most suitable digital implementation technology (FPGA, semi-custom or custom IC) based on product volume

Lecture 18: USB

- explain the advantages of USB interfaces over "RS-232" serial interfaces