CSEE 3827: Homework 2 — Due: 2/21/11

Complete the following problems. Be sure to show your work for partial credit.

1. A two-bit wide, 2:1 mux selects between two inputs, A and B, each of which is two bits wide, placing one of the inputs onto the two bit output C. Given this interface (IN: A_1, A_0, B_1, B_0, S, OUT: C_1, C_0, draw a circuit implementing such a mux.

2. Show how $F(ABC) = \Sigma m(0, 2, 6, 7)$ can be implemented using

   (a) an 8:1 mux

   (b) a 4:1 mux

3. In this problem you are going to design an RNA translation circuit. In biology, RNA translation is the process through which trios of nucleotides (also called codons) from a strand of RNA are translated into the corresponding amino acid. We have decomposed the translation into three component steps, followed by a final assembly of each of the component modules. To implement each module, you may either draw a circuit or give the boolean expressions for the circuitry. For our and your sakes, please pick reasonably natural notations.

   (a) **BP_DEC (Base Pair Decoder)**: The BP_DEC module has a two bit input I_1, I_0 and four bits of output A, C, G, U. The two bits encode one of four possible base pairs, according to the following code:

I_1	I_0	A	C	G	U
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

   Give an implementation for BP_DEC. You may use any of the modules we've discussed in class in your design.

   (b) Trios of base pairs (or codons) can be interpreted as codewords for particular amino acids. The table below shows the codon to amino acid mappings. Note that (1) the code is redundant, i.e., there are 4^3 possible codons, but only 20 amino acids, meaning multiple codons code for the same amino acid, and (2) in addition to the amino acids, there are special Stop codons which terminate the string of amino acids that forms a protein.

   Next implement the **CODON_CODE (Codon Code)** module which takes in a trio of base pairs as twelve bits of input ($First_A$,$First_C$, $First_G$,$First_U$,$Second_A$,$Second_C$,$Second_G$,$Second_U$,$Third_A$, $Third_C$,$Third_G$,$Third_U$) and produces twenty one bits of output (Phe,Leu,Ile,Met,Val,Ser,Pro,Thr, Ala,Tyr,$Stop$,His,Gln,Asn,Lys,Asp,Glu,Cys,Trp,Arg,Gly). As above, implement CODON_CODE. Please do not give a minterm expansion, but instead simplify where possible. The structure of the table below that shows the mapping from codon to amino acid should provide some hints.

   (c) **AA_ENC (Amino acid encoder)**: This module takes twenty one bits in with one bit per amino acid plus one for the *Stop* (the same twenty one signals that are generated by the CODON_CODE module) and produces five bits of encoded output (A_4, A_4, A_2, A_1, A_0), where $Phe = 00000$,$Leu = 00001$,$Ile = 00010$,$Met = 00011$,...,$Arg = 10011$,$Gly = 10100$.

First Position	Second Position: U		C		A		G		Third Position
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
	UUC	Phe	UCC	Ser	UAC	Tyr	UGC	Cys	C
	UUA	Leu	UCA	Ser	UAA	*Stop*	UGA	*Stop*	A
	UUG	Leu	UCG	Ser	UAG	*Stop*	UGG	Trp	G
C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg	U
	CUC	Leu	CCC	Pro	CAC	His	CGC	Arg	C
	CUA	Leu	CCA	Pro	CAA	Gln	CGA	Arg	A
	CUG	Leu	CCG	Pro	CAG	Gln	CGG	Arg	G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
	AUC	Ile	ACC	Thr	AAC	Asn	AGC	Ser	C
	AUA	Ile	ACA	Thr	AAA	Lys	AGA	Arg	A
	AUG	Met	ACG	Thr	AAG	Lys	AGG	Arg	G
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly	U
	GUC	Val	GCC	Ala	GAC	Asp	GGC	Gly	C
	GUA	Val	GCA	Ala	GAA	Glu	GGA	Gly	A
	GUG	Val	GCG	Ala	GAG	Glu	GGG	Gly	G

(d) Finally, show how BP_DEC, CODON_CODE, and AA_ENC can be used to implement RNA_TRANSLATE, which translates six bits of input describing a codon (i.e., the two bit code for each of three base pairs: $I_1, I_0, J_1, J_0, K_1, K_0$) into five bits indicating the corresponding amino acid or stop code.