Homework 1

Prof. Loh
CS3220 - Processor Design - Spring 2005
Handed Out: 20 Jan 2005
Due: 27 Jan 2005

1. **Boolean functions, K-maps, Minimization, Gates**

   Consider the a five-input function $f(x_4, x_3, x_2, x_1, x_0)$ where the inputs form a 5-bit unsigned value (i.e. $X = x_4x_3x_2x_1x_0 = \sum_{i=0}^{4} x_i \cdot 2^i$). Let $f()$ be a function that evaluates to true if X is either a prime number (0 and 1 are not prime) or if X is a Fibonacci number (1,2,3,5,8,...).

   (a) Use a 5-variable Karnaugh map to minimize $f()$.

   (b) Use the Quine-McCluskey method to minimize $f()$.

   (c) Use DeMorgen's theorem to write a product-of-sums form for $\overline{f()}$.

   (d) Use the solution from 1(c) to implement a CMOS gate for $f()$. Note that the NMOS pulldown network will pull down if $\overline{f()}$ evaluates to true, which will result in a gate that implements the function $f()$.

2. **Parallel Prefix Adders**

   In class we discussed a Look-Ahead Carry Adder where the carry-propagate circuit was implemented as a binary tree. As the circuit makes its way up the tree, the wire distances that must be traveled between nodes increases. Sometimes it makes sense to collect multiple inputs together, perform a larger computation, and then move on to the next node; that is, use a trinary- or even a quartinary-tree.

   (a) Consider a node for a trinary parallel-prefix adder tree (shown on next page in Figure 1). The inputs from the "bottom" are pgk_L, pgk_M, pgk_R and the input from the top is pgk_{in}. The outputs out the bottom to the subtrees are PGK_L, PGK_M, PGK_R and the output up to the next root is PGK_{out}. Fill in three "truth tables" for the output PGK_{out}, PGK_L and PGK_M (you don't need to do PGK_R since it is always equal to pgk_{in}). Input values should be either K, P, G or X (don't care) and the output values should be K, P or G.

   Each tables should be of the form (I've filled in an arbitrary row from each as an example):

pgk_R	pgk_{in}	PGK_M
K	X	K
⋮	⋮	⋮

pgk_M	pgk_R	pgk_{in}	PGK_L
P	P	K	K
⋮	⋮	⋮	⋮

pgk_L	pgk_M	pgk_R	PGK_{out}
P	G	X	G
⋮	⋮	⋮	⋮

   Note that the inputs for each table are different. For example, the PGK_{out} signal going out of the top of the tree never cares about pgk_{in}.

   (b) Draw a 9-bit look-ahead carry adder using a trinary parallel-prefix tree (for each node, you can just use blank blocks like the figure on the next page). Like the example in class, annotate each wire in your 9-bit parallel prefix tree with a P, G or K assuming inputs of $x = 011011100$ and $y = 010110010$.

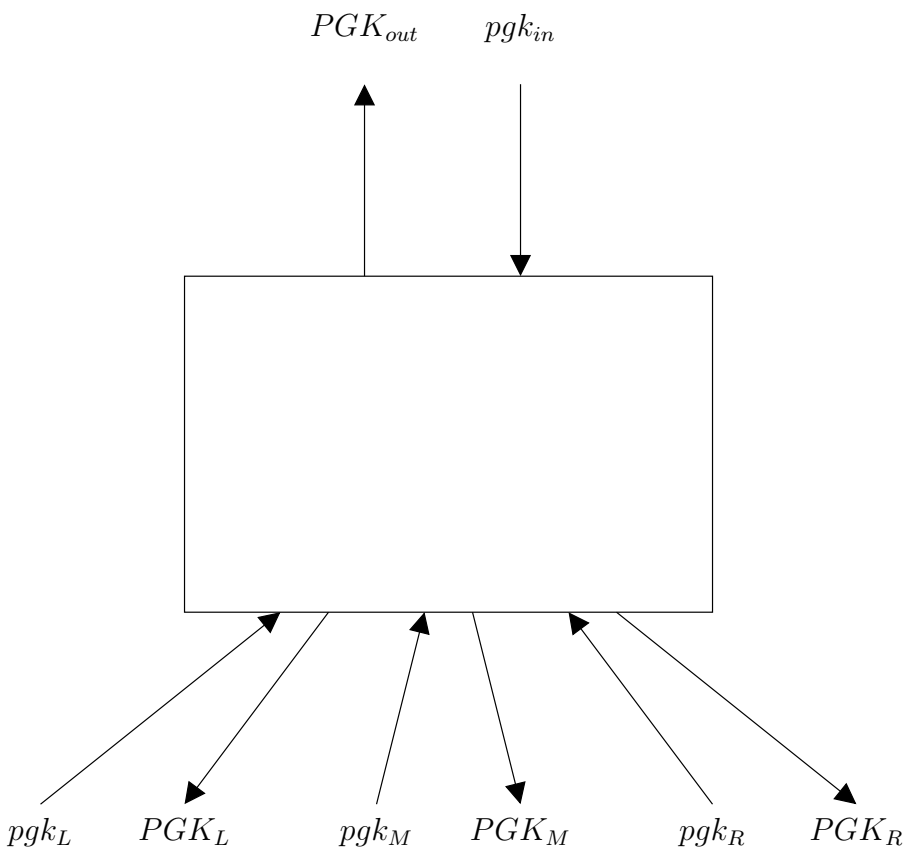

Figure 1. The trinary parallel-prefix tree node.