APPENDIX A
THE C GATE AND THE VIRTEX-5 FPGA

The Muller C gate is a sequential circuit element which switches only when all its inputs have the same value. When the inputs disagree, the gate output retains its previous value.
Any multi-input C gate can be expressed as a cascade of 2-input C gates (henceforth MULLER). This appendix describes the MULLER gate and derives its realization on the Xilinx Virtex-5 FPGA.

Boolean expressions of sequential elements need special notation to distinguish the current value and next value of signals. We denote both the signal and its current value by the signal name. Hence, a denotes the value of signal 'a' at the present instant. Let a' denote its value at the next instant and $\overline{a}$ its logical negation. Then a'' will be the value of 'a' after a'. Also, let $\oplus$, $\cdot$ and $+$ represent logical XOR, AND and OR respectively.

The MULLER gate can be modelled as a latch which is transparent to one of its inputs only when both of them agree:

$$\texttt{MULLER}(a,b)' = \overline{(a \oplus b)} \cdot a \; + \; (a \oplus b) \cdot \texttt{MULLER}(a,b) \quad (1)$$

A.1 Xilinx Virtex-5 Primitives

The Xilinx Virtex-5 FPGA contains 4 storage elements in each SLICE. A storage element can be configured as a level-sensitive latch with input driven by a LUT in the same SLICE. The latch is transparent when the control signal clock CLK is LOW. Any latch element can either be used directly as a *Transparent Data Latch with Asynchronous Clear and Present and Gate Enable* (LDCPE) primitive or inferred by the Xilinx Synthesis Tool (XST). Although XST enables optimization across modules, inferred synthesis may combine handshake stages to produce unwanted behaviour [TODO: experiment to see].

For guaranteed functionality, the LDCPE primitve must be used. The module inputs are asynchronous clear (CLR) and preset (PRE), gate enable (GE), gate (G) and data (D):

$$\texttt{LDCPE}' = \overline{\texttt{CLR}} \cdot (\texttt{PRE} + (\texttt{GE} \cdot \texttt{G}) \cdot \texttt{D}) + \overline{\texttt{CLR}} \cdot (\overline{\texttt{GE} \cdot \texttt{G}}) \cdot \texttt{LDCPE} \quad (2)$$

A straightforward way of synthesizing MULLER on Virtex-5 would be to program the LUT for the XNOR gate $\overline{(a \oplus b)}$ and connect its output to CLK.

A.2 C Gate using RS and SR Latches

Although MULLER can be implemented by a single latch and LUT, it is prone to spurious transitions in inputs. Using two latches reduces the risk of an output transition due to glitching. A well known implementation of MULLER using RS and SR latches due to Murphy [TODO: cite murphy] can be used for this purpose.

Both SR and RS latches have two inputs set (S) and reset (R). SR latch is equivalent to the RS latch with R and S inputs interchanged and the output inverted.

$$\texttt{RSLatch}(\texttt{S},\texttt{R})' = \overline{\texttt{R}} \cdot (\texttt{S} + \texttt{RSLatch}) \quad (3)$$

$$\texttt{SRLatch}(\texttt{S},\texttt{R})' = \texttt{S} + \overline{\texttt{R}} \cdot \texttt{SRLatch} \quad (4)$$

For brevity $\texttt{MULLER}(a,b)$ is denoted by c. One can easily simplify eqn. 1 (using identity $x + \overline{x} \cdot y = x + y$) to obtain:

$$\begin{aligned} \texttt{MULLER}(a,b)' &= c' \\ &= (a \cdot b + \overline{a} \cdot \overline{b}) \cdot a + (a \cdot \overline{b} + \overline{a} \cdot b) \cdot c \\ &= a \cdot (b + \overline{b} \cdot c) + \overline{a} \cdot b \cdot c \\ &= a \cdot c + b \cdot (a + \overline{a} \cdot c) \\ &= a \cdot b + (a + b) \cdot c \end{aligned} \quad (5)$$

To express MULLER in terms of SR and RS latches, observe that eqn. 5 contains only true values of inputs but R appears in both SR and RS latches (eqs. 4 and 3) as $\overline{R}$. Hence, R of the latches will be $\overline{a}$ or $\overline{b}$. Only the expression for RS latch (eqn. 3) has a minterm containing both R and S, which will produce the minterm $a \cdot b$ in eqn. 5.
Therefore, the RS latch appears in the first stage producing the minterms $a \cdot b$ and $a \cdot c$ or $b \cdot c$ (depending on R being $\overline{a}$ or $\overline{b}$ respectively). The SR latch forms the second stage. Since S appears alone in its expression, we connect the RS output to S and the input other than R of previous stage to the R of the second stage.

Representing the RS output by p and the subsequent SR output by q, we have:

$$\begin{aligned} \texttt{RSLatch}(\texttt{S} = a, \texttt{R} = \overline{b}) &= p' = a \cdot b + b \cdot p \\ \texttt{SRLatch}(\texttt{S} = p, \texttt{R} = \overline{a}) &= q' = p + a \cdot q \end{aligned} \quad (6)$$

From above, we have $q'' = p' + a' \cdot q' = (a \cdot b + b \cdot p) + a' \cdot (p + a \cdot q)$. Assuming input steady state i.e. $a'' = a' = a$ and $b'' = b' = b$,

$$\begin{aligned} q'' &= a \cdot b + a \cdot p + b \cdot p + a \cdot q \\ &= a \cdot b \cdot (1 + q) + (a + b) \cdot p + a \cdot q \\ &= a \cdot b + (a + b) \cdot p + a \cdot q + a \cdot b \cdot q \\ &= a \cdot b + (a + b) \cdot p + (a + b) \cdot a \cdot q \\ &= a \cdot b + (a + b) \cdot (p + a \cdot q) \\ &= a \cdot b + (a + b) \cdot q' \end{aligned} \quad (7)$$

This proves the equivalence of the latch pair (eqn. 6) to MULLER (eqn. 5).

It may appear that the heuristic used to connect the pair just happened to prove equivalent to MULLER. However, the circuit really was synthesized using reductions on a Signal Transition Graphs (STG) specification [TODO: cite murphy]. STGs are directed graphs with nodes corresponding to minterms in eqn. 5 [TODO: Find citation]. The synthesis algorithm does something similar to matching leaf nodes of the latch STGs with that of MULLER – like our heuristic [TODO: Find citation]. Also, note that the difference of SR and RS latch expressions (eqs. 4 and 3) helped using the heuristic. Though an RS latch can be converted to SR latch easily, it would be much harder to build the pair using 2 SR latches.

The LDCPE primitive can be easily configured as RS and SR latches. To set $\texttt{LDCPE} = \texttt{RSLatch}(\texttt{S},\texttt{R})$ by comparing eqs. 2 and 3, we set $\texttt{CLR} = \texttt{R}$ and $\overline{\texttt{GE} \cdot \texttt{G}} = 1$. Then, $\texttt{PRE} = \texttt{S}$. Similarly, for SRLatch set $\overline{\texttt{CLR}} = 1, (\texttt{GE} \cdot \texttt{G}) = \texttt{R}, \texttt{D} = 0$ and $\texttt{PRE} = \texttt{S}$.