QAN2
Counter Designs in the pASIC Device

HIGHLIGHTS

- **Free running counters** – High-speed counters optimized for binary counting at frequencies in excess of 100 MHz.
- **Counters with added features** – Binary counters with LOAD for data inputs, COUNT ENABLE, UP/DOWN count capability, 3-State output control, synchronous and asynchronous clear inputs.
- **Counter Macro Library** – A comprehensive library of QuickLogic counters exists as ready-made designs for instant systems applications.
- **Counter Design Methodology for pASIC™ devices** – Introduction of techniques to enhance the performance of counter design. How to use look-ahead and pipelined carry to decrease propagation delay between counter modules.

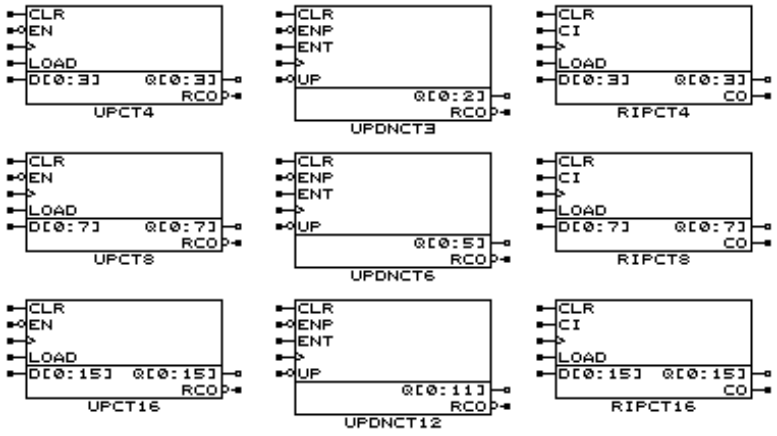
INTRODUCTION

The following application note introduces several high-speed techniques for both counter and state machine design using the QuickLogic pASIC 1 family of FPGA's. This application note uses the QL8x12 1.2 micron device in all of the examples. Since the original publication of this application note, QuickLogic has moved to a 0.65 micron process which has decreased propogation delays by 30%-50% and can achieve worst case counter speeds in excess of 180 Mhz. The techniques displayed here are worthy of merit and the application note has been retained in its original form.

The low-impedance ViaLink™ interconnect element employed in the pASIC device architecture enables higher performance operation than any other FPGA family. This is particularly evident in the design of high-speed counters. The pASIC macro library contains a range of predesigned counters covering a wide variety of needs. Examples of nine of these are shown below. This QuickNote is intended as an overview of alternative approaches to the pASIC design methodology of functions that cannot be satisfied by these counter modules.

QUICKLOGIC

Counters from QuickLogic's Macro Library



Designs in this application note were implemented with the QuickLogic pASIC Toolkit operating under Microsoft Windows™ 3.0 on the PC. This comprehensive set of CAE software includes third-party design entry (ECS from Data I/O, Inc.) and timing and functional simulation (X-SIM from SAS, Inc.). Both operate efficiently and interactively with QuickLogic SpDE place and route, delay modeling and physical viewer tools. The ability to enter and simulate in the same graphical environment provides the user with a quick and efficient way of generating and debugging counter designs.

After SpDE place and route of the pASIC device, timing values may be generated. Compiling net, gate and ViaLink propagation delays provides timing parameters that are a function of the layout and partitioning of the cells in the pASIC device. These timing values can be annotated back into the simulator. Having done this, the designer can evaluate the performance of his counter with regard to maximum count frequency, clock to output, input set up and hold times. If the counter's performance needs improvement the ECS schematic capture environment may be invoked to manually improve the placement of registers or clock input buffers. An optimum placement for counters would have registers placed in a column with clock inputs driven from a clock express line to minimize clock skew.

Figure 1 shows a block diagram of a Moore state machine. The next state of the registered Q outputs is a function of combinatorial inputs gated with the current state of the same Q outputs. A counter is a state machine that conforms to this structure. The state of the inputs combine with the Q registered feedback to provide the next output state. Depending on counter complexity, a design can have combinatorial inputs; CLEAR, HOLD, COUNT ENable, UP/DOWN control, and for loadable counters DATA inputs and a LOAD control.

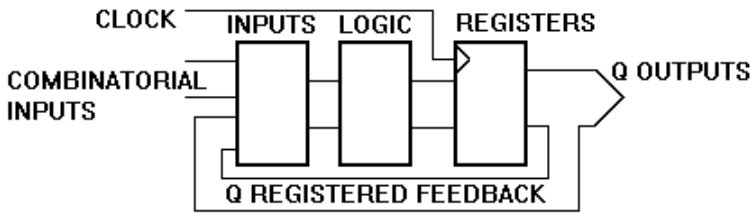


FIGURE 1
Moore State Machine

...State machine design in the pASIC device

Implementation of this type of state machine is well suited to the pASIC device that has internal logic cells comprising logic gates, multiplexers and registers. Universal cell interconnect is possible through vertical and horizontal routing channels and programmable ViaLink sites. Logic and registers combine with interconnect to realize state machines and counters of varying complexity. Position constraints of the registered cells can be entered into the schematic along with the design itself, thus ensuring optimum placement of cell groups.

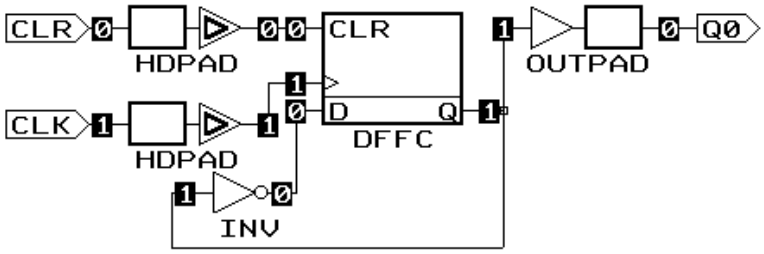
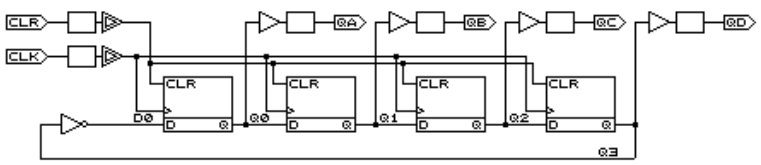


FIGURE 2
Binary Flip-flop

Figure 2 shows a binary flip-flop as entered in the ECS schematic capture system. It conforms to the Moore state machine of Figure 1. The CLR line is used as a direct input to reset the register and the Q0 output is inverted and fed back to drive the D input of the same register which causes it to toggle after each clock rising edge. This circuit forms the least significant bit of a free-running binary counter. To optimize a counter design for maximum performance the designer should take advantage of the high drive input buffer (HDPAD) which rapidly charges/discharges the low capacitance on the dedicated express clock lines. High current drive minimizes clock skew on these lines that cover the entire length and breadth of the pASIC device.

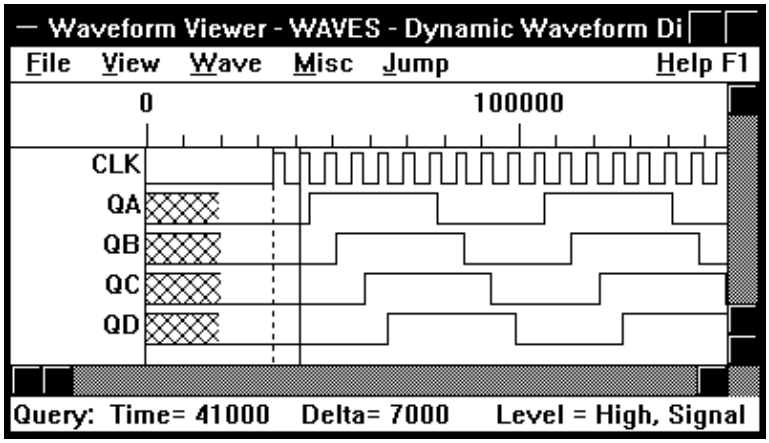
The X-SIM simulator allows logic ONEs and ZEROs to be displayed over the signal lines so the designer can trace the behavior of his system dynamically, as in Figure 2.

FIGURE 3
Four-bit Johnson counter design



The circuit shown in Figure 3 is a Johnson counter implemented in the pASIC device. It consists of four D-type registers linked as a shift register with an inverting feedback from Q3 into the D0 input of register Q0. After the registers have been cleared, the first clock pulse will strobe a logic HIGH into Q0 and a further three clock pulses will propagate that HIGH through the other registers. When Q3 goes HIGH the feedback to D0 will be inverted and a LOW will be clocked into Q0. The subsequent clock pulses will sequentially clock a LOW condition through the shift register as shown in the following simulation. This design comprises four registers and provides eight distinct states through which the counter can transition.

Simulation of the Johnson counter



The advantages of the Johnson counter are found in its simplicity and very high performance. The light capacitive loading on each register output and the lack of combinatorial logic delays in the feedback loop make this design capable of clocking at well over 100 MHz in the pASIC device. A disadvantage is in the number of states through which the counter can transition. If n represents the number of registers in the counter, then the Johnson counter can clock through 2n states as opposed to 2^n found in a binary counter with n registers.

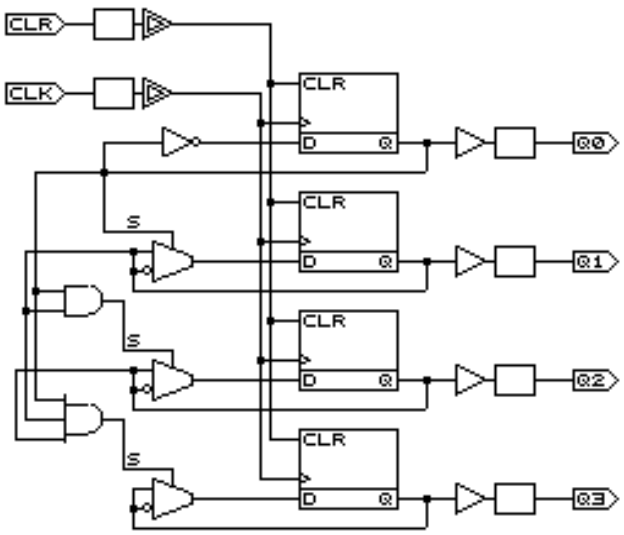
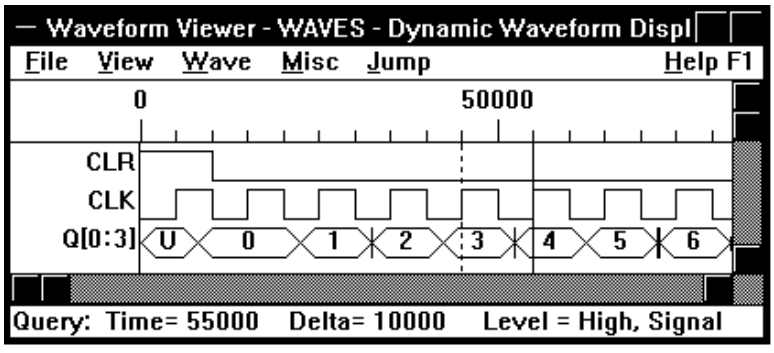


FIGURE 4
Free-running four-bit binary counter with clear

Figure 4 shows a binary counter with an asynchronous CLEAR input. In this design a register is required to maintain or HOLD its current contents until all the lesser significant registers become HIGH. Then the register is required to change its state or TOGGLE after the next clock edge. The least significant stage of this design features the binary flip-flop that is given in Figure 2 and provides the output for Q0.



Simulation waveforms of the four-bit free-running counter clocking at 100 MHz

The TOGGLE and HOLD functions for registered outputs Q1, Q2 and Q3 are achieved with a 2:1 multiplexer feedback. An AND gate control drives the S (Select) input of each multiplexer. When the AND gate output is HIGH, the inverting feedback path through the multiplexer is selected causing the register output to TOGGLE, otherwise a HOLD function is maintained through the non-inverting route. This four-bit counter shows a very efficient use of the pASIC cell because only four of them are required to perform this function. The design when simulated with a clock input with a period of 10,000 pico seconds (10ns), showed 100 MHz counter performance.

A Gray Code sequence allows only one bit in the pattern to change as one state proceeds to the next. This type of encoding can safeguard against simultaneous output driver transition. By definition only one bit can change so only one output buffer will transition after each clock pulse.

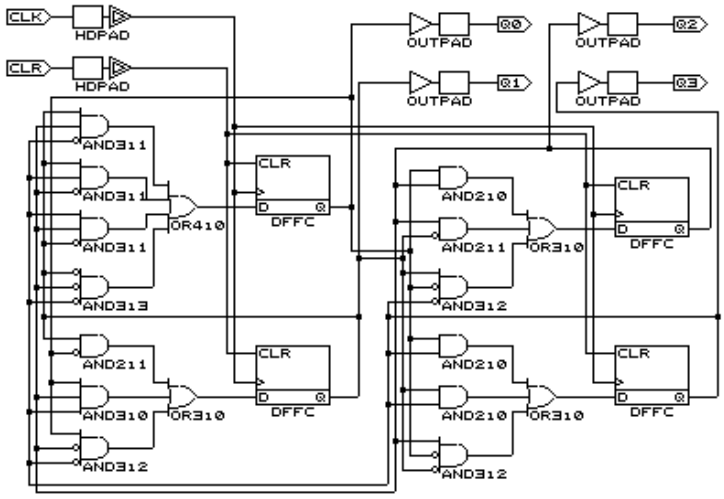
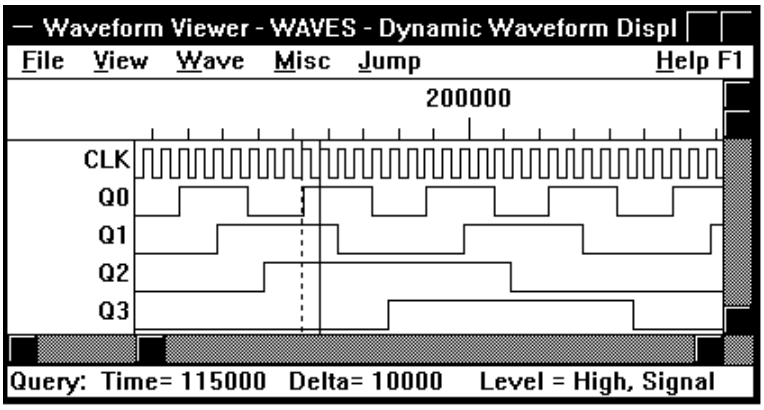


FIGURE 5
Four-bit Gray code counter in pASIC device

Figure 5 shows a Gray code counter as a state machine design. The conventional "sum of product terms" has been used to encode the registered feedback and provide the correct sequence states.



Four-bit Gray code counter simulation

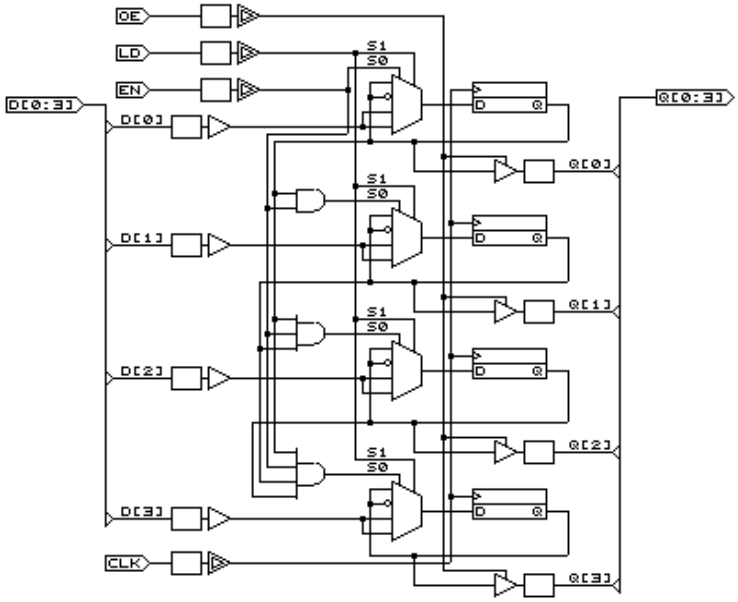
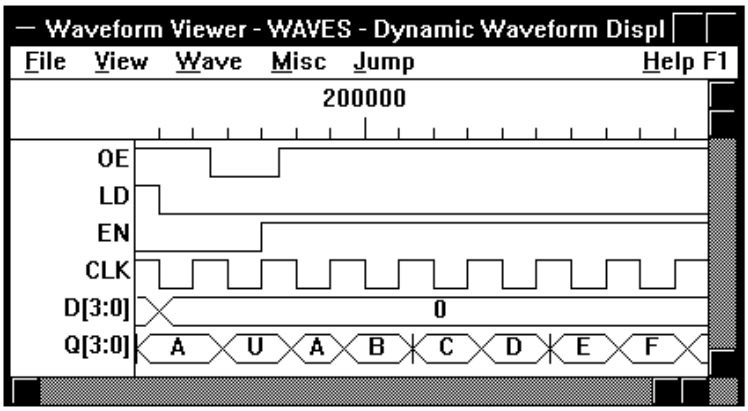


FIGURE 6
Loadable counter with count and output enable

Figure 6 shows a counter having a LOAD/COUNT function selected through 4:1 multiplexers that drive each register in the counter. When the LOAD input is HIGH, the data on the D0:D3 input bus is selected through the multiplexers. The clock rising edge will synchronously load the registers Q0:Q3. To disable a LOAD function, the LD input must be LOW, enabling the COUNT function. The EN input allows the counter to increment or hold. Finally the OE input gives an active HIGH enable to the 3-State output buffers (TRIPAD) given in Figure 6.

Counter simulation load hold 3-State count



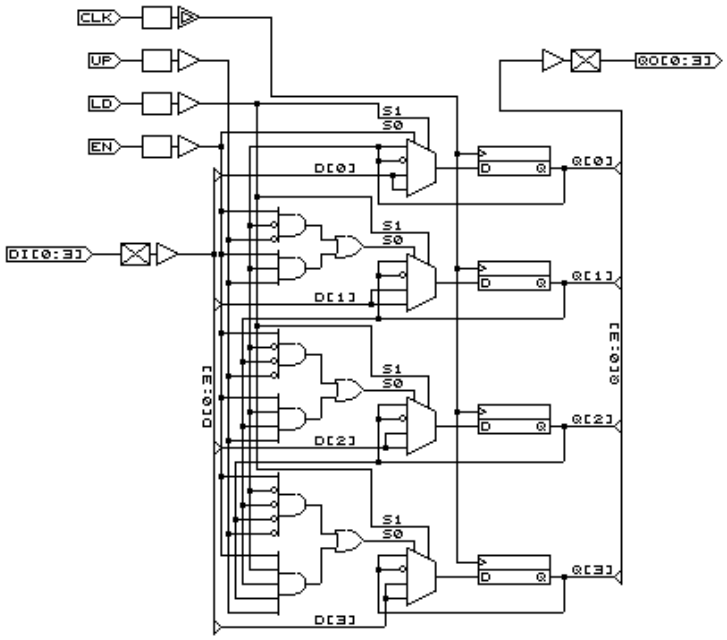


FIGURE 7 Loadable up/down counter

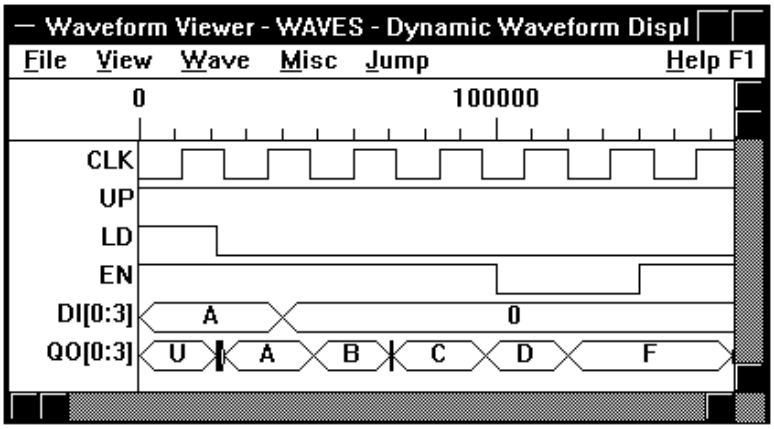
Synchronous loadable up/down counter

Figure 7 shows a binary four-bit loadable UP/DOWN counter with a count enable, EN, input. Data inputs DI[0:3] are tied to a bus and enter the pASIC device through a four-bit wide input buffer. Individual data lines D[0], D[1], D[2], and D[3] drive inputs 3 and 4 of the multiplexer circuits. The LD input,

when HIGH, will select these data lines to drive the D-type register inputs. A synchronous clock loads D[0:3] Data inputs to the registered Q[0:3] outputs. When the registers are loaded, the LD input may be taken inactive LOW. To enable the count function, the EN input must be driven HIGH. This input provides an enabling HIGH to all the AND gates shown in Figure 7. When LOW, this signal maintains a HOLD condition on all four registers. In a binary counter a register is required to TOGGLE after all the less significant registers become HIGH, but this circuit can count either way, UP or DOWN. In a down count a register TOGGLES after all the less significant registers become LOW. The circuit controlling the TOGGLE function of each register comprises a sum of two product terms and drives the S0 input of each multiplexer. One of the two AND gates will be enabled by a logic HIGH on the UP control and the second AND gate enabled for the DOWN count. The UP input is a dual function pin, UP and NOT DOWN. When HIGH, this input selects the UP count and when LOW, selects the down count.

Synchronous loadable up/down counter

It should be noted that in both Figures 6 and 7, buses have been used for data inputs and register outputs. Combining signals on a bus and using components in a group can help simplify the schematic diagrams in complex circuits. For deeper counters such as eight bits, sixteen bits and above, it is recommended that the designer use buses to improve the clarity of the circuit design.

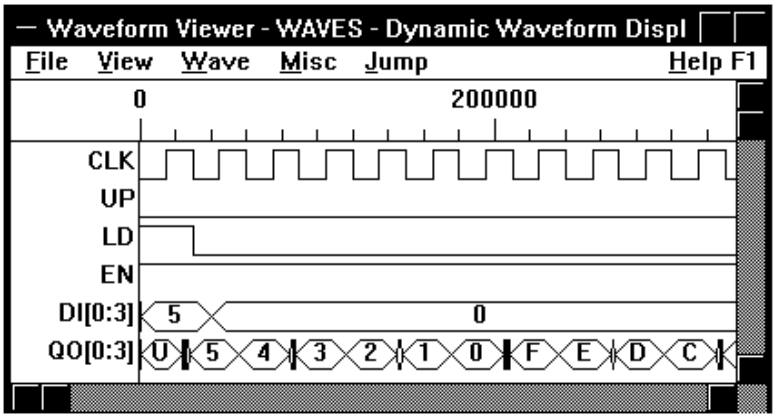


Simulation showing up count, load, count enable

5

Application Notes

Simulation showing counter load, down count, wrap-around



Eight-bit binary counter with synchronous clear

The design methodology used for integrating deeper counters into the pASIC device employs a technique of interlacing internal registered HIGH and LOW conditions. The wider AND gates in the pASIC cell library have both true and complement inputs up to the widest AND gate which is the AND14i7. This gate has a total of fourteen inputs, seven inverting and seven noninverting. The counter shown in Figure 8 has outputs Q0:Q7 and has been designed such that Q0:Q3 function on LOW logic levels, and Q4:Q7 on HIGHs. When the SCL input is driven active HIGH it will select a logic LOW for registers Q0:Q3, and a HIGH for registers Q4:Q7 via the multiplexer inputs 3 and 4. If registered outputs Q4:Q7 drive inverting output buffers and Q0:Q3 noninverting buffers then all the buffer output pins will be driven LOW. To an external system, this counter is designed to function as a conventional binary counter, but internally the groups Q0:Q3 and Q4:Q7 function on interlaced LOW and HIGH logic group levels, respectively.

During synchronous count operations, the AND gates controlling the TOGGLE function also require interlaced true and complement inputs. Just as Q4:Q7 require inverting output buffers, the same internal TOGGLE control AND gates invert Q4:Q7 signals. An example is given in the TOGGLE control gate for Q6 which combines Q0:Q3 and NOT Q4 and NOT Q5 as an AND gate function of all six input variables.

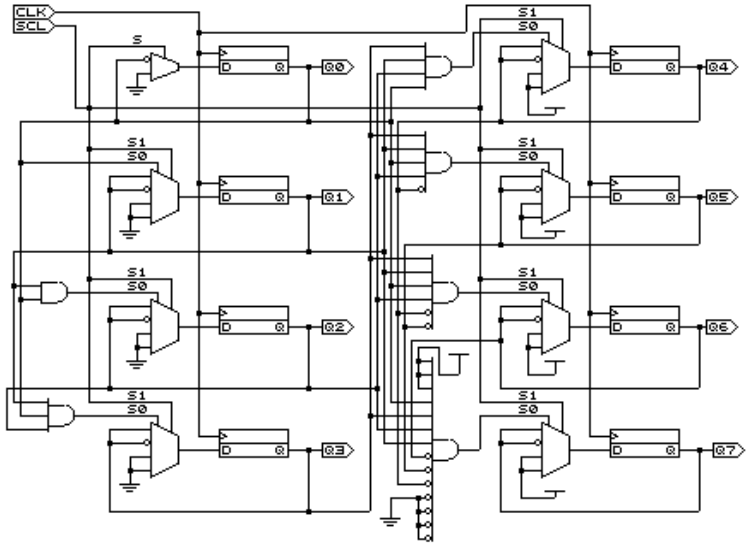


FIGURE 8
Eight-bit binary counter

...Tie unused logic inputs to Vcc or GND

The TOGGLE function to register Q7 uses the 14i7 AND gate. In any circuit design the unused inverting or noninverting inputs must always be tied to Vcc or GND. This is shown for the 14i7 AND gate in Figure 8. Any unused input to a logic cell can be tied to create a permanent HIGH or LOW enable condition for the other signal inputs. The way in which the designer interlaces the logic HIGH and LOW conditions in the counter design is also important. An obvious way would be to consider even outputs Q0, Q2, Q4 ... functioning on logic HIGH levels and odd outputs Q1, Q3, Q5 ... on logic LOWs. This way of interlacing a counter is on a bit-by-bit basis and could cause problems if there were a requirement for the registers to be bussed onto pin driver groups. Interlacing a four-bit-wide group , Q0:Q3, and Q4:Q7 allows a bus of four bits wide to drive a pin driver group four bits in width. Interlacing on alternate bits would prevent the designer taking advantage of the bus feature.

The circuit given in Figure 8 can be made into a symbol and used as an eight-bit counter module or block with CLK and SCL inputs. The design has no count ENable input and is a free-running counter toggling on each clock cycle. A designer can develop deeper counters, 16, 24 and 32 bits by combining eight-bit counter modules. If a 16-bit counter is created from two eight-bit modules, then the higher order eight-bit module must be prevented from incrementing until the lower order counter has reached its final count. An ENable input is an additional control that is required by the higher order byte counter. The counter shown in Figure 8 may be developed to incorporate an additional ENable input. If each TOGGLE AND gate has one additional input to control the TOGGLE or HOLD function, then that control can be used

as the ENable input. An AND gate will be required for the register and multiplexer combination driving Q1. The S0 input of the multiplexer will be selected by Q0 AND the ENable function. For Q0, a four-input multiplexer should be added with the ENable input driving its S0 input and the S1 input being driven by SCL. This multiplexer has the same connections as Q1, Q2 and Q3.

Pipelined carry generation

The way in which a carry output is generated, from the lower order eight-bit counter, and propagated to the ENable input of the next stage is crucial in determining the performance of the two stages. One method would be to detect the final count of the lower order counter and feed an enabling signal to the next stage. This requires one clock period for gating and propagation. A more subtle approach would be to detect the count value prior to the final count. This penultimate carry bit is generated one clock pulse before the last count, so a D-type register is required to delay its propagation to the next eight-bit stage. Carry generation and propagation can be extended over two clock cycles, effectively halving the delay path.

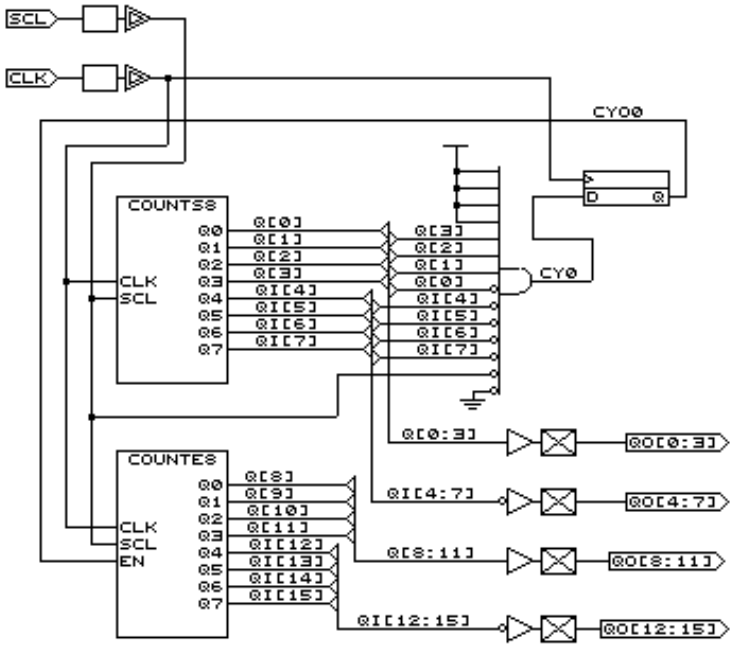


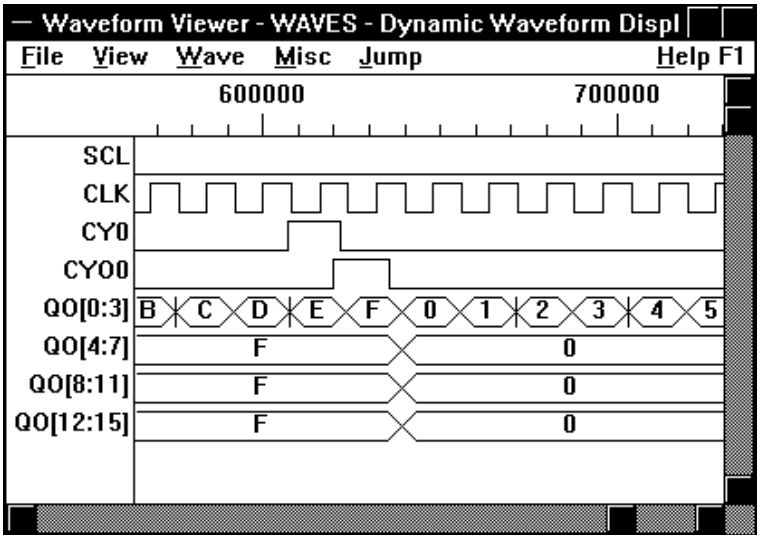
FIGURE 9
16-bit counter from two eight-bit modules and a pipelined look-ahead carry

One AND gate and a D-type register combination is used to decode and generate and pipeline the carry CY0/CYO0. This is decoded from the least significant eight-bit counter output bus. The technique is also called 'carry anticipate' and its application is to reduce the carry propagation delay to the input of the next counter stage. The eight-bit counter module in Figure 8 is given in Figure 9 as a macro COUNTS8. An additional eight-bit counter has been added to provide outputs QO8-QO15. The macro COUNTS8 is identical to COUNTE8 apart from one additional input. The EN input is an active HIGH count enable input and is driven from the look-ahead pipeline carry register. This counter will HOLD its current contents when EN is LOW and increment when EN goes HIGH. From Figure 9 the bus nets Q[0:3] and Q[8:11] are non-inverted while QI[4:7] and QI[12:15] give inverted outputs. The output buffers correct the logic polarity to provide a conventional binary output code.

When considering the performance of this counter design it should be noted that the internal set up time for the registered outputs Q[8:11] and QI[12:15], of module COUNTE8, is 256 clock cycles. The propagation of the carry bit then becomes the critical performance factor in the linking of the two counters.

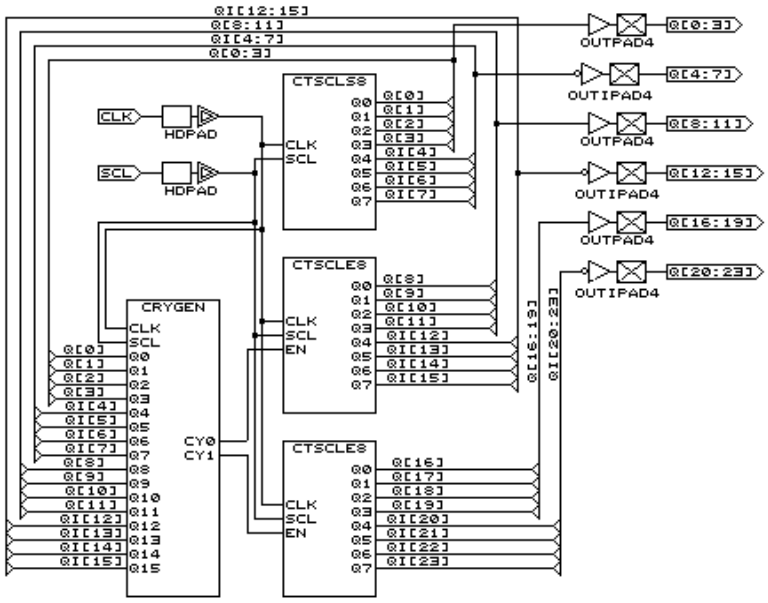
...Carry generation and propagation is critical in determining performance

The simulation shows the sixteen-bit counter outputs and the look-ahead carry CYO which decodes the hexadecimal count 'FFFE'. The D-type register delays this signal by one clock edge, so its output CYO0 goes HIGH when the counter reaches 'FFFF'. The counter outputs "roll over" to '0000' after the next clock transition and a logic zero is clocked into the carry register. So carry generation and propagation takes two clock cycles.



Simulation of the 16-bit counter showing the pipelined carry

FIGURE 10
24-bit counter form three eight-bit modules



Look-ahead carry for a 24-bit counter

Figure 10 takes the counter depth to 24 bits using a similar look-ahead technique. In this example a carry generator macro CRYGEN has been created to decode the count before the penultimate count. If 'FD' hexadecimal is decoded from the least significant byte, as opposed to 'FE' then two registers are required to pipeline the early carry. Its propagation delay is then extended to three clock cycles increasing the time available to enable the higher order counters. The higher order byte counters in the chain have a set up time through internal feedback of 256 count periods. This makes the look-ahead carry generation propagation delay the significant determining factor in the counter's overall performance.

For a 32-bit design, the designer could decode 'FC' from the least significant eight-bit stage and pipeline the carry through three registered stages if it gives any benefit to the system performance. In a design that uses the technique of a pipelined carry it should be noted that a synchronous clear should flush the carry registers as well as the counter registers. The CRYGEN MACRO has a SCL (synchronous clear) input to perform this function, Figure 11.

Additional considerations

For deep loadable counters that use look-ahead carry, the designer should distinguish between a count and a load function. When the count ENable is valid, the carry bit is generated from the look-ahead condition at the registered outputs. For a LOAD, the carry input is created from a HIGH condition on all the data inputs of the lesser significant stage.

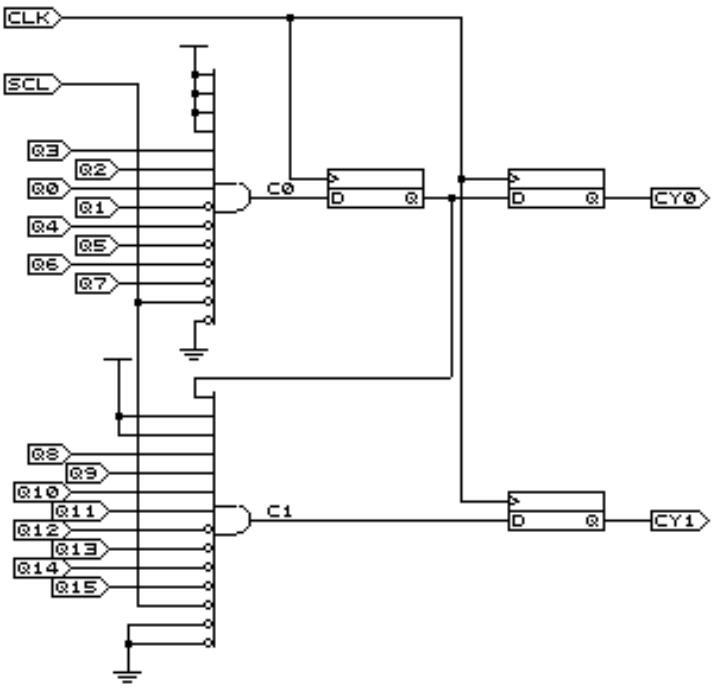
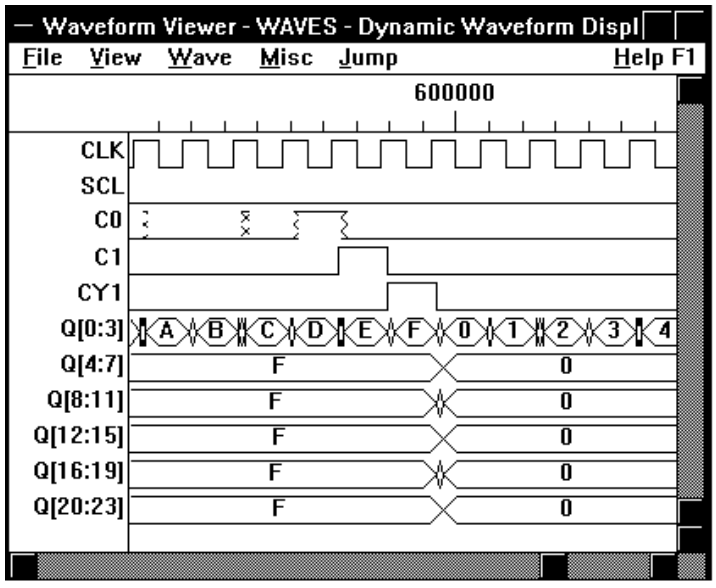


FIGURE 11
Pipelined look-ahead carry generator CRYGEN



Simulation of the 24-bit counter