Overview of NULL Convention Logic (NCL)

NCL is a ***delay-insensitive (DI)*** asynchronous (i.e. clockless) paradigm, which means that NCL circuits will operate correctly regardless of when circuit inputs become available; therefore NCL circuits are said to be correct-by-construction (i.e. no timing analysis is necessary for correct operation). NCL circuits utilize dual-rail or quad-rail logic to achieve delay-insensitivity. A dual-rail signal, D, consists of two wires, D^0 and D^1, which may assume any value from the set {DATA0, DATA1, NULL}. The DATA0 state ($D^0 = 1$, $D^1 = 0$) corresponds to a Boolean logic 0, the DATA1 state ($D^0 = 0$, $D^1 = 1$) corresponds to a Boolean logic 1, and the NULL state ($D^0 = 0$, $D^1 = 0$) corresponds to the empty set meaning that the value of D is not yet available. The two rails are mutually exclusive, such that both rails can never be asserted simultaneously; this state is defined as an illegal state. A quad-rail signal, Q, consists of four wires, Q^0, Q^1, Q^2, and Q^3, which may assume any value from the set {DATA0, DATA1, DATA2, DATA3, NULL}. The DATA0 state ($Q^0 = 1$, $Q^1 = 0$, $Q^2 = 0$, $Q^3 = 0$) corresponds to two Boolean logic signals, X and Y, where X = 0 and Y = 0. The DATA1 state ($Q^0 = 0$, $Q^1 = 1$, $Q^2 = 0$, $Q^3 = 0$) corresponds to X = 0 and Y = 1. The DATA2 state ($Q^0 = 0$, $Q^1 = 0$, $Q^2 = 1$, $Q^3 = 0$) corresponds to X = 1 and Y = 0. The DATA3 state ($Q^0 = 0$, $Q^1 = 0$, $Q^2 = 0$, $Q^3 = 1$) corresponds to X = 1 and Y = 1, and the NULL state ($Q^0 = 0$, $Q^1 = 0$, $Q^2 = 0$, $Q^3 = 0$) corresponds to the empty set meaning that the result is not yet available. The four rails of a quad-rail NCL signal are mutually exclusive, such that no two rails can ever be asserted simultaneously; these states are defined as illegal states. Both dual-rail and quad-rail signals are space optimal 1-hot delay-insensitive codes, requiring two wires per bit.

The framework for NCL systems consist of DI combinational logic sandwiched between DI registers, as shown in Figure 1, which is very similar to synchronous systems.

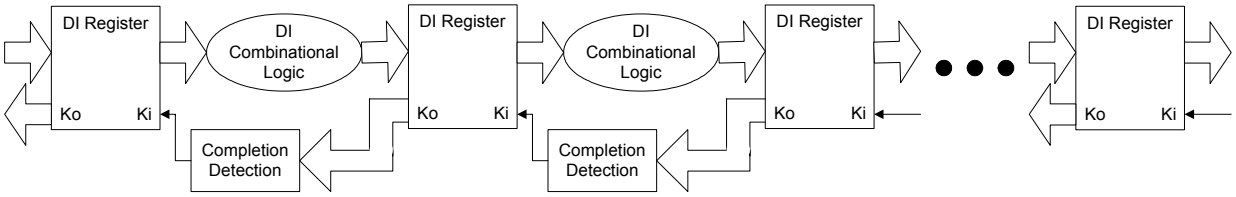


Figure 1. NCL system framework: input wavefronts are controlled by local handshaking signals and Completion Detection instead of by a global clock signal. Feedback requires at least three DI registers in the feedback loop to prevent deadlock.

NCL circuits are comprised of 27 fundamental gates, as shown in Table 1, which constitute the set of all functions consisting of four or fewer variables. Since each rail of an NCL signal is considered a separate variable, a four variable function is not the same as a function of four literals, which would normally consist of eight variables. The primary type of threshold gate, shown in Figure 2, is the *THmn gate*, where $1 \le m \le n$. THmn gates have n inputs. At least m of the n inputs must be asserted before the output will become asserted. In a THmn gate, each of the n inputs is connected to the rounded portion of the gate; the output emanates from the pointed end of the gate; and the gate's threshold value, m, is written inside of the gate.

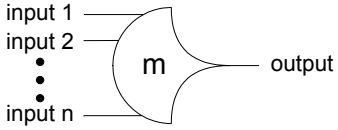


Figure 2. THmn threshold gate.

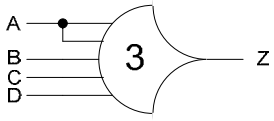


Figure 3. TH34w2 threshold gate: Z = AB + AC + AD + BCD

Another type of threshold gate is referred to as a weighted threshold gate, denoted as $THmnWw_1w_2\ldots w_R$. Weighted threshold gates have an integer value, $m \ge w_R > 1$, applied to *inputR*. Here $1 \le R < n$; where n is the number of inputs; m is the gate's threshold; and $w_1, w_2, \ldots w_R$, each > 1, are the integer weights of *input1, input2*, … *inputR*, respectively. For example, consider the TH34W2 gate, whose $n = 4$ inputs are labeled A, B, C, and D, shown in Figure 3. The weight of input A, $W(A)$, is therefore 2. Since the gate's threshold, m, is 3, this implies that in order for the output to be asserted, either inputs B, C, and D must all be asserted, or input A must be asserted along with any other input, B, C, or D. NCL threshold gates are designed with *hysteresis* state-holding capability, such that after the output is asserted, all inputs must be deasserted before the output will be deasserted. Hysteresis ensures a

complete transition of inputs back to NULL before asserting the output associated with the next wavefront of input data. Therefore, a THnn gate is equivalent to an n-input C-element (i.e. when all inputs are asserted the output is asserted; the output then remains asserted until all inputs are deasserted, at which time the output becomes deasserted); and a TH1n gate is equivalent to an n-input OR gate. NCL threshold gates may also include a *reset* input to initialize the output. Circuit diagrams designate resettable gates by either a *d* or an *n* appearing inside the gate, along with the gate's threshold. *d* denotes the gate as being reset to logic 1; *n*, to logic 0. These resettable gates are used in the design of DI registers.

Table 1. 27 fundamental NCL gates.

NCL Gate	Boolean Function
TH12	A + B
TH22	AB
TH13	A + B + C
TH23	AB + AC + BC
TH33	ABC
TH23w2	A + BC
TH33w2	AB + AC
TH14	A + B + C + D
TH24	AB + AC + AD + BC + BD + CD
TH34	ABC + ABD + ACD + BCD
TH44	ABCD
TH24w2	A + BC + BD + CD
TH34w2	AB + AC + AD + BCD
TH44w2	ABC + ABD + ACD
TH34w3	A + BCD
TH44w3	AB + AC + AD
TH24w22	A + B + CD
TH34w22	AB + AC + AD + BC + BD
TH44w22	AB + ACD + BCD
TH54w22	ABC + ABD
TH34w32	A + BC + BD
TH54w32	AB + ACD
TH44w322	AB + AC + AD + BC
TH54w322	AB + AC + BCD
THxor0	AB + CD
THand0	AB + BC + AD
TH24comp	AC + BC + AD + BD

NCL systems contain at least two DI registers, one at both the input and at the output. Two adjacent register stages interact through their request and acknowledge signals, K_i and K_o, respectively, to prevent the current DATA wavefront from overwriting the previous DATA wavefront, by ensuring that the two DATA wavefronts are always separated by a NULL wavefront. The acknowledge signals are combined in the Completion Detection circuitry to produce the request signal(s) to the previous register stage. NCL registration is realized through cascaded arrangements of single-bit dual-rail registers or single-signal quad-rail registers, depicted in Figures 4 and 5, respectively. These registers consist of TH22 gates that pass a DATA value at the input only when K_i is *request for data* (rfd) (i.e. logic 1) and likewise pass NULL only when K_i is *request for null* (rfn) (i.e. logic 0). They also contain a NOR gate to generate K_o, which is *rfn* when the register output is DATA and *rfd* when the register output is NULL. The registers shown below are reset to NULL, since all TH22 gates are reset to logic 0. However, either register could be instead reset to a DATA value by replacing exactly one of the TH22n gates with a TH22d gate.

An N-bit register stage, comprised of *N* single-bit dual-rail NCL registers, requires *N* completion signals, one for each bit. The NCL completion component, shown in Figure 6, uses these *N* K_o lines to detect complete DATA

and NULL sets at the output of every register stage and request the next NULL and DATA set, respectively. In full-word completion, the single-bit output of the completion component is connected to all K_i lines of the previous register stage. Since the maximum input threshold gate is the TH44 gate, the number of logic levels in the completion component for an N-bit register is given by $\lceil \log_4 N \rceil$. Likewise, the completion component for an N-bit quad-rail registration stage requires $\frac{N}{2}$ inputs, and can be realized in a similar fashion using TH44 gates. Figures 7 and 8 show the flow of DATA and NULL wavefronts through an NCL combinational circuit (i.e. an AND function) and an arbitrary pipeline stage, respectively.

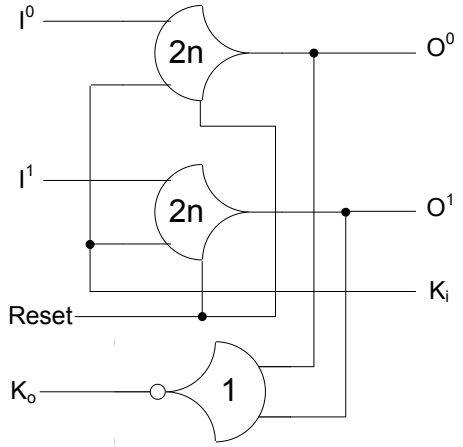


Figure 4. Single-bit dual-rail register.

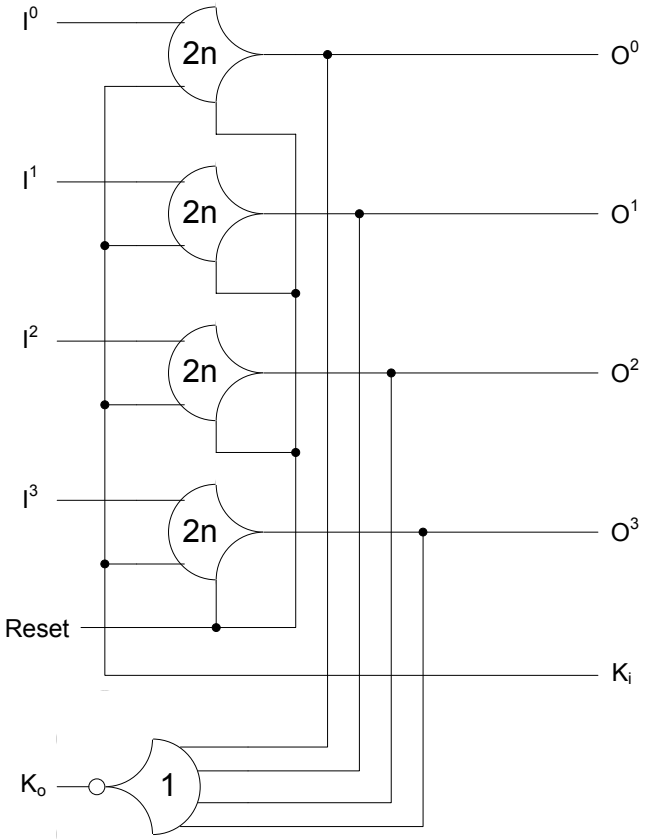


Figure 5. Single-signal quad-rail register.

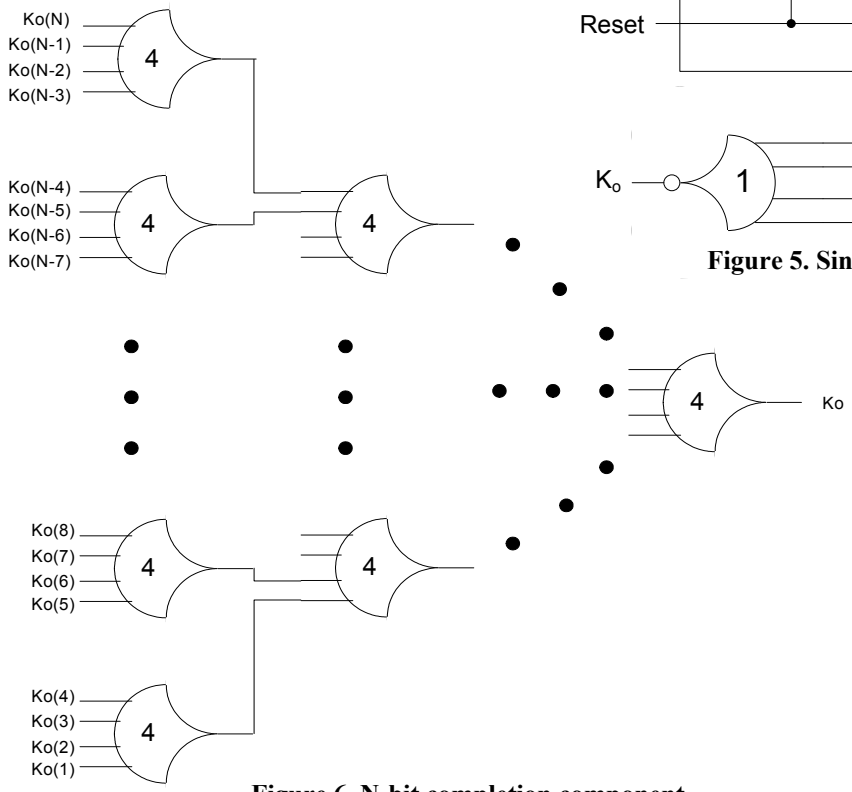


Figure 6. N-bit completion component.

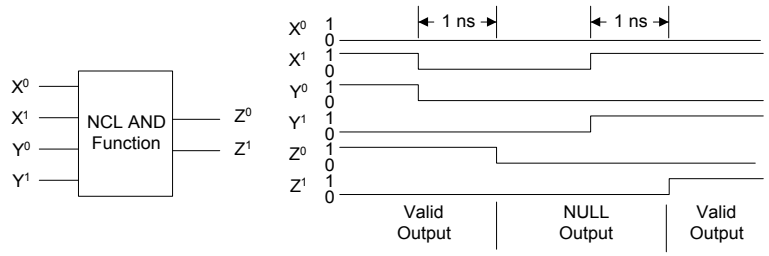
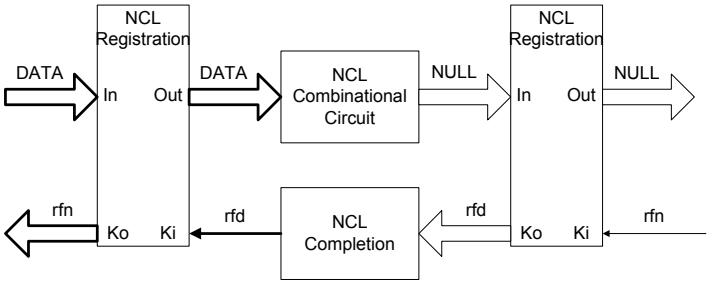
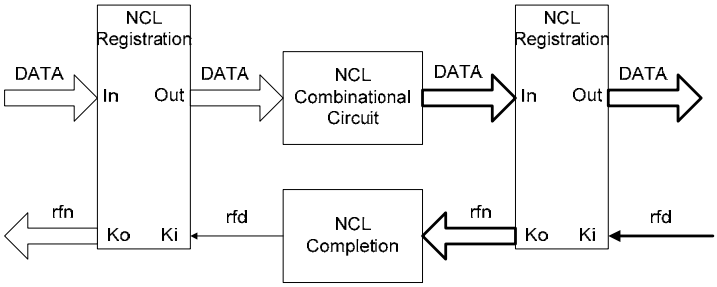


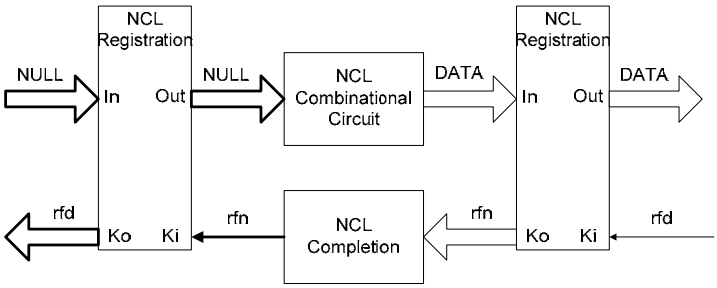
Figure 7. NCL AND function: Z = X • Y: initially *X*=DATA1 and *Y*=DATA0, so *Z*=DATA0; next *X* and *Y* both transition to NULL, so *Z* transitions to NULL; then *X* and *Y* both transition to DATA1, so *Z* transitions to DATA1.



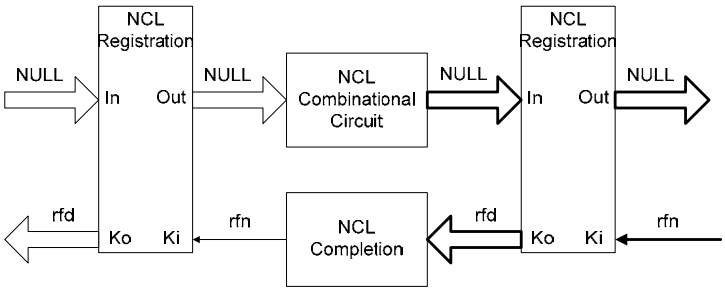
a) DATA flows through input register and combinational circuit



b) DATA flows through output register and *rfn* flows through completion circuit



c) NULL flows through input register and combinational circuit



d) NULL flows through output register and *rfd* flows through completion circuit

Figure 8. NCL DATA/NULL cycle.