A Complex Logic for Computation with Simple Interpretations for Physics

Richard G. Shoup

Interval Research
Palo Alto, CA 94304
shoup@interval.com

Abstract

This paper derives a simple Boolean logic for representing digital circuits. The logic is then extended to self-referential circuits and shown to generate "imaginary" logic values, analogs to imaginary numbers in ordinary algebra. The generalized logic can represent both combinatorial and sequential circuits, and also demonstrates interesting properties suggestive of both classical and quantum physics including superposition, interference, and quantum correlation. A solution is given to the "Square Root of NOT" problem, a function not realizable with isolated classical circuits, and it is shown how this circuit may also be interpreted to represent the equivalent quantum computer circuit. Possible interpretations of the logic for physical systems are discussed.

1. Introduction[1]

1.1 Models of computation

In an effort to learn how to design better computer systems, and especially better languages for expressing computation, we have found ourselves engaged in a lengthy search toward the fundamentals which underlie both hardware and software. Our current models, based loosely on Boolean algebra, set theory, automata theory, etc. are seen to be somewhat inadequate to describe effectively the highly-concurrent, data- and demand-driven, distributed systems emerging today. Concurrency and synchronization are especially difficult to define and model, suggesting that our current ideas about the most basic concepts of space, time, and the meaning of objects and events are in need of some improvement.

This paper develops a simple logic suitable for computing, including self-reference, which begins beneath ordinary Boolean algebra and extends well beyond it in terms of expressive power. We then suggest how this logic might be relevant to physical theory as well, especially quantum mechanics.

1.2 Computation and Physics

Loosely speaking, Computer Science can be thought of as the study of constructing "universes", while Physics tries to understand the universe in which we seem to find ourselves, so-called "physical reality". We wish to explore how these universes are related.

While a thorough investigation of the relationship between Computation and Physics is well beyond the scope of this paper, it is suggested that the two may be fundamentally identical -- two ways of thinking about exactly the same (ultimately mathematical) reality. Thus *neither depends on the other for its explanation at the deepest possible levels*.

However, it is apparent that the fundamental underpinnings of both Physics and Computation are incompletely understood. Each needs new theory, and this is exactly where any ultimate union would be accomplished.[2]

We see below how the simplest assumptions, if explored thoroughly, lead to systems with quite interesting properties from both the computational and the physical viewpoints. At this stage, we can be fairly specific about circuits and computational structures, but only speculative about the meaning for physical systems. Although the correspondences are sometimes striking, considerably more work will be required to firmly establish and complete the link between the two disciplines. But it does seem that the closer we look, and the more fundamental the questions we ask, the more Computation and Physics look remarkably alike.

---

[1] This paper is a slightly revised (and, one hopes, improved) version of the one presented at the 1992 IEEE Workshop on Physics and Computation.

[2] See [7,8,16] for examples of new thinking in this direction.

2. Distinction and the real Booleans

We begin at the simplest possible place, the *Void*. The Void is not a mathematical or physical space, nor is it an object or "thing" of any sort. It is the *absence* of any thing, and thus it possesses no attributes of any kind, including emptiness[1]. Yet the Void will play an important role in our logic, and in fact will serve as one of the two logical values in a Boolean system.

From the Void we create the fundamental computational object: the *distinction*[2], a binary decision, a choice usually represented as a bit of information[3], By drawing a distinction, we form a *boundary* with two sides, the first object in our space. We can visualize this first distinction by a closed curve in the plane as shown in Figure 1.

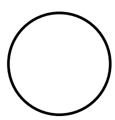


Figure 1. The first distinction.

The distinction may be taken both as an *object* (an area defined in the space), and as an *action* (an injunction to cross from the outside to the inside). At this ultimately-simple level, object and action, state and event, thing and change are united in a single concept.

The functionality of the distinction may be seen more clearly when we draw a second distinction (Figure 2). Given a single object distinguished from the Void, only two possibilities exist for the second distinction: 1) as an additional identical distinction from the Void, or 2) as a distinction from (within) the existing object.

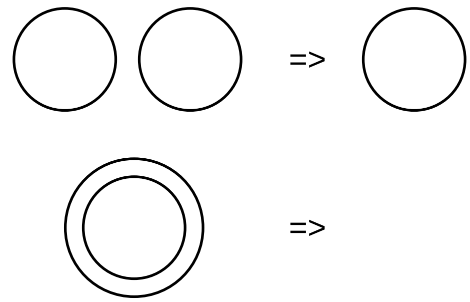


Figure 2. Two ways of combining two distinctions -- the axioms of the logical arithmetic.

In the first case, we have two identical distinctions from the Void taken together[4], which is just the same as one distinction. (At this level of extreme simplicity, either we have distinguished from the Void or we have not.) In the second case, a distinction *from the first distinction* can only refer back to the Void. The second arrangement can also be thought of as a function -- one distinction "applied to" or "surrounding" another. Since in both arrangements the two distinctions are equivalent, both combinations are symmetrical, that is, trivially commutative.

Using these two axioms, any arrangement of boundaries in the plane can be simplified to either one distinction or the Void. One possible interpretation of this arithmetic is a traditional Boolean logic.

For ease of reference and familiarity, we will denote the two sides of the first distinction as **0** and **1**, representing Void and object respectively. As values, they will be interpreted as the *constants* of our logic. When applied as functions, **0** will be interpreted as the *null function* (logical identity, or no operation at all) and **1** will be interpreted as logical *inversion* (crossing from Void to object or vice-versa).

Names or tokens such as A, B, ... will be used to stand for unknown values or *variables*. As with the first two distinctions above, it is possible for two variables to be taken together and it is also possible for them to be applied to one another.[5] Given several variables and the above

---

[1]Technically, we can speak only about our *concept* of the Void, not the Void itself, since the Void is what we describe by not speaking at all.

[2]This logic begins with, and is inspired by, G. Spencer-Brown's remarkable book Laws of Form[14]. Unfortunately, space does not permit a detailed discussion of the steps leading from Spencer-Brown's first distinction to the Boolean arithmetic and algebra. Particularly lucid explanations of this logic can be found in Bricken [4,5], by the equally-appropriate name "boundary logic".

[3]We must be careful here to distinguish between a bit of information and a bit of storage. At this fundamental level, a bit of information is just a logical value (the result of drawing a distinction), a signal which may or may not have been involved in a self-referential memory circuit -- a bit of storage. These two concepts are often used interchangeably, particularly in discussions of computer languages, where named bits ("variables") usually correspond to a location in the computer's memory. Yet a third concept is the "carrier" of the bit, referring to electrons or other physical manifestation.

[4]We avoid phrases such as "simultaneously" or "at the same time" here, since neither time nor even sequence has been defined or implied at this stage. In general, however, it is nearly impossible to completely rid our vocabulary of time-related words and phrases because the idea is so deeply ingrained in our everyday perceptions of reality.

[5]Spencer-Brown's original Calculus of Indications[14] fails to take advantage of one of the symmetries available to it in the algebra, namely that of an expression as a function of another expression, as in the lambda calculus. This was first pointed out

two functions, we can construct an algebra of expressions of arbitrary complexity equivalent to propositional logic.

Although the two values **0** and **1** will be represented by a variable and thus are often treated symmetrically in what follows, it is important to bear in mind that one of them (**0**) stands for the Void, is thus logically null, and ultimately represents the *absence* of a value or function.

One significance of the drawing of a distinction as the basis for the logic is that it emphasizes that each object in the system exists not *in vacuo*, but only by being distinguished from some context of other objects or the Void.

2.1 Simple gates

We will use simple circuits (directed dependency graphs) for our notation[1]. A *gate* (node) represents a logical function which *depends on* (is defined in terms of) zero or more inputs. Each *signal* (arc) is the output of a gate and represents a logical value.

Figure 3 shows two two-input gates corresponding to the two axioms of arithmetic given in Figure 2 above. Two values taken together are, in effect, ORed, while two values applied to each other constitute a function equivalent to the familiar Exclusive OR or XOR. Other traditional gates such as NOR and AND may be constructed similarly.

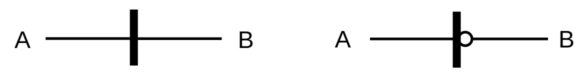
	0	1
0	0	1
1	1	1

	0	1
0	0	1
1	1	0

Figure 3. Two-input OR and XOR gates with real values.

Gates with more or less than two inputs are related in the natural way. For example, with one input the OR gate reduces to the identity function, which will be called "DUP" (for "duplicate"), as shown in Figure 4. Note that while the DUP gate's output is logically identical to its input, the DUP gate is *not* the same as a direct connection (continuous graph arc), since its output is a *new signal* or object, as are all gate outputs.

In general, a **0** input can be considered absent entirely, since it represents Void. Thus with one input set equal to **0**, both the two-input OR and XOR gates reduce to DUP. With one input held to **1**, XOR is equivalent to a simple inversion or NOT gate. A generalization of the NOT gate to more than one input gives the usual NOR gate. With more than two inputs, the XOR becomes the familiar parity function.



0	1
0	1

0	1
1	0

Figure 4: DUP and NOT gates with real values.

We could have, of course, axiomatized a Boolean logic with a simple set of primitives such as OR and NOT or NOR and False, etc., but we wish to retain the symmetry at the algebraic level which we derived from simple (in fact unavoidable) assumptions at the arithmetic level. While the single inverting gate of Spencer-Brown is more compact and perhaps more elegant, we will see that a *non*-inverting gate is not only convenient but essential for dealing with the self-referential logic values.

From the logic given so far, we can build up common computing elements such as adders, multiplexers, registers, etc., in the usual way.[2] But as with all standard Boolean logics, we have no way to represent and analyze *sequential* circuits (self-referential circuits with feedback) except step-by-step, as clocked finite-state automata.

3. Self-reference and the imaginary Booleans

3.1 Classical self-reference

In ordinary algebra, the simplest equation requiring complex numbers for its solution is

$$x^2 + 1 = 0.$$

by Banaschewski[2]. The XOR circuit given here restores this symmetry for all cases, including the imaginary values.

[1] We have deliberately neglected to provide an equivalent textual or equation form of this circuit logic because 1) many expressions in this logic are inherently >1-dimensional, especially self-referential ones, and are thus awkward or impossible to represent in text form, and 2) clarity and understanding are significantly improved by viewing the expressions as two-dimensional directed graphs. Even the highly-compact notation of Spencer-Brown's single mark is inherently two-dimensional, and necessarily so, even without self-referential expressions.

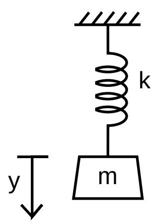
[2] Figure A1 in the Appendix shows the correspondence between our new notation, conventional circuit diagrams, Boolean algebraic notation and Spencer-Brown's Calculus of Indications. The last two rows in the chart represent logic values arising from self-reference.

As Spencer-Brown[14] has pointed out, this can easily be rewritten as

$$x = -1/x,$$

showing the solution of the equation defined in terms of itself, and thus emphasizing its *self-reference* or re-entrance.

Many situations in classical physics and engineering are easily and effectively represented using complex numbers, but it is seldom made clear how the need for these numbers follows directly from the *self-dependency* or feedback in the arrangement.[1] Figure 5 shows a simple example of a classical system where a quantity depends upon a function of itself in the form $y = -f(y)$. The position y of the weight depends upon the force applied to it (via $F = ma = m\, d^2y/dt^2$) which in turn depends on the position (by $F = -ky$). The well-known solution to this differential equation is an oscillation because of the inverse dependence, that is, negative feedback.



$$y = -\frac{m}{k}\frac{d^2y}{dt^2} = C1\, e^{+i\omega t} + C2\, e^{-i\omega t} = C \sin(\omega T + \phi)$$

Figure 5. Self-reference in a classical system.

3.2 Boolean self-reference

In our binary logic, self-reference is simply the presence of a path from the output of a gate back to its own input -- a loop in the dependency graph. Figure 6 shows the two simplest cases using the DUP and NOT gates. With feedback, these two gates produce, respectively: *consistent* self-reference (an even number of inversions in the loop), and *contradictory* self-reference (an odd number of inversions). The former results in an *autology*[2] (a memory or storage element), and the latter yields a *paradox* (an oscillator or clock).[3]

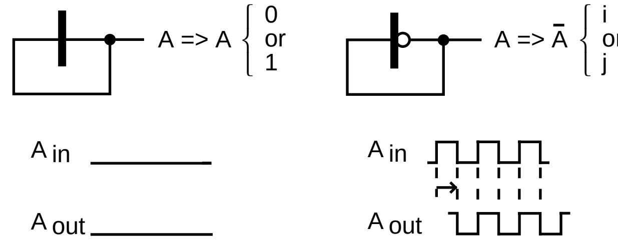


Figure 6. Two cases of self-reference: autological (a memory) and paradoxical (an oscillator).

In the autological case, the circuit implements a flip-flop or one-bit memory, and is stable with a value of *either* **0** or **1**. If we had the concept of Time (which we have neither required nor assumed), we would say the value of the circuit is determined or resolved at $t = -\infty$, that is, before Time began, or "always". We can also think of this setting as an *initial condition*, part of the circuit's definition.

In the second, paradoxical case, the circuit is stable with *neither* value **0** nor **1**, and can be thought of as continually inverting itself, thus *superposing* the two values. The only way to sensibly think about this situation in terms of the real Boolean values of **0** and **1** is to allow the circuit to assume the two values alternately, in a waveform or oscillation. The circuit is thus resolved at $t = +\infty$, that is, "never", and is equivalent to the familiar Liar Paradox "This sentence is false".

It is important to realize that just as in the autological case, an initial condition is required in the paradoxical case as well, and can be thought of as determining the phase of the oscillation. Thus in addition to the real values of **0** and **1**, there are *two* oscillatory values, which we will call **i** and **j**, as shown in Figure 7. They each represent a superposition of the real logical values *and are indistinguishable except when combined (interfered) with each other*.

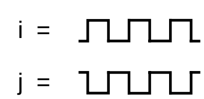


Figure 7. Imaginary values as waveforms.

---

[1] Moreover, Spencer-Brown notes that while we routinely use complex values in ordinary algebra, we have resolutely disallowed them in the domain of logic *and thus in all our reasoning processes as well.* The full implications of this are far from being appreciated. It should be possible, for example, to construct valid mathematical proofs whose intermediate steps have imaginary truth values but which reach real consequences.

[2] From *auto-* (self) and *-logos* (word or speech), thus meaning "self-informing" or "self-derived".

[3] The imaginary Booleans were first introduced by Spencer-Brown[14] and further developed by Kauffman[10-12]. The logic presented here is similar to these authors', but with a number of important differences and extensions, and a new notation.

In order to be able to calculate with these self-referential circuits, we will take the oscillations **i** and **j** to be new values in the logic and call them *imaginary*, thus yielding a four-valued logic { **0**, **1**, **i**, **j** }, with two reals, two imaginaries and properties quite similar to those of the complex numbers[1]. Note however that we require our logical values to be distinctly real or imaginary. They cannot be a combination of real and imaginary components, as with complex numbers.

It should be stressed that we have not merely chosen these values and their interactions so as to make them interesting, but arrived at them as direct consequences of the primitive acts of distinction and self-reference.

It is also clear that a similar and perhaps equivalent logic to that presented here (including imaginaries) could be formulated in terms of more traditional group or set elements, etc. We prefer the basis given here because it 1) begins with absolutely primitive assumptions, 2) is readily and consistently interpreted as binary logic circuits, 3) naturally incorporates self-reference, and, most important, 4) can be related in a meaningful way to physical concepts.

3.3 Sequence and the concept of Time

One way to visualize or justify the operation of the basic gates on the imaginaries is to think of the gates as causing a small "delay" or "phase shift" from input to output, as shown in Figure 8. In the case of the DUP gate, the shift results in an inversion or *conjugation* of the imaginary, transforming **i** to **j** and vice-versa. In the NOT gate, a shift *and an inversion* take place, returning the waveform to itself -- just as required to satisfy the definition of the imaginary given in Figure 6.

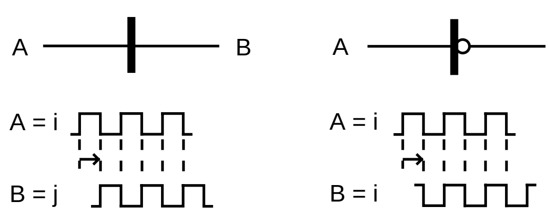
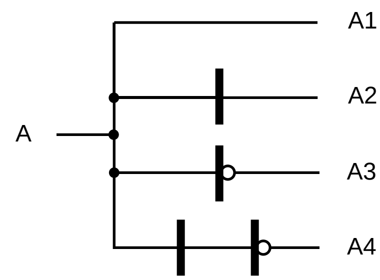


Figure 8. DUP and NOT of imaginary values. Both gates introduce "delay", and NOT also inverts.

[1]Specifically, the correspondence to the complex number group is {1,-1,i,-i} respectively, with XOR taken as multiplication. Louis Kauffman[12] and David Keenan (private communication) have pointed out that the four logic values can also be represented effectively using ordered pairs of the real values.

Figure 9 summarizes the logic values and basic one-input gates: DUP passes reals and conjugates imaginaries, while NOT inverts reals and passes imaginaries. None of the gate configurations shown is logically equivalent to the wire (direct connection).



	0	1	i	j
A1	0	1	i	j
A2	0	1	j	i
A3	1	0	i	j
A4	1	0	j	i

Figure 9. Comparison of a wire (A1), DUP gate (A2), NOT gate (A3), and DUP followed by NOT (A4).

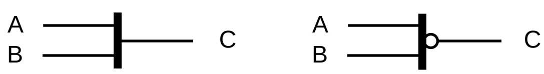
In real physical circuits, there is always a finite delay which serves to spread the alternation out "in time". However, here in our simple mathematical world, there is no dimension of time, only *sequence* due to dependency of the output of a circuit on its inputs. We represent this sequence (partial ordering) by drawing along a horizontal axis in order to display the alternation. It is important to bear in mind, however, that this is really just a mental convenience which makes the superposition easier to visualize. It is not meaningful to ask, for example, what the frequency of oscillation is in these circuits or whether any two independent delays are equal or have the same phase. The relative delay and phase of any two imaginary signals are undefined *unless they have a common dependency* (i.e., unless they have a common source or have interacted in the "past", earlier in the circuit).

Another way of thinking about sequence, for example in the case of self-referential NOT (Figure 6), is that it is necessary to distinguish the signal *from itself* since it is present both at the input and at the output of the inverting gate. An additional (partial) dimension is required to "hold" the paradox in the same way the Argand plane is used in representing complex numbers. In a sense, the gate *is* the distinction (between input and output) which we visualize as delay, a distinction in time. By this view, we have created one bit of discrete "time", a zero value ("Now") and a non-zero value ("Not-now" or "Then").

We can now see that the self-referential situation *requires* additional logical values and excursion into a new time-like dimension. Although this is surely the origin of our concept of Time, we do *not* imply or require at this stage any *continuous* dimension of global or local Time in the classical or relativistic sense, only *sequence*, which is entirely discrete, local and relative to its dependencies.

3.4 Generalized gates

The basic gates can all be extended in a similar way for the imaginary values, giving circuits which are natural generalizations of the usual Boolean gates. Figure 10 shows the two-input OR and XOR gates complete with imaginary values. Note that while these gates retain the names of their real counterparts, they do not necessarily compose in the expected ways. For example, the NOR gate is not equivalent to OR followed by NOT, and the OR gate is not associative in all cases. In computing the function tables, bear in mind that each gate causes the imaginary values to be phase-shifted or "delayed" as discussed above. Also note that when conjugate imaginaries are combined in OR or NOR the result is a *real* value, as desired.



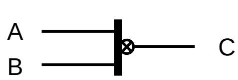
	0	1	i	j
0	0	1	j	i
1	1	1	1	1
i	j	1	j	1
j	i	1	1	i

	0	1	i	j
0	1	0	i	j
1	0	0	0	0
i	i	0	i	0
j	j	0	0	j

Figure 10. Complete two-input OR and NOR gates.

A more interesting circuit is the XOR or "Apply" gate extended to the imaginaries and shown in Figure 11. This gate is highly symmetric, commutative as well as associative ((A ⊗ B) ⊗ C = A ⊗ (B ⊗ C)). It implements the usual Exclusive OR function for the reals, and equality (interference) for the imaginaries.[1]

Note also that **i** ⊗ **i** = **1** (analogous to √-1 * √-1 = -1 in complex numbers), A ⊗ **0** = A (identity for both reals and imaginaries), and A ⊗ **1** is the complement for both reals and imaginaries.



	0	1	i	j
0	0	1	i	j
1	1	0	j	i
i	i	j	1	0
j	j	i	0	1

Figure 11. Complete XOR or Apply gate.

[1]It is an instructive exercise to construct the above XOR gate from the simpler gates, as the solution is (necessarily) not simple. The smallest such construction currently known contains about 20 gates. A different and simpler XOR gate definition can be derived naturally from NOR gates (this is also left as an exercise), but does not have such a clear correspondence to √-1.

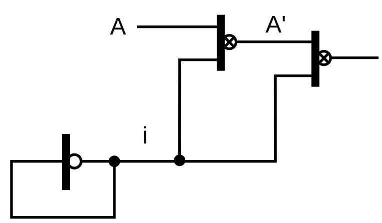
3.5 The Square Root of NOT

We have seen briefly above how the imaginary values represent superposition of real values and how they can "interfere" with one another to create real results. While these circuits possess properties suggestive of quantum-mechanical phenomena, all our circuits are strictly classical, in that they do not rely on any quantum phenomena for their operation.[2]

However, a stronger link with quantum phenomena can be shown in consideration of the "Square Root of NOT" problem posed by David Deutsch in his work on quantum computing[6,7]. The problem is to design a circuit which when cascaded with itself gives the inversion of the original (real) input. That is, construct a gate or function **f** such that

$$\mathbf{f}(\mathbf{f}(A)) = \mathrm{NOT}(A), \quad A \in \{\mathbf{0}, \mathbf{1}\}.$$

It is claimed that such a function is a simple example of one which cannot be implemented by conventional classical circuits, but could be implemented by a circuit operating on quantum mechanical principles. It seems to be true that no *isolated* classical circuit cascaded with itself can have this property, but Figure 12 shows that a classical solution is possible *if both gates receive an additional common imaginary input.*



A	A'	A"
0	i	1
1	j	0

Figure 12. Two "Square Root of NOT" circuits (XOR with common imaginary input) cascade to form NOT.

The "Square Root of NOT" function then is

$$\mathbf{f}(A) = A \otimes \mathbf{i}$$

and XOR is associative, so that

[2]In fact, these circuits can be built today using available logic modules so long as the real electronic delays are properly compensated for -- a tedious task. Instead, practical computers are synchronized globally via a clock signal (an imaginary value, in our terms), which maintains global coherence by gating all self-referential memory elements, or locally by asynchronous or self-timed techniques[13,15].

$$f(f(A)) = (A \otimes \mathbf{i}) \otimes \mathbf{i} = A \otimes (\mathbf{i} \otimes \mathbf{i})$$
$$= A \otimes \mathbf{1} = \text{NOT}(A).$$

Note that the two gates are identical and that the extra input is independent and in common so that *it carries no information about A or about which gate is which*. In essence, the value of A is encoded[1] in the *phase* of the imaginary at A'. This signal is then decoded in the second gate by interfering with the *same phase* imaginary to produce a real result which is the inversion of A. Note that the output A" persists through oscillations in **i** and only changes when A changes. Note also that the common imaginary input is not the same as a global "clock", since no memory elements are involved and no transitions are being used.

Recalling that XOR is just the Apply function or "surround" of one value by another, we see that ⊗ **i** means **i** applied as a function. Thus we can say that **i** *is* the "Square Root of NOT"[2], and the desired circuit is possible within a classical logic *if* the additional input is permitted. (However, see below for a non-classical interpretation.)

4. Interpretations for Physics

Various suggestive properties of the above logic naturally lead to questions about its meaning for Physics. This section consists largely of informal, highly speculative interpretations of the above logic to various aspects and phenomena of classical and quantum physics. The real utility of this logic will be determined as it is extended to more complex waveforms and used to model real physical experiments, particularly quantum-mechanical ones.

4.1 Superposition and Quantum Correlation

An alternate interpretation of the circuit of Figure 12 above is that it represents exactly the (hidden) correlation necessary to realize the intended function in a quantum-mechanical implementation. This logic may therefore be useful in describing quantum computing circuits and thus quantum physical experiments as well by explicitly representing the correlation of quantum states as common imaginary inputs to spatially separated elements.

[1] See similarities to the use of quantum states by Bennett[3] in quantum cryptography.

[2] The idea of "square root of negation" was also studied by Kauffman[11,12], who showed how applying an imaginary value can be seen as a rotation in the complex Boolean plane, entirely analogously to multiplication by √-1 in the complex number plane.

4.2 Distinctions and Objects

Since the formulation of quantum mechanics, the nature of physical reality has been the subject of considerable debate. Our notions of "object" and "event" suddenly became far less clear than before. An evident implication of the wave/particle conundrum is that *our idea of object itself needs to be somehow modified and deepened*.

The simplest interpretation of the above logic onto physical reality is this: Logic values correspond to "objects" or states in the physical world, which we create by "observing" them. The real logic values are the observables, the discrete results of measurement. The imaginary values correspond to superpositions of states, and are not observable without first being collapsed to real values by combining (interfering) them with other values. Note that in our logic, interference phenomena appear just where physical theory suggests they should, between imaginary values or wave functions.

From this interpretation we might also infer that Space and Time are simply concepts which we have invented to refer to the properties of ordinary distinctions and of self-referential distinctions, respectively. Quantities in neither Space nor Time actually "exist" independent of our actions, and such evidence will not be found in the "real" physical world. But every experimental arrangement (circuit) involves making discrete measurements (drawing distinctions in Space) and is necessarily self-referential, thus requiring Time-like distinctions as well to describe it.

It might be said that this principle (existence is distinction) is the ultimate extension of the concept of *relativity*, in the following sense: The history of physical theory may be regarded as a sequence of realizations, each made more reluctantly than the last, that our notions of cherished absolutes are illusory. First, positions and velocities of objects were seen to be quantities which could only be measured relatively. Next, distances in space and time were shown to be observer-dependent. Then deterministic outcomes and complete knowledge of state had to be abandoned. And finally, even the existence of the object itself is seen to be merely a consequence of our actions. At each step we, the observers, have had to increasingly acknowledge our unavoidable entanglement with our experiments and responsibility for what we see.

4.3 Measurement

The essence of the measurement interaction has been described by Wheeler[16] and others as "the making of a record" -- something with permanence which can affect the outside world. In the present theory, this corresponds simply to capturing a real value in a memory (autology circuit). If we insert one or more gates in the feedback path of a memory circuit, we can interfere a measurement signal or waveform with the signal under observation and set the memory (usually called a "latch") accordingly. In this way, a derived value (an observation or measurement) can persist and affect its environment *without further dependence on its source*.

It is indeed notable that in the logic the extension to imaginary values occurs exactly where self-reference is introduced, that is, just where the observer/observed boundary can be crossed and the measurement interaction can take place. Remarkably, a single act of self-reference at once makes possible memory phenomena necessary for the measurement "record", the superposition of states (imaginary values) necessary for quantum indeterminacy, and the mechanism (interference) necessary for the collapse to real observables.

4.4 Randomness and Correlation

The computational model above does not allow for the definition of a signal which is not dependent on anything else (except for constants). In this theory, there are no causeless events. Thus if this model is to be successfully interpreted for physical reality, some hidden variables or unobserved influences must be responsible for the apparent randomness of certain quantum phenomena.

Such influences may already be present in the measurement interaction itself. For example, if we sample an imaginary waveform with a short *uncorrelated* waveform such as a pulse, the result could appear arbitrary, that is, random. This would imply that apparent randomness in nature *is a function of the observer*, and depends upon of the lack of correlation (a common source or interaction in the past) or an unknown correlation between observer and observed.

Conversely, it is well known that there are certain excess correlations between apparently random sequences which have been predicted by quantum mechanics and confirmed in experiments[1] similar to those imagined by Einstein, Podolsky and Rosen[8]. In analysis of such experiments, it has been conjectured by several researchers (see for example Feynman[9]) that perhaps we and our experiment, sharing a common past, are already in some way correlated. For this reason, we observe higher correlations in our measurements than we would if the sequences were unrelated and perfectly random.

The above logical theory would appear to support just such a mechanism, and suggests that it may be possible to explain these correlations in detail. If God plays dice with the universe, it must be *He* who sets the dice to a particular outcome. More likely, it is *we* who set the dice by observing just the way we do.

5. Conclusions

We have derived from the most primitive assumptions a logical arithmetic and algebra of simple circuits, extended it to include self-reference, and shown how this leads naturally to circuits which are useful in modelling computational systems.

Our logic is also strongly suggestive of physical systems, including quantum mechanics. But, like quantum mechanics, the psychological price is high: We have had to give up the idea of independent existence, abandon our usual concept of Time, and fully embrace the paradox.

It is hoped that further explorations in this direction will lead to more-powerful design techniques for digital computers (including quantum computers), to new representations of physical systems, and above all to the greater appreciation of the commonality between the two.

Acknowledgements

I am indebted to William Bricken for assistance in clarifying the nature of the imaginaries and their presentation, and to David Keenan for many useful comments and spirited discussions. Nick Herbert and Adolph Smith contributed significantly to my education in physics. I am also grateful to Louis Kauffman, Eric Minch, David Robson, Fred Furtek, and Greg Wolff for numerous informative and enthusiastic discussions on these subjects.

References

[1] Aspect, A., Dalibard, J., and Roger, G., "Experimental Test of Bell's Inequalities Using Time-Varying Analyzers", Phys. Rev. Letters, Vol. 49, 1982.

[2] Banaschewski, B., "On G. Spencer Brown's Laws of Form", Notre Dame J. of Formal Logic, Vol. XVIII, No. 3, July 1977.

[3] Bennett, C. H., "Quantum Crytography: Uncertainty in the Service of Privacy", Science, Vol. 257, August 7, 1992.

[4] Bricken, W., "A Deductive Mathematics for Efficient Reasoning", Human Interface Technology Lab. Tech. Report R-86-2, U. of Washington.

[5] Bricken, W., "A Simple Space", Human Interface Technology Lab. Tech. Report R-86-3, U. of Washington.

[6] Deutsch, D., "Quantum Computation", Physics World, June 1992.

[7] Deutsch, D., "Quantum Computational Networks", Proc. R. Soc. London, A425, 1989.

[8] Einstein, A., Podolsky, B., and Rosen, N., "Can Quantum-mechanical Description of Physical Reality Be Considered Complete?", Phys. Rev., Vol. 7, p. 777, 1935.

[9] Feynman, R. P., "Simulating Physics with Computers", Int'l. J. Th. Phys., Vol. 21, Nos. 6/7, 1982.

[10] Kauffman, L. H., "Complex Numbers and Algebraic Logic", Proc. 10th Intl. Symp. Multivalued Logic, IEEE, 1980.

[11] Kauffman, L. H., "Imaginary Values in Mathematical Logic", Proc. 17th Intl. Symp. Multivalued Logic, IEEE, 1987.
[12] Kauffman, L. H., "Self-reference and Recursive Forms", J. Social Bio. Struct., Vol. 10, pp. 53-72, 1987.
[13] Seitz, C. L., "System Timing", in Introduction to VLSI Systems, C. A. Mead and L. A. Conway (eds.), Addison-Wesley, 1980.
[14] Spencer-Brown, G., Laws of Form, George Allen and Unwin Ltd., London, 1969; Julian Press, New York, 1972; Bantam Books, New York, 1973; E. P. Dutton, New York, 1979.
[15] Sutherland, I., "Micropipelines", Comm. Assoc. Comp. Mach., Vol. 32, No. 6, June 1989.
[16] Wheeler, J. A., "The Computer and the Universe", Int'l. J. Th. Phys., Vol. 21, Nos. 6/7, 1982.

Appendix

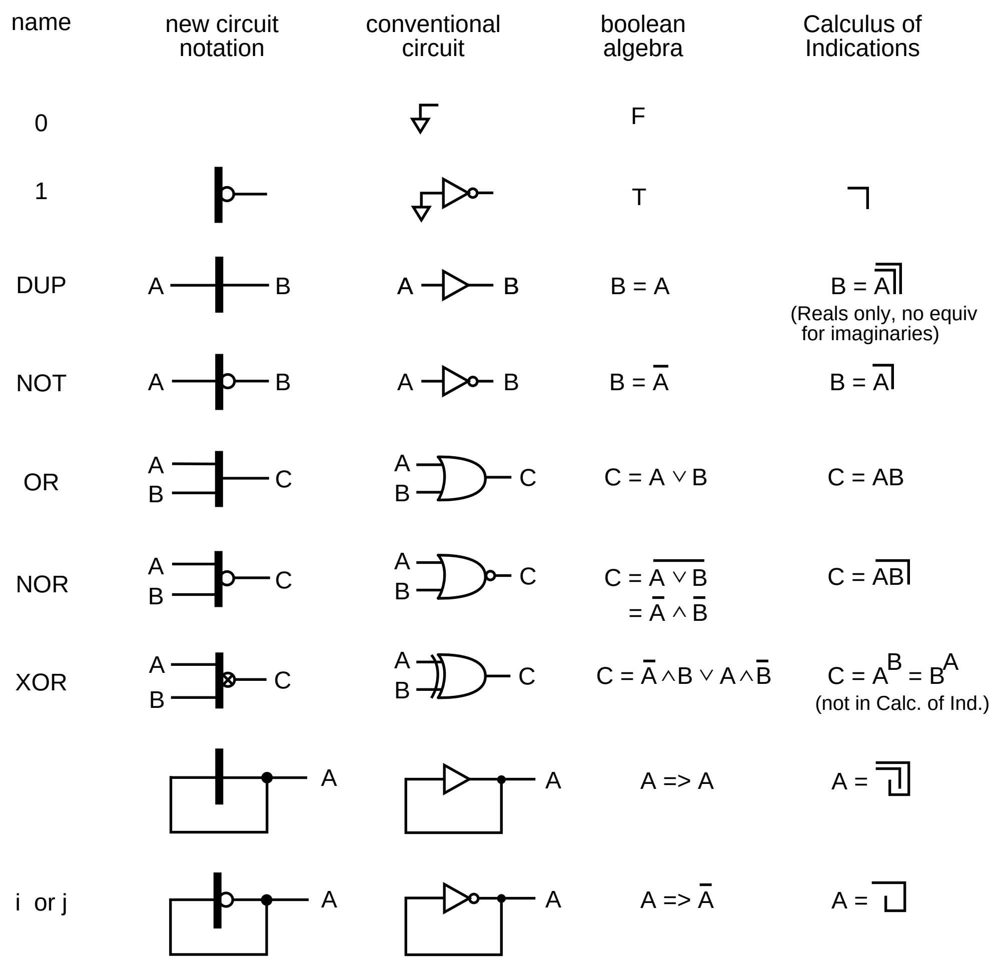


Figure A1. Comparison of logic notations.