FORMALIZED MATHEMATICS
Volume 10, Number 1, 2002
University of Białystok

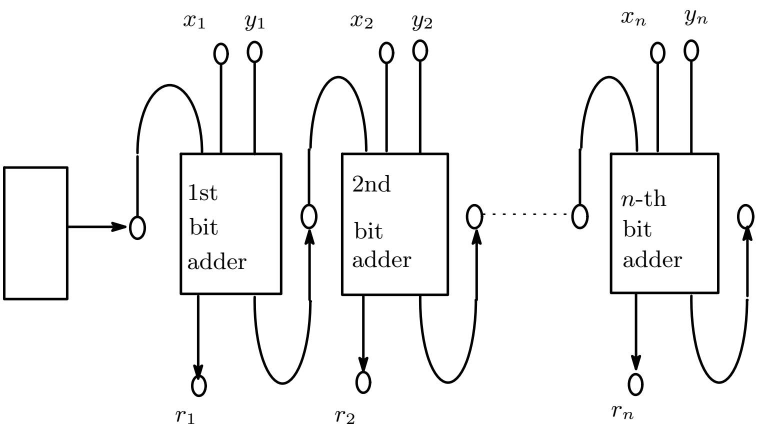
Full Adder Circuit. Part II

Grzegorz Bancerek
Białystok Technical University

Shin'nosuke Yamaguchi
Shinshu University
Nagano

Katsumi Wasaki
Shinshu University
Nagano

Summary. In this article we continue the investigations from [6] of verification of a design of adder circuit. We define it as a combination of 1-bit adders using schemes from [7]. n-bit adder circuit has the following structure



As the main result we prove the stability of the circuit. Further works will consist of the proof of full correctness of the circuit.

MML Identifier: FACIRC_2.

The articles [13], [20], [1], [4], [10], [11], [14], [6], [5], [8], [9], [2], [21], [16], [12], [18], [3], [17], [22], [19], and [15] provide the notation and terminology for this paper.

ISSN 1426–2630

The following three propositions are true:

(1) For all sets x, y, z such that $x \neq z$ and $y \neq z$ holds $\{x, y\} \setminus \{z\} = \{x, y\}$.

(2) For all non empty sets X, Y and for all natural numbers n, m such that $n \neq m$ holds $X^n \neq Y^m$.

(3) For all sets x, y, z holds $x \neq \langle\langle x, y\rangle, z\rangle$ and $y \neq \langle\langle x, y\rangle, z\rangle$.

Let us note that every many sorted signature which is void is also unsplit and has arity held in gates and Boolean denotation held in gates.

One can check that there exists a many sorted signature which is strict and void.

Let x be a set. The functor SingleMSS x yielding a strict void many sorted signature is defined as follows:

(Def. 1) The carrier of SingleMSS $x = \{x\}$.

Let x be a set. Note that SingleMSS x is non empty.

Let x be a set.

(Def. 2) SingleMSA x is a Boolean strict algebra over SingleMSS x.

We now state three propositions:

(4) For every set x and for every many sorted signature S holds SingleMSS $x \approx S$.

(5) Let x be a set and S be a non empty many sorted signature. Suppose $x \in$ the carrier of S. Then SingleMSS $x +\cdot S =$ the many sorted signature of S.

(6) Let x be a set, S be a non empty strict many sorted signature, and A be a Boolean algebra over S. If $x \in$ the carrier of S, then SingleMSA $x +\cdot A =$ the algebra of A.

$\emptyset$ is a finite sequence with length 0. We introduce ε as a synonym of $\emptyset$.

Let n be a natural number and let x, y be finite sequences. The functor n-BitAdderStr(x, y) yields an unsplit non void strict non empty many sorted signature with arity held in gates and Boolean denotation held in gates and is defined by the condition (Def. 3).

(Def. 3) There exist many sorted sets f, g indexed by $\mathbb{N}$ such that

(i) n-BitAdderStr$(x, y) = f(n)$,

(ii) $f(0) =$ 1GateCircStr$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$,

(iii) $g(0) = \langle\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false}\rangle$, and

(iv) for every natural number n and for every non empty many sorted signature S and for every set z such that $S = f(n)$ and $z = g(n)$ holds $f(n+1) = S +\cdot$ BitAdderWithOverflowStr$(x(n+1), y(n+1), z)$ and $g(n+1) =$ MajorityOutput$(x(n+1), y(n+1), z)$.

Let n be a natural number and let x, y be finite sequences. The functor n-BitAdderCirc(x, y) yields a Boolean strict circuit of n-BitAdderStr(x, y) with denotation held in gates and is defined by the condition (Def. 4).

(Def. 4) There exist many sorted sets f, g, h indexed by $\mathbb{N}$ such that

(i) n-BitAdderStr$(x, y) = f(n)$,

(ii) n-BitAdderCirc$(x, y) = g(n)$,

(iii) $f(0) = $ 1GateCircStr$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$,

(iv) $g(0) = $ 1GateCircuit$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$,

(v) $h(0) = \langle \varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false} \rangle$, and

(vi) for every natural number n and for every non empty many sorted signature S and for every non-empty algebra A over S and for every set z such that $S = f(n)$ and $A = g(n)$ and $z = h(n)$ holds $f(n+1) = S +\cdot$ BitAdderWithOverflowStr$(x(n+1), y(n+1), z)$ and $g(n+1) = A +\cdot$ BitAdderWithOverflowCirc$(x(n+1), y(n+1), z)$ and $h(n+1) = $ MajorityOutput$(x(n+1), y(n+1), z)$.

Let n be a natural number and let x, y be finite sequences. The functor n-BitMajorityOutput(x, y) yielding an element of InnerVertices(n-BitAdderStr (x, y)) is defined by the condition (Def. 5).

(Def. 5) There exists a many sorted set h indexed by $\mathbb{N}$ such that

(i) n-BitMajorityOutput$(x, y) = h(n)$,

(ii) $h(0) = \langle \varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false} \rangle$, and

(iii) for every natural number n and for every set z such that $z = h(n)$ holds $h(n+1) = $ MajorityOutput$(x(n+1), y(n+1), z)$.

We now state several propositions:

(7) Let x, y be finite sequences and f, g, h be many sorted sets indexed by $\mathbb{N}$. Suppose that

(i) $f(0) = $ 1GateCircStr$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$,

(ii) $g(0) = $ 1GateCircuit$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$,

(iii) $h(0) = \langle \varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false} \rangle$, and

(iv) for every natural number n and for every non empty many sorted signature S and for every non-empty algebra A over S and for every set z such that $S = f(n)$ and $A = g(n)$ and $z = h(n)$ holds $f(n+1) = S +\cdot$ BitAdderWithOverflowStr$(x(n+1), y(n+1), z)$ and $g(n+1) = A +\cdot$ BitAdderWithOverflowCirc$(x(n+1), y(n+1), z)$ and $h(n+1) = $ MajorityOutput$(x(n+1), y(n+1), z)$.

Let n be a natural number. Then n-BitAdderStr$(x, y) = f(n)$ and n-BitAdderCirc$(x, y) = g(n)$ and n-BitMajorityOutput$(x, y) = h(n)$.

(8) For all finite sequences a, b holds 0-BitAdderStr$(a, b) = $ 1GateCircStr$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$ and 0-BitAdderCirc$(a, b) = $ 1GateCircuit$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$ and 0-BitMajorityOutput$(a, b) = \langle \varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false} \rangle$.

(9) Let a, b be finite sequences and c be a set. Suppose $c = \langle \varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false} \rangle$. Then 1-BitAdderStr$(a, b) = $ 1GateCircStr$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false}) +\cdot$ BitAdderWithOverflowStr$(a(1), b(1), c)$ and 1-BitAdderCirc$(a, b) = $ 1GateCircuit$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false}) +\cdot$ BitAdderWithOverflowCirc$(a(1),$

$b(1), c)$ and 1-BitMajorityOutput$(a, b) =$ MajorityOutput$(a(1), b(1), c)$.

(10) For all sets a, b, c such that $c = \langle\!\langle \varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false} \rangle\!\rangle$ holds 1-BitAdderStr$(\langle a \rangle, \langle b \rangle) =$ 1GateCircStr$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$ $+\cdot$ BitAdderWithOverflowStr(a, b, c) and 1-BitAdderCirc$(\langle a \rangle, \langle b \rangle) =$ 1GateCircuit$(\varepsilon, \mathit{Boolean}^0 \longmapsto \mathit{false})$$+\cdot$ BitAdderWithOverflowCirc(a, b, c) and 1-BitMajorityOutput$(\langle a \rangle, \langle b \rangle) =$ MajorityOutput(a, b, c).

(11) Let n be a natural number, p, q be finite sequences with length n, and p_1, p_2, q_1, q_2 be finite sequences. Then n-BitAdderStr$(p \frown p_1, q \frown q_1) = n$-BitAdderStr$(p \frown p_2, q \frown q_2)$ and n-BitAdderCirc$(p \frown p_1, q \frown q_1) = n$-BitAdderCirc$(p \frown p_2, q \frown q_2)$ and n-BitMajorityOutput$(p \frown p_1, q \frown q_1) = n$-BitMajorityOutput$(p \frown p_2, q \frown q_2)$.

(12) Let n be a natural number, x, y be finite sequences with length n, and a, b be sets. Then $(n + 1)$-BitAdderStr$(x \frown \langle a \rangle, y \frown \langle b \rangle) = (n$-BitAdderStr$(x, y))$$+\cdot$ BitAdderWithOverflowStr$(a, b,$ n-BitMajorityOutput$(x, y))$ and $(n + 1)$-BitAdderCirc$(x \frown \langle a \rangle, y \frown \langle b \rangle) = (n$-BitAdderCirc$(x, y))$$+\cdot$ BitAdderWithOverflowCirc $(a, b, n$-BitMajorityOutput$(x, y))$ and $(n + 1)$-BitMajorityOutput$(x \frown \langle a \rangle, y \frown \langle b \rangle) =$ MajorityOutput$(a, b, n$-BitMajorityOutput$(x, y))$.

(13) Let n be a natural number and x, y be finite sequences. Then $(n + 1)$-BitAdderStr$(x, y) = (n$-BitAdderStr$(x, y))$$+\cdot$ BitAdderWithOverflowStr $(x(n+1), y(n+1), n$-BitMajorityOutput$(x, y))$ and $(n+1)$-BitAdderCirc(x, y) $= (n$-BitAdderCirc$(x, y))$$+\cdot$ BitAdderWithOverflowCirc$(x(n+1), y(n+1),$ n-BitMajorityOutput$(x, y))$ and $(n + 1)$-BitMajorityOutput$(x, y) =$ MajorityOutput$(x(n + 1), y(n + 1), n$-BitMajorityOutput$(x, y))$.

(14) For all natural numbers n, m such that $n \leqslant m$ and for all finite sequences x, y holds InnerVertices$(n$-BitAdderStr$(x, y)) \subseteq$ InnerVertices$(m$-BitAdderStr$(x, y))$.

(15) For every natural number n and for all finite sequences x, y holds InnerVertices$((n+1)$-BitAdderStr$(x, y)) =$ InnerVertices$(n$-BitAdderStr$(x, y)) \cup$ InnerVertices(BitAdderWithOverflowStr$(x(n + 1), y(n + 1),$ n-BitMajorityOutput$(x, y)))$.

Let k, n be natural numbers. Let us assume that $k \geqslant 1$ and $k \leqslant n$. Let x, y be finite sequences. The functor (k, n)-BitAdderOutput(x, y) yielding an element of InnerVertices$(n$-BitAdderStr$(x, y))$ is defined by:

(Def. 6) There exists a natural number i such that $k = i + 1$ and (k, n)-BitAdderOutput$(x, y) =$ BitAdderOutput$(x(k), y(k),$ i-BitMajorityOutput$(x, y))$.

Next we state several propositions:

(16) For all natural numbers n, k such that $k < n$ and for all finite sequences x, y holds $(k + 1, n)$-BitAdderOutput$(x, y) =$ BitAdderOutput$(x(k +$

$1), y(k+1), k$-BitMajorityOutput$(x, y))$.

(17) For every natural number n and for all finite sequences x, y holds InnerVertices(n-BitAdderStr(x, y)) is a binary relation.

(18) For all sets x, y, c holds InnerVertices(MajorityIStr(x, y, c)) $= \{\langle\langle x, y\rangle, \&\rangle, \langle\langle y, c\rangle, \&\rangle, \langle\langle c, x\rangle, \&\rangle\}$.

(19) For all sets x, y, c such that $x \neq \langle\langle y, c\rangle, \&\rangle$ and $y \neq \langle\langle c, x\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \&\rangle$ holds InputVertices(MajorityIStr(x, y, c)) $= \{x, y, c\}$.

(20) For all sets x, y, c holds InnerVertices(MajorityStr(x, y, c)) $= \{\langle\langle x, y\rangle, \&\rangle, \langle\langle y, c\rangle, \&\rangle, \langle\langle c, x\rangle, \&\rangle\} \cup \{$MajorityOutput$(x, y, c)\}$.

(21) For all sets x, y, c such that $x \neq \langle\langle y, c\rangle, \&\rangle$ and $y \neq \langle\langle c, x\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \&\rangle$ holds InputVertices(MajorityStr(x, y, c)) $= \{x, y, c\}$.

(22) For all non empty many sorted signatures S_1, S_2 such that $S_1 \approx S_2$ and InputVertices(S_1) = InputVertices(S_2) holds InputVertices$(S_1 +\cdot S_2)$ = InputVertices(S_1).

(23) For all sets x, y, c such that $x \neq \langle\langle y, c\rangle, \&\rangle$ and $y \neq \langle\langle c, x\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \mathrm{xor}\rangle$ holds InputVertices(BitAdderWithOverflowStr(x, y, c)) $= \{x, y, c\}$.

(24) For all sets x, y, c holds InnerVertices(BitAdderWithOverflowStr(x, y, c)) $= \{\langle\langle x, y\rangle, \mathrm{xor}\rangle$, 2GatesCircOutput$(x, y, c, \mathrm{xor})\} \cup \{\langle\langle x, y\rangle, \&\rangle, \langle\langle y, c\rangle, \&\rangle, \langle\langle c, x\rangle, \&\rangle\} \cup \{$MajorityOutput$(x, y, c)\}$.

Let us mention that every set which is empty is also non pair.

Observe that $\emptyset$ is nonpair yielding. Let f be a nonpair yielding function and let x be a set. Observe that $f(x)$ is non pair.

Let n be a natural number and let x, y be finite sequences. Note that n-BitMajorityOutput(x, y) is pair.

The following propositions are true:

(25) Let x, y be finite sequences and n be a natural number. Then (n-BitMajorityOutput$(x, y))_{\mathbf{1}} = \varepsilon$ and (n-BitMajorityOutput$(x, y))_{\mathbf{2}}$ = $\underline{Boolean^0 \longmapsto false}$ and π_1((n-BitMajorityOutput$(x, y))_{\mathbf{2}}) = Boolean^0$ or (n-BitMajorityOutput$(x, y))_{\mathbf{1}} = 3$ and (n-BitMajorityOutput$(x, y))_{\mathbf{2}}$ = or_3 and π_1((n-BitMajorityOutput$(x, y))_{\mathbf{2}}) = Boolean^3$.

(26) For every natural number n and for all finite sequences x, y and for every set p holds n-BitMajorityOutput$(x, y) \neq \langle p, \&\rangle$ and n-BitMajorityOutput$(x, y) \neq \langle p, \mathrm{xor}\rangle$.

(27) Let f, g be nonpair yielding finite sequences and n be a natural number. Then InputVertices($(n+1)$-BitAdderStr(f, g)) = InputVertices(n-BitAdderStr(f, g)) $\cup$ (InputVertices (BitAdderWithOverflowStr$(f(n+1), g(n+1), n$-BitMajorityOutput(f, g))) $\setminus$ $\{n$-BitMajorityOutput$(f, g)\}$) and InnerVertices(n-BitAdderStr(f, g)) is a binary relation and InputVertices(n-BitAdderStr(f, g)) has no pairs.

(28) For every natural number n and for all nonpair yielding finite sequences x, y with length n holds InputVertices(n-BitAdderStr(x, y)) $=$ rng $x \cup$ rng y.

(29) Let x, y, c be sets, s be a state of MajorityCirc(x, y, c), and a_1, a_2, a_3 be elements of *Boolean*. If $a_1 = s(\langle\langle x, y\rangle, \&\rangle)$ and $a_2 = s(\langle\langle y, c\rangle, \&\rangle)$ and $a_3 = s(\langle\langle c, x\rangle, \&\rangle)$, then (Following$(s)$)(MajorityOutput$(x, y, c)$) $= a_1 \vee a_2 \vee a_3$.

(30) Let x, y, c be sets. Suppose $x \neq \langle\langle y, c\rangle, \&\rangle$ and $y \neq \langle\langle c, x\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \mathrm{xor}\rangle$. Let s be a state of MajorityCirc(x, y, c). Then Following$(s, 2)$ is stable.

(31) Let x, y, c be sets. Suppose $x \neq \langle\langle y, c\rangle, \&\rangle$ and $y \neq \langle\langle c, x\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \mathrm{xor}\rangle$. Let s be a state of BitAdderWithOverflowCirc(x, y, c) and a_1, a_2, a_3 be elements of *Boolean*. Suppose $a_1 = s(x)$ and $a_2 = s(y)$ and $a_3 = s(c)$. Then (Following$(s, 2)$)(BitAdderOutput(x, y, c)) $= a_1 \oplus a_2 \oplus a_3$ and (Following$(s, 2)$)(MajorityOutput(x, y, c)) $= a_1 \wedge a_2 \vee a_2 \wedge a_3 \vee a_3 \wedge a_1$.

(32) Let x, y, c be sets. Suppose $x \neq \langle\langle y, c\rangle, \&\rangle$ and $y \neq \langle\langle c, x\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \&\rangle$ and $c \neq \langle\langle x, y\rangle, \mathrm{xor}\rangle$. Let s be a state of BitAdderWithOverflowCirc(x, y, c). Then Following$(s, 2)$ is stable.

(33) Let n be a natural number, x, y be nonpair yielding finite sequences with length n, and s be a state of n-BitAdderCirc(x, y). Then Following$(s, 1 + 2 \cdot n)$ is stable.

References

[1] Grzegorz Bancerek. Cardinal numbers. *Formalized Mathematics*, 1(**2**):377–382, 1990.

[2] Grzegorz Bancerek. Curried and uncurried functions. *Formalized Mathematics*, 1(**3**):537–541, 1990.

[3] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[4] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.

[5] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[6] Grzegorz Bancerek and Yatsuka Nakamura. Full adder circuit. Part I. *Formalized Mathematics*, 5(**3**):367–380, 1996.

[7] Grzegorz Bancerek, Shin'nosuke Yamaguchi, and Yasunari Shidama. Combining of multi cell circuits. *Formalized Mathematics*, 10(**1**):47–64, 2002.

[8] Czesław Byliński. Finite sequences and tuples of elements of a non-empty sets. *Formalized Mathematics*, 1(**3**):529–536, 1990.

[9] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[10] Yatsuka Nakamura and Grzegorz Bancerek. Combining of circuits. *Formalized Mathematics*, 5(**2**):283–295, 1996.

[11] Yatsuka Nakamura, Piotr Rudnicki, Andrzej Trybulec, and Pauline N. Kawamoto. Introduction to circuits, II. *Formalized Mathematics*, 5(**2**):273–278, 1996.

[12] Yatsuka Nakamura, Piotr Rudnicki, Andrzej Trybulec, and Pauline N. Kawamoto. Preliminaries to circuits, II. *Formalized Mathematics*, 5(**2**):215–220, 1996.

[13] Takaya Nishiyama and Yasuho Mizuhara. Binary arithmetics. *Formalized Mathematics*, 4(**1**):83–86, 1993.
[14] Andrzej Trybulec. Enumerated sets. *Formalized Mathematics*, 1(**1**):25–34, 1990.
[15] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.
[16] Andrzej Trybulec. Tuples, projections and Cartesian products. *Formalized Mathematics*, 1(**1**):97–105, 1990.
[17] Andrzej Trybulec. Many-sorted sets. *Formalized Mathematics*, 4(**1**):15–22, 1993.
[18] Andrzej Trybulec. Many sorted algebras. *Formalized Mathematics*, 5(**1**):37–42, 1996.
[19] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.
[20] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.
[21] Edmund Woronowicz. Many–argument relations. *Formalized Mathematics*, 1(**4**):733–737, 1990.
[22] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**):73–83, 1990.

Received March 22, 2002