FORMALIZED MATHEMATICS
Volume 7, Number 2, 1998
University of Białystok

The Basic Properties of SCM over Ring

Artur Korniłowicz
University of Białystok

MML Identifier: SCMRING2.

The articles [6], [7], [12], [1], [8], [2], [3], [10], [4], [11], [9], and [5] provide the terminology and notation for this paper.

1. SCM over Ring

In this paper I is an element of $\mathbb{Z}_8$, S is a non empty 1-sorted structure, t is an element of the carrier of S, and x is a set.

Let R be a good ring. The functor $\mathbf{SCM}(R)$ yields a strict AMI over {the carrier of R} and is defined by the conditions (Def. 1).

(Def. 1)(i) The objects of $\mathbf{SCM}(R) = \mathbb{N}$,

(ii) the instruction counter of $\mathbf{SCM}(R) = 0$,

(iii) the instruction locations of $\mathbf{SCM}(R) = \text{Instr-Loc}_{\text{SCM}}$,

(iv) the instruction codes of $\mathbf{SCM}(R) = \mathbb{Z}_8$,

(v) the instructions of $\mathbf{SCM}(R) = \text{Instr}_{\text{SCM}}(R)$,

(vi) the object kind of $\mathbf{SCM}(R) = \text{OK}_{\text{SCM}}(R)$, and

(vii) the execution of $\mathbf{SCM}(R) = \text{Exec}_{\text{SCM}}(R)$.

Let R be a good ring, let s be a state of $\mathbf{SCM}(R)$, and let a be an element of $\text{Data-Loc}_{\text{SCM}}$. Then $s(a)$ is an element of R.

Let R be a good ring. An object of $\mathbf{SCM}(R)$ is called a Data-Location of R if:

(Def. 2) It $\in$ (the objects of $\mathbf{SCM}(R)$) $\setminus$ ($\text{Instr-Loc}_{\text{SCM}} \cup \{0\}$).

For simplicity, we use the following convention: R is a good ring, r is an element of the carrier of R, a, b, c, d_1, d_2 are Data-Location of R, and i_1 is an instruction-location of $\mathbf{SCM}(R)$.

Next we state the proposition

ISSN 1426–2630

(1) x is a Data-Location of R iff $x \in \text{Data-Loc}_{\text{SCM}}$.

Let R be a good ring, let s be a state of $\mathbf{SCM}(R)$, and let a be a Data-Location of R. Then $s(a)$ is an element of R.

We now state several propositions:

(2) $\langle 0, \varepsilon \rangle \in \text{Instr}_{\text{SCM}}(S)$.

(3) $\langle 0, \varepsilon \rangle$ is an instruction of $\mathbf{SCM}(R)$.

(4) If $x \in \{1, 2, 3, 4\}$, then $\langle x, \langle d_1, d_2 \rangle \rangle \in \text{Instr}_{\text{SCM}}(S)$.

(5) $\langle 5, \langle d_1, t \rangle \rangle \in \text{Instr}_{\text{SCM}}(S)$.

(6) $\langle 6, \langle i_1 \rangle \rangle \in \text{Instr}_{\text{SCM}}(S)$.

(7) $\langle 7, \langle i_1, d_1 \rangle \rangle \in \text{Instr}_{\text{SCM}}(S)$.

Let R be a good ring and let a, b be Data-Location of R. The functor $a{:=}b$ yielding an instruction of $\mathbf{SCM}(R)$ is defined by:

(Def. 3) $a{:=}b = \langle 1, \langle a, b \rangle \rangle$.

The functor $\text{AddTo}(a, b)$ yielding an instruction of $\mathbf{SCM}(R)$ is defined by:

(Def. 4) $\text{AddTo}(a, b) = \langle 2, \langle a, b \rangle \rangle$.

The functor $\text{SubFrom}(a, b)$ yielding an instruction of $\mathbf{SCM}(R)$ is defined by:

(Def. 5) $\text{SubFrom}(a, b) = \langle 3, \langle a, b \rangle \rangle$.

The functor $\text{MultBy}(a, b)$ yielding an instruction of $\mathbf{SCM}(R)$ is defined as follows:

(Def. 6) $\text{MultBy}(a, b) = \langle 4, \langle a, b \rangle \rangle$.

Let R be a good ring, let a be a Data-Location of R, and let r be an element of the carrier of R. The functor $a{:=}r$ yields an instruction of $\mathbf{SCM}(R)$ and is defined by:

(Def. 7) $a{:=}r = \langle 5, \langle a, r \rangle \rangle$.

Let R be a good ring and let l be an instruction-location of $\mathbf{SCM}(R)$. The functor goto l yielding an instruction of $\mathbf{SCM}(R)$ is defined by:

(Def. 8) $\text{goto}\ l = \langle 6, \langle l \rangle \rangle$.

Let R be a good ring, let l be an instruction-location of $\mathbf{SCM}(R)$, and let a be a Data-Location of R. The functor **if** $a = 0$ **goto** l yielding an instruction of $\mathbf{SCM}(R)$ is defined as follows:

(Def. 9) **if** $a = 0$ **goto** $l = \langle 7, \langle l, a \rangle \rangle$.

One can prove the following proposition

(8) Let I be a set. Then I is an instruction of $\mathbf{SCM}(R)$ if and only if one of the following conditions is satisfied:

(i) $I = \langle 0, \varepsilon \rangle$, or

(ii) there exist a, b such that $I = a{:=}b$, or

(iii) there exist a, b such that $I = \text{AddTo}(a, b)$, or

(iv) there exist a, b such that $I = \text{SubFrom}(a, b)$, or

(v) there exist a, b such that $I = \mathrm{MultBy}(a, b)$, or

(vi) there exists i_1 such that $I = \mathrm{goto}\ i_1$, or

(vii) there exist a, i_1 such that $I = \mathbf{if}\ a = 0\ \mathbf{goto}\ i_1$, or

(viii) there exist a, r such that $I = a{:=}r$.

In the sequel s denotes a state of $\mathbf{SCM}(R)$.

Let us consider R. Observe that $\mathbf{SCM}(R)$ is von Neumann.

The following two propositions are true:

(9) $\mathbf{IC}_{\mathbf{SCM}(R)} = 0$.

(10) For every **SCM**-state S over R such that $S = s$ holds $\mathbf{IC}_s = \mathbf{IC}_S$.

Let R be a good ring and let i_1 be an instruction-location of $\mathbf{SCM}(R)$. The functor $\mathrm{Next}(i_1)$ yields an instruction-location of $\mathbf{SCM}(R)$ and is defined by:

(Def. 10) There exists an element m_1 of $\mathrm{Instr\text{-}Loc}_{\mathrm{SCM}}$ such that $m_1 = i_1$ and $\mathrm{Next}(i_1) = \mathrm{Next}(m_1)$.

The following propositions are true:

(11) For every instruction-location i_1 of $\mathbf{SCM}(R)$ and for every element m_1 of $\mathrm{Instr\text{-}Loc}_{\mathrm{SCM}}$ such that $m_1 = i_1$ holds $\mathrm{Next}(m_1) = \mathrm{Next}(i_1)$.

(12) Let I be an instruction of $\mathbf{SCM}(R)$ and i be an element of $\mathrm{Instr}_{\mathrm{SCM}}(R)$. If $i = I$, then for every **SCM**-state S over R such that $S = s$ holds $\mathrm{Exec}(I, s) = \mathrm{Exec\text{-}Res}_{\mathrm{SCM}}(i, S)$.

2. Users Guide

Next we state several propositions:

(13) $(\mathrm{Exec}(a{:=}b, s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ and $(\mathrm{Exec}(a{:=}b, s))(a) = s(b)$ and for every c such that $c \neq a$ holds $(\mathrm{Exec}(a{:=}b, s))(c) = s(c)$.

(14) $(\mathrm{Exec}(\mathrm{AddTo}(a, b), s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ and $(\mathrm{Exec}(\mathrm{AddTo}(a, b), s))(a) = s(a) + s(b)$ and for every c such that $c \neq a$ holds $(\mathrm{Exec}(\mathrm{AddTo}(a, b), s))(c) = s(c)$.

(15) $(\mathrm{Exec}(\mathrm{SubFrom}(a, b), s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ and $(\mathrm{Exec}(\mathrm{SubFrom}(a, b), s))(a) = s(a) - s(b)$ and for every c such that $c \neq a$ holds $(\mathrm{Exec}(\mathrm{SubFrom}(a, b), s))(c) = s(c)$.

(16) $(\mathrm{Exec}(\mathrm{MultBy}(a, b), s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ and $(\mathrm{Exec}(\mathrm{MultBy}(a, b), s))(a) = s(a) \cdot s(b)$ and for every c such that $c \neq a$ holds $(\mathrm{Exec}(\mathrm{MultBy}(a, b), s))(c) = s(c)$.

(17) $(\mathrm{Exec}(\mathrm{goto}\ i_1, s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = i_1$ and $(\mathrm{Exec}(\mathrm{goto}\ i_1, s))(c) = s(c)$.

(18) If $s(a) = 0_R$, then $(\mathrm{Exec}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_1, s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = i_1$ and if $s(a) \neq 0_R$, then $(\mathrm{Exec}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_1, s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ and $(\mathrm{Exec}(\mathbf{if}\ a = 0\ \mathbf{goto}\ i_1, s))(c) = s(c)$.

(19) $(\mathrm{Exec}(a{:=}r, s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ and $(\mathrm{Exec}(a{:=}r, s))(a) = r$ and for every c such that $c \neq a$ holds $(\mathrm{Exec}(a{:=}r, s))(c) = s(c)$.

3. Halt Instruction

The following two propositions are true:

(20) For every instruction I of $\mathbf{SCM}(R)$ such that there exists s such that $(\mathrm{Exec}(I, s))(\mathbf{IC}_{\mathbf{SCM}(R)}) = \mathrm{Next}(\mathbf{IC}_s)$ holds I is non halting.

(21) For every instruction I of $\mathbf{SCM}(R)$ such that $I = \langle 0, \varepsilon \rangle$ holds I is halting.

Let us consider R, a, b. One can check the following observations:

* $a{:=}b$ is non halting,
* $\mathrm{AddTo}(a, b)$ is non halting,
* $\mathrm{SubFrom}(a, b)$ is non halting, and
* $\mathrm{MultBy}(a, b)$ is non halting.

Let us consider R, i_1. Observe that goto i_1 is non halting.

Let us consider R, a, i_1. Observe that **if** $a = 0$ **goto** i_1 is non halting.

Let us consider R, a, r. Note that $a{:=}r$ is non halting.

Let us consider R. One can check that $\mathbf{SCM}(R)$ is halting definite data-oriented steady-programmed and realistic.

One can prove the following propositions:

(29)[1] For every instruction I of $\mathbf{SCM}(R)$ such that I is halting holds $I = \mathbf{halt}_{\mathbf{SCM}(R)}$.

(30) $\mathbf{halt}_{\mathbf{SCM}(R)} = \langle 0, \varepsilon \rangle$.

References

[1] Grzegorz Bancerek. König's theorem. *Formalized Mathematics*, 1(**3**):589–593, 1990.
[2] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.
[3] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.
[4] Artur Korniłowicz. The construction of **SCM** over ring. *Formalized Mathematics*, 7(**2**):295–300, 1998.
[5] Eugeniusz Kusak, Wojciech Leończuk, and Michał Muzalewski. Abelian groups, fields and vector spaces. *Formalized Mathematics*, 1(**2**):335–342, 1990.
[6] Yatsuka Nakamura and Andrzej Trybulec. A mathematical model of CPU. *Formalized Mathematics*, 3(**2**):151–160, 1992.
[7] Yatsuka Nakamura and Andrzej Trybulec. On a mathematical model of programs. *Formalized Mathematics*, 3(**2**):241–250, 1992.
[8] Andrzej Trybulec. Enumerated sets. *Formalized Mathematics*, 1(**1**):25–34, 1990.

[1]The propositions (22)–(28) have been removed.

[9] Andrzej Trybulec. Tarski Grothendieck set theory. *Formalized Mathematics*, 1(**1**):9–11, 1990.
[10] Wojciech A. Trybulec. Vectors in real linear space. *Formalized Mathematics*, 1(**2**):291–296, 1990.
[11] Zinaida Trybulec. Properties of subsets. *Formalized Mathematics*, 1(**1**):67–71, 1990.
[12] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

Received November 29, 1998