

Formal syntax and semantics of SUGAR/PSL

Cindy Eisner¹ Dana Fisman^{1,2} John Havlicek³

¹ IBM Haifa Research Lab ² Weizmann Institute of Science ³ Motorola, Inc.

DRAFT - Please do not distribute!!!

August 26, 2003

Abstract. This document describes a proposal for the formal syntax and semantics of Accellera PSL.

The clocked semantics for the LTL subset follows the clocks paper [?], with the exception that strength is applied at the boolean level rather than at the propositional level.

The clocked semantics for SERES is very similar to the one in the LRM version 1.01 [?]. The main difference is a correction of the clocked semantics of $r[*0]$.

The abort semantics follows the abort paper [?] but is presented differently, as suggested in the sva-semantics [?].

1 Syntax of Accellera PSL

The logic Accellera PSL is defined with respect to a non-empty set of atomic propositions P and a given set of boolean expressions B over P . We assume two designated boolean expression *true* and *false* belong to B .

Definition 1 (SERES).

- Every boolean expression $b \in B$ is a SERE.
- If r , r_1 , and r_2 are SERES, and c is a boolean expression, then the following are SERES:
 - $\{r\}$ • $r_1 ; r_2$ • $\{r_1\} : \{r_2\}$ • $\{r_1\} | \{r_2\}$
 - $\{r_1\} \&\& \{r_2\}$ • $r[*0]$ • $r[*]$ • $r@c$

Definition 2 (Accellera PSL formulas).

- If b is a boolean expression then both b and $b!$ are Accellera PSL formulas.
- If φ and ψ are Accellera PSL formulas, r, r_1, r_2 are non-degenerate SERES¹, and b a boolean expression, then the following are Accellera PSL formulas:
 - (φ) • $\neg\varphi$ • $\varphi \wedge \psi$ • $X! \varphi$ • $[\varphi U \psi]$
 - $\varphi \text{ abort } b$ • $\varphi@c$ • $\{r\}(\varphi)$ • $\{r_1\} \mapsto \{r_2\}$

¹ See Section 2 for the definition of non-degenerate SERE.

2 Semantics of Accellera PSL

The semantics of Accellera PSL is defined with respect to finite and infinite words over $\Sigma = 2^P \cup \{\top, \perp\}$. We denote a letter from Σ by ℓ (possibly with subscripts) and an empty, finite, or infinite word from Σ by u , v , or w . We denote the length of word v as $|v|$. An empty word $v = \epsilon$ has length 0, a finite word $v = (\ell_0 \ell_1 \ell_2 \dots \ell_n)$ has length $n + 1$, and an infinite word has length ∞ . We use i , j , and k to denote non-negative integers. We denote the i^{th} letter of v by v^{i-1} (since counting of letters starts at zero). We denote by $v^{i..}$ the suffix of v starting at v^i . That is, for every $i < |v|$, $v^{i..} = v^i v^{i+1} \dots v^n$ or $v^{i..} = v^i v^{i+1} \dots$. We denote by $v^{i..j}$ the finite sequence of letters starting from v^i and ending in v^j . That is, for $j \geq i$, $v^{i..j} = v^i v^{i+1} \dots v^j$ and for $j < i$, $v^{i..j} = \epsilon$. We use ℓ^ω to denote an infinite-length word, each letter of which is ℓ .

We use \bar{v} to denote the word obtained by replacing every \top with a \perp and vice versa. We call \bar{v} the *complement* of v .

The semantics of Accellera PSL *formulas* over *words* is defined inductively, using as the base case the semantics of *boolean expressions* over *letters* in Σ . The semantics of boolean expression is assumed to be given as a relation $\models \subseteq \Sigma \times B$ relating letters in Σ with boolean expressions in B . If $(\ell, b) \in \models$ we say that the letter ℓ *satisfies* the boolean expression b and denote it $\ell \models b$. We assume the two special letters \top and \perp behave as follows: for every boolean expression b , $\top \models b$ and $\perp \not\models b$. For every other letter $\ell \in 2^P$, we assume that $\ell \models \text{true}$ and $\ell \not\models \text{false}$.

2.1 Unlocked Semantics

Semantics of unlocked SERES

Unlocked SERES are defined over finite words from the alphabet Σ . The notation $v \models r$, where r is a SERE and v a finite word means that v is *in the language* of r . The semantics of unlocked SERES are defined as follows, where b denote a boolean expression, and r , r_1 , and r_2 denote unlocked SERES.

- $v \models \{r\} \iff v \models r$
- $v \models b \iff |v| = 1$ and $v \models b$
- $v \models r_1; r_2 \iff \exists v_1, v_2$ s.t. $v = v_1 v_2$, $v_1 \models r_1$, and $v_2 \models r_2$
- $v \models \{r_1\}:\{r_2\} \iff \exists v_1, v_2$, and ℓ s.t. $v = v_1 \ell v_2$, $v_1 \ell \models r_1$, and $\ell v_2 \models r_2$
- $v \models \{r_1\}\{r_2\} \iff v \models r_1$ or $v \models r_2$
- $v \models \{r_1\}\&\&\{r_2\} \iff v \models r_1$ and $v \models r_2$
- $v \models r[*0] \iff v = \epsilon$
- $v \models r[*] \iff$ either $v \models r[*0]$ or $\exists v_1, v_2$ s.t. $v_1 \neq \epsilon$, $v = v_1 v_2$, $v_1 \models r$ and $v_2 \models r[*]$

An unlocked SERE r is *non-degenerate* if there exists a non-empty finite word w over Σ such that $w \models r$.

Semantics of unlocked Accellera PSL

We refer to a formula of Accellera PSL with no $\textcircled{\text{A}}$ operator as an *unlocked formula*. Let v be a finite or infinite word, b be a boolean expression, r, r_1, r_2 non-degenerate unlocked SERES, and φ, ψ unlocked Accellera PSL formulas. We use \models to define the semantics of unlocked Accellera PSL formulas: If $v \models \varphi$ we say that v *models* (or *satisfies*) φ .²

1. $v \models (\varphi) \iff v \models \varphi$
2. $v \models b! \iff |v| > 0$ and $v^0 \models b$
3. $v \models b \iff |v| = 0$ or $v^0 \models b$
4. $v \models \neg\varphi \iff \bar{v} \not\models \varphi$
5. $v \models \varphi \wedge \psi \iff v \models \varphi$ and $v \models \psi$
6. $v \models \mathbf{X!} \varphi \iff |v| > 1$ and $v^{1..} \models \varphi$
7. $v \models [\varphi \mathbf{U} \psi] \iff \exists k < |v|$ s.t. $v^{k..} \models \psi$, and $\forall j < k$, $v^{j..} \models \varphi$
8. $v \models \varphi \mathbf{abort} b \iff$ either $v \models \varphi$ or $\exists k < |v|$ s.t. $v^k \models b$ and $v^{0..k-1} \top^\omega \models \varphi$
9. $v \models \{r\}(\varphi) \iff \forall j < |v|$ s.t. $\bar{v}^{0..j} \models r$, $v^{j..} \models \varphi$
10. $v \models \{r_1\} \mapsto \{r_2\} \iff \forall j < |v|$ s.t. $\bar{v}^{0..j} \models r_1$, *either* $\exists k$ s.t. $j \leq k < |v|$ and $v^{j..k} \models r_2$ *or* $\forall k$ s.t. $j \leq k < |v|$, $\exists v'$ s.t. $v^{j..k} v' \models r_2$

2.2 Clocked Semantics

We say that finite word v *is a clock tick of* c iff $|v| > 0$ and $v^{|v|-1} \models c$ and for every natural number $i < |v| - 1$, $v^i \models \neg c$.

Semantics of clocked SERES

Clocked SERES are defined over finite words from the alphabet Σ and a boolean expression that serves as the clock context. The notation $v \stackrel{c}{\models} r$, where r is a SERE and c is a boolean expression, means that v is *in the language of* r *in context of clock* c . The semantics of clocked SERES are defined as follows, where b, c , and c_1 denote boolean expressions, r, r_1 , and r_2 denote clocked SERES.

- $v \stackrel{c}{\models} \{r\} \iff v \stackrel{c}{\models} r$
- $v \stackrel{c}{\models} b \iff v$ is a clock tick of c and $v^{|v|-1} \models b$
- $v \stackrel{c}{\models} r_1; r_2 \iff \exists v_1, v_2$ s.t. $v = v_1 v_2$, $v_1 \stackrel{c}{\models} r_1$, and $v_2 \stackrel{c}{\models} r_2$
- $v \stackrel{c}{\models} \{r_1\}; \{r_2\} \iff \exists v_1, v_2$, and ℓ s.t. $v = v_1 \ell v_2$, $v_1 \ell \stackrel{c}{\models} r_1$, and $\ell v_2 \stackrel{c}{\models} r_2$

² The semantics given here is equivalent to the truncated semantics given in [?] which is interpreted over 2^P rather than over $2^P \cup \{\top, \perp\}$. Using \models_\bullet for the semantics in [?] the following proposition states the equivalence: Let w be a finite word over 2^P , and let φ be a formula of $\text{LTL}^{\text{trunc}}$. Then

1. $w \models_\bullet^- \varphi \iff w \top^\omega \models \varphi$
2. $w \models_\bullet^+ \varphi \iff w \perp^\omega \models \varphi$
3. $w \models_\bullet \varphi \iff w \models \varphi$

- $v \models^c \{r_1\} | \{r_2\} \iff v \models^c r_1 \text{ or } v \models^c r_2$
- $v \models^c \{r_1\} \&\& \{r_2\} \iff v \models^c r_1 \text{ and } v \models^c r_2$
- $v \models^c r[*0] \iff v = \epsilon$
- $v \models^c r[*] \iff \text{either } v \models^c r[*0] \text{ or } \exists v_1, v_2 \text{ s.t. } v_1 \neq \epsilon, v = v_1 v_2, v_1 \models^c r \text{ and } v_2 \models^c r[*]$
- $v \models^c r@c_1 \iff v \models^c r$

A SERE r is *non-degenerate* if there exists a non-empty word w over Σ and a boolean expression c such that $w \models^c r$.

Semantics of clocked Accellera PSL

The semantics of (clocked) Accellera PSL formulas is defined with respect to finite/infinite words over Σ and a boolean expression c which serves as the clock context. Let v be a finite or infinite word, b, c, c_1 boolean expressions, r, r_1, r_2 non-degenerate SERES, and φ, ψ Accellera PSL formulas. We use \models^c to define the semantics of Accellera PSL formulas. If $v \models^c \varphi$ we say that v *models* (or *satisfies*) φ *in the context of clock* c .

1. $v \models^c (\varphi) \iff v \models^c \varphi$
2. $v \models^c b! \iff \exists j < |v| \text{ s.t. } v^{0..j} \text{ is a clock tick of } c \text{ and } v^j \models b$
3. $v \models^c b \iff \text{if } \exists j < |v| \text{ s.t. } \bar{v}^{0..j} \text{ is a clock tick of } c \text{ then } v^j \models b$
4. $v \models^c \neg\varphi \iff \bar{v} \not\models^c \varphi$
5. $v \models^c \varphi \wedge \psi \iff v \models^c \varphi \text{ and } v \models^c \psi$
6. $v \models^c \mathbf{X!} f \iff \exists j < k < |v| \text{ s.t. } v^{0..j} \text{ and } v^{j+1..k} \text{ are clock ticks of } c \text{ and } v^{k..} \models^c f$
7. $v \models^c [\varphi \mathbf{U} \psi] \iff \exists k < |v| \text{ s.t. } v^k \models c, v^{k..} \models^c \psi, \text{ and } \forall j < k \text{ s.t. } v^j \models c, v^{j..} \models^c \varphi$
8. $v \models^c \varphi \text{ abort } b \iff \text{either } v \models^c \varphi \text{ or } \exists k < |v| \text{ s.t. } v^k \models b \text{ and } v^{0..k-1} \top^\omega \models^c \varphi$
9. $v \models^c \varphi@c_1 \iff v \models^c \varphi$
10. $v \models^c \{r\}(\varphi) \iff \forall j < |v| \text{ s.t. } \bar{v}^{0..j} \models^c r, v^{j..} \models^c \varphi$
11. $v \models^c \{r_1\} \mapsto \{r_2\} \iff \forall j < |v| \text{ s.t. } \bar{v}^{0..j} \models^c r_1, \text{ either } \exists k \text{ s.t. } j \leq k < |v| \text{ and } v^{j..k} \models^c r_2 \text{ or } \forall k \text{ s.t. } j \leq k < |v|, \exists v' \text{ s.t. } v^{j..k} v' \models^c r_2$

Rewrite Rules

The rewrite rules for SERES are:

1. $\mathcal{R}^c(b) = \{\neg c[*]; c \wedge b\}$
2. $\mathcal{R}^c(r_1 ; r_2) = \mathcal{R}^c(r_1) ; \mathcal{R}^c(r_2)$
3. $\mathcal{R}^c(\{r_1\} : \{r_2\}) = \mathcal{R}^c(r_1) : \mathcal{R}^c(r_2)$
4. $\mathcal{R}^c(\{r_1\} | \{r_2\}) = \mathcal{R}^c(r_1) | \mathcal{R}^c(r_2)$
5. $\mathcal{R}^c(\{r_1\} \&\& \{r_2\}) = \mathcal{R}^c(r_1) \&\& \mathcal{R}^c(r_2)$

6. $\mathcal{R}^c(r[*0]) = \{\mathcal{R}^c(r)\}[*0]$
7. $\mathcal{R}^c(r[*]) = \{\mathcal{R}^c(r)\}[*]$
8. $\mathcal{R}^c(r@c_1) = \mathcal{R}^{c_1}(r)$

The rewrite rules for Sugar formulas are:

1. $\mathcal{F}^c(b!) = [\neg c \cup (c \wedge b)]$
2. $\mathcal{F}^c(b) = [\neg c \cup (c \wedge b)]$
3. $\mathcal{F}^c(\neg\varphi) = \neg\mathcal{F}^c(\varphi)$
4. $\mathcal{F}^c(\varphi \wedge \psi) = (\mathcal{F}^c(\varphi) \wedge \mathcal{F}^c(\psi))$
5. $\mathcal{F}^c(X!\varphi) = [\neg c \cup (c \wedge X! [\neg c \cup (c \wedge \mathcal{F}^c(\varphi))])]$
6. $\mathcal{F}^c(\varphi \cup \psi) = [(c \rightarrow \mathcal{F}^c(\varphi)) \cup (c \wedge \mathcal{F}^c(\psi))]]$
7. $\mathcal{F}^c(\varphi \text{ abort } b) = \mathcal{F}^c(\varphi) \text{ abort } b$
8. $\mathcal{F}^c(\varphi@c_1) = \mathcal{F}^{c_1}(\varphi)$
9. $\mathcal{F}^c(\{r\}(f)) = \{\mathcal{R}^c(r)\}(\mathcal{F}^c(f))$
10. $\mathcal{F}^c(\{r_1\} \mapsto \{r_2\}) = \{\mathcal{R}^c(r_1)\} \mapsto \{\mathcal{R}^c(r_2)\}$

3 Syntactic Sugaring

Exactly as in the LRM (V1.01) plus

- $\{r_1\} \mapsto \{r_2\}! \stackrel{\text{def}}{=} \underbrace{\{r_1\}(\neg\{r_2\})(\text{false}@true))}_{k \text{ times}}$
- $r[*k] = \overbrace{r; r; \dots; r}^k$ (recall that in the LRM $k \geq 1$)

and minus

- existing syntactic sugaring for $r[*i]$

4 Some characteristics of Accellera PSL

Lemma 1. *Let v be a finite word over Σ .*

v is a clock tick of true iff $v = \top^k \alpha$, where $k \geq 0$ and $\alpha \neq \perp$.

Lemma 2. *Let v be a non-empty word over Σ , c a boolean expression, and r an unlocked SERE. If $v \stackrel{c}{\models} r$ then $v^{|v|-1} \models c$*

Proposition 1. *Let v be a finite word over Σ and let r be an unlocked SERE.*

If $v \models r$, then $v \stackrel{\text{true}}{\models} r$.

Remark 1. The converse of the preceding proposition does not hold. For example, $\top^2 \stackrel{\text{true}}{\models} \text{true}$, but $\top^2 \not\models \text{true}$. The converse does hold if v is a word over 2^P .

Lemma 3. *Let v be a finite word over Σ and let r be an unlocked SERE. Then $v \models r[+]$ iff there exist $k > 0$ and v_1, \dots, v_k such that $v = v_1 \dots v_k$ and $v_j \models r$ for each $1 \leq j \leq k$.*

Lemma 4. *Let v be a finite word over Σ , let c be a boolean expression, and let r be a SERE. Then $v \models^c r[+]$ iff there exist $k > 0$ and v_1, \dots, v_k such that $v = v_1 \dots v_k$ and $v_j \models^c r$ for each $1 \leq j \leq k$.*

Lemma 5. *If v is a finite word over Σ and if $v \models r$, where r is an unlocked SERE, then no letter of v is \perp .*

Lemma 6. *If v is a finite word over Σ , c is a boolean expression, and $v \models^c r$, where r is a SERE, then no letter of v is \perp .*

Lemma 7. *Let r be an unlocked SERE and let v be a word over Σ . Then the following are equivalent:*

1. $v \models \neg\{r\}(\text{false})$
2. there exists $0 \leq j < |v|$ such that $v^{0..j} \models r$

Lemma 8. *Let r be an unlocked SERE, let c be a boolean expression, and let v be a word over Σ . Then the following are equivalent:*

1. $v \models^c \neg\{r\}(\text{false})$
2. there exists $0 \leq j < |v|$ such that $v^{0..j} \models^c r$

Lemma 9. *Let r be a SERE, let c be a boolean expression, and let v be a word over Σ . Then the following are equivalent:*

1. $v \models^c \neg\{r\}(\text{false@true})$
2. there exists $0 \leq j < |v|$ such that $v^{0..j} \models^c r$

Lemma 10. *Let r_1, r_2 be unlocked SEREs and let v be a word over Σ . Then the following are equivalent:*

1. $v \models \{r_1\} \mapsto \{r_2\}!$
2. for every $0 \leq j < |v|$ such that $\bar{v}^{0..j} \models r_1$, there exists $j \leq k < |v|$ such that $v^{j..k} \models r_2$
3. $v \models \{r_1\}(\neg\{r_2\}(\text{false}))$

Lemma 11. *Let r_1, r_2 be SEREs, let c be a boolean expression, and let v be a word over Σ . Then the following are equivalent:*

1. $v \models^c \{r_1\} \mapsto \{r_2\}!$
2. for every $0 \leq j < |v|$ such that $\bar{v}^{0..j} \models^c r_1$, there exists $j \leq k < |v|$ such that $v^{j..k} \models^c r_2$
3. $v \models^c \{r_1\}(\neg\{r_2\}(\text{false@true}))$

Convnetion When negation is applied to a boolean expression, the resulting formula is treated as boolean expression negation.

Lemma 12. *Let b be a boolean expression and let v be a word over Σ . Then*

1. $v \models b!$ iff $v \models \neg\{b\}$ (false).
2. $v \models b$ iff $v \models \{-b\}$ (false).

Lemma 13 (Duality of Boolean Formulas). *Let b, c be boolean expressions and let v be a word over Σ . Then*

1. $v \models^c \neg(\neg b)!$ iff $v \models^c b$
2. $v \models^c \neg(b!)$ iff $v \models^c (\neg b)$

Lemma 14. *Let b, c be boolean expressions and let v be a word over Σ .*

1. $v \models^c \{b\}$ (false) iff $v \models^c (\neg b)$
2. $v \models^c \{-b\}$ (false) iff $v \models^c b$

Lemma 15. *Let b, c be boolean expressions and let v be a word over Σ such that $\bar{v}^0 \models c$. Then the following are equivalent:*

1. $v \models^c b$
2. $v \models^c b!$

Lemma 16. *Let φ be an Accellera PSL formula, let c be a boolean expression, and let v be a word over Σ . The following statements are equivalent:*

1. $v \models^{\text{true}} (\mathbf{G} \varphi) @c$
2. for all $0 \leq i \leq |v|$ such that $\bar{v}^i \models c$, $v^{i..} \models^c \varphi$

Definition 3. *A word w over $2^P \cup \{\top, \perp\}$ is called proper if it is of the form $w = uv$, where u is a word over 2^P and v is one of the following:*

1. ϵ
2. \top^ω
3. \perp^ω

Lemma 17. *Let φ be an Accellera PSL formula, let b, c be boolean expressions, and let v be a word over Σ . The following statements are equivalent:*

1. $v \models^c (b! @ \text{true} \rightarrow \varphi)$
2. if there exists $0 \leq j \leq |v|$ such that $\bar{v}^{0..j}$ is a clock tick of true and $\bar{v}^j \models b$, then $v \models^c \varphi$

If, in addition, φ is non-degenerate and v is proper, then 1 and 2 are equivalent to

3. if $|v| > 0$ and $\bar{v}^0 \models b$, then $v \models^c \varphi$
4. $v \models^c \{b @ \text{true}\}(\varphi)$

If, in addition, $|v| > 0$ and $\bar{v}^0 \models c$, then 1-4 are equivalent to

5. $v \stackrel{c}{\models} b! \rightarrow \varphi$
6. $v \stackrel{c}{\models} b \rightarrow \varphi$
7. $v \stackrel{c}{\models} \{b\}(\varphi)$

Lemma 18. *Let φ be a non-degenerate Accellera PSL formula, let b and c be boolean expressions, and let v be a proper word over Σ . Then the following are equivalent:*

1. $v \stackrel{\text{true}}{\models} G(\{b@true\}(\varphi))@c$
2. for all $0 \leq j < |v|$, if $\bar{v}^j \models c$ and $\bar{v}^j \models b$ then $v^{j..} \stackrel{c}{\models} \varphi$
3. $v \stackrel{\text{true}}{\models} G(b! \rightarrow \varphi)@c$
4. $v \stackrel{\text{true}}{\models} G(b \rightarrow \varphi)@c$
5. $v \stackrel{\text{true}}{\models} G(\{b\}(\varphi))@c$