$vvar$
$name$
$enum\_ty$
$enum$
nat, num_bytes
bvi
bv
str
j, k

annot ::= Annotation

int ::=
- | *nat*
- | $-nat$

ty ::=
- | **Bool**
- | (_ **BitVec** *nat*)
- | *enum_ty*
- | (**Array** ty_1 ty_2)

bool ::=
- | **true**
- | **false**

unop ::=
- | **not**
- | **bvnot**
- | **bvredand**
- | **bvredor**
- | **bvneg**
- | (_ **extract** *nat* *nat*′)
- | (_ **zero_extend** *nat*)
- | (_ **sign_extend** *nat*)

bvarith ::=
- | **bvnand**
- | **bvnor**
- | **bvxnor**
- | **bvsub**
- | **bvudiv**
- | **bvudiv_i**
- | **bvsdiv**
- | **bvsdiv_i**
- | **bvurem**
- | **bvsrem**
- | **bvsmod**
- | **bvshl**
- | **bvlshr**
- | **bvashr**

bvcomp ::=
- | **bvult**
- | **bvslt**
- | **bvule**
- | **bvsle**
- | **bvuge**
- | **bvsge**

	| **bvugt**
	| **bvsgt**
binop	::=
	| =
	| *bvarith*
	| *bvcomp*
bvmanyarith	::=
	| **bvand**
	| **bvor**
	| **bvxor**
	| **bvadd**
	| **bvmul**
manyop	::=
	| **and**
	| **or**
	| *bvmanyarith*
	| **concat**
base_val	::=
	| *vvar*
	| *bool*
	| *bv*
	| *enum*
exp_aux	::=
	| *base_val*
	| (*unop exp*)
	| (*binop* exp_1 exp_2)
	| (*manyop* exp_1 .. exp_j)
	| (**ite** exp_1 exp_2 exp_3)
exp	::=
	| *exp_aux annot*
smt	::=
	| $(declare - constvvar\ ty)$
	| $(define - constvvar\ exp)$
	| (**assert** *exp*)
	| $(define - enumint)$
valu, *addr*, *data*, *rkind*, *wkind*, *bkind*, *ckind*, *opcode*	::=
	| *base_val*
	| (*_bvi int*)
	| *str*
	| (_ **unit**)
	| (*name*(*_unit*))

	|	$(\_\,\textbf{vec}\ valu_1\ ..\ valu_k)$	
	|	$(\_\,\textbf{list}\ valu_1\ ..\ valu_k)$	
	|	$(\_\,\textbf{struct}\ selem_1\ ..\ selem_k)$	
	|	$(\_\,\textbf{poison}\,)$	
selem	::=		
	|	$(name\ valu)$	
accessor	::=		
	|	$(\_\,\textbf{field}\ name)$	
accessor_list	::=		
	|	**nil**	
	|	$(accessor_1\ ..\ accessor_k)$	
valu_option, *tag_value*	::=		
	|	**None**	
	|	**Some** *valu*	
event_aux	::=		
	|	*smt*	
	|	$(\,\textbf{branch}\ int\ str)$	Sail trace fork
	|	$(read - regname\ accessor\_list\ valu)$	read value 'va
	|	$(write - regname\ accessor\_list\ valu)$	write value 'v
	|	$(read - memvalu\ rkind\ addr\ num\_bytes\ tag\_value)$	read value 'va
	|	$(write - memvalu\ wkind\ addr\ data\ num\_bytes\ tag\_value)$	write value 'v
	|	$(branch - address addr)$	announce bra
	|	$(\,\textbf{barrier}\ bkind)$	memory barri
	|	$(cache - opckind\ addr)$	cache mainter
	|	$(mark - regname\ str)$	instrumentati
	|	$(\,\textbf{cycle}\,)$	instruction bo
	|	$(\,\textbf{instr}\ opcode)$	records the in
	|	$(\,\textbf{sleeping}\ vvar)$	Arm sleeping
	|	$(wake - request)$	Arm wake req
	|	$(sleep - request)$	Arm sleep req
event	::=		
	|	*event_aux annot*	
trc	::=		
	|	$(\,\textbf{trace}\ event_1\ ..\ event_j)$	
trcs	::=		
	|	$trc_1\ ..\ trc_k$	
formula	::=		
	|	*judgement*	
judgement	::=		

user_syntax ::=

- *vvar*
- *name*
- *enum_ty*
- *enum*
- *nat*
- *bvi*
- *bv*
- *str*
- *j*
- *annot*
- *int*
- *ty*
- *bool*
- *unop*
- *bvarith*
- *bvcomp*
- *binop*
- *bvmanyarith*
- *manyop*
- *base_val*
- *exp_aux*
- *exp*
- *smt*
- *valu*
- *selem*
- *accessor*
- *accessor_list*
- *valu_option*
- *event_aux*
- *event*
- *trc*
- *trcs*