SSBC Ver 5 Rev 1 (ARTN)

Processor State

PC<15 . . 0>:	Program Counter
SP<15 . . 0>:	Stack Pointer
MR<15 . . 0>:	Memory Register
R[0 . . 3]<7 . . 0>:	General Registers
IR<7 . . 0>:	Instruction Register
Z:	1-bit Zero flag
N:	1-bit Negative flag
Fault:	1-bit Fault Indicator
Halt:	1-bit Halt Indicator
Reset:	Reset Signal

Main Memory

Mem[0 . . $2^{16} - 1$]<7 . . 0>:
M[x]<7 . . 0> ≔ Mem[x]:

Instruction Format

op<3 . . 0> ≔ IR<3 . . 0>:

Operands/Address

ii<7 . . 0> ≔ M[PC]:
s1<7 . . 0> ≔ M[SP + 1]:
s2<7 . . 0> ≔ M[SP + 2]:
ext<15 . . 0> ≔ M[PC]#M[PC + 1]:

Program Status Word

PSW<7 . . 0> ≔ Z#N#(6@0):

Fault Detection

set_fault ≔ Fault ⟵ ¬ (0 ≤ op ≤ 0xA):

Instruction Interpretation

```
ins_int ≔ (
  Reset ⟶ ((PC ⟵ 0x0 : SP ⟵ 0xFFFA : Halt ⟵ 0x0 : Fault ⟵ 0x0) ; ins_int):
  (¬ Reset) ∧ (¬ Fault) ⟶ (IR ⟵ M[PC] ; set_fault ; (¬ Fault) ⟶ (PC ⟵ PC + 1 ; ins_exe))
);
```

Instruction Execution

```
ins_exe ≔ (
noop        (≔ op= 0)  ⟶ :
halt        (≔ op= 1)  ⟶ Halt ⟵ 0x1:
pushimm     (≔ op= 2)  ⟶ M[SP] ⟵ ii ; (SP ⟵ SP-1 : PC ⟵ PC + 1):
pushext     (≔ op= 3)  ⟶ M[SP] ⟵ M[ext] ; (SP ⟵ SP − 1 : PC ⟵ PC + 2):
popinh      (≔ op= 4)  ⟶ SP ⟵ SP + 1:
popext      (≔ op= 5)  ⟶ M[ext] ⟵ s1 ; (SP ⟵ SP + 1 : PC ⟵ PC + 2):
jnz         (≔ op= 6)  ⟶ (¬ Z) ⟶ PC ⟵ ext:
jnn         (≔ op= 7)  ⟶ (¬ N) ⟶ PC ⟵ ext:
add         (≔ op= 8)  ⟶ M[SP+2] ⟵ s1 + s2 ; (Z ⟵ ¬ (R2<7> ∨ R2<6> . . ∨ R2<0>) : N ⟵ R2<7> : SP ⟵ SP + 1):
sub         (≔ op= 9)  ⟶ M[SP+2] ⟵ s1 − s2 ; (Z ⟵ ¬ (R2<7> ∨ R2<6> . . ∨ R2<0>) : N ⟵ R2<7> : SP ⟵ SP + 1):
nor         (≔ op= 10) ⟶ M[SP+2] ⟵ s1 ⊽ s2 ; SP ⟵ SP + 1:
); ins_int
```

SSBC Ver 5 Rev 1 (ARTN)

Memory Map

PSW mapped to 0xFFFB
Port A (read only) mapped to 0xFFFC
Port B (write only) mapped to 0xFFFD
Port C (read only) mapped to 0xFFFE
Port D (write only) mapped to 0xFFFF

Notation (From Computer Systems Design and Architecture Heuring and Jordan)

⟶ If - Then: if the LHS is true then evaluate the RHS.

⟵ Register transfer: the register on the LHS stores the value from the RHS.

: Parallel separator: actions or evaluations on the LHS and the RHS are carried out simultaneously. Restated, the order of execution typically does not matter. However, if the LHS relies on the RHS then the RHS is evaluated before the LHS is changed. Eg) IR ⟵ MEM[PC] : PC ⟵ PC + 4

; Sequential separator: LHS evaluated and/or performed before the RHS.

@ Replication: RHS is replicated N times as specified by LHS, all concatenated.

≔ Definition: text substitution (alias).

# Concatenation: LHS and RHS are combined.

¬ Logical NOT

∧ Logical AND

∨ Logical OR

⊽ Logical NOR

Definitions

RTN:	Register Transfer Notation
ARTN:	Abstract RTN
CRTN:	Concrete RTN
LHS:	Left-hand Side
RHS:	Right-hand Side
MSB:	Most Significant Bit
LSB:	Least Significant Bit
SSBC:	Simple Stack Based Computer
ISA:	Instruction Set Architecture