Sonne-8 Microcontroller Reference Schematics Rev. Myth

"VLSI Chip" Footprint

Clock Phases

D(OWN COUNTER) Inner-loops

Instruction Decoder

Branching

Predecode signals for early triggers

Pre-Decoders PAIR Target

Pre-Decoder PAIR Source

Start, latch instruction

Instruction-Word Latch

Scrounger

Remap ("scrounge") inherent NOPs such as AA, EE etc. and impracticable opcodes such as NM to other instructions

Pair DST Decoder

Pair SRC Decoder

SYS Decoder

TRAP decoder Immediate calls

IO Decoder

GETPUT Decoder

Opcode Type Decoder

Instruction byte:

```
all0: OPC_SYS     00000   xxx    See table @ SYS decoder
else: OPC_IO      00001   xxx    See table @ IO decoder
      OPC_ALU     0001    xxxx   See table @ ALU
      OPC_TRAP    001     xxxxx  b0-4: DEST PAGE
      OPC_GETPUT  01 x x xxx     b0-2 OFFS, b3:GET/PUT, B4-5:REG(GETPUT)
      OPC_PAIR    1  xxx xxxx    b0-3 DST, b4-6 SRC
```

(PC Program Counter)

On RET: Restore PC value from O register (Center right)

On JUMP: Load target offset on DB into PC

On RET/xC/xC: Load instruction offset from O into PC

Inc PC after each instruction cycle

Delay SUBR until CLOSE

CLR: Set PC to 0 during TRAP. Execution starts at offset zero, "head" of target page

Control Tree

A receives ALU result A and O are input operands to the ALU

A(ccumulator) Register

Zero detection

ALU Instructions

```
0
1
2 DCA ("Ones Complement A" / A := ~A)
3 DCO ("Ones Complement O" / A := ~O)
4 SLA ("Shift left A" / A := A << 1)
5 SLO ("Shift left O" / A := O << 1)
6 SRA ("Shift right A" / A := A >>1)
7 SRO ("Shift right O" / A := O >> 1)

8 AND (A := A & O)
9 IOR (A := A | O)
10 EOR (A := A ^ O)
11 ADD ("Add" / A := A + O, bits 0-7)
12 CAR ("Carry Bit" / A := 9th bit of A + O) 00h or 01h
13 ALO ("Flag: A less than O" / A := (A<O)? 0xFF:0
14 AEO ("Flag: A equals O" / A := (A==O)? 0xFF:0
15 AGO ("Flag: A greater than O" / A := (A>O)? 0xFF:0
```

Alternatively use two SST39SF040 NOR-Flash

BLUR3 LUT ROM layout:

16 maps @256x256

See list of Instructions

ALU

Register at CLOSE via Instruction adder would feedback

B0-7 goes to PC module latched during RET

(Page OFFSET) Register

low

high

O has five purposes:
- Saves offset of current instruction byte during call/trap/go
- Outputs return offset during RET
- Together with A is ALU input operand
- Is the (only!) implied address offset for memory read-write, with the corresponding page index stored in B (Base)
- Together with B forms a 16-bit address pointer, which can be saved/retrieved using SAVE/USE instructions into JET/SKY

B (Base) Page Index

Pointer Regs Low Order (0)

Pointer Regs High Order (B)

B has four purposes:
- Saves page index of current instruction byte during call/trap/go
- Outputs return page index during RET
- Is the implied page index for memory read-write, with the corresponding byte offset stored in O (Offset)
- Together with O forms a 16-bit address pointer, which can be saved/retrieved using SAVE/USE instructions into JET/SKY

Input/Output

SIR

SOR

Serial IO

Implements SPI functionality with dedicated SER/DES registers

Device Enable

Two independent 4-bit groups (high/low) feed into hc16 decoders for device selection

PIR

POR

Parallel IO

8-bit GPIO

5731 Transparent Latch follows input

L (Local Page Index reg)

{LL[7..0]}

{LO[7..0]}

{HI[7..0]}

{DB[7..0]}

RAM/ROM

16K

C (ODE) Page index

C active during fetch or interpage control flow

B active during xM Mv data access instructions.

Register at CLOSE (MB is a valid instruction)

Based on working prototype PCB: https://github.com/michaelmangelsdorf/sonne8
Rev. Step "New"
Project homepage: https://github.com/michaelmangelsdorf/Myth
Author: Michael Mangelsdorf
Contact: mim@ok-schalter.de
Sheet: /
File: myth.kicad_sch

Title: Sonne-8 Micro-Controller

Size: A0	Date: 2025-06-01	Rev: Myth
KiCad 8.0.4		Id: 1/1