Memory (RAM/ROM)

The module provides the CPU with *2MB of volatile memory* and *32k of persistent EEPROM* memory. You are able to address locations larger than 16-bit by making use of the *memory segment registers*. With these, you can access the full 19-bits of address space and use multiple RAM chips..

Power: **5V Vcc**

Overview

- 1x EEPROM
  - 32k
  - 150ns
- 4x RAM
  - 512k
  - 25ns
- 3 Segment Registers
  - Code
  - Data
  - Stack
- Memory Address Register

Control & Instructions

Read & Write

Address	Segment [0..4]	Read	Write	Use Code	Use Data	Use Stack	Description
n.c.	n.c.	1	1	1	1	1	NOP
<= 0x7ffff	0x0	0	n.c.	0	1	1	Read Code EEPROM
<= 0x7ffff	0x0	0	n.c.	1	0	1	Read Data EEPROM
0x0...0x20000	0x1...0x1f	0	1	0	1	1	Read Code RAM
0x0...0x20000	0x1...0x1f	0	1	1	0	1	Read Data RAM
0x0...0x20000	0x1...0x1f	0	1	1	1	0	Read Stack RAM
>= 0x8000	n.c.	0	1	0	1	1	Read Code RAM
>= 0x8000	n.c.	0	1	1	0	1	Read Data RAM
>= 0x8000	n.c.	0	1	1	1	0	Read Stack RAM
0x0...0x20000	0x1...0x1f	1	0	1	0	1	Write Data RAM

>= 0x8000	n.c.	1i	0	1	0	1	Write Data RAM
0x0…0x20000	0x1…0x1f	1	0	1	1	0	Write Stack RAM
>= 0x8000	n.c.	1	0	1	1	0	Write Stack RAM

Memory Segment & Address Register

CSI	CSO	DSI	DSO	SSI	SSO	MAR	CLK	Description
1	1	1	1	1	1	1	_/	NOP
0	1	1	1	1	1	1	_/	Latches Code Segment
1	1	0	1	1	1	1	_/	Latches Data Segment
1	1	1	1	0	1	1	_/	Latches Stack Segment
1	0	1	1	1	1	1	_/	Assert Code Segment
1	1	1	0	1	1	1	_/	Assert Data Segment
1	1	1	1	1	0	1	_/	Assert Stack Segment
1	1	1	1	1	1	0	_/	Latch Memory Address Register

Memory Map

There is more detailed information about how Interrupts, Resets and the Entrypoint are handled can be found on the following sections: Instruction Cycle & Microcode section

Description	Address Range	Bytes	Location
Interrupt Vector Addresses [0..7]	$0:00000 … $0:00007	8	ROM
BIOS / PROGRAM	$0:00008 … $0:07fff	32k	ROM
Working Ram	$0:08000 … $0:fffff	491k	RAM 0
	$1:80000 … $1:fffff	524k	RAM 1
	$2:80000 … $2:fffff	524k	RAM 2
Working Ram / Stack	$3:80000 … $3:fffff	524k	RAM 3

Addressing Memory

Address Bits

Active Segment	Memory Address Register

RAM						EEPROM														
S4	S3	S2	S1	S0	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0

Code Segment (CS)

This is active when fetching the next instruction. It is only changed by a far jmp instruction and cannot be manipulated by mov.

```
// Example
jmp 0x2:0x8d32;
```

Data Segment (DS)

This is active during any read's or write's to Memory (excluding program code). This can be set using mov.

```
// Example
// Write
mov ds, 0x4;
// Active (uses the data segment value)
mov a, $8f40;
```

Stack Segment (SS)

This is active when executing a PUSH or POP instruction to and from the stack. This can be set using mov ss, 0xf

```
// Example
// Writes (sets the stack segment value)
mov ss, 0x2
pop ss, 0x2
// Active (uses the stack segment value)
push a;
pop a;
```

Parts & Components List

Part #	Description	Qty	Datasheet	Link
IS61C5128AS-25TLI	512K x 8 HIGH-SPEED CMOS STATIC RAM	4	DATA	Mouser
AT28C256	256K (32K x 8) Paged Parallel EEPROM	1	DATA	Mouser
74HC574	8-Bit, Edge-Triggered, flip-flop's	2	DATA	Mouser
74HC173	Quad D-Type Flip-Flop	2	DATA	Mouser
74HC245	8-bit, Tri-State Transceiver	7	DATA	Mouser
CD4072	Dual 4-Input OR gates	1	DATA	Mouser
74HC139	4 to 16 line, decoder	1	DATA	Mouser
74HC00	Dual 4-input NAND gates	1	DATA	Mouser
-	Ceramic Capacitor	8	-	-
-	Resistor (LED)	49	-	-
-	Light Emitting Diodes (LED)	49	-	-
-	PLCC Socket, 32 pin - MOUSER	1	-	-
-	Header Pins	-	-	-