**PHYSICAL CHANGES**

RAM upgraded to 32K ==> 15 address bits

* Double register (MSB + LSB)
* Extra parts: 6x 74LS173 for address + address prep registers, 1x 62256-80 (new ram chip), 2x octuple DIP switch for address, 2x 74LS157 for additional selecting, 2x 74LS245 for prep reg output
* Needless parts: 2x 74LS04 because of non-inverting ram output, 2x 74LS189 (old ram chip), 1x quadruple DIP switch for address
* Control signals: +MHI, +MLI, -MI, +ZHI, +ZLI, +ZHO, +ZLO **CLK NAND RI => /WE**

Instruction length upgraded to 7 bits

* Arguments are now seperate bytes
* Needless parts: 1x 74LS245 because of no output
* Control signals: -IO

Counter length upgraded to 256

* Extra parts: 1x 74LS161 for extra counter bits

Memory Pages + Counter

* Extra parts: 2x 74LS161 for page count, 1x 74LS245 for bus output
* Control signals: +PI, +PO

Conditional instructions

* Extra parts: 2x 74LS85 for comparing, 1x 74LS173 to store result, 1x74LS126 to output results
* Control signals: +CMP, +CMPO, +CPB0, +CPB1

Shift register

* Binary shift instructions
* Extra parts: 1x 74HC299 (register), 1x 74LS08 for controlling roll-over, 1x 74LS245 for bus output
* Control signals: +SM0, +SM1, +SRO, +SO

Buffer register

* Used for microcode-level buffering
* Extra parts: 2x 74LS173 (register), 1x 74LS245 for bus output
* Control signals: +FI, +FO

User input

* Stops the clock to wait for user input
* Extra parts: 1x button for confirming, 1x octuple DIP switch for input, 1x 74LS00 for halt latch + logic, 1x 74LS245 for bus output
* Control signals: +NO

Bitwise operations

* Extra parts: 4x 74LS153 as output selectors, 1x 74LS245 for output, 2x 74LS04 (inverters), 2x 74LS08 (and), 2x 74LS32 (or), 2x 74LS86 (xor)
* Control signals: +T0, +T1, +TO

Call value register

* Return values from functions or pass arguments
* Extra parts: 2x 74LS173 (register), 1x 74LS245 for bus output
* Control signals: +VI, +VO

Swapped output register

* Extra parts: 2x 74LS173
* Needless parts: 1x 74LS273, 1x 74LS08

Numeric output mode

* Extra parts: 1x 74LS173
* Control signals: +OMI

Reset Circuit

* Extra parts: 1x button, 1x 74LS00 for inverting and gating, 1x 1k resistor (or other value) for pull-down

**CONTROL SIGNAL OVERVIEW**

|  |  |  |
| --- | --- | --- |
| Inputs | (no in), MHI, MLI, RI, II, AI, BI, CI, FI, OI, VI, PI, XI, ZHI, ZLI, OMI | = 16 multiplexed to 4 |
| Controls | HLT, SUB, CE, CMP, SM0, SM1, SRO, T0, T1, RM, XR, RMSB, FE, CPB0, CBP1, CPTC | = 16 |
| Outputs | (no out), RO, AO, EO, CO, SO, FO, NO, TO, VO, PO, OO, CMPO, ZHO, ZLO, OUT\_15 (unused) | = 16 multiplexed to 4 |
| Total |  | = 48 (with multiplexing 24) |

Multiplexing requires 2x 74HCT 4514, RM is “Reset Microinstruction Counter”, OO is “one out” (sets bus to 0x01)

FE sets the bus to 0xFE (use with OO to get 0xFF)

**INSTRUCTIONS**

1. NOP
2. MOV literal – A
3. MOV literal – B
4. MOV literal – S
5. MOV literal – V
6. MOV literal – ZH
7. MOV literal – ZL
8. MOV literal – Z
9. MOV literal – RAM
10. MOV literal – RAML
11. MOV RAM – A
12. MOV RAM – B
13. MOV RAM – S
14. MOV RAM – V
15. MOV RAM – ZH
16. MOV RAM – ZL
17. MOV RAM – RAML
18. MOV RAML – A
19. MOV RAML – B
20. MOV RAML – S
21. MOV RAML – V
22. MOV RAML – ZH
23. MOV RAML – ZL
24. MOV RAML – RAM
25. MOV RAML – RAML
26. MOV A – B
27. MOV A – S
28. MOV A – V
29. MOV A – ZH
30. MOV A – ZL
31. MOV A – RAM
32. MOV A – RAML
33. MOV S – A
34. MOV S – B
35. MOV S – V
36. MOV S – ZH
37. MOV S – ZL
38. MOV S – RAM
39. MOV S – RAML
40. MOV V – A
41. MOV V – B
42. MOV V – S
43. MOV V – ZH
44. MOV V – ZL
45. MOV V – RAM
46. MOV V – RAML
47. MOV ZH – A
48. MOV ZH – B
49. MOV ZH – S
50. MOV ZH – V
51. MOV ZH – RAM
52. MOV ZH – RAML
53. MOV ZL – A
54. MOV ZL – B
55. MOV ZL – S
56. MOV ZL – V
57. MOV ZL – RAM
58. MOV ZL – RAML
59. MOV C – A
60. MOV C – B
61. MOV C – S
62. MOV C – V
63. MOV C – RAM
64. MOV C – RAML
65. MOV P – A
66. MOV P – B
67. MOV P – S
68. MOV P – V
69. MOV P – RAM
70. MOV P – RAML
71. SWP A – S
72. SWP A – V
73. SWP A – ZH
74. SWP A – ZL
75. SWP A – RAML
76. SWP S – V
77. SWP S – ZH
78. SWP S – ZL
79. SWP S – RAML
80. SWP V – ZH
81. SWP V – ZL
82. SWP V – RAML
83. SWP ZH – RAML
84. SWP ZL – RAML
85. OUT
86. OMD [0-8]
87. ZRO
88. ONE
89. MONE
90. ADD
91. INC
92. SUB
93. DEC
94. JMP
95. JLT
96. JEQ
97. JGT
98. CALL
99. CMP
100. CMPM
101. CMS
102. CMSM
103. ILT
104. IEQ
105. IGT
106. ICA
107. NOT
108. AND
109. OR
110. XOR
111. NEG
112. INP
113. PAU
114. SHR
115. SHL
116. ROR
117. ROL
118. TXI
119. TXD
120. HLT

Fetch Instruction:

* MHI PO
* MLI CO
* II CE RO

Access RAML:

* MLI CO
* MLI CE RO
* Read/Write RMSB

Access RAM any (using Z):

* ZHO MHI
* ZLO MLI
* Read/Write

MOV literal – Z:

* MLI CO
* ZHI CE RO
* MLI CO
* ZLI CE RO

Write to S:

* SM0 SM1 \_O

Swap:

* 1 into F
* 2 into 1
* F into 2

Add/Subtract:

* Add: AI EO
* Subtract: AI SUB EO

Zero/One:

* AI (OO)

Jump:

* [FE] into P (using FE signal)
* [FF] into C (using FE and OO signals)

Call:

* P ⬄ [FE]
* C ⬄ [FF]
* Swap using F

Return = Jump

Compare unsigned (CMP, CMPM = with subtract)

* CMP (SUB)

Compare signed (CMS, CMSM = with subtract)

* CMP CPTC (SUB)

Is =, >, <, carry:

* LT: AI CMPO
* EQ: AI CPB0 CMPO
* GT: AI CPB1 CMPO
* CA: AI CPB0 CPB1 CMPO

Bitwise:

* NOT: AI TO
* AND: AI T0 TO
* OR: AI T1 TO
* XOR: AI T1 T0 TO

Negate:

* BI AO
* AI
* AI SUB EO

Output:

* OI AO

Output mode:

* MLI CO
* OMI CE RO

Input:

* NO (stops clock)
* AI NO (transfers data)
* Setting NO stops the clock until input is given

Pause:

* NO (without input)

Shift:

* No roll-over
* Right: SM0
* Left: SM1
* Roll-over
* Right: SM0 SRO
* Left: SM1 SRO

Halt:

* HLT
* Halts the clock until complete CPU reset

Any opcode must be contained within one memory page.

Memory: 0xFE, 0xFF are reserved for jump target page and address

