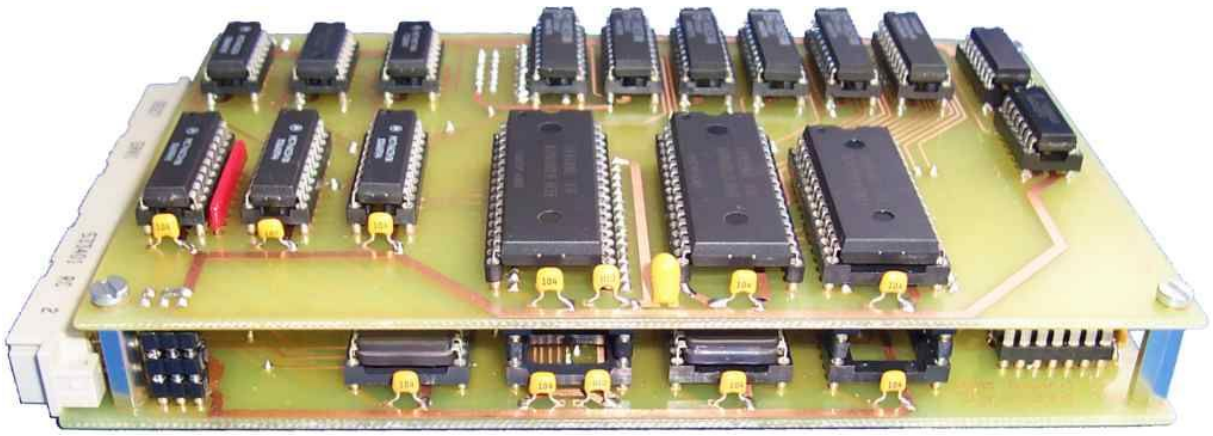
Memory Unit v2.1

For the MyCPU

- Selfbuild Guide -



uses `0000h-1FFFh` and `3800h-FFFFh`

D. Kuschel · Corintostraße 21 · 28279 Bremen · Germany · www.mycpu.eu · dennis_k@freenet.de 2015-07-13

Memory Unit Technical Details

➢ Memory Base Board

- 512kb universal RAM memory
- 8kb system memory (in a 25% of 62256 chip)
- 32kb ROM for the operating system kernel
- 128kb ROM for user application software
- the memory is paged (16kb and 32kb sized pages)
- high speed, up to 8MHz CPU frequency possible

➢ Memory Extension Board

- adds 512kb more universal RAM memory
- makes 100% of the 62256 chip accessible (adds 24kb more memory)
- adds support for multitasking operating systems through extended MMU
- adds support for loading other operating systems from external storage

Content

Figures

Tables

Author:

Dennis Kuschel
Corintostraße 21
28279 Bremen
Germany

web : http://www.mycpu.eu
email: dennis_k@freenet.de

1 Boards

1.1 Memory Base Board

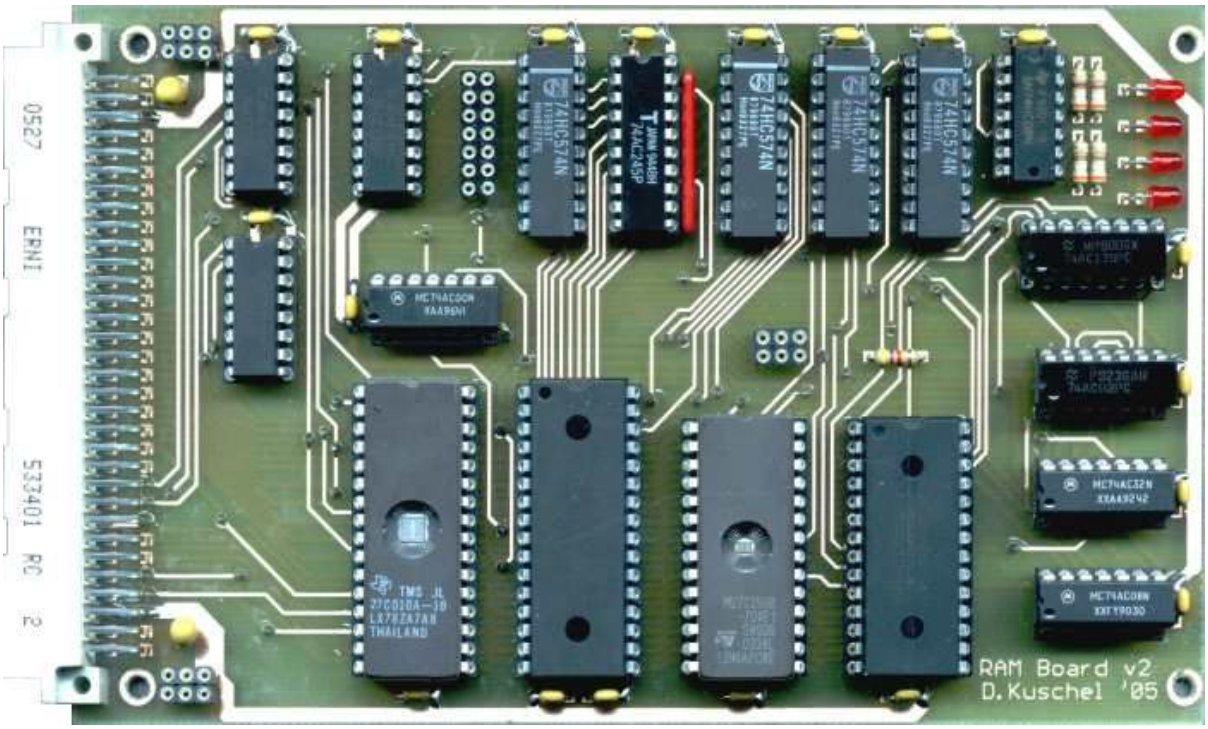


Fig. 1: Memory Base Board

1.1.1 Description

The memory base board contains the main memory of the processor system. The memory is divided into the RAM (520kb) that is used for temporary data storage, and the ROM (160kb) that contains the operating system and some useful application software. Furthermore the board contains some glue logic that does the memory page mapping. Memory paging is required because the MyCPU can only access 64kb of RAM and 64kb of ROM (128kb in total) directly. Please see Fig. 7 on Page 10 for an overview how the memory paging works.

1.1.2 Selection of Components

There are several 74ACxxx types listed in the partlist. The AC-type parts are required if you want to overclock the computer system and take it to the limit. Because the Advanced CMOS (AC) parts are hard to get and very expensive, you could also use the cheaper HC types. Dependent of the quality of the components, you can also reach clock rates up to 8MHz by using only the cheaper 74HCxxx types.

1.1.3 Placement of Components

After you have soldered all VIA's, you can continue with the integrated circuits. I suggest you not to use sockets for the IC's, except for the EPROM's and IC2. If you wish to use sockets for all IC's, you must use precision sockets. Only the high quality sockets allow you to solder the pads on the top side of the board. I strongly recommend you to follow the placement order I have noted in the placeplan below (see blue numbers, and start with the IC that has the blue number 1). When all IC's are placed and soldered, you can continue to place the capacitors. In the last round the resistors, diodes and board connectors are placed and soldered.

Note: It is required to use sockets for the IC's 2, 13 and 16!

ATTENTION!
Please be careful, and don't forget to solder a pad on the top side of the board. I have marked all critical pads with red paint in the placeplan below. Please check if you have really soldered these pads!

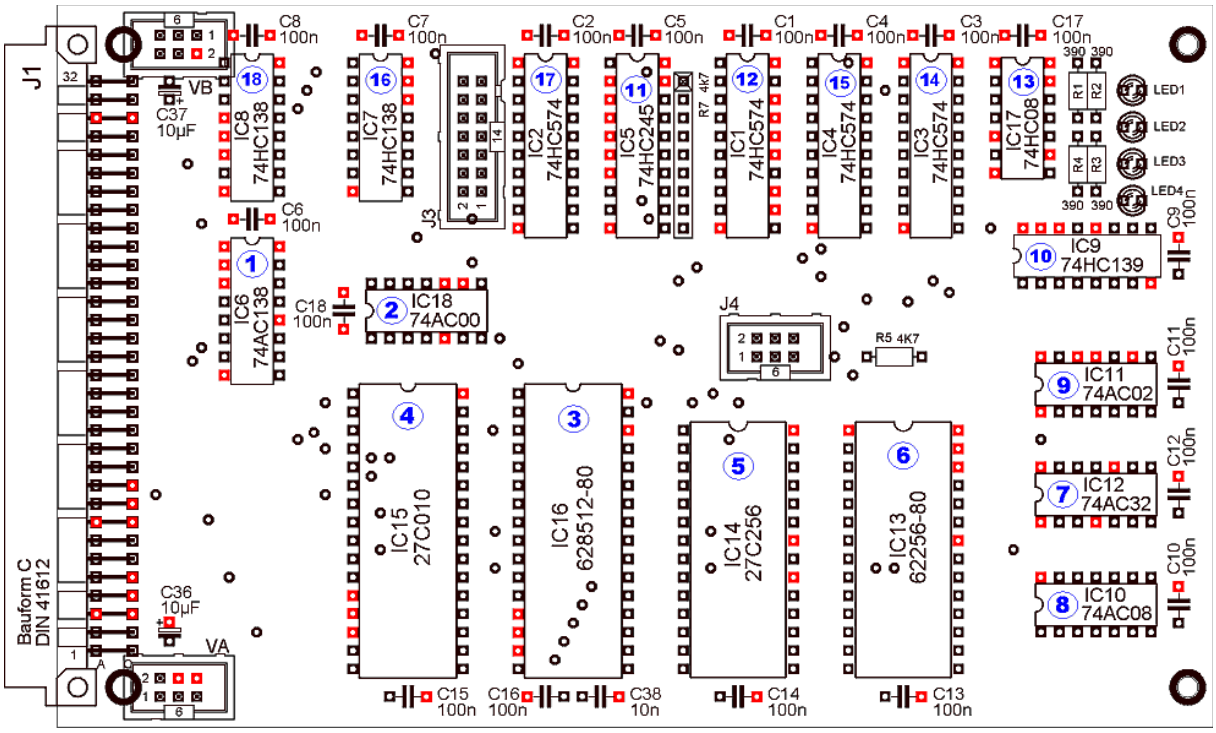


Fig. 2: Memory Base Board - Placeplan

1.1.4 Partlist

74AC00	IC18	
74AC02	IC11	
74AC08	IC10	
74HC08	IC17	
74AC32	IC12	
74AC138	IC6, IC7, IC8	
74HC139	IC9	
74HC245	IC5	
74HC574	IC1, IC2, IC3, IC4	
62256, 80ns	IC13	***80** ns are required for 8Mhz*
628512, 80ns	IC16	***80** ns are required for 8Mhz*
27C256, 70ns	IC14	***70** ns are required for 8Mhz*
27C010 **-or-** 27C1001, 70ns	IC15	***70** ns are required for 8Mhz*
LED red, 3mm	LED1, LED2, LED3, LED4	
390 Ohm	R1, R2, R3, R4	
4,7 kOhm	R5	
SIL 8 x 4.7 kOhm (8x 4k7)	R7	
10nF, ceramic capacitor	C38	
100nF, ceramic capacitor	C1 – C18	
10µF / 16V, tantal	C36, C37	
6 pin header	VA, VB, J4	
14 pin header	J3	
IC Socket	IC2, IC13, IC16	
DIN 41612 Connector	J1	

1.2 Memory Extension Board

*The Memory Extension Board is **not required**. It is **not a must**, but **just an option**.*

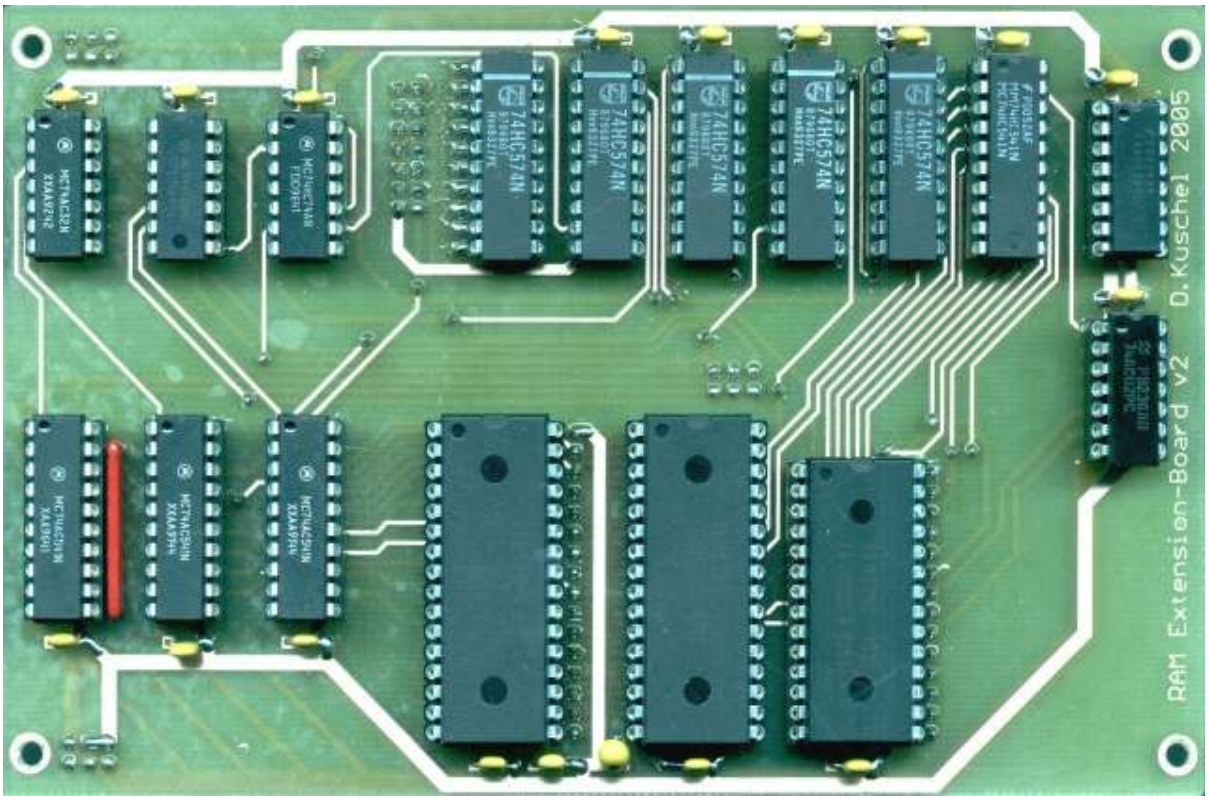


Fig. 2: Memory Extension Board (Prototype v2)

1.2.1 Description

The extension board adds 512kb more RAM to the system. Furthermore it provides support for multitasking operating systems and allows to load other operating systems from external data storage devices.

1.2.2 Selection of Components

The 74ACxxx types are required for the proper operation. I strongly recommend to use the AC types listed in the partlist. In fact it is possible to replace the 74ACxxx parts by their 74HCxxx counterparts, but this will result in dramatically decrease of reachable speed.

1.2.3 Placement of Components

After you have soldered all VIA's, you can continue with the integrated circuits. I suggest you not to use sockets for the IC's, except for the RAM's. If you wish to use sockets for all IC's, you must use precision sockets. Only the high quality sockets allow you to solder the pads on the top side of the board. I strongly recommend you to follow the placement order I have noted in the placeplan below (see blue and green numbers, and start with the IC that has the blue number 1). When all IC's (with blue numbers till 16) are placed and soldered, you can continue to place the capacitors and resistors.
After that, you can continue with placing the board connectors (green symbols and numbers). Be careful, this is a really difficult job! I strongly recommend you to place the parts in the order of the green numbers, else you may not be able to solder all pads.

ATTENTION!
Please be careful, and don't forget to solder a pad on the top side of the board. I have marked all critical pads with red paint in the placeplan below. Please check if you have really soldered these pads!

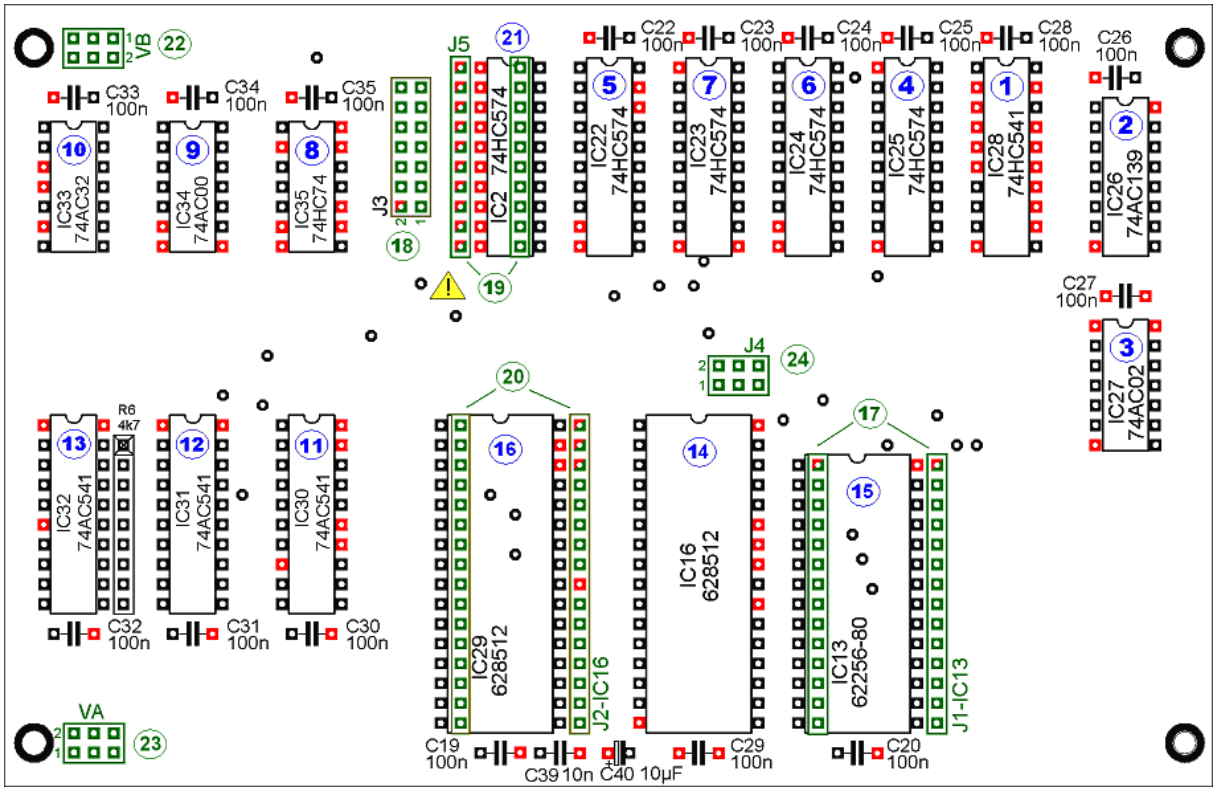


Fig. 3: Memory Extension Board v3 - Placeplan

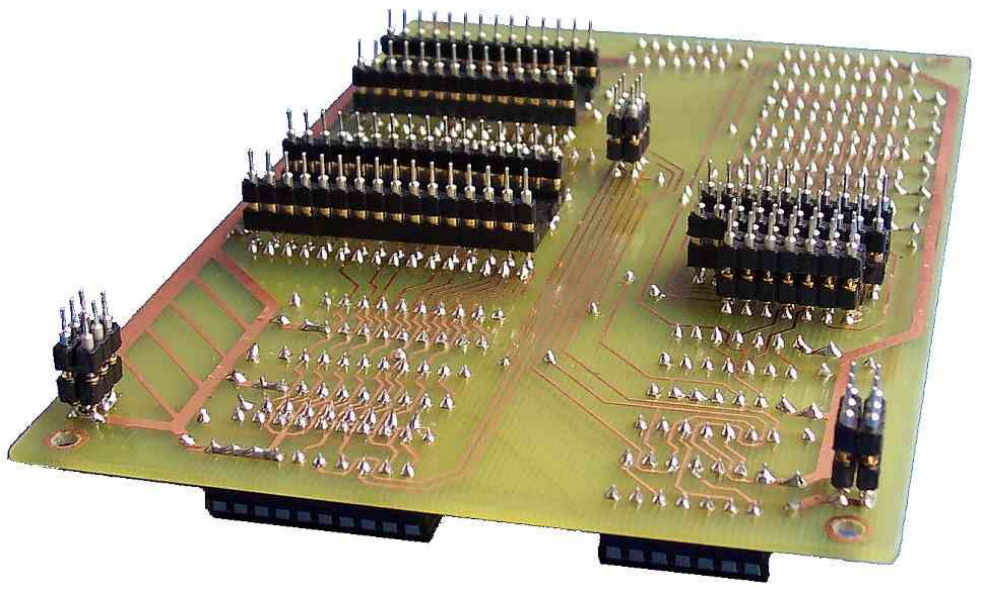


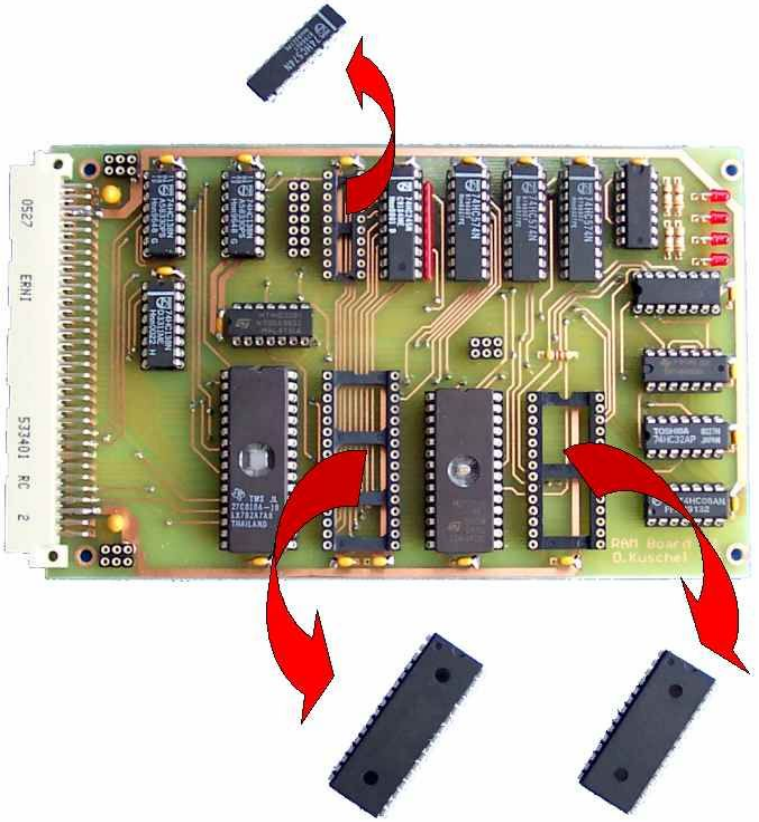
Fig. 4: View on bottom side of Extension Board

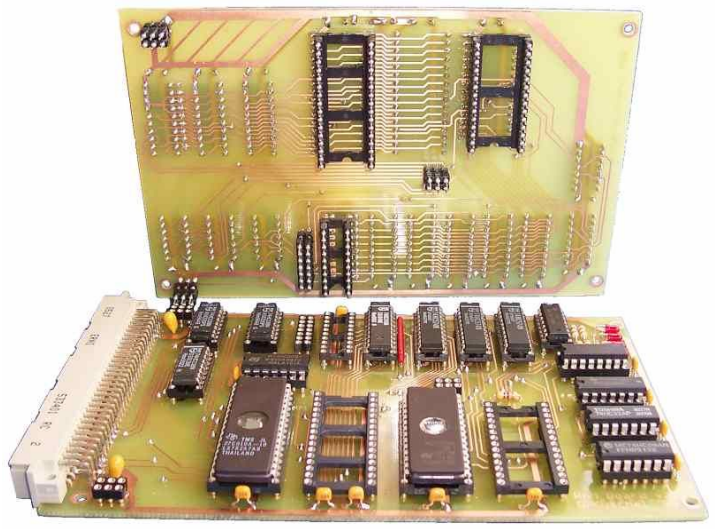
1.2.4 Partlist

74AC00	IC34
74AC02	IC27
74AC32	IC33
74HC74	IC35
74AC139	IC26
74AC541	IC30, IC31, IC32
74HC541	IC28
74HC574	IC2, IC22, IC23, IC24, IC25
62256, 80ns	IC13
628512, 80ns	IC16, IC29
SIL 8 x 4.7 kOhm (8x 4k7)	R6
10nF, ceramic capacitor	C39
100nF, ceramic capacitor	C22 – C35
10µF / 16V, tantal	C40
6 pin header	VA, VB, J4
14 pin header	J3
20 pin IC socket header	J5
28 pin IC socket header	J1
32 pin IC socket header	J2

2 Joining the Boards

Please remove the IC's 2, 13 and 16 from their sockets. Put this IC's into the appropriate sockets on the memory extension board.





Now put the extension board on top of the memory base board. You will need some force to press the boards together.

Fig. 5: Memory Base- and Extension-Board joined together

3 First Test

3.1 Test Setup

To test the memory base board, you have to connect the memory board with the CPU. No other peripherals must be connected to the bus!
Please apply the supply power of 5V to the bus connector pins A1 (GND) and A3 (+5V). If the LEDs show a moving light (a dot moving from right to left and back again) the first test is passed. Otherwise, please see the description of the post status code on the next page.

Hint: For verification, you can simulate this test setup with the MyCPU Emulator v4. The appropriated command line for the emulator is " `cpuemu4 -v1` ".

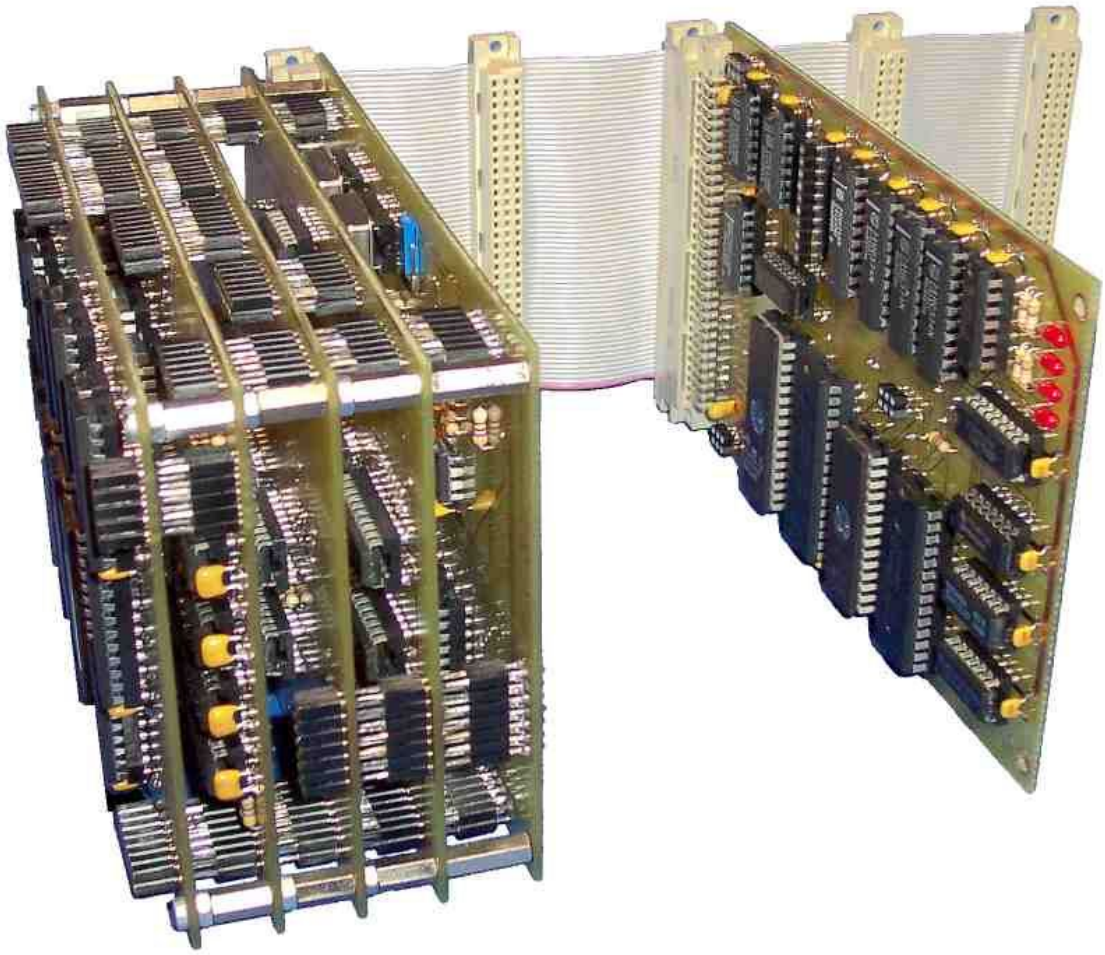
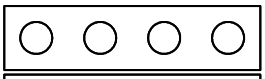
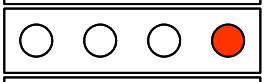
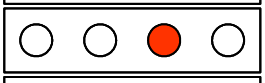
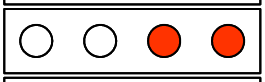
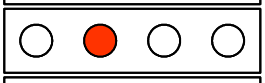
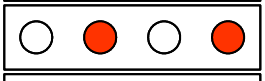
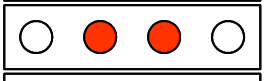
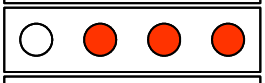
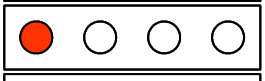
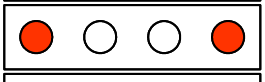
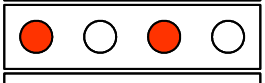
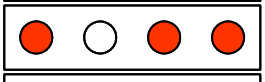
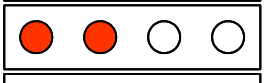
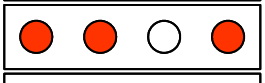
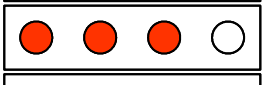
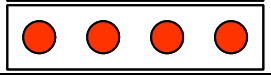


Fig. 6: Test Setup with MyCPU and Memory Base Board

3.2 Post Status

After power-on, the four LEDs D1 - D4 show the status of the self test. If the LEDs show a moving light (a dot moving from right to left and back again) the first test is passed successfully. Otherwise, the LEDs display a post code. Here is a list of possible post codes and their meanings:

POST-Code	LEDs LED4 – 3 – 2 – LED1	Error Description
0		No error. But if no LED did flash while initialization phase, there may be a big error in the CPU or bus.
1		ALU is defective: OR / AND does not work, or Z/C-flag wires are damaged
2		ALU: flag-adjustment of Z and V -flag does not work, or V-flag is defective
3		one byte ALU functions do not work, or Z-flag does not work properly
4		two byte ALU functions do not work or error in internal C-flag handling
5		X-register does not work properly
6		Y-register does not work properly
7		P-register does not work properly
8		stack-register does not work properly
9		basic system RAM failed the test, or address/data-wires are defective (RAM area $0000-$01FF, this is IC13)
10		stack does not work properly, error in stack-pointer handling
11		memory mapping register REG_RAMPAGE (IC1 / IC2) or REG_ROMPAGE (IC3 / IC4) failed
12		memory mapping register logic failed (search the error around IC1-IC5, IC9-IC12)
13		extension RAM failed, or address/data-wires are defective (RAM-pages $4000-$7FFF, IC16)
14		the memory extension board has register errors
15		the memory extension board has address logic errors

Tab. 1: POST Error Codes

IMPORTANT NOTE:
When more modules than only the CPU and Memory Unit are connected together (e.g. the Multi-I/O-Unit is also installed), the LEDs are no more showing the post code or the moving light! Instead the LEDs are showing a random or static flickering that indicates that the CPU is accessing the memory.

3.3 Test of the Extension Board

When you have already console access to the MyCPU operating system, you should enter the "`mem`" command to verify the presence of the memory extension board. Simply type "`mem`" at the kShell command prompt. You should get the following output (the amount of free memory may vary a bit):

```
8:/>mem

free program ram     : 31499 bytes
free paged data ram  : 983040 bytes
free user zeropage   : 32 bytes
free zero/stack pages: 92
```

When the memory extension board works correctly, the "`free paged data ram`" must be bigger than 512000 bytes, and the amount of "`free zero/stack pages`" must not be zero. Note: To get correct values, you need to disable all memory intensive software. When you have a RAM-drive installed, please remove the RAM-drive by typing "`ramdrive -r`".

When the value of "`free paged data ram`" is wrong, please check the integrated circuits IC29 - IC32. When the amount of "`free zero/stack pages`" is less than 64 or zero, please check the parts IC22 - IC28, IC33 - IC35.

If neither the value of "`free paged data ram`" nor the amount of "`free zero/stack pages`" is in the valid area, please check J3, J5, IC2 and IC33 - IC35.

There is one more feature of the memory extension board that needs to be tested: The ability to load an other operating system while the MyCPU is just running. To test this feature there exists a tool called "`copyroms`" in the selfbuild-guide zip-file. This tool copies the content of the EPROMs into the extension RAM and executes the OS from there. After entering "`copyroms`" at the command prompt, the CPU should reboot without any problems. When the CPU crashes, please check IC30 - IC35 or reduce the CPU core frequency.

4 Memory Map

4.1 MyCPU Memory Overview

The MyCPU can directly address 64kb code- and 64kb data-memory. Code-memory (mostly ROM) is addressed by asserting the /RD_ROM line, and data-memory (RAM) is addressed by asserting the /RD_RAM line. Overall the CPU can address 128kb of memory directly. But this memory unit provides up to 160kb ROM and 1056kb RAM, so there must be a method of mapping the huge amount of memory into the small address space of the CPU.

The memory unit uses the paging scheme to map small blocks of memory into the address space that is visible to the CPU.

The following figure shows an overview of the memory layout of the board:
(Note: RAM-Code-Pages are sometimes also called “ROM-pages”)

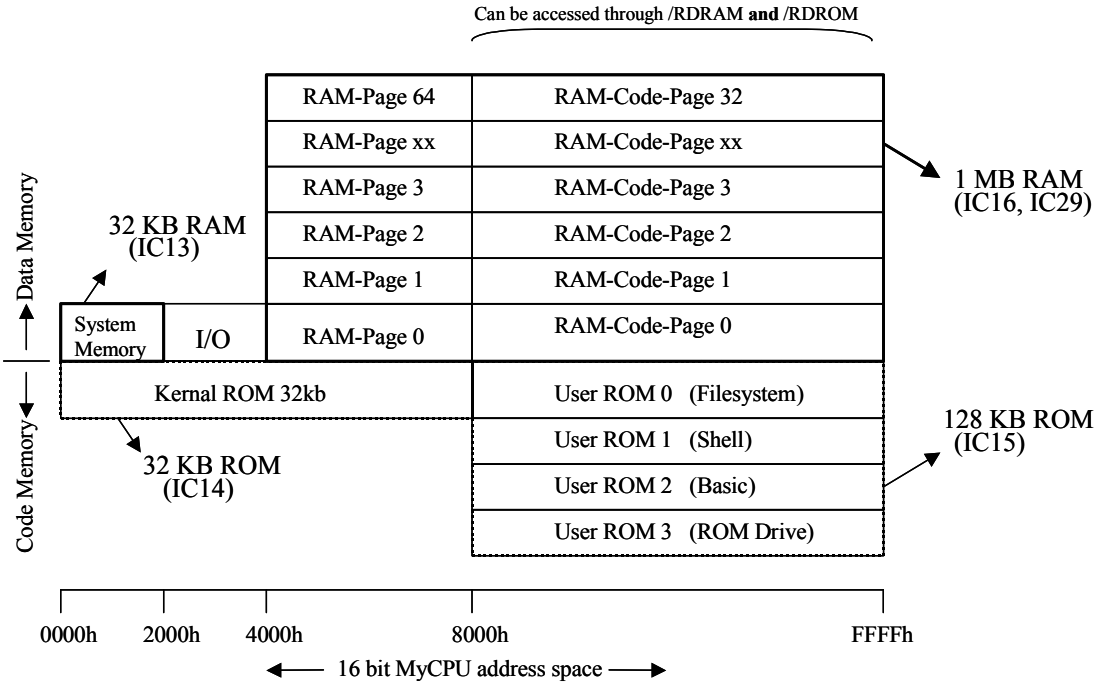


Fig. 7: Memory Map

The code memory area below 8000h is reserved for the kernel ROM (selected by /RDROM). The kernel ROM can not be switched off, it is present all the time. The data memory area below 8000h is splitted into three blocks: The lower 8kb contain the RAM that is required for stack memory, between 8kb and 16kb (2000h-3fffh) is the multi-purpose I/O-area, and from 4000h to 7FFFh is paged user RAM (max. 1Mb).

The address-space above 8000h is reserved for program memory. The program memory can either be ROM memory (the EPROM IC15, 128kb) or RAM memory (paged RAM from IC16 and IC19, the same memory that is shadowed at 4000h – 7FFFh). Note that there is no strict border between code and data memory above address 8000h. That means, the CPU is able to execute code from RAM, what is required to be able to load programs into and execute from RAM memory.

4.2 Internal Organization of the 1Mb Extension SRAM

The RAM memory (IC16 on the base board and IC29 on the extension board) is divided into several small pages. The pages are mapped between address 4000h and 7FFFh (RAM-pages, 16kb) and between address 8000h and FFFFh (RAM-Code-pages, 32kb) into the address space of the MyCPU. The mapping is made by using the page registers 3800h (REG_RAMPAGE) and 3900h (REG_ROMPAGE).

Note that RAM-pages and RAM-code-pages share the same space in memory. The figure below shows this. For example, selecting RAM page 62 or 63 uses the same memory in IC29 like the RAM code page 31 and the same memory like user ROM 3 in RAM. This makes the use of RAM very flexible.

IC16 on the base board stores RAM-pages 0 to 31 and RAM-code-pages 0 to 15 (=512 kb). IC29 on the extension board stores the RAM-pages 32 to 63 and RAM-code-pages 16 to 31 and optional the shadowed user EPROM pages (=512 kb in sum).

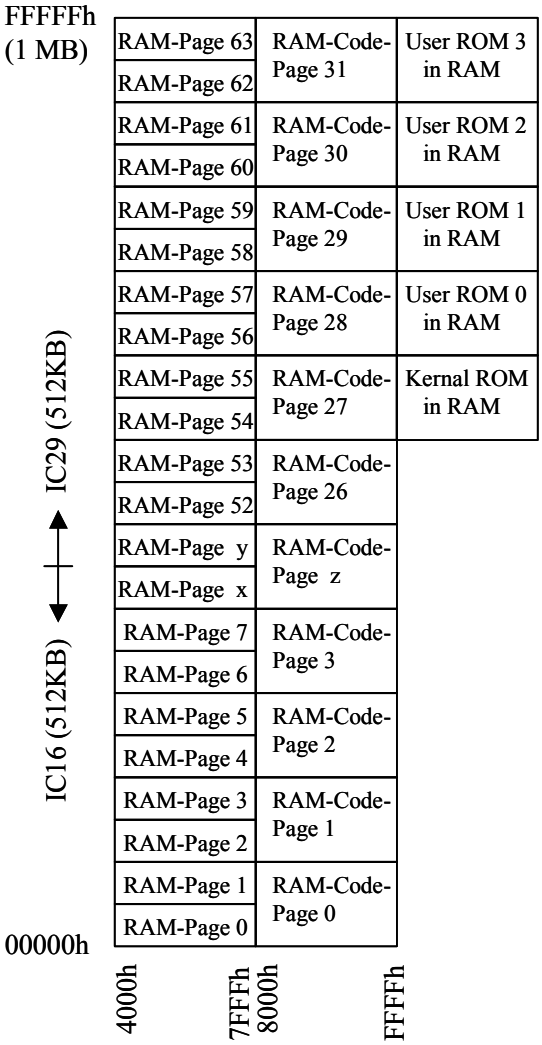


Fig. 8: Internal Organization of the main RAM chips

4.3 Internal Organization of the 32kb System SRAM

The system SRAM (IC13 on the base board) is used to store the call-stack, the zero-page and kernel variables. The MyCPU can directly access the lower 8kb of the SRAM.
The extension board adds a paging mechanism so that also the remaining rest of the 32kb SRAM can be used by the CPU. A page has a size of 256 bytes, and in sum 128 pages are available. A 256 bytes sized page can either be mapped into the address area 0000h – 00FFh or into the address area 0100h – 01FFh. This enables replacing the zero-page and stack-page by other pages, which is required for running realtime operating systems, for example.

The paging registers are at address 3A00h (REG_ZEROPAGE) and at address 3B00h (REG_STACKPAGE). This registers are only present if you have the memory extension board installed.

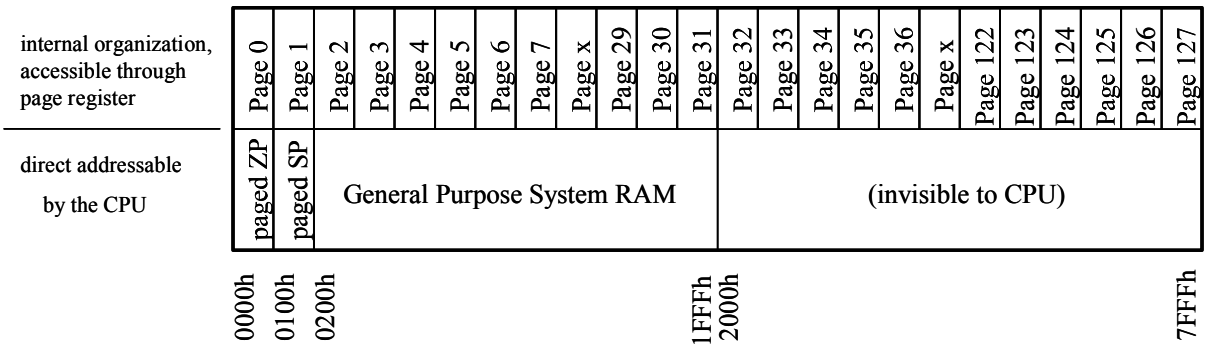


Fig. 9: Internal Organization of the 32kb System Memory Chip

4.4 I/O-Area

The memory unit drives three lines on the bus connector: /IOEN1, /IOEN2 and /IOEN3. This lines are asserted when the CPU accesses one of the three multi-purpose I/O-areas at the addresses 2000h-27FFh (I/O-Space1, line /IOEN1), 2800h-2FFFh (I/O-Space2, line /IOEN2) and 3000h-37FFh (I/O-Space3, line /IOEN3). The address area 3800h to 3FFFh is private to the memory unit and is used for internal registers.

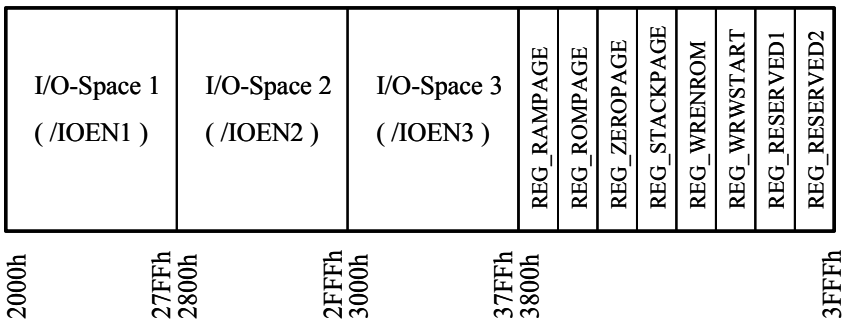


Fig. 10: I/O-Space Memory Mapping

5 Registers

5.1 Registerbank on Memory Unit (Overview)

Address	Register Name	Read/Write	Function
3800h	REG_RAMPAGE	R / W	Select RAM-Page (16kb data memory page)
3900h	REG_ROMPAGE	R / W	Select ROM-Page (32kb data and code page)
3A00h*	REG_ZEROPAGE	R / W	Select Zero-Page (0000-00FFh, 256 bytes)
3B00h*	REG_STACKPAGE	R / W	Select Stack-Page (0100-01FFh, 256 bytes)
3C00h*	REG_WRENROMBIT	W	1 Bit: switch between EPROM and RAM
3D00h*	REG_WRWSTARTBIT	W	1 Bit: warmstart flag (set to 0 after hw-reset)
3E00h			(reserved)
3F00h			(reserved)

* Only available with the RAM extension board.

Note: Page registers can easily be accessed with the MyCPU "RBK" and "SBK" OP-Codes.

5.2 Registerbank on Memory Base Board

5.2.1 REG_RAMPAGE

The RAM-page-register is used to select a 16 kbyte sized memory page that is mapped into the data memory address space from 4000h to 7FFFh. Please see also Fig. 7 on page 10 and Fig. 8 on page 11.

Address: 3800h (read/writeable)

MSB(D7)							LSB(D0)
		RAP_5	RAP_4	RAP_3	RAP_2	RAP_1	RAP_0

RAP_5 – RAP_0 RAM-Page number. One of 64 possible RAM-memory-pages can be selected, which is then mapped into the CPU address range from 4000h to 7FFFh. The CPU can access this memory by using the /RD_RAM signal.
Note: Without the memory extension board, RAP_5 is not available and must be set to zero.

RAP_3 – RAP_0 User LEDs. There are 4 user LEDs on the memory unit that are connected to this bits. If one of this bits is set to 1 and the CPU fetches code from the lower address area 0000h – 7FFFh, the corresponding LED is lighting.

5.2.2 REG_ROMPAGE

The ROM-page-register is a bit more special. Commonly said, it tells the CPU where to execute the program code from. But if you look closer, you can select two different memory planes that are available in parallel! How does this work? When you look to Fig. 7 on page 10, you will recognize that there is no strict separation between code- and data-memory above address 8000h. When a writeable memory page is mapped into this address area, the CPU can not only write data to the memory, but it can also execute program code from it! This allows an operating system to load and execute programs from a removable storage device. This so called "RAM-Code-Pages" can be selected through the bits ROP_0 – ROP_4 in this register.

Furthermore there is a 128kb sized EPROM on the memory unit. The EPROM is partitioned into four pages with 32kb each. This EPROM-pages can also be mapped into the address-area 8000h - FFFFh. Which of this four pages is mapped into the CPU visible space is selected through the bits URP_0 and URP_1.

The remaining bit, bit REN, is used to tell the CPU from which source (EPROM or RAM) it shall execute the program code. When REN is set to 0 the CPU executes code from RAM (selected through ROP_0 – ROP_4), and the EPROM is disabled in this configuration. But when REN is set to 1, the CPU executes code from EPROM (selected through URP_0 – URP_1), and when the CPU reads or writes data in the same address area, the data is read or written from/to the RAM-code-page that is selected by the ROP bits.

Address: 3900h (read/writeable)

MSB(D7)							LSB(D0)
URP_1	URP_0	ROP_4	ROP_3	ROP_2	ROP_1	ROP_0	REN

ROP_4 – ROP_0 RAM-code-page number. One of 32 possible RAM-memory-pages can be selected, which is then mapped into the CPU address range from 8000h to FFFFh. The CPU can access this memory by using both the /RD_RAM and the /RD_ROM signal. Code is only executed from RAM when REN is set to 0.
Note: Without the memory extension board, ROP_4 is not available and must be set to zero.

URP1 – URP_0 EPROM page number. 4 "User-" EPROM pages are available that can be mapped into the address area from 8000h to FFFFh. The EPROM is only visible to the CPU when REN is set to 1. The EPROM is IC15.

REN EPROM enable bit. Set this bit to 1 when the CPU shall execute code from the EPROM IC15. When this bit is set to 0, the CPU will attempt to execute code from the RAM-code-page that is selected by ROP_0 – ROP_4.

Notice the bit special RAM-code-page memory mapping (see Fig. 8 on page 10 for details). The RAM-code-page memory can also be mirrored to the data address range 4000h – 7FFFh. This is done by copying the content of register REG_ROMPAGE into the register REG_RAMPAGE. Note that bit D0 then chooses the lower or upper half of the 32kb RAM-code-page to be mirrored to 4000h-7FFFh.

5.3 Registerbank on Memory Extension Board

5.3.1 REG_ZEROPAGE

This register is only available with the memory extension board. This register is used to select one of 128 possible 256 byte sized blocks for the processors zeropage (0000h – 00FFh). Please see Fig. 9 on page 10 for details how the 32kb RAM IC13 is structured.

Address: 3A00h (read/writeable)

MSB(D7)							LSB(D0)
EEN	ZPP_6	ZPP_5	ZPP_4	ZPP_3	ZPP_2	ZPP_1	ZPP_0

ZPP_6 – ZPP_0 Zeropage-page number. One of 128 possible 256 byte sized pages can be selected, which is then mapped into the CPU data address range from 0000h to 00FFh. The CPU usually uses this memory area for its "Zeropage". After reset, the CPU should initialize this register with 00h. Because the CPU can address the lower 8kb of IC13 directly, only the page numbers 32 to 127 should be written to this register. Pay attention that this page number does not interfere with the stackpage page number.

EEN Read only bit. If this bit is 1, the EPROMs are enabled and the CPU executes the program code from this ROMs (this is the case after a hardware reset). When this bit is 0, the EPROMs are switched off (they are invisible) and the CPU executes the program from a RAM shadow in IC29. Please see Fig. 8 for details. Note: To update this bit so it reflects the actual status, it is required to do a write-cycle to this register. See also register REG_WRENROMBIT.

5.3.2 REG_STACKPAGE

This register is only available with the memory extension board. This register is used to select one of 128 possible 256 byte sized blocks for the processors stackpage (0100h – 01FFh). Please see Fig. 9 on page 10 for details how the 32kb RAM IC13 is structured.

Address: 3B00h (read/writeable)

MSB(D7)							LSB(D0)
WST	SPP_6	SPP_5	SPP_4	SPP_3	SPP_2	SPP_1	SPP_0

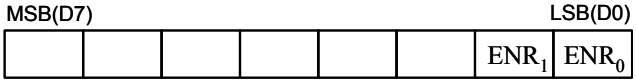
SPP_6 – SPP_0 Stackpage-page number. One of 128 possible 256 byte sized pages can be selected, which is then mapped into the CPU data address range from 0100h to 01FFh. The CPU usually uses this memory area for its call-stack. After reset, the CPU should initialize this register with 01h. Because the CPU can address the lower 8kb of IC13 directly, only the page numbers 32 to 127 should be written to this register. Pay attention that this page number does not interfere with the zeropage page number.

WST Read only bit. This bit is automatically reset to zero with a hardware reset. It can be set to 1 by writing the value 01h to the register REG_WRWSTARTBIT. This bit can be used to detect a processor warmstart. Note: To update this bit so it reflects the actual status, it is required to do a write-cycle to this register.

5.3.3 REG_WRENROMBIT

This register is only available with the memory extension board. It is used to switch off the EPROMs IC14 and IC15. When the EPROMs are disabled, a ROM-shadow in RAM-code-pages 27 through 31 is enabled. This mechanism allows "overloading" the EPROMs at runtime, so other operating systems can be loaded and executed "from ROM". For example this can be very useful for debugging purpose.

Address: 3C00h (write only)

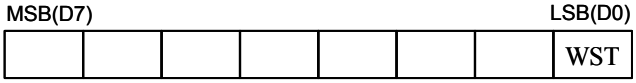


`ENR1 – ENR0` To disable the EPROMs and enable the RAM shadow, it is required to write the value 03h to this register. If the EPROMs are enabled or not can be tested by reading back the bit `EEN` in register REG_ZEROPAGE. You can enable the EPROMs again by writing the value 00h to this register.

5.3.4 REG_WRWSTARTBIT

This register is only available with the memory extension board. The memory extension board supports a mechanism to detect if the CPU has done a cold- or a warmstart. The "warmstart"-bit is automatically reset to zero with a hardware reset. The bit can manually be set to 1 by writing to this register.

Address: 3D00h (write only)



`WST` This is the "warmstart" detect bit. It can be set or reset by writing 01h or 00h to this register. If you want to read back the state of this bit you must use the register REG_STACKPAGE.

6 Schematics

On the following two pages you will find the schematics of the Memory Base Board and the Memory Extension Board.

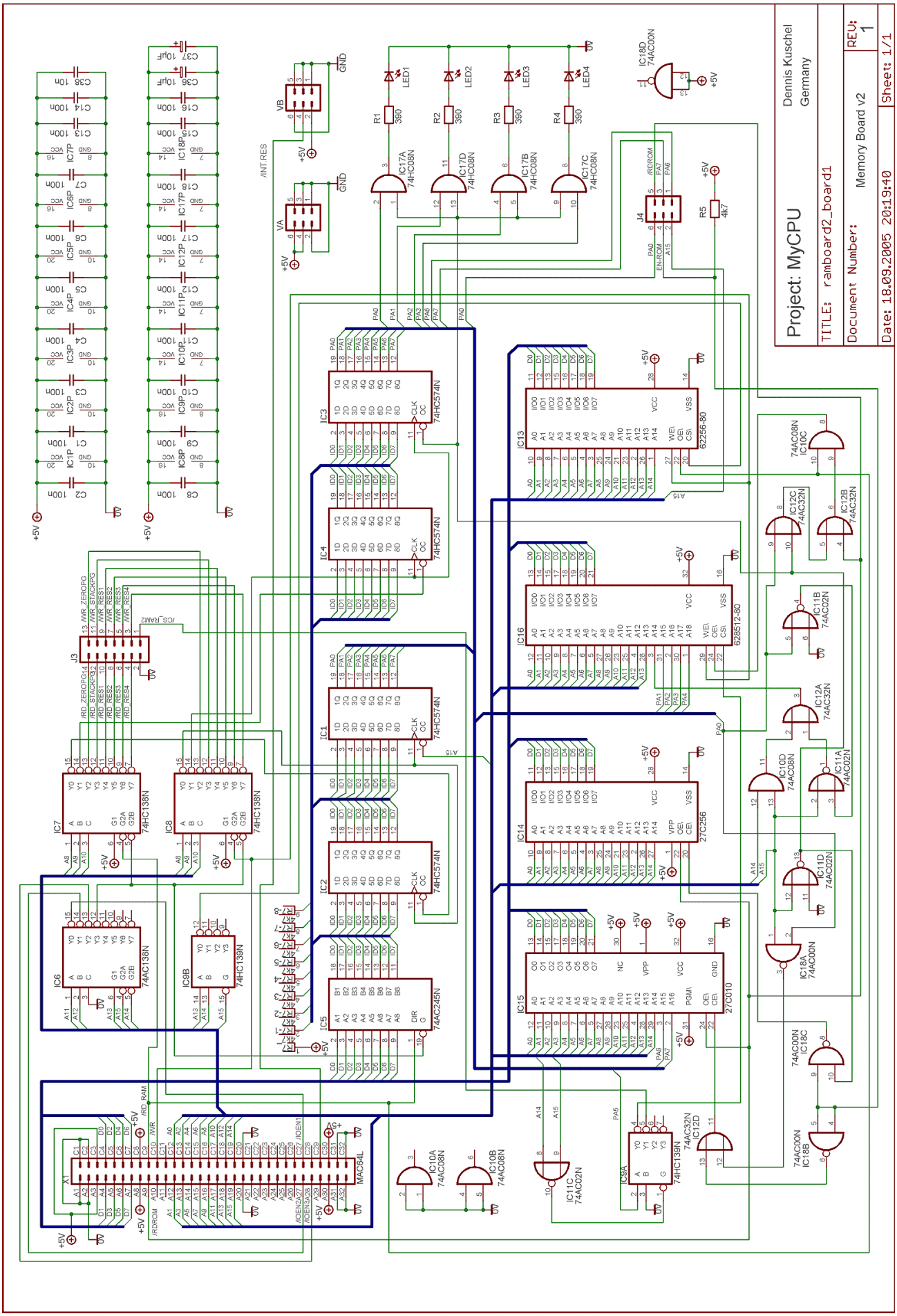


Fig. 11: Schematic of Memory Base Board

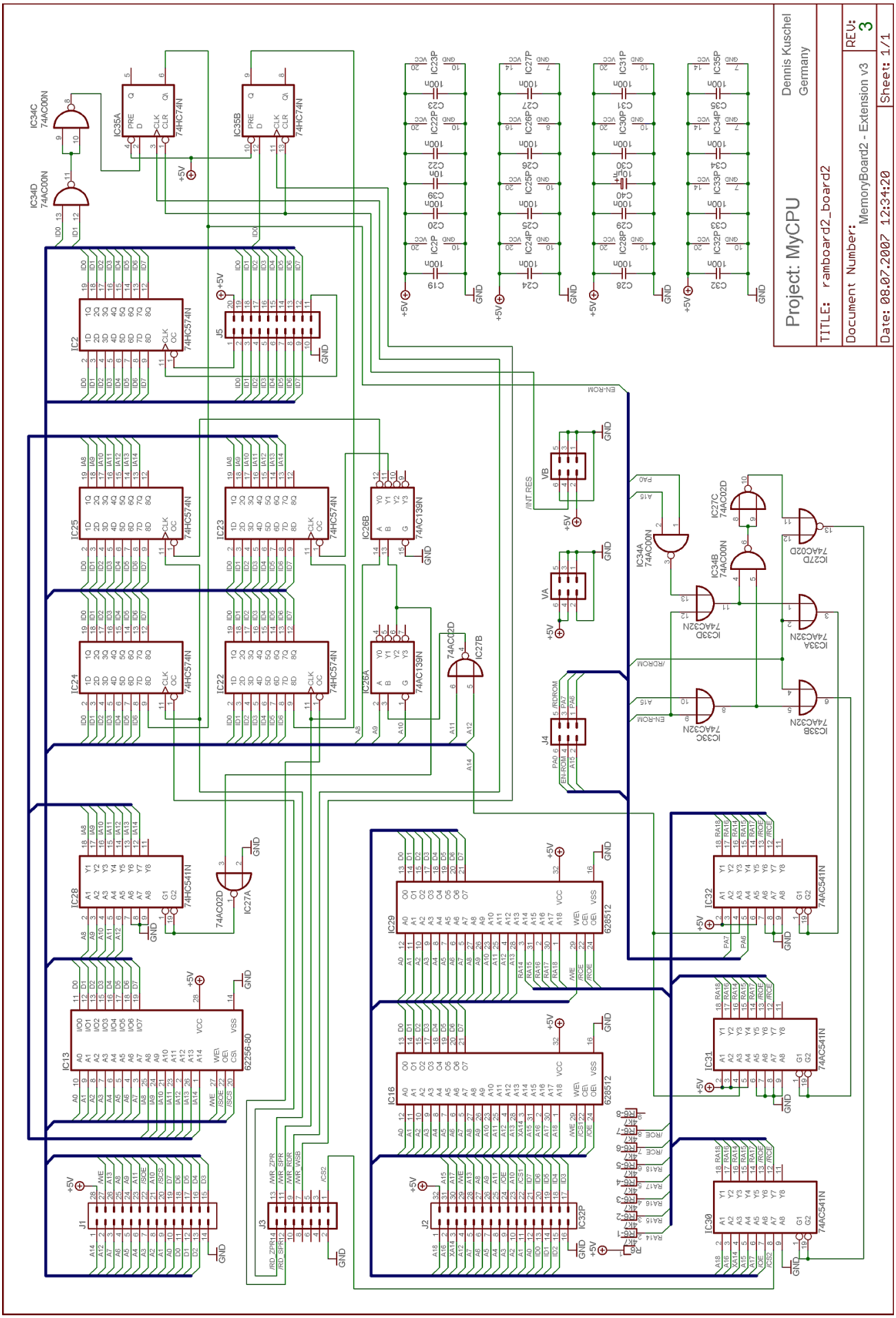


Fig. 12: Schematic of Memory Extension Board

7 Change Log

7.1 Changes in the Memory Unit design

1899-12-30	Name	Chapter	Description
2007-07-08	D.Kuschel		Bug on Memory Extension Board fixed. Schematic and pictures in chapter 1.2 updated.
2008-12-22	D.Kuschel	all	Document converted to OpenOffice format
2015-07-13	D.Kuschel	3.2	Added a note about the LEDs