Subject: New board: V20-MBC, an easy to build V20 SBC (and a prototype on breadboard)
Posted by just4fun on Sun, 19 Jan 2020 09:21:41 GMT
View Forum Message <> Reply to Message

I've just finished the first draft of the schematic for an experimental test based on a V20 CPU.
The objective is to make an easy to build board V20 based, using a MCU to virtualize all the I/O chips as I've done with the Z80-MBC2.
The V20 is the CMOS full static version uPD70108H, so the minimum clock can be DC.

Currently I'm assembling the circuit on a breadboard, using as MCU a STM32F030R8 with a custom board (ARMando) that I've done to play with old CPU like this one (but the final board will probably use an Atmega32 MCU).
ARMando can be directly inserted into 830 points breadboards (supported from 1 to 4 breadboards with direct insertion):

Video using three breadboards here: https://www.youtube.com/watch?v=v_FUc8-PeMM

I'll post updates here...

Subject: Re: Starting V20 CPU on breadboard.
Posted by gbm on Mon, 20 Jan 2020 11:55:41 GMT
View Forum Message <> Reply to Message

Please look up my topic on SDC_One.
 https://www.retrobrewcomputers.org/forum/index.php?t=msg&amp ;goto=4206&#msg_4206
I have a 8086/8088/V20/V30 PCB for SDC_One but no time to write the firmware. If you are interested in putting the V20 on top of a Nucleo64, I may send you an empty board with schematic, so you could avoid all the wiring troubles.

Subject: Re: Starting V20 CPU on breadboard.

Posted by just4fun on Mon, 20 Jan 2020 19:16:51 GMT
View Forum Message <> Reply to Message

gbm wrote on Mon, 20 January 2020 12:55Please look up my topic on SDC_One...
Hi gmb,
I've seen your topic and it is very interesting.

What I'd like to do is a little board that runs at "normal" speed (i.e 4/8MHz) with 1MByte RAM (all the address space) where the "companion" MCU do all the I/O stuff.
The breadboard is only for the first phase to check the boot sequence and the I/O "handshaking" between the V20 and the MCU (and others details), so using a breadboard (that is almost wired now) is needed to do "experiments".

In the final design I've the intention to use an Atmega32 as MCU (because I want use only THP parts in the final board), so the STM32 is only for the first step.

Anyway thanks you for your kind offer.

Fabio

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Mon, 03 Feb 2020 18:37:17 GMT
View Forum Message <> Reply to Message

Starting to do first experiments (small scale) and seeing interesting things...

It seems that the V20H (uPD70108H) CPU needs 8 clock pulses from the RESET signal de-asserted up to fetch the first instruction (T1 of the first instruction fetch, when the ASTB = ALE latch signal does a 1 -> 0 transition):

V20 on breadboard - A090120
IOS - I/O Subsystem (STM32F030R8) - S010220

 * RESET Done *
i = 0 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 1 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 2 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 3 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 4 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 5 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 6 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 7 HLDAK = 0
ASTB = 0 IOM_ = 1 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 1
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 1

i = 8 HLDAK = 0
ASTB = 1 IOM_ = 0 RD_ = 1 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 0
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 0

i = 9 HLDAK = 0
ASTB = 0 IOM_ = 0 RD_ = 0 WR_ = 1
A4 = 1 A3 = 0 A2 = 0 A1 = 0 A0 = 0
D7 = 1 D6 = 1 D5 = 1 D4 = 1 D3 = 0 D2 = 0 D1 = 0 D0 = 0

i = 9 is the T1 bus state of the first fetched instruction.

This is not reported in any document I've read. May be it executes a couple of NOP before first real instruction is fetched to allow an early HLDRQ to be served before first instruction (as stated in the datasheet).

An other interesting thing is the effect of the "hold" circuit in every 3-stated pin. If you leave the bus disconnected, reset and start sending clock pulses, after fetching the address FFFFF4 the CPU goes in HALT state because it reads F4 that is the HALT instruction as side effect of the holding circuitry on the multiplexed AD0-AD7 data bus...

File Attachments

1) 20200203_185726b.jpg, downloaded 2769 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Wed, 05 Feb 2020 09:01:33 GMT
View Forum Message <> Reply to Message

UPDATE: The 8 clock pulses needed after RESET goes LOW are described in the original 8088 datasheet at page 11 (see attachment):

... The low-going transition of RESET triggers an internal reset sequence for approximately 7 clock cycles.
After this interval the 8088 operates normally, beginning with the instruction in absolute locations FFFF0H

File Attachments

1) i8088 - i8088-2 datasheet.pdf, downloaded 465 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Wed, 19 Feb 2020 20:11:18 GMT
View Forum Message <> Reply to Message

I've done some HW changes and added an "I/O" led:

Now it is possible boot directly from a binary file on a microSD end execute it with the V20 CPU running at 8MHz.
Here the classic "Hello world" test:

I'm using now the NASM assembler:

```
;
; Test HelloWord - HW ref: A090120 (V20 on breadboard)
```

```
;
; REQUIRED: IOS S010220 DEVEL_16 (STM32F030R8)
;
; Assemble with "nasm -f bin  filename.asm -o filename.bin"
;
    [map mapFile.txt]   ; Set output map file
    CPU     8086        ; Set 8086/8088 opcodes only
    BITS    16          ; Set default 16 bit

eos         equ     0x00    ; End of string

; IOS equates
EXC_WR_OPCD equ    0x00      ; Address of the EXECUTE WRITE OPCODE write port
EXC_RD_OPCD equ    0x00      ; Address of the EXECUTE READ OPCODE read port
STO_OPCD    equ    0x01      ; Address of the STORE OPCODE write port
SERIAL_RX   equ    0x01      ; Address of the SERIAL RX read port
SYSFLAGS    equ    0x02      ; Address of the SYSFLAGS read port
SERTX_OPC   equ    0x01      ; SERIAL TX opcode

    org  0
    ;
    ; Init
    ;
    mov     ax, cs          ; DS = SS = CS
    mov     ds, ax
    mov     ss, ax
    mov     sp, stack       ; Set the stack
    ;
    ; Print a message and halt
    ;
    mov     bx, msg
    call    puts
halt:
    hlt
    jmp     halt

;
================================================================================
= ;
;
; Send a string to the serial line, BX contains the pointer to the string.
; NOTE: Only AL and BX are used
;
;
================================================================================
= ;
puts:
    mov     al, [bx]        ; AL = current char to print
```

```
    cmp     al, eos         ; End of string reached?
    jz      puts_end        ; Yes, jump
    mov     al, SERTX_OPC   ; AL = SERIAL TX opcode
    out     STO_OPCD, al    ; Write the opcode
    mov     al, [bx]        ; AL = current char to print
    out     EXC_WR_OPCD, al ; Print AL
    inc     bx              ; Increment character pointer
    jmp     puts            ; Transmit next character

puts_end:
    ret


msg     db  'Hello, World!', 0x0d, 0x0a, eos    ; 0x00 is the message terminator
        times   0x100   db  0;
stack:
```

Now more tests but it's probably time to start thinking at the final PCB version... :)

File Attachments

1) 20200219_200954.jpg, downloaded 2584 times
2) Screenshot_20200219_192837.png, downloaded 2578 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sat, 22 Feb 2020 11:59:50 GMT
View Forum Message <> Reply to Message

Little step ahead...
I've "manually ported" the iLoad loader from the Z80 assembler to 8086/8088, so now I've a decent toolchain to assemble, upload (with the serial port) and execute with one command (using TeraTerm):

The limit of iLoad is that only a "flat file" can be used (so all the code and data inside a single 64KB segment), but it is more than enough for now.

This little system starts to be usable...

File Attachments

1) Screenshot_20200222_110851.png, downloaded 2591 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Thu, 05 Mar 2020 18:11:12 GMT

View Forum Message <> Reply to Message

Just to test the 8080 mode of this CPU, I've "ported" the Altair Basic (8KB version).
I've done a new version of iLoad, called iLoad-80, to load a Intel-Hex 8080 executable.
After the load phase iLoad-80 switches the V20 CPU into 8080 mode and jumps to the 8080 code.

Here a short video: https://www.youtube.com/watch?v=yM8VKNeYMu8&feature=yout u.be

File Attachments
1) Altair_Basic.JPG, downloaded 2482 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 06 Mar 2020 13:16:15 GMT
View Forum Message <> Reply to Message

...and IMSAI 8KB Basic too... :)

File Attachments
1) IMSAI_Basic.JPG, downloaded 2462 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sat, 07 Mar 2020 11:11:58 GMT
View Forum Message <> Reply to Message

I must to admit that I'm having a lot of fun with this V20 on breadboard (distracting me to design the "final" board...).

So I couldn't resist to try the MON85 monitor from here.

I've done a change to the bin -> ASCII routine because it seems not to work on the V20 CPU in 8080 mode.
Here the change:

```
;
; Display A in hexidecimal
;
HPR:    PUSH    PSW     ; Save low digit
    RRC         ; Shift
    RRC         ; high
```

```
   RRC        ; digit
   RRC        ; into low
   CALL    HOUT        ; Display a single digit
   POP PSW     ; Restore low digit
HOUT:   ANI 0Fh     ; Remove high digit
   ;CPI 10      ; Convert to ASCII
   ;SBI 2Fh
   ;DAA
 daa
 cpi 10
 cmc
 aci 48
   JMP OUT     ; And output it
```

It is a great monitor!!

File Attachments

1) Mon85_0.JPG, downloaded 2510 times
2) Mon85_1.JPG, downloaded 2422 times
3) Mon85_2.JPG, downloaded 2453 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Wed, 11 Mar 2020 17:32:14 GMT
View Forum Message <> Reply to Message

I've just finished the first draft of the "final" board schematic (see the the draft pdf).
Here I've used an Atmega32 as MCU (so only THP parts).

Here the main specs:

- V20HL CPU(uPD70108HCZ)
- RAM can be configured with 128/512/1024KB;
- optional RTC and microSD modules (the same used in the Z80-MBC2);
- optional 16x GPIO port;
- I2C expansion port;
- serial port;
- ISP connector (for the Atmega32);
- clock can be configured at 4/8MHz.

It should work with a 80C88 too (of course loosing the 8080 mode).

BTW: In the 80C88 datasheet there is an uncommon bypass cap between two GND pin:

If anyone has more info about this please tell me (in the uPD70108H datasheet there is no mention of this cap).

File Attachments

1) V20-MBC - PREVIEW.pdf, downloaded 510 times
2) 80c88_cap.jpg, downloaded 2384 times
3) 80C88.pdf, downloaded 1101 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Thu, 12 Mar 2020 18:19:50 GMT
View Forum Message <> Reply to Message

Hello,
I have not worked with a v20 in a few years...
I went back to some old to some old paper schematics and there is only a 0.1µF capacitor between pins 20 and 40.
I think that the "A 0.1µF capacitor between pins 1 and 20 is recommended for decoupling." is a typo....

FYI
As I still have a few v20 and all of the other chips in your schematic, I would LOVE to play with this...

THX
CM
cm@cm.ca

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 13 Mar 2020 11:43:51 GMT
View Forum Message <> Reply to Message

Hi,
I don't think that is a typo...

If you see Fig. 3 in the datasheet it is clearly reported in the schematic:

Take in consideration that * only * the CMOS version 80C88 has this cap.

If you have got some V20 chips, you can use here only the full static version V20HL = uPD70108H (the "H" makes the differences...)

"Normal" versions will not work (very likely) here...

I've attached here the uPD70108H (aka V20HL) datasheet.

File Attachments
1) 80c88_cap2.jpg, downloaded 2358 times
2) uPD70108H_NEC (V20H).pdf, downloaded 450 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Fri, 13 Mar 2020 17:18:37 GMT
View Forum Message <> Reply to Message

Hello,
I checked the 8086-8088 parts box...
QTY 1 uPD70108D-8
QTY 5 uPD70108C-10
QTY 5 uPD70108HCZ-16

QTY 1 P8088 (INTEL)
QTY 3 D8088 (INTEL CERAMIC)
QTY 3 D8088 (AMD CERAMIC)
QTY 8 uPD8088D-2
QTY 37 MBL8088-2

QTY 1 D8086-2 (INTEL CERAMIC)
QTY 2 D8086-2 (AMD CERAMIC)

QTY 5 uPD71084C-10
QTY 2 MD8284A-1

Do you think the uPD70108HCZ-16 will work ?

THX
CM

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 13 Mar 2020 17:58:23 GMT
View Forum Message <> Reply to Message

cmacarthur wrote on Fri, 13 March 2020 18:18Hello,
Do you think the uPD70108HCZ-16 will work ?

For sure! 8)

See the photo in the first post...

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Sat, 14 Mar 2020 04:53:24 GMT
View Forum Message <> Reply to Message

Hello,
I know you are looking at using a "Atmega32 as MCU (because I want use only THP parts in the final board)"
BUT have you considered using something like a "Blue Pill" STM32F103C8T6 board that fits in a socket on your v20 PCB.

It would give you all of the performance of the STM32 you are currently using and a small footprint on the final PCB...
Just a thought...
CM

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sat, 14 Mar 2020 14:19:45 GMT
View Forum Message <> Reply to Message

Hmmm... I think that an Atmega32 is enough... it must do very similar tasks as in the Z80-MBC2.
A STM32F030R8 is an overkill, and a STM32F103C8T6 even more, and IMHO a Blue Pill footprint is not less than a DIP40 after all...

BTW: just started the PCB design... I'll try to make it compatible with the uTerm and uCom boards...

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sat, 14 Mar 2020 17:14:23 GMT
View Forum Message <> Reply to Message

Here a first attempt to do a placement without routing (only ratnest), just to have raw idea...

File Attachments
1) A250220.jpg, downloaded 2270 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Sat, 14 Mar 2020 20:47:55 GMT
View Forum Message <> Reply to Message

Looks GREAT...

CM

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Wed, 18 Mar 2020 15:10:45 GMT
View Forum Message <> Reply to Message

The PCB routing is complete. Now some refinements and checks:

File Attachments
1) A250220_B.jpg, downloaded 2261 times
2) A250220_B2.jpg, downloaded 2218 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Wed, 18 Mar 2020 18:45:56 GMT
View Forum Message <> Reply to Message

Looks like you are having "WAY to much FUN"!
and the Board looks GREAT!
CM

Subject: Re: Starting V20 CPU on breadboard.
Posted by ab0tj on Wed, 18 Mar 2020 19:06:37 GMT
View Forum Message <> Reply to Message

If you have extra boards when you get some fabbed, I'd love to acquire one from you.

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 20 Mar 2020 12:22:34 GMT
View Forum Message <> Reply to Message

ab0tj wrote on Wed, 18 March 2020 20:06If you have extra boards when you get some fabbed, I'd love to acquire one from you.

Hi, if you live outside EU I think that it's probably more convenient for you order a small batch directly.

More, as my country is currently "locked down" and this situation will continue for a while, it isn't a good idea to go at a post office (if you can find one open) these days... :(

Here the final aspect of the board:

I'll publish all the details as usual after some testing (hoping to be able to get the packet).

File Attachments
1) A250220_B2.jpg, downloaded 2167 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by rvumbaca on Sat, 21 Mar 2020 01:33:12 GMT
View Forum Message <> Reply to Message

Hi just4fun,

Just a warning that the chip markings on your V20 CPUs in your first post are consistent with re-marked chips.

That is, they are probably not really 16MHz parts, but re-marked parts that were originally rated at a lower speed (such as 8Mhz or 10Mhz). I can tell you more about this if you are interested.

This is a common issue with many retro and legacy parts. If you bought these parts from EBay and they came from Asia (for example China), then there is a high likelihood of this situation.

Therefore I just wanted to warn you that if you ever clock your chips at a high speed and you run into unusual behaviour, you do not have to blame your design! :)

Best regards and good luck with your project!

Subject: Re: Starting V20 CPU on breadboard.
Posted by ab0tj on Sat, 21 Mar 2020 03:48:05 GMT

View Forum Message <> Reply to Message

just4fun wrote on Fri, 20 March 2020 06:22Hi, if you live outside EU I think that it's probably more convenient for you order a small batch directly.

More, as my country is currently "locked down" and this situation will continue for a while, it isn't a good idea to go at a post office (if you can find one open) these days... :(
Fair enough. I can order some and then there will be extras for those on "this side of the pond"

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Sat, 21 Mar 2020 21:17:25 GMT
View Forum Message <> Reply to Message

Hello,
Are the Gerbers available, I am about to release this months jlcpcb order and would like to add this board.
THX
CM

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sun, 22 Mar 2020 10:24:42 GMT
View Forum Message <> Reply to Message

@rvumbaca:

Thanks very much for the warning! Yes, I've bought the CPU on ebay (as usual)...
The final board will use a 4/8 MHz clock, so accordingly with the V20HL datasheet the lowest speed grade is 10MHz and it is fine.

Up to now I've used no more than 8MHz as clock, anyway the breadboard prototype can set higher clock using a STM32F030 running at 48MHz, so I'll do some test just to see.
If you can give more info about the labeling I'm curious to know...

@cmacarthur:

I haven't tested the board yet, so I can't say if it works or not. However if you want try to produce it "at your own risk" I've attached the gerber for JLCPCB and the corresponding schematic.

File Attachments
1) A250220 - PRELIMINARY UNTESTED Gerber.zip, downloaded 384 times
2) A250220 - SCH_Preliminary.pdf, downloaded 459 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by cmacarthur on Sun, 22 Mar 2020 15:06:22 GMT
View Forum Message <> Reply to Message

Thank-You
I will add it to the monthly jlcpcb order.
CM

Subject: Re: Starting V20 CPU on breadboard.
Posted by rvumbaca on Sun, 22 Mar 2020 22:22:31 GMT
View Forum Message <> Reply to Message

Hi just4fun,

just4fun wrote on Sun, 22 March 2020 03:24@rvumbaca:

If you can give more info about the labeling I'm curious to know...

Long back story: In recent years there has probably been some renewed demand for legacy ICs such as CPUs and EPROMs, which are no longer manufactured. The IC counterfeiters of the world took the opportunity to profit from this. I can only assume that people often search for higher speed rated parts, as this seems to be a common reason for many re-marked Motorola, Zilog, Intel and other CPUs. Another common item is that slower speed EPROMs are re-marked as higher speed parts (and also the brand is changed). So if you search for "ST 27C400", you will find many, many re-marked parts (my programmer can read the ID code, often I have found that these are re-marked Macronix or National parts).

Most of the supply of these parts comes from a few warehouses in Asia. 99.9% of the sellers on EBay do not hold stock, they simply act as a seller and when you order a part they either buy it from these warehouses (or a distributor) or (more commonly) they arrange for drop-shipping from the warehouse/distributor direct to you. Often I found that ICs I bought from China or HK, are posted to me directly from the Philippines, Brunei, and other countries in Asia. When I have complained about the parts being counterfeit, the seller will often perform an immediate refund with no questions asked. The sellers have little knowledge of what they are selling, they often offer thousands of different items (sometimes not just ICs). Sometimes the picture looks good, but the part you receive is not the marked the same way.

The re-marking of chips exists at different "quality levels". The simplest method, is to paint over and use ink. If you rub these ICs with IPA or Acetone the ink often can be removed.

-> Try to rub the top of your IC with acetone or IPA and check if the marking is removed.

"Higher quality" re-marked chips usually have been ground on top, then a "black top" is applied. They then use laser engravers to re-mark the chip. Usually the markings have mistakes in them such as:
- Colour of marking does not match a part from that time
- Invalid date codes (for example a MC68010 with date code of 2019!)

- Many, many parts have the same batch code
- The marking does not match the original part

To see a great example of what "black top" and laser engraving looks like on an MC68000, see this photo from a colleague: https://twitter.com/SushiL0gic/status/1116219450708021248/photo/1

Recently I made a special effort to buy a real NEC V20. I was tired of receiving counterfeit parts, so I bought it from a seller in the USA. This is the appearance of my part:

Here is an image of your NEC V20 with signs of re-marking highlighted:

The counterfeiters have improved their process, but they still lack attention to detail :)

File Attachments
1) Re-marked.png, downloaded 2063 times
2) NEC-V20.png, downloaded 2124 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Tue, 24 Mar 2020 08:39:42 GMT
View Forum Message <> Reply to Message

Thanks you. Very detailed!

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sun, 12 Apr 2020 14:19:47 GMT
View Forum Message <> Reply to Message

Starting some testing:

File Attachments
1) V20-MBC_1.PNG, downloaded 1912 times
2) V20-MBC_2.JPG, downloaded 2036 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Mon, 13 Apr 2020 10:46:53 GMT
View Forum Message <> Reply to Message

I've created a page on the V20-MBC here:

https://hackaday.io/project/170924

BTW: If someone want to test a V20-MBC board (HW ref. A250220) I've attached an IOS beta release (zip file) and the ILOAD.BIN executable to put into the microSD.

File Attachments
1) S260320_DEVEL-2RB.zip, downloaded 364 times
2) iload.bin, downloaded 364 times
3) Screenshot_20200413_124358.png, downloaded 1993 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by ab0tj on Tue, 14 Apr 2020 01:55:00 GMT
View Forum Message <> Reply to Message

Thanks for sharing your design. I placed an order with JLCPCB today and threw in the Gerbers for this board. That said I'll have a few extras available for the price of shipping if anyone stateside wants one as long as just4fun doesn't object :)

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 17 Apr 2020 15:50:25 GMT
View Forum Message <> Reply to Message

Some little improvements...

Here the new boot/system menu:

iLoad: loads and executes a 8088 Intel-Hex executable from the serial;
iLoad-80: loads and executes a 8080 Intel-Hex executable from the serial;
Autoboot: loads and executes an 8088 binary file (AUTOBOOT.BIN) on SD;
Autoboot-80: loads and executes an 8080 binary file (AU80BOOT.BIN) on SD.

iLoad (option 1) is now embedded into the FW, so it is the only possible boot choice if there is no SD installed.

Options 5, 6 and 8 are useless for now...

Here the load of the AU80BOOT.BIN 8080 binary file from SD. In this case a dual-stage boot is needed.
In the first stage is loaded the 8080 executable, in the second one the Switch-80 utility that takes care to activate the 8080 mode and executes the 8080 code (I've used the Altair Basic executable as 8080 binary):

Here a fully assembled board:

File Attachments
1) V20-MBC_3.JPG, downloaded 1883 times
2) V20-MBC_4.JPG, downloaded 1889 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Fri, 17 Apr 2020 16:15:23 GMT
View Forum Message <> Reply to Message

Are any changes to the PC board pending, or is it "safe" to go ahead and order using the A250220 - PRELIMINARY UNTESTED Gerber.zip file?

Great looking project.

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 17 Apr 2020 16:40:26 GMT
View Forum Message <> Reply to Message

Thanks!

Up to now there isn't any pending PCB change, so it should be enough safe... (fingers crossed...)

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Mon, 20 Apr 2020 18:25:42 GMT
View Forum Message <> Reply to Message

I've tested an 80C88 CPU too and seems to work:

...and if you try to run an 8080 executable this is what you get: :lol:

File Attachments
1) Screenshot_20200420_201628.png, downloaded 1890 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Mon, 20 Apr 2020 19:58:28 GMT
View Forum Message <> Reply to Message

Any reason not to use 74HCT series? I already have those here...

Subject: Re: Starting V20 CPU on breadboard.
Posted by mikemac on Mon, 20 Apr 2020 22:05:44 GMT
View Forum Message <> Reply to Message

Dumb question, what are those two boards sticking straight up in the upper left corner? There used to be only one but a second one just showed up. Just curious as none of my 8088s are the CMOS version. :(

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Tue, 21 Apr 2020 06:31:13 GMT
View Forum Message <> Reply to Message

- optional RTC and microSD modules (the same used in the Z80-MBC2);

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Tue, 21 Apr 2020 15:55:13 GMT
View Forum Message <> Reply to Message

I've just uploaded on the V20-MBC page (https://hackaday.io/project/170924#menu-files) all the details to build it, including the Assembly guide, BOM, etc.

Also I've released the IOS and the content of the related SD.

There isn't any OS for now, but you can use a couple of Basic interpreters and the Fig-Forth v1.3 (8080 mode).

The option 4 loads the IMSAI Basic as default configuration.

The option 3 loads a HelloWord demo as default configuration.

@gkaufman:
yes, you can use HCT. I've used some too.

@mikemac:
as gkaufman already said...

Subject: Re: Starting V20 CPU on breadboard.
Posted by Yves-D. on Wed, 22 Apr 2020 08:14:31 GMT
View Forum Message <> Reply to Message

@just4fun
What would be the prerequisites of running MS-DOS and DOS-text-based appliations on the V20-MBC?
There you could tap into a large software repository.
I guess, DOS relies heavily on (non-existent) BIOS-Calls for user and I/O interaction.
A complex editor like the one in TurboPascal 7.0 perhaps also relies on a (non-existent) graphics-card, if it would bypass the BIOS-Calls for speed and switching to 80x43 character mode for example.

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Thu, 23 Apr 2020 07:28:56 GMT
View Forum Message <> Reply to Message

@Yves-D.

Using a custom BIOS probably it should be possible to run MSDOS with restrictions due to the lack of a video card, so only "text programs" could run here (using only BIOS calls).
This means that the custom BIOS should "remap" the text video I/O on the serial port (using VT100-like cursor control).
I haven't analyzed that in detail yet...

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sat, 02 May 2020 11:26:37 GMT
View Forum Message <> Reply to Message

The current IOS (S263020) has a bug preventing a running program to read date/time as expected.

To fix overwrite line 2060 of S263020.ino:

if ((ioOpcode != 0x0A) && (ioOpcode != 0x0C)) ioOpcode = 0xFF; // All done for the single byte opcodes.

with the following:

if ((ioOpcode != 0x84) && (ioOpcode != 0x86)) ioOpcode = 0xFF; // All done for the single byte opcodes.

It will be fixed in the next revision.

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Sun, 17 May 2020 23:29:57 GMT
View Forum Message <> Reply to Message

I've been having trouble locating D70108HCZ processors that weren't Chinese remarked parts.

Decided to try some V20's I had lying around including D70108D-8 and D70107C-8.

Surprisingly both booted right up at 8mhz and seem to be working fine, at least with some minimal testing.

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Mon, 18 May 2020 08:36:30 GMT
View Forum Message <> Reply to Message

gkaufman wrote on Mon, 18 May 2020 01:29Decided to try some V20's I had lying around including D70108D-8 and D70107C-8.
Surprisingly both booted right up at 8mhz and seem to be working fine, at least with some minimal testing.

Interesting. I've some D70108C too...

BTW: starting to play with CP/M 2.2 using the 8080 mode:

File Attachments
1) V20-MBC_cpm22_1.JPG, downloaded 1651 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Wed, 20 May 2020 17:22:23 GMT
View Forum Message <> Reply to Message

The new IOS with CP/M 2.2 (8080 mode) support is out here:

File Attachments
1) Screenshot_20200520_190345.png, downloaded 1575 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Sat, 23 May 2020 15:53:37 GMT
View Forum Message <> Reply to Message

CP/M came up nicely, thanks again for your efforts!

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Thu, 04 Jun 2020 18:10:12 GMT
View Forum Message <> Reply to Message

CP/M-86 is out!

Update here both IOS and the SD image.

File Attachments
1) Screenshot_20200604_193724.png, downloaded 1484 times
2) V20-MBC_11.JPG, downloaded 1547 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by Yves-D. on Thu, 04 Jun 2020 19:32:19 GMT
View Forum Message <> Reply to Message

Yay, great. Thanks for all the work you did there. Lovely!

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Fri, 05 Jun 2020 15:31:40 GMT
View Forum Message <> Reply to Message

CP/M-86 came up easily.

Reminds me of the old CompuPro 8085/8088 S-100 board.

Thanks!

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Wed, 10 Jun 2020 15:46:03 GMT
View Forum Message <> Reply to Message

I've just released a new SD image that adds in the drive F: of the CP/M-86 Disk Set a CP/M-80 emulator under CP/M-86 using the V20 8080 mode (VCPM15.CMD).

In this way it is also possible run Catchum or send/receive a file with XMODEM:

As a curiosity I've also added a pure SW emulator (8080.CMD)

For more info read the README.TXT file in the F: drive.

File Attachments
1) V20-MBC_14JPG.JPG, downloaded 1397 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by bigpete on Mon, 03 Aug 2020 14:29:43 GMT
View Forum Message <> Reply to Message

I ordered my PCBs for this today. This is going to be my first build and am looking forward to getting it up and running. Now just to source the V20...

Subject: Re: Starting V20 CPU on breadboard.
Posted by dittman on Sun, 23 Aug 2020 05:08:57 GMT
View Forum Message <> Reply to Message

Does anyone have a pre-built BOM for Mouser or Digi-Key for this board?

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Mon, 24 Aug 2020 09:56:24 GMT
View Forum Message <> Reply to Message

Just destroyed the "V20 on breadboard"...

to start a new toy...

but this one will be another story...

File Attachments
1) 20200824_105053_R.jpg, downloaded 1163 times
2) 20200824_112514_R.jpg, downloaded 1191 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by guus.assmann on Thu, 27 Aug 2020 18:25:55 GMT
View Forum Message <> Reply to Message

Hello,

What files should I program into the Mega32A?
I have a Hi-Lo ALL programmer that can program the Mega.
How should I set the fuses, what file goes into the Eeprom and what goes in the flash?

Thanks for a very nice design.
I'm working on getting ist to the same bus as the SBC-85.

BR/
Guus Assmann

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Fri, 28 Aug 2020 07:24:04 GMT
View Forum Message <> Reply to Message

@guus.assmann:

Hi, all the related files are here: https://hackaday.io/project/170924/files

You have to flash the .hex file into the Atmega32. Nothing more. The fuse bits are written in the description of the .hex file (High Byte 0xD6, Low Byte 0xAF, Lock Byte 0xCF).

Regards.

Subject: Starting V20 CPU on breadboard.
Posted by guus.assmann on Thu, 10 Sep 2020 07:52:12 GMT
View Forum Message <> Reply to Message

Hello,

For the "sister project" with the Z80, it's possible to go to 20Mhz.
Can this be done for the V20 as well?

And I've been bussy for a bit.
You may know of the SBC-85 project, that has a backpanel.
Both the Z80 and V20 have been routed by me, to fit this bus.
The PCB's have not been tested yet. And some checking of the schematics would be nice.
I'll try to publish these schematics this evening.

BR/
Guus

File Attachments
1) V20_Back_design.pdf, downloaded 333 times
2) V20_Front_design.pdf, downloaded 348 times
3) Z80_Back_design.pdf, downloaded 342 times
4) Z80_Front_design.pdf, downloaded 346 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Thu, 10 Sep 2020 10:04:24 GMT
View Forum Message <> Reply to Message

Hi, the clock is provided by the Atmega32, so 8MHz is the max.

Please note that the Z80-MBC and V20-MBC design is not intended for the use on a external bus. This could require a deep re-design. How deep depends on the reference bus logic and electrical requirements.

Subject: Starting V20 CPU on breadboard.
Posted by guus.assmann on Thu, 10 Sep 2020 13:21:09 GMT
View Forum Message <> Reply to Message

Hello,

Thanks for the comment.

Maybe I should be a bit more clear. :blush:
For the Z80, it may be possible to overclock the Mega32. As this is out of spec, some luck is needed.
And I wonder if the same can be done for the uPD70108 board.

To me, both designs look very similar in the Mega32A part. (And the uPD70108 I have is rated for 16Mhz and the memory is faster than 55nS)

As for the PCB's I've routed, basically what I've done is get very many signals to the bus.
I assumed that it should be possible to connect some I/O on the bus.
For that, I'm thinking of a 82C55 and other "simple" I/O.
Maybe some memory is possible as well. Of course the addresses should not "clash".
The bus monitor may also work for the major part. (I often like blinking lights and numbers. :lol:)

BR/
Guus

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Thu, 10 Sep 2020 16:05:59 GMT
View Forum Message <> Reply to Message

Ah... yes, you can overclock the Atmega32 @ 20MHz and this should work without any particular problem.

I've done it with the Z80-MBC2 (read here) and could also be done with the V20-MBC (but I haven't tested in this case).

About the external bus, you need to make changes in the design to allow external I/O to access the bus without data clash.
It is not possible just to bring the signals to an external connector... (for both the boards)

E.g.: on the Z80-MBC2 the Atmega32 just "senses" the A0 line simply ignoring all the others address lines. So in your case you must change this logic.
You have to re-check and analyse the whole I/O addressing logic of both the boards to find what to change in your case. Also the logic that puts the CPU in wait must be changed.

About using external memory, all the memory addressing space is already fitted.

Good luck!

Subject: Starting V20 CPU on breadboard.
Posted by guus.assmann on Thu, 10 Sep 2020 17:41:25 GMT
View Forum Message <> Reply to Message

Thanks for the info.
And just in case, I'll upload the schematics.
So this setup cannot do too much, other than having a peek on the busses.
The monitor card of the SB-85 may (partially) work, as a display device anyway.
I'm not sure if I'll be able to change the logic / program to do some more usefull things.
It will be quite a challenge.....

BR/
Guus

File Attachments
1) Z80_A40618_sch.oxps, downloaded 324 times
2) V20_SBC.oxps, downloaded 335 times

Subject: Re: Starting V20 CPU on breadboard.
Posted by ab0tj on Tue, 15 Sep 2020 22:51:34 GMT
View Forum Message <> Reply to Message

Has anyone gotten MSDOS booted on one of these yet?

Subject: Re: Starting V20 CPU on breadboard.
Posted by bigpete on Tue, 06 Oct 2020 15:30:17 GMT
View Forum Message <> Reply to Message

So I finally got my PCBs after they were stranded on the US West Coast for a bit due to the fires. I have ordered most of my components but am having difficulty locating the Samsung SRAM. Will I have to resort to Ebay or Alibaba for these or is anyone aware of a supplier in the US that might have some?

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Tue, 06 Oct 2020 18:11:50 GMT
View Forum Message <> Reply to Message

Last ones ordered here.

Subject: Re: Starting V20 CPU on breadboard.
Posted by gkaufman on Wed, 07 Oct 2020 01:20:40 GMT
View Forum Message <> Reply to Message

AS6C4008-55PCN work fine, about $5 from Mouser or Digikey

Subject: Re: Starting V20 CPU on breadboard.
Posted by bigpete on Wed, 07 Oct 2020 03:03:25 GMT
View Forum Message <> Reply to Message

Awesome!

Thank you for the responses, all the components are now on the way.

I am very much looking forward to building this and learning some things in the process.

Subject: Re: Starting V20 CPU on breadboard.
Posted by guus.assmann on Sat, 10 Oct 2020 13:30:24 GMT
View Forum Message <> Reply to Message

Hello,

Using a Mighty-core board, I've been able to compile and upload a program to the Mega32.
But I can't get the 20Mhz bootloader programmed. (Windows10 somehow prevents this :( )
Could you please send me a hex-file with this 20Mhz bootloader or point me to where I can find it?
I can programm that with my ALL007 programmer.
Alternatively, a hex-dump of the Z80-MBC2 would also do fine. (20Mhz version) This does contain the bootloader, so would help as well.
Thanks for the support.
BR/
Guus

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sat, 10 Oct 2020 17:28:27 GMT
View Forum Message <> Reply to Message

@guus.assmann:

Not clear to me for which board do you need the 20MHz .hex FW... V20-MBC or Z80-MBC2?
:roll:

Subject: Re: Starting V20 CPU on breadboard.
Posted by guus.assmann on Sat, 10 Oct 2020 18:34:27 GMT
View Forum Message <> Reply to Message

Hello,
Actually, I have both boards and want to get them to run at 10Mhz with the 20Mhz bootloader.
What I cannot get done is program a bootloader into the Mega32.
But I can copy the Mega32.
If I have either, 1) bootloader for 20Mhz, or 2) Z80-MBC2 or 3) V20-MBC, I should be able to generate the other ones.
My programmer can accept the Intel-Hex files and programm that into the Mega32.

On the Might-Core development board, there's a Mega32 with 16Mhz booloader. And I sucessfully made a copy of this processor and uploaded the Z80-MBC2 into that, after compiling it. But I cannot get another bootloader into the Mega32. The programmer (USB-ASP) is recognised, but I keep getting errors.

BR/
Guus

Subject: Re: Starting V20 CPU on breadboard.
Posted by just4fun on Sun, 11 Oct 2020 11:01:10 GMT
View Forum Message <> Reply to Message

Here both the IOS (Z80-MBC2 and V20-MBC) at 20MHz.

They should be last revisions...

File Attachments
1) IOS_Z80_V20_20MHz.zip, downloaded 248 times

Subject: Starting V20 CPU on breadboard.
Posted by guus.assmann on Sun, 11 Oct 2020 14:02:08 GMT
View Forum Message <> Reply to Message

Thanks, I'll inform of the results.
BR/
Guus

Subject: Re: Starting V20 CPU on breadboard.
Posted by guus.assmann on Thu, 15 Oct 2020 20:46:22 GMT
View Forum Message <> Reply to Message

Hello,
Programming the files into the Mega32 didn't work out.

I finally found out why I couldn't write the bootloader. The driver for Windows10 was corrupt.
After reinstalling, I could program the bootloader and assemble both Z80 and V20 code.
So now they both work at 10Mhz clock. (Mega32 at 20Mhz of course.)
Thanks for the help. :)

BR/
Guus Assmann
