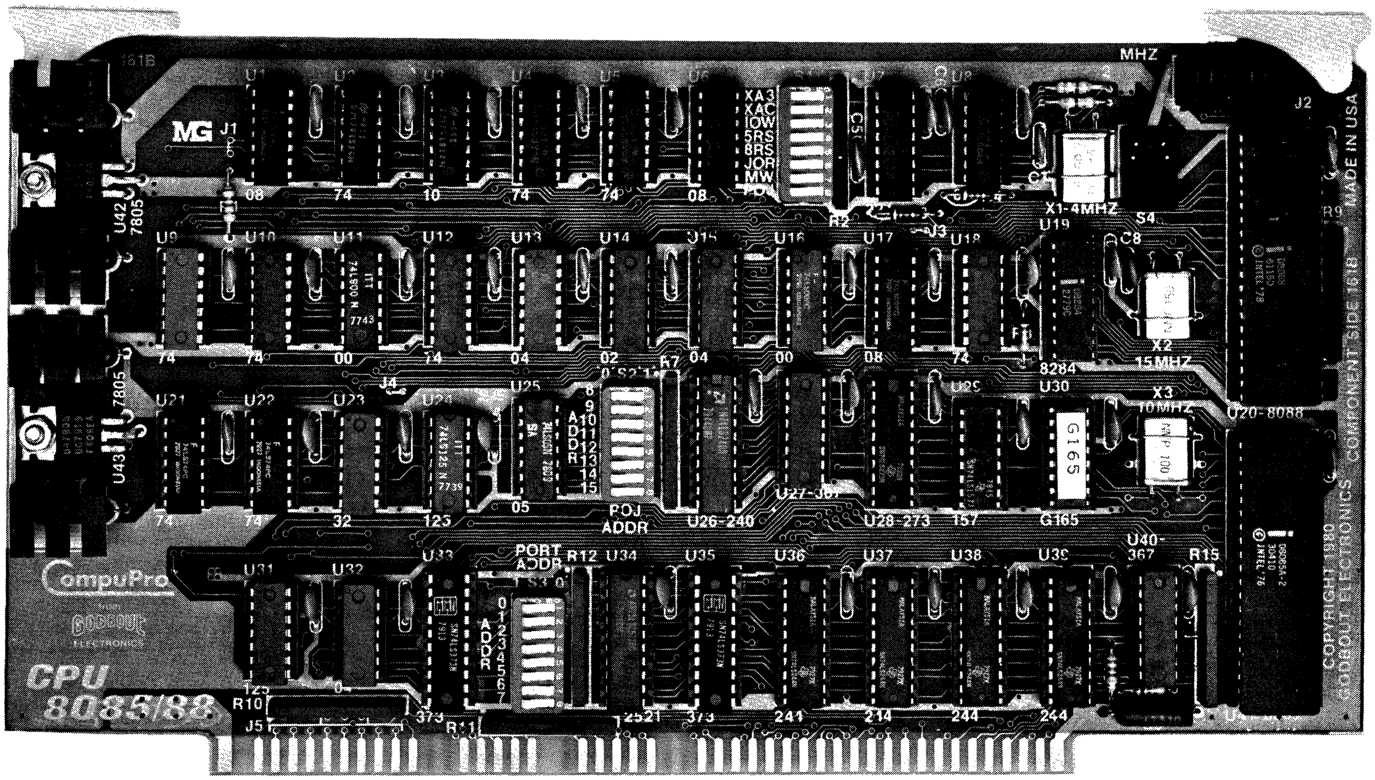
CPU 8085/88
USER'S MANUAL



IEEE • S-100
DUAL CPU
5MHz - 8 bit 8085
5MHz - 16 bit 8088

161
mac

CompuPro™ from **GODBOUT**

7/80

Table of contents

HOW TO GET YOUR CPU 8085/88 TO RUN IN UNDER 5 MINUTES WITHOUT READING THE MANUAL

This section is for those of you who are so anxious to see if your new CPU board works that you can't wait long enough to read the manual. This section will tell you how to set the switches up so that the **CPU 8085/88** will look most like your older CPU board.

WE STRONGLY RECCOMEND THAT YOU RELAX, AND READ THE MANUAL!! If, after reading and following the directions in this section, your system doesn't work, DON'T CALL!!! READ THE MANUAL FIRST!!!

8085 ONLY

If your CPU 8085/88 was factory assembled, or you are using a version that contains both CPUs, don't read this section. Skip to the next section. This section is for those of you who purchased an 8085 only version, UNKIT.

You will need to install two jumpers, J3 and J4. (If you followed the assembly instructions carefully, you should have already installed them.) J3 is located above and to the right of U17 and J4 is located above and to the right of U23.

Use a piece of component lead for the jumpers and leave about a 1/4 inch loop above the board so that they may be easily cut when you want to add an 8088.

SWITCH SETTINGS

Dip switch 1 is located between U6 and U7. It is used to select various operational modes of the **CPU 8085/88**.

DIP SWITCH 1

SWITCH POSITION	LABELED	HOW TO SET IT
1	XA3	**OFF.**
2	XAC	**OFF.**
3	IOW	**OFF.**
4	5RS	**ON.**
5	8RS	**ON.**
6	JOR	**ON.**
7	MW	**OFF** if you have a front panel, ON otherwise.
8	POJ	**ON** if you need power-on-jump, **OFF** otherwise.

Dip switch 2 is located between U25 and U26 and is used to set the power-on-jump address. If you don't need power-on-jump, you should have set Dip Switch 1, position 8 to OFF, and you can skip this section.

DIP SWITCH 2

SWITCH POSITION	LABELED	HOW TO SET IT
1	ADDR 8	**ON** to match addr. bit 8.
2	ADDR 9	**ON** to match addr. bit 9.
3	ADDR 10	**ON** to match addr. bit 10.
4	ADDR 11	**ON** to match addr. bit 11.
5	ADDR 12	**ON** to match addr. bit 12.
6	ADDR 13	**ON** to match addr. bit 13.
7	ADDR 14	**ON** to match addr. bit 14.
8	ADDR 15	**ON** to match addr. bit 15.

Dip switch 3 is used to set the I/O port that is used by the Memory Manager and swapping processors. The Compupro 'standard' is port address FD hex, and here's how to set it:

DIP SWITCH 3

SWITCH POSITION	LABELED	HOW TO SET IT
1	ADDR 0	OFF
2	ADDR 1	ON
3	ADDR 2	OFF
4	ADDR 3	OFF
5	ADDR 4	OFF
6	ADDR 5	OFF
7	ADDR 6	OFF
8	ADDR 7	OFF

Switch 4 is located in the upper right hand corner of the board and is the large switch with the red paddle. It is used to select either 2 or 5 Mhz operation for the 8085.

TURN THIS SWITCH SO THAT THE PADDLE IS IN THE LEFT-MOST POSITION. This will select 2 Mhz operation, which is a good place to start.

If you have a IMSAI-type front panel, plug the cable into J2. J2 is a 16 pin socket located in the upper right hand corner of the board. You will also want to turn position 7 of Dip Switch 1 (MW), OFF.

ABOUT THE CPU 8085/88

Congratulations on your purchase of the **CPU 8085/88** - an advanced processor board designed specifically for full electrical and mechanical compatibility with the IEEE S-100 bus standard. The S-100 bus is the professional level choice for commercial, industrial and scientific applications. This bus provides for ready expansion and modification as the state of the art improves. We believe that this board, with the rest of the **CompuPro** family, is one of the best boards available for that bus.

Features such as 5 Mhz operation, 24 bits of extended addressing, power-on jump, and the ability to have 16 bit power in an S-100 system at a reasonable price, make the **CPU 8085/88** another proud member of the **CompuPro** family.

Thank you for choosing a **CompuPro** product.... welcome to the club!

TECHNICAL OVERVIEW

The **CPU 8085/88** Dual Processor board was specifically designed to make it easy for the S-100 Bus user to get into the world of 16 bit micros, while at the same time preserving compatiblity with existing hardware and software.

We accomplished this goal by choosing the Intel 8088 16 bit CPU, (an 8 bit bus version of the 8086), and the 8085A 8 bit CPU. The 8085 provides both hardware and software compatibility with the current crop of S-100 peripherals, and the 8088 provides for greatly enhanced software capability while maintaining an 8 bit external bus for hardware compatibility.

The user may switch back and forth between the two processors with a simple software command. For example, this allows the user to let the 8085 run his currently available (and familiar) disk operating system while letting the 8088 run the more advanced applications software. One processor would then "call" the other to handle the task most suited to it.

This environment is also extremely effective when trying to develop new software for the 8088. One may use tools available that run under the 8085 (such as CP/M and Microsoft's 8086/88 Macro Cross-Assembler that runs under CP/M) to write the new code and then simply switch over to the 8088 to try it out. PROM's need not be burned and erased and systems pulled apart to transfer the code to the 8088 system.

Both processors currently run at 5 Mhz which maximizes bus throughput. A switch is provided to slow the 8085 down to 2 Mhz for software dependent timing loops that want to run at that speed and are not easily changed. Intel says that an 8 Mhz 8088 and faster 8085's are on the way, and this board was designed with the faster chips in mind. By merely changing a crystal you may upgrade the board to use a faster processor when they become available. Current state of the art in UART's (used in serial I/O boards) is barely able to cope with this 5 Mhz bus rate, so a switch is provided on the **CPU 8085/88** to add one wait state to every I/O cycle.

Power-on-jump circuitry is provided that allows the CPU to begin it's execution at any 256 byte boundary (within the lower 64K). A switch is provided to disable this feature. A switch is also provided to allow the power-on-jump circuitry to be active at power-on only, or to work at power-on and every time a RESET occurs (jump-on-reset).

The 8085 can directly address 64K bytes of memory, but our built-in Memory Manager scheme allows access to the full 16 megabytes available per the IEEE S-100 standard. The 8088 can directly address 1 megabyte, but our Memory Manager is smart enough to know which processor is in control. Thus the 8088 uses only the upper four bits of the Memory Manager so it too can access the 16 megabyte address space. This will be described in greater detail later.

The **CPU 8085/88** rigidly adheres to the IEEE S-100 standard to insure compatibility with future S-100 components, but should also work quite well with most well designed pre-IEEE hardware. For example, provision is made to use the IMSAI front panel even though it doesn't exactly fit into the new standard.

Many long hours of thought and revision went into this product and we at Compupro are confident that it will provide years of solid service. We sincerely hope that you will enjoy it.

8085 Only OPERATION

If you purchased the 8085-only version of this board you will need to install two jumpers. If you have the assembled and tested or CSC versions, these jumpers are installed for you already.

One is labeled J3 and is located above and slightly to the right of U17. The other is labeled J4 and is located above and slightly to the right of U23.

Use a small piece of wire, such as that left over from a component lead. Be sure and leave enough of a loop (about 1/4 inch) so that it may be easily cut when upgrading to the 8088. Solder the jumpers on the solder side of the board and trim off the excess leads.

If you are providing your own 8088 and 8284, do not install J3 and J4!

SWITCH S1

Switch S1 is located between U6 and U7 and is used to select the various options and operational modes of the **CPU 8085/88**.

EXTENDED ADDRESS OPTIONS

The first two positions of S1 are used to control how the upper 8 address bits (A16-23) respond to certain system operations.

Switch position 1 is labeled **XA3** and is used to control whether or not A16-23 will be tri-stated when ADSBL* is asserted on the S-100 Bus. ADSBL* is used to tri-state the system address bus, usually during a DMA.

Newer DMA devices that meet the IEEE S-100 specs are required to provide the full 24 bit address to the bus during a DMA, but most of the older devices do not.

If you have a device that provides all 24 address bits then turn switch position 1 ON. Otherwise turn switch position 1 OFF.

Switch position 2 is labeled **XAC** and is used to control whether or not the extended address bits (A16-23) are cleared when RESET* is asserted on the S-100 Bus.

The extended address register will always be cleared on a power-up, but after your program sets it's value you may not want it to be cleared (set to 0) each time a RESET occurs.

If you desire the bits to be cleared on RESET* then turn switch position 2 ON. If you want to leave the register unchanged after a RESET* then turn switch position 2 OFF.

I/O WAIT STATE SELECTION

Switch position S3 is labeled **IOW** and is used to control whether or not a wait state will be inserted into every I/O cycle.

This is particularly useful when running at 5 Mhz and beyond. Older design I/O boards may have trouble running at 5 Mhz, especially those with UARTs. To deal with this problem the **CPU 8085/88** allows automatic insertion of one wait state to each I/O cycle.

To allow wait state generation turn switch position 3 ON. To inhibit automatic wait state generation turn switch position 3 OFF.

PROCESSOR INITIALIZATION AFTER SWAP OPTIONS

The **CPU 8085/88** contains circuitry that handles orderly change-overs between the 8085 and the 8088. Part of this circuitry determines what state the respective processor will be in when it comes "on line".

One mode of operation will cause a RESET to be issued to the processor as it comes on line. This will cause the 8085 to go through a power-on-jump sequence (if enabled) or to begin execution at address 0000 hex. The 8088 will begin execution at FFFF0 hex.

Otherwise, when each processor comes on line it will begin it's execution at the place where it went off line. In other words, it just picks up where it left off.

If you desire the 8085 to always be reset when it comes on line then turn switch position 4 ON (labeled 5RS for 8085-Reset-On-Swap). If you want the 8085 to resume operations in place when coming on line, then turn switch position 4 OFF.

If you desire the 8088 to always be reset when it comes on line then turn switch position 5 ON (labeled 8RS for 8088-Reset-On-Swap). If you want the 8088 to resume operations in place when coming on line, then turn switch position 5 OFF.

NOTE: When both processors come on line the first time they will go through their normal power-up sequence regardless of the settings of these two switches.

JUMP-ON-RESET

The **CPU 8085/88** contains a "power-on-jump" circuit that allows the 8085 to begin it's execution at any 256 byte page boundary. In the strictest sense of the definition, a power-on-jump circuit should only be active at power on. But sometimes it is convenient to perform the jump each time a RESET* occurs. We have provided this option for you.

Switch position 6 is labeled **JOR** and is used to determine whether or not the 8085 will jump on reset or only on power on. If you desire a jump sequence to occur for resets then turn switch position 6 ON. If you want the jump to occur at power on only then turn switch position 6 OFF.

MWRITE ENABLE

Switch position 7 is labeled **MW** and is used to determine if the **CPU 8085/88** will generate the S-100 signal MWRITE.

The IEEE S-100 standard states that MWRITE shall be generated only in one place in a given system. Some systems, notably those with front panels (ie:IMSAI 8080), have circuitry that generate MWRITE. This circuitry is not easily disabled. Some systems do not have an MWRITE generator, so it is up to the CPU to provide MWRITE to the bus.

The important point here is that you only want one MWRITE generator in your system. If your current system has an MWRITE generator in it, you will want to turn switch position 7 OFF. If your current system has no MWRITE generator in it (or if you are about to remove your old CPU card that generated it) you will want to turn switch position 7 ON.

To sumarize, turning switch position 7 ON will allow the **CPU 8085/88** to generate MWRITE and turning switch position 7 OFF will inhibit generation of MWRITE by the **CPU 8085/88**.

POWER-ON-JUMP ENABLE

Switch position 8 is labeled **POJ** and is used to enable or disable the power-on-jump feature of the **CPU 8085/88**.

If you desire to have the power-on-jump feature active then turn switch position 8 ON. If you don't want the power-on-jump feature to be active then turn switch position 8 OFF.

If the power-on-jump feature is not utilised then the 8085 will begin it's execution at 0000 hex, which is it's normal mode of operation.

DIP SWITCH 2

POWER-ON-JUMP ADDRESS

Dip switch 2 is located between U25 and U26 and it is used to set the address that the **CPU 8085/88** will jump to on power-on if enabled.

The **CPU 8085/88** can jump to any 256 byte page boundary in the lower 64K of address space. Switch 2 is set to correspond to the bit pattern of A8-15 of the page you want to jump to. When a switch is ON it represents a "one" and conversly when a switch is OFF it represents a "zero".

SWITCH 2

Switch Position	Address Bit	
1	A8	
2	A9	
3	A10	ON = "1"
4	A11	
5	A12	
6	A13	OFF = "0"
7	A14	
8	A15	

For example, if you want to jump to E000 you would turn switch positions 1 through 5 OFF and switch positions 6 through 8 ON.

DIP SWITCH 3

MEMORY MANAGER/PROCESSOR SWAP PORT ADDRESS

Dip switch 3 is located between U33 and U34 and is used to set the address of the I/O port that is used to control the memory manager (output) and to swap processors back and forth (input).

One port address is used by the **CPU 8085/88** and it is determined by setting the positions of dip switch 3 to correspond to the lower 8 address bits (A0-7) of the desired port address.

When a switch is turned ON it represents a "zero" and conversly when a switch is turned OFF it represents a "one". Note that this is the opposite of switch 2.

SWITCH 3

Switch Position	Address Bit	
1	A0	
2	A1	
3	A2	ON = "0"
4	A3	
5	A4	
6	A5	OFF = "1"
7	A6	
8	A7	

The "standard" port address is FD hex. Any software that is provided by Compupro will assume that this switch is set for port FD. To set the switch for port FD turn switch position 2 ON and all other switch positions OFF.

2 OR 5 MHz OPERATION

SWITCH 4

Switch 4 is used to select whether the 8085 will run at 2 Mhz or 5 Mhz. In some older systems, with slower memory, it may be necessary to run at 2 Mhz. Some older generation hardware will not operate correctly or reliably at 5 Mhz (such as the IMSAI front panel). Also some software has timing loops that depend on a 2 Mhz processor. If possible these loops should be modified, but some are undocumented making them hard to find and therefore change. If this is the case, you will probably want to run at 2 Mhz.

Switch 4 is located at the upper right hand corner of the board and is the large paddle switch with the red handle.

Putting the switch in the left-most postion will cause the 8085 to run at 2 Mhz. Putting Switch 4 in the right-most position will cause the 8085 to run at 5 Mhz.

This switch is not designed to be changed while running. It may work while running, but your program may also bomb. We make no guarantees!

Note that this switch affects the 8085 only. The 8088 always runs at 5 Mhz.

USING THE MEMORY MANAGER

The Compupro Memory Manager scheme works as follows:

A standard 8 bit microprocessor (such as the 8085) can only directly address 64K bytes of memory. That takes a 16 bit address bus. But the new IEEE S-100 standard provides for 24 bits of address bus which allows addressing of 16 megabytes. The problem is how to allow a processor with only 16 bits of address to appear to have 24 bits.

What we have done is to take an eight bit output port and latch any data byte that is written to it. This latched information is then bufferred and placed on the upper 8 address lines on the S-100 bus (A16-23). Now an 8 bit processor has access to 16 megabytes instead of it's usual 64K bytes.

Those that are familiar with bank select schemes will appreciate that this is quite similar, but instead of having the port duplicated on every memory board, it appears in only one place in the system. The biggest advantage that this has over bank select schemes is that the physical memory boards are standardized because of the IEEE defined address lines.

So when using the 8085 one just writes an 8 bit word out to the output port that is selected by the setting of Dip Switch 3 and then by the begining of the next cycle that address will appear on A16-23.

But things are a bit different when using the 8088. The 8088 has 20 address bits so it can directly address 1 megabyte of memory. The Memory Manager is smart enough to know if the 8088 is in control of the bus. If the 8088 is in control, only the upper four bits of the memory manager port are used and the lower four come direct from the 8088. In other terms, A0-19 come from the 8088 and A20-23 come from the Memory Manager latch.

As with the 8085, the 8088 will write a full byte to the port, but only the upper four bits will ever see the bus. The full byte is latched, however, and all eight bits will appear on the bus when the 8085 comes back on line.

All eight bits will always be cleared (set to 0) on power up. See the Option Selection section of this manual for more options concerning the Memory Manager. (Dip Switch 1, positions 1 and 2, and Dip Switch 3.)

SWAPPING PROCESSORS

On power-up, the 8085 will always be in control of the bus and the 8088 will be asleep, just as if it was never turned on. To change over to the 8088, and to subsequently change back and forth, all that need be done is a simple INPUT instruction from the Processor Swap Port.

This port is shared with the memory manager port in that they share the same address space. Hence they are both set with dip switch 3. The memory manager uses the output side, and the swap function uses the input side. The "standard" port address is FD hex. See the Option Selection section of this manual to find out how to set this address.

When the 8088 comes on line for the first time, it will go through it's normal initialization sequence, which is that it will begin to execute code at address FFFF0 (0FFFF0 if you take the Memory Manager into account). After that first time it comes on line there are two optional ways for it to come on line thereafter.

One is just like the first, in that a RESET will be issued to the 8088 before any execution can ensue. This will cause it to again begin to execute code at FFFF0.

The other mode is one where the 8088 just "sleeps" in place until it is reawakened, when it picks up where it left off as if nothing had happenned.

Either mode can be useful depending on the application. The RESET mode may be useful in a developement atmosphere, but the sleep mode may be more practical in a real-time environment.

The 8085 can work in just the same way as the 8088. It will either do a normal power-on sequence (where it executes at 0000) or it can do a power-on-jump if enabled, when the system is powered up.

Ater the 8088 gains control, by the 8085 doing an IN to the processor swap port, you have the same options for when it comes back on line as you do with the 8088.

You may have the 8088 in one mode and the 8085 in the other or both the same. See the Option Selection section of this manual (Dip Switch 1, locations 4 and 5) to determine how to select the modes you desire.

Note that if you elect the RESET mode for the 8085 and the power-on-jump is enabled, the 8085 will do a power-on-jump sequence each time it comes on line.

Since the command to swap processors is an IN instruction, and the purpose is not really to read any information from the input port, FF hex will be returned in the A register. This means that any previous contents will be lost. So if the contents of the A register are important to you, be sure to store it somewhere first (a PUSH instruction would be the most likely).

The following is an example of a typical program flow using both processors, assume sleep mode for both:

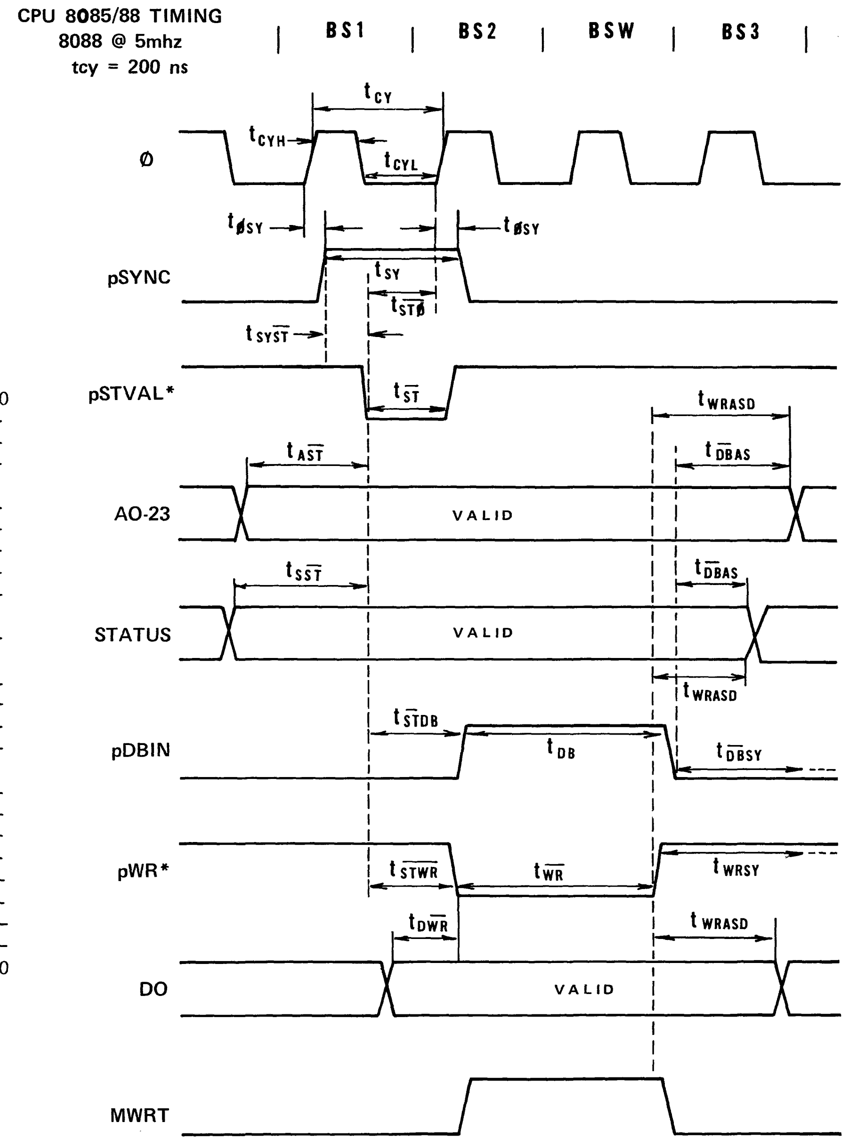
Computer Powers Up
8085 does a power-on-jump
Program begins to execute
Time to swap processors
Load in object module for 8088
Set up jump to 8088 code at FFFF0 hex
Push A register
Do an IN from port FD hex
(on board hardware puts 8085 to sleep and wakes up 8088)
8088 now on line
8088 begins execution at FFFF0 hex
Jump to object module
Perform task
Time to swap processors
(ie: need something from disk)
Put info to be passed in RAM
(ie: task info: drive number, sector etc.)
Push A register
Do an IN from port FD hex
8085 back on line
Pop A register
Perform task
Push A register
Do an IN from port FD hex
8088 back on line
etc.

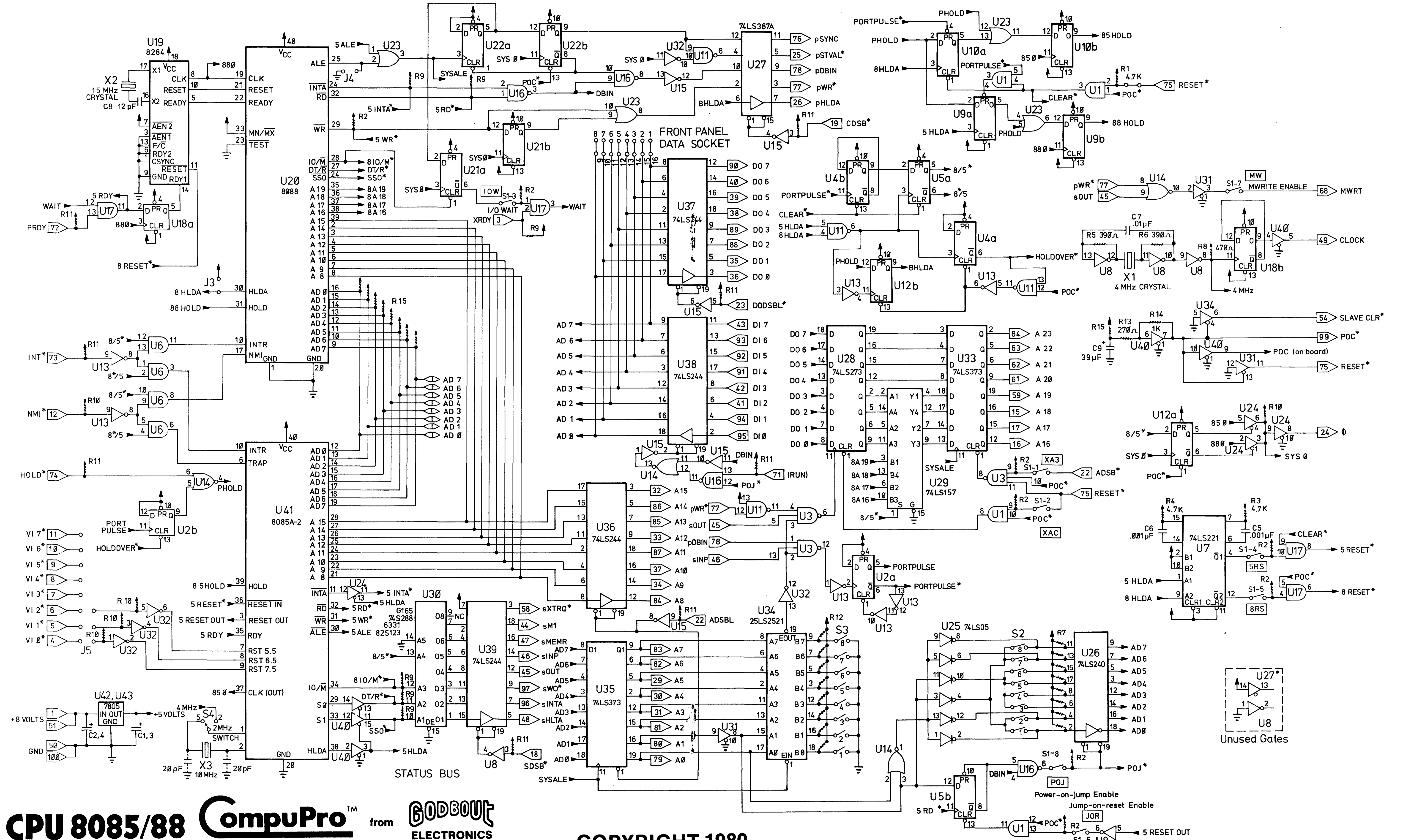
The last obvious consideration is that one processor should generally not modify the execution or stack areas of the other processor. This requires careful planning on the part of the programmer.

The complete software picture would require an entire book in itself and maybe someone out there would like to write one, but this manual will not attempt to delve into it any further.

8085 READ/WRITE CYCLE TIMING
BASED ON IEEE S-100 TIMING PARAMETERS
(see timing diagram for meaning of signal mnemonics)

		IN NSECS			
		8085		IEEE SPEC	
		2MHZ	5MHZ	MIN	MAX
tCY	Ø PERIOD	500	200	166	2000
tCYH	Ø PULSE WIDTH HIGH	245	100	0.4tCY	-
tCYL	Ø PULSE WIDTH LOW	225	80	0.4tCY	-
tØSY	DELAY Ø HIGH TO pSYNC HIGH: DELAY Ø HIGH TO pSYNC LOW	18	18	10	-
tSY	pSYNC PULSE WIDTH HIGH	500	200	0.7tCY	-
tSTØ	pSTVAL* LOW PRIOR TO Ø HIGH DURING pSYNC	215	83	0	-
tST	pSTVAL* PULSE WIDTH LOW	225	90	50	-
tSYST	DELAY pSYNC HIGH TO pSTVAL* LOW	240	100	20	-
tAST	ADDRESSES STABLE PRIOR TO pSTVAL* LOW DURING pSYNC HIGH	500	200	70	-
tSST	STATUS STABLE PRIOR TO pSTVAL* LOW DURING pSYNC HIGH	400	160	40	-
tDB	pDBIN PULSE WIDTH HIGH	465	180	0.9tCY	-
tSTDB	DELAY pSTVAL* LOW TO pDBIN HIGH	240	90	20	-
tDBSY	DELAY pDBIN LOW TO pSYNC HIGH	500	200	0	-
tDBAS	HOLD TIME FOR ADDRESSES AND STATUS AFTER pDBIN LOW	240	80	50	-
tACC	DELAY pSTVAL* LOW TO DATA VALID	600	150	minimum	-
tWR	pWR* PULSE WIDTH LOW	500	190	0.9tCY	-
tSTWR	DELAY pSTVAL* LOW TO pWR* LOW	240	98	30	-
tWRSY	DELAY pWR* HIGH TO pSYNC HIGH	480	200	0	-
tDWR	SETUP TIME DO VALID TO pWR* LOW	250	100	0.1tCY	-
tWRASD	HOLD TIME ADDRESSES FROM pWR* HIGH	220	75	0.2tCY	-
tWRASD	HOLD TIME STATUS FROM pWR* HIGH	245	100	0.2tCY	-
tWRASD	HOLD TIME DO FROM pWR* HIGH	225	100	0.2tCY	-
tWRMR	DELAY pWR* LOW TO MWRT HIGH; pWR* HIGH TO MWRT LOW	5	5	-	30





CPU 8085/88 CompuPro™ from GODBOUT ELECTRONICS

The **CPU 8085/88** contains four basic sections of circuitry. They are: The Processor and Bus Interface Circuitry, the Processor Swap Circuitry, the Memory Manager Circuitry and the Power-on-Jump Circuitry. We will discuss each section individually.

The processor and S-100 Bus interface circuitry performs the necessary primitive functions such as providing clocks to the CPUs, signal buffering and timing conversion to fit the S-100 IEEE standard. Some signals are "synthesized" for the S-100 bus since neither CPU generates those signals directly.

The 8085 has a built-in clock generator which the **CPU 8085/88** takes advantage of when running at 5 Mhz. A crystal that is two times the desired frequency is hooked across the appropriate pins. In this case the crystal is X3 and is 10 Mhz which gives us the desired 5 Mhz operating frequency.

But in-between the crystal and the 8085 is a SPDT switch, S4. S4 allows either the crystal to be hooked up directly to the CPU or allows insertion of a 4 Mhz clock signal (more about that signal later). This causes the CPU to run at 2 Mhz.

The 8088 requires a completely different clock circuit. It requires a clock frequency three times the desired operating frequency and it must have a 63/33 % duty cycle. For this we use the Intel 8284 IC, a clock generator designed specifically for the 8088/86. The 15 Mhz crystal, X2 is hooked up to the 8284 which then provides the necessary division by three and the proper duty cycle. No provision has been made, other than by changing the crystal, to alter the 8088's operating frequency.

The S-100 bus requires a 2 Mhz clock signal on pin 49 regardless of the operating frequency of the processor. This is provided by an oscillator comprised of three sections of U8 and X1, a 4 Mhz crystal. The 4 Mhz output of the oscillator is used above to run the 8085 at 2 Mhz. The 4 Mhz signal is also divided by a flip-flop to 2 Mhz, is buffered and goes out to the bus.

Both processors have a bi-directional data bus. Both are tied together (one is always tri-stated). The resultant data bus is buffered by U37 and then goes out onto the DO bus. U37 can be tri-stated by the DODSB* signal on the S-100 Bus. The DI bus is buffered by U38 and U38's outputs are tied to the internal data bus. Those outputs are enabled by the DBIN signal, but will be disabled by the POJ* signal or the RUN line on the S-100 bus. RUN is used by IMSAI-type front panels to force data into the CPU over the 16 pin cable (J2) so the DI buffer should be turned off. A pull-up resistor is provided on the RUN line so that systems not utilising this line will still work. Note that RUN is no longer a specified signal on the S-100 Bus.

Both the 8085 and the 8088 put the lower 8 address bits on the data bus during the first part of the cycle. The ALE signal from both processors is used to latch this information. Both ALE signals are ORed by a section of U23 to generate the SYSALE signal. We now have a common ALE signal for the whole card that will represent the 8085 ALE when it is in control and the 8088 ALE when it is in control.

The data bus is tied to the inputs of of U35, a 74LS373 transparent latch. The latch control is hooked to the SYSALE signal. This latches the address information from the data bus. The outputs of U35 become A0-A7 on the S-100 Bus.

A8-15 from both processors are tied together (one is always tri-stated) and are buffered by U36 and go out onto the S-100 bus. Both U35 and U36 may be tri-stated by the ADSB* signal.

The 8085 has three lines; S0, S1, and IO/M* from which all of the possible states of operation (status) may be decoded. The 8088 has three similar lines; SSO, IO/M* and DT/R*. The code on the two sets of lines is completely different. All the lines from the 8088 are tri-stated during a HOLD, but S0 and S1 from the 8085 are not. Two sections of U40 allow these two signals to be tri-stated, being contolled by the HLDA signal from the 8085.

The two sets of three lines may now be tied together. These are fed into a bipolar PROM, U30, that decodes the signals into the S-100 status signals. But the codes on the two lines are different. The **CPU 8085/88** has a signal called 8/5* that is high when the 8088 is in control and low when the 8085 is in control. This signal is used to select between two sets of data inside the PROM so that the three lines are decoded differently depending on which processor is driving the lines.

The status signals are buffered by U39 and then go out to the S-100 bus. The outputs of U39 may be tri-stated by the SDSB* signal.

The processor control signals on the S-100 Bus, pSYNC and the like, are the most difficult to synthesize, so we will discuss each signal seperately.

The signal pSYNC is used to signify the begining of each machine cycle. The SYSALE signal occurs at the begining of each machine cycle so it makes sense to use that for pSYNC.

It cannot be used directly because it does not have the proper relationship to the bus clock.

SYSALE sets a flip-flop by clocking in a "1" as it rises. The output of the flip-flop is connected to the D input of another flip-flop. The second flip-flop is clocked by the system clock so that after the next rising edge of the clock it's output will go high. This is the pSYNC signal.

When the inverting output of the second flop goes low it clears the first flip-flop which makes the D input to flop 2 go low. On the next rising edge of the system clock this low will be clocked out ending the pSYNC signal.

The first flip-flop is needed because ALE is not guaranteed to be stable during that first rising edge of the clock.

The signal pSTVAL* goes low to signify that the address and status bus are stable and contain valid information. This signal is generated by NANDing pSYNC with inverted system clock.

The signal pDBIN is used to signify that the CPU wants to read data on the DI bus. It is the read strobe for the S-100 bus. The S-100 bus specification states that pDBIN should go high during a memory read, input or interrupt acknowledge cycle.

The RD* signals from the CPUs will go low for a memory or I/O cycle, but not for an interrupt acknowledge cycle. The INTA* signal goes low for interrupt acknowledge cycles.

So to synthesize the pDBIN signal the INTA* signals from both CPUs are tied together, as are the RD* signals. (As with the data bus, only one is ever active). The composite RD* and INTA* signals go into the inputs of a NAND gate which provides inversion and an OR function at the same time. So either RD* <u>or</u> INTA* can cause pDBIN.

The output of this gate is the DBIN signal used on-board, but occurs too early to meet the S-100 Bus spec. Therefore the DBIN signal is gated by the inversion of pSYNC so that pDBIN cannot start until pSYNC is low. The resultant signal is inverted and becomes pDBIN.

The signal pWR* is used to signify that valid data is on the DO bus to be written into a memory or I/O device. It is the generalized write strobe for the S-100 bus.

Both WR* lines from the CPUs are tied together (one is always tri-stated). The signal direction is OK for the S-100 bus, but the timing is not. The S-100 Bus spec states that data must be valid before the leading edge of the write strobe and after the trailing edge of the write strobe.

The 8085 and 8088 guarantee data to be valid after the trailing edge, but not before the leading edge. Therefore we must delay the leading edge of pWR*. Delaying pWR* until the next positive clock edge will meet the timing requirements nicely, so we do just that.

The write strobe is presented simultaneously to one input of an OR gate and the D input of a flip-flop. The clock input to the flip-flop is the system clock which has been clocking in a high from the inactive write strobe, so the output of the flop will be high. The output of the flop is connected to the other input of the OR gate.

So when the write strobe is inactive (high), both inputs to the OR gate will be high making the output high. When the write strobe goes low, one input to the OR gate will go low but the output will remain high. After the next rising edge of the clock the output of the flip-flop will go low making the other input to the OR gate low. The two lows will cause the output of the OR gate to go low making our pWR* signal. When the write strobe returns high, so will the output of the OR gate ending pWR* at the right time.

The pHLDA signal is used to signify that the processor has relinqushed the bus to another temporary master, usually a DMA device. The generation of this signal will be covered under the section concerning the processor swap circuitry.

All of these signals are buffered by U27 and go out to the S-100 Bus. U27 may be tri-stated by assertion of the CDSB* signal.

The two RDY signals on the S-100 Bus are used to extend the current bus cycle for slow memory, single stepping etc. They are ANDed together along with the on-board I/O wait state generator. The resultant output is then connected directly to the 8085 RDY input, but is synchronized to the clock by a flip-flop for the 8088. The output of the flip-flop goes to the ready input of the 8284 and then to the 8088 itself. The flip-flop is necessary because of a timing idiosyncracy in the 8284.

The on-board wait state generator is a flip-flop whose D input is an extended ALE derived from the pSYNC generator. This is clocked in and out by the system clock, causing one wait state to be generated. The CLR input to the flip-flop is tied to the IO/M line from both processors which is low during memory cycles. This will cause the flip-flop to be inactive during memory cycles so that wait states will only be generated during I/O cycles.

The output of the flip-flop is connected/disconnected from the RDY circuits by a dip switch.

The interrupt lines, NMI* and pINT* are inverted and then gated by some AND gates that are turned on and off by the 8/5* line and it's inversion. This logic is used to steer the interrupt inputs to the processor currently on line.

The MWRT signal is the memory write strobe for the S-100 bus and is generated by NORing pWR* and the status line sOUT. MWRT may be disconnected from the bus by a dip switch. MWRT is bufferes by a section of U31.

The power-on-clear circuitry is used to initialize on board logic and also to generate the POC* signal for the S-100 Bus. Tri-state buffers are enabled by the POC* signal to also drive RESET* and SLVCLR* low at power-on, per the S-100 bus spec.

A simple RC time constant is formed by R15 and C9 which is buffered by a section of U40 acting as a schmitt trigger.

This completes the description of the Processor and Bus Interface Circuitry. Next we will describe how the Processor Swap Circuitry operates.

PROCESSOR SWAP CIRCUITRY

It is the job of the processor swap circuitry to handle the orderly change-over between the two CPUs on the **CPU 8085/88.**

The basic theory is this: One processor is "put to sleep" by pulling it's HOLD line high. This line is normally used for DMA. We are told by the CPU that it has gone to sleep (relinquished the bus by tri-stating it's outputs and suspending all internal operations) by the HLDA line that is active high.

What the processor swap circuitry does is to alternatively make the HOLD line to each processor high and low to control which one is on line. Since the HOLD line is also normally used for DMA, some logic must be provided to steer the pHOLD* line from the S-100 bus to the processor on line, and to arbitrate between a hold from the bus and an internal hold.

There must also be a "command" signal that tells the circuitry to change processors. This is done by decoding an I/O port and making a command pulse each time an access is done. This port address is shared with the Memory Manager circuitry (described later).

At power-on, the signal 85HOLD from the output of flip-flop U10 will be low, and the signal 88HOLD from the output of filp-flop U9 will be high. This means that the 8085 will come up running and the 8088 will be held. Also a signal named 8/5* will be low signifying that the 8085 is in control. This signal originates from flip-flop U5.

The I/O port is decoded by the 25LS2521 eight bit comparator, U34. The output of U34 is inverted and applied to one input of U3, a three input NAND gate. The other two inputs are tied to pDBIN and sINP. When all three signals are high the output of U3 will go low. This is inverted and used to clock a flip-flop, U2. The flip-flop's D input is tied high making the non-inverting output of the flop go high. The non-inverting output is tied, through two inverters for delay, to the clear input of the flop. So after two gate dealys the flip-flop will be cleared. This produces a short pulse at it's outputs. This signal is called PORTPULSE*, and is the swap command pulse we need.

When PORTPULSE* occurs, it sets the output of U10a to a one. Assuming there is no DMA request from the bus PHOLD (an internal board inversion of the bus pHOLD*) will be low. This allows the high from U10a to pass through the OR gate U23, presenting a high to the D input of U10b. On the next positive transition of the 8085 clock, the output of U10b (85HOLD) will go high. This causes a hold request to be issued to the 8085.

When the 8085 is done with the current cycle or cycles it will raise it's HLDA line, 5HLDA. This indicates that the 8085 outputs are tri-stated and internal operations have been suspended. When 5HLDA rises it will clock a low out of flip-flop U9a. This low will pass through OR gate U23 and appear at the D input to U9b. After the next rising edge of the 8088 clock, the output of U9b (88HOLD) will go low. This will allow the 8088 to begin operations.

On the next occurence of PORTPULSE* the whole process will be reversed.

PORTPULSE* will also clock U4b which is set up as a divide by two toggle. U4b's output will be low at power-on. This output is tied to the D input of U5a. The output of U5a (8/5*) will also be low at power-on. The clock input of U5a will be clocked by either 5HLDA or 8HLDA going low, signifying a processor coming on line.

So when PORTPULSE* occurs U4b will change states, and when a processor comes on line the state of U4b will be clocked through U5a, changing the state of the 8/5* line. This lets the on-board circuitry know which processor is in control.

The signal that clocks U5a (signifying a CPU coming out of a hold state) also clocks U4a. U4a then produces a pulse called HOLDOVER*. The signal that clocks U5a and U4a is inverted (which means it now signifies a CPU going into the hold state) and used to clock U12b. The D input of U12b is the PHOLD signal which will be low if there is no bus DMA request.

The output of U12b is BHLDA, which after being buffered becomes pHLDA on the S-100 Bus. So if there is no bus hold request (PHOLD low), when U12b is clocked, BHLDA will remain low. This is exactly what we want- we don't want the bus to see the internal hold operations.

But if PHOLD is high, BHLDA will go high. When the HOLDOVER* pulse occurs, at the end of the DMA, U12b will be reset causing BHLDA to return low.

Some arbitration is needed to hold off a hold request from the S-100 Bus during a processor swap. This is accomplished with flip-flop U2b and a NOR gate U14.

The pHOLD* signal from the S-100 Bus goes to one input of the NOR gate. The other input comes from the output of U2b. This output is normally low, allowing the pHOLD* signal through the gate (with inversion). The output of the gate becomes PHOLD. When PORTPULSE* occurs, a high will appear at the output of U2b which will inhibit pHOLD* from going through the NOR gate. The HOLDOVER* pulse will clear U2b allowing pHOLD* through the gate again.

PHOLD is allowed to ripple through the sections of U9 and U10 because of OR gate U23. This allows the hold request to get to the processors, unless it is held off by the previously discussed circuitry.

When the processor changes, so must the system clock. This is accomplished with a flip-flop, U12a, and some tri-state buffers from U24.

The flip-flop is constantly being clocked by the current system clock. The non-inverting output is hooked to the tri-state control of one section of U24 (U24b) and the inverting output is hooked to the tri-state control of another section of U24, U24a. The D input to the flop is hooked to the 8/5* line.

The input to U24b is the clock from the 8085 and the input to U24a is the clock from the 8088. Since the tri-state inputs are connected to opposite outputs of U12a, only one section of U24 will be turned on at a time.

When 8/5* changes, the change will be reflected in the outputs of U12a which will turn on the appropriate section of U24 allowing the correct clock to become the system clock.

The last section of the processor swap circuitry concerns the Reset-On-Swap option. This circuit allows a RESET to be issued to the processor that is just coming on line.

U7 is a 74LS221 dual one-shot. A one-shot produces a pulse of a fixed duration in response to an edge-triggered input. The duration of the pulse is set by a resistor and capacitor.

The resistors and capacitors in this case are R3 and 4, and C5 and C6. Their values will produce a pulse of about two microseconds.

The trigger inputs are the HLDA signals from the CPUs. The trigger inputs of U7 are set up to trigger on the negative going edge. So, for example, when 5HLDA falls (signifying that the 8085 is about to come on line) a 2 microsecond pulse will be issued from U7. This pulse goes to the 8085 reset input which causes the processor to reset.

The outputs of U7 may be connected/disconnected from operation by means of a dip switch.

This completes the section on the processor swap circuitry. Next we will explain the Memory Manager.

MEMORY MANAGER CIRCUITRY

The function of the memory manager is more clearly defined in the section entitled <u>USING THE MEMORY MANAGER</u>. Please refer to that section for an explanation of what the memory manager's functions are. Here we will only explain how the circuitry works.

As explained in the previous section, the memory manager's port address decoder is shared with the processor swap port. The processor swap port uses the 'input' side, while the memory manager uses the 'output' side.

The output is decoded by the signal from U34, sOUT and the inversion of pWR*. These three signals are connected to the inputs of a three input NAND gate, U3. The output of U3 will go low when all three inputs are high, signifying an output to the selected port. The output will return high at the end of the output cycle, but data will still be valid.

This positive going edge is used to latch data from the data bus into U28, a 74LS273 octal latch. The upper four outputs of the latch go to U33, a 74LS373 octal transparent latch. The lower four outputs of U28 go to half the inputs of U29, a 74LS157 quad two input multiplexer.

The control input to the multiplexer is the 8/5* line. When 8/5* is low, signifying that the 8085 is in control, the four outputs of U28 pass through the multiplexer to the other four inputs of U33. But when the 8088 is in control (8/5* high), the four outputs of the multiplexer no longer represent the outputs of U28, but instead represent A16-19 from the 8088.

SYSALE is used to latch the outputs from U28 and U29 into U33. The outputs of U33 become the S-100 extended address bits A16-23. SYSALE is needed because the 8088 only puts address info on the A16-19 lines during the first part of the cycle (as it does with the data/address lines). It also ensures that the address on the bus changes at the first part of the next cycle, instead of the last part of the current cycle.

U28 will be cleared at power-on. It may also be cleared by each succesive bus RESET* depending on the position of a dip switch.

The outputs of U33 will be tri-stated by a POC*, RESET* or ADSB*. ADSB* can be ignored by the memory manager by setting a dip-switch.

This completes the description of the memory manager circuit. Next we will discuss the operation of the power-on-jump circuit.

POWER-ON-JUMP CIRCUITRY

The power-on-jump circuitry used in the **CPU 8085/88** is designed to allow the 8085 to begin execution of a program at some address other than 0000. It does this by forcing a jump op-code (C3 hex) followed by a byte of zeroes and then an eight bit value that contains the starting address of the page you want to jump to.

At power-on, flip-flop U5b is cleared, setting it's non-inverting output high. This output is connected to one input of NAND gate U16. The other input to U16 is connected to DBIN. When both signals are high, the output (POJ*) will go low. This will enable the octal buffer U26 and disable the normal DI driver, U38.

At address 0, a C3 hex will be presented to the inputs of U26 and thus to the 8085. C3 is the 'jump' op-code. At address 1, a byte of zeroes will be presented to the inputs of U26. At address 2, the setting of dip switch 2 will be presented to the inputs of U26. The setting of dip switch 2 should correspond to the upper byte of the desired memory address.

Finally, at the next address, U5 will clock it's non-inverting output low, causing U26 to be tri-stated and the normal DI buffer to be re-enabled. This allows the next byte to be read from the bus, and normal execution ensues.

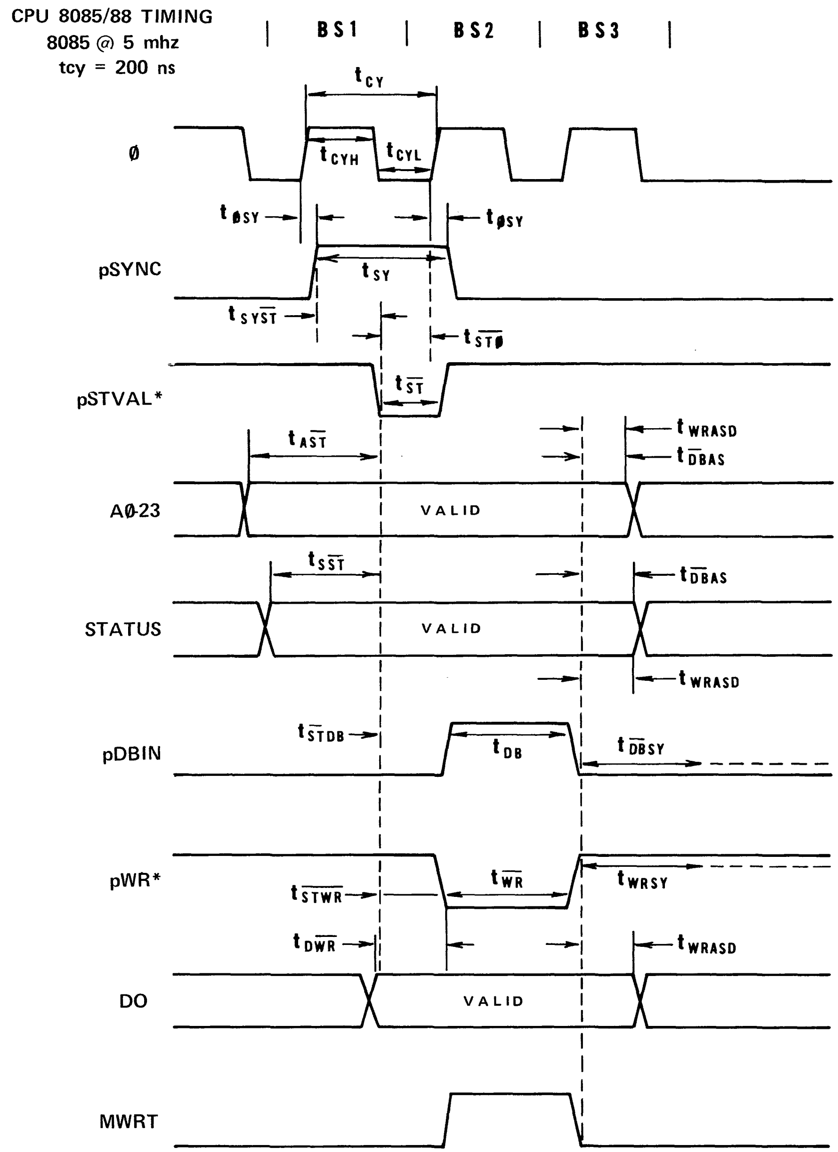
The output of U16 (POJ*), can be disabled by a dip switch, which will cause the power-on-jump circuitry to be inactive.

The power-on-jump circuitry will activate after a POC*, and if desired, after each RESET*. This option is also set with a dip switch.

This completes the circuit description of the **CPU 8085/88.**

8088 READ/WRITE CYCLE TIMING
BASED ON IEEE S-100 TIMING PARAMETERS
(see timing diagram for meaning of signal mnemonics)

		IN NSECS		
		8088	IEEE SPEC	
		5MHZ	MIN	MAX
tCY	Ø PERIOD	200	166	2000
tCYH	Ø PULSE WIDTH HIGH	65	0.4tcy	-
tCYL	Ø PULSE WIDTH LOW	110	0.4tCY	-
tØSY	DELAY Ø HIGH TO pSYNC HIGH: DELAY Ø HIGH TO pSYNC LOW	20	10	-
tSY	pSYNC PULSE WIDTH HIGH	180	0.7tCY	-
tSTØ	pSTVAL* LOW PRIOR TØ O HIGH DURING pSYNC	120	0	-
tST	pSTVAL* PULSE WIDTH LOW	120	50	-
tSYST	DELAY pSYNC HIGH TO pSTVAL* LOW	50	20	-
tAST	ADDRESSES STABLE PRIOR TO pSTVAL* LOW DURING pSYNC HIGH	180	70	-
tSST	STATUS STABLE PRIOR TO pSTVAL* LOW DURING pSYNC HIGH	200	40	-
tDB	pDBIN PULSE WIDTH HIGH	280	0.9tCY	-
tSTDB	DELAY pSTVAL* LOW TO pDBIN HIGH	150	20	-
tDBSY	DELAY pDBIN LOW TO pSYNC HIGH	280	0	-
tDBAS	HOLD TIME FOR ADDRESSES AND STATUS AFTER pDBIN LOW	180,120	50	-
tACC	DELAY pSTVAL* LOW TO DATA VALID	400 minimum		
tWR	pWR* PULSE WIDTH LOW	260	0.9tCY	-
tSTWR	DELAY pSTVAL* LOW TO pWR* LOW	140	30	-
tWRSY	DELAY pWR* HIGH TO pSYNC HIGH	320	0	-
tDWR	SETUP TIME DO VALID TO pWR* LOW	100	0.1tCY	-
tWRASD	HOLD TIME ADDRESSES FROM pWR* HIGH	180	0.2tCY	-
tWRASD	HOLD TIME STATUS FROM pWR* HIGH	140	0.2tCY	-
tWRASD	HOLD TIME DO FROM pWR* HIGH	180	0.2tCY	-
tWRMR	DELAY pWR* LOW TO MWRT HIGH; pWR* HIGH TO MWRT LOW	10	-	30



Parts List

☐	(1)	Circuit Board		

INTEGRATED CIRCUITS (NOTE: the following parts may have letter suffixes and prefixes along with the key numbers given below.

☐	(2)	74LS00	quad 2 input NAND	(U11,16)
☐	(1)	74LS02	quad 2 input NOR	(U14)
☐	(4)	74LS04	hex inverter	(U8,13,15,32)
☐	(1)	74LS05	hex inverter O.C.	(U25)
☐	(3)	74LS08	quad 2 input AND	(U1,6,17)
☐	(1)	74LS10	triple 3 input NAND	(U3)
☐	(1)	74LS32	quad 2 input OR	(U23)
☐	(9)	74LS74	dual D flip-flop	(U2,4,5,9,10 12,18,21,22)
☐	(2)	74LS125	quad tri-state driver	(U24,31)
☐	(1)	74LS157	quad 2 input MUX	(U29)
☐	(1)	74LS221	dual one shot	(U7)
☐	(1)	74LS240	octal inv. bus driver	(U26)
☐	(4)	74LS244	octal bus driver	(U36-39)
☐	(1)	74LS273	octal latch	(U28)
☐	(2)	74LS367	hex bus driver	(U27,40)
☐	(2)	74LS373	octal trnsprnt latch	(U33,35)
☐	(1)	25LS2521	octal comparator	(U34)
☐	(1)	G165	32x8 bipolar PROM	(U30)
☐	(1)	8085A-2	5 Mhz microprocessor	(U41)
☐	(2)	7805	5 volt regulator	(U42,43)

OTHER ELECTRICAL COMPONENTS

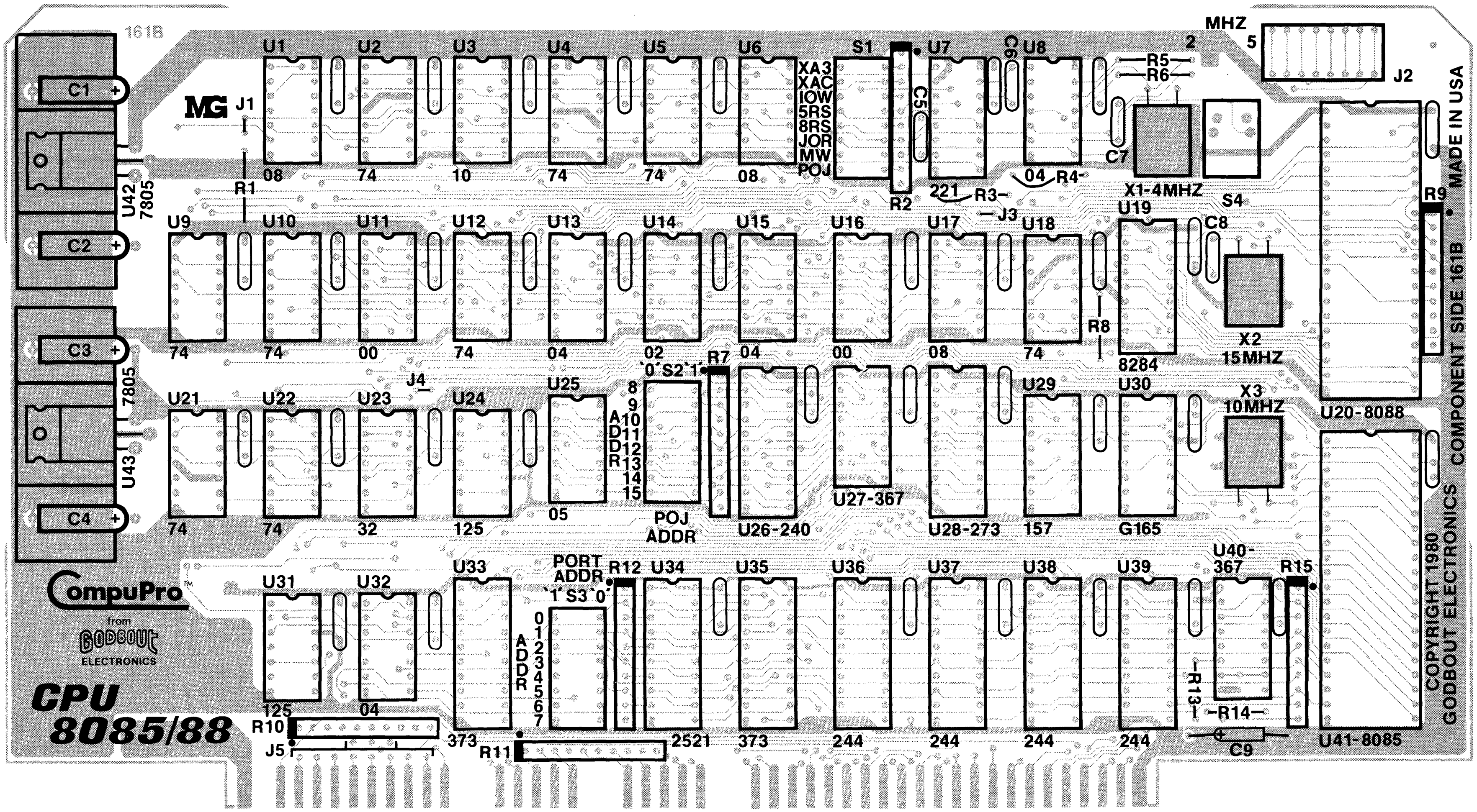
☐	(1)	crystal, 4 Mhz	(X1)
☐	(1)	crystal, 10 Mhz	(X3)
☐	(5)	39mfd, 10v tantalum capacitor	(C1-4,9)
☐	(2)	.001mfd disc capacitor	(C5,6)
☐	(1)	.01mfd disc capacitor	(C7)
☐	(36)	bypass, disc capacitor	
☐	(1)	270 ohm resistor	(R13)
☐	(2)	390 ohm resistor	(R5,6)
☐	(2)	1K ohm resistor	(R8,14)
☐	(3)	4.7K ohm resistor	(R1,3,4)
☐	(7)	SIP resistor networks	(R2,7,9-12,15)

MECHANICAL COMPONENTS

☐	(42)	low profile sockets	
☐	(3)	8 position DIP switches	(S1-3)
☐	(1)	paddle switch	(S4)
☐	(2)	heat sinks	
☐	(2)	sets 6-32 hardware	
☐	(2)	card ejectors	
☐	(1)	set user's manuals	

ADDITIONAL COMPONENTS FOR 8088 OPTION

☐	(1)	8088	5 Mhz 16 bit MPU	(U20)
☐	(1)	8284	clock generator	(U19)
☐	(1)	crystal, 15 Mhz		(X2)
☐	(1)	10pfd disc capacitor		(C8)



Component Layout

UP

CUSTOMER SERVICE INFORMATION

Our paramount concern is that you be satisfied with any Godbout CompuPro product. If this product fails to operate properly, it may be returned to us, see warranty information below.

If you have any questions about assembly, performance, specifications or need further information feel free to write us at:

P.O. Box 2355, Oakland Airport, CA 94614.

When writing, please be as specific as possible concerning the nature of your query. We maintain a 24 hour a day phone, for taking orders, (415) 562-0636. If you have problems or questions which cannot be handled by mail, this number can be used to connect you with our technical people ONLY during normal business hours (10am-5pm Pacific Time). Unfortunately, we cannot return calls, or accept collect calls.

LIMITED WARRANTY INFORMATION

Godbout Electronics will repair or replace, at our option, any parts found to be defective in either materials or workmanship for a period of 1 year from date of invoice. Defective parts *must* be returned for replacement.

If a defective part or design error causes a Godbout Electronics product to operate improperly during the 1 year warranty period, we will service it free (original owner only) if delivered and shipped at owner's expense to Godbout Electronics. If improper operation is due to an error or errors on the part of the purchaser, there may be a repair charge. Purchaser will be notified if this charge exceeds $10.00.

We are not responsible for damage caused by use of solder intended for purposes other than electronic equipment construction, failure to follow printed instructions, misuse or abuse, unauthorized modifications, use of our products in applications other than those intended by Godbout Electronics, theft, fire, or accidents.

Return to purchaser of a fully functioning unit meeting all advertised specifications in effect as of date of purchase is considered to be complete fulfillment of all warranty obligations assumed by Godbout Electronics. This warranty covers only products marketed by Godbout Electronics and does not cover other equipment used in conjunction with said products. We are not responsible for incidental or consequential damages.

Prices and specifications are subject to change without notice, owing to the volatile nature and pricing structure of the electronics industry.