The GIMIX 6809+ CPU BOARD is the heart of the GIMIX 6809 computer system. It is an extremely versatile board that offers the user a great many features and options which make it an ideal choice for a variety of systems and applications.

FEATURES

+ 4 PROM/ROM/RAM sockets for monitors and user software (up to 32K)
+ PROM/ROM/RAM sockets individually jumper selectable for single or multiple supply voltage and 1,2,4 or 8K byte devices
+ 1K bytes of scratchpad RAM (optional)
  (CMOS RAM W/Battery backup optional if the 58167 option is installed)  
+ 6840 Programmable timer with provisions for external clock, gate and output connections.
+ Time of Day Clock (58167) W/Battery backup (optional)
+ 9511A or 9512 Arithmetic processor W/jumper selectable 2, 3, or 4 MHz. clock speeds. (optional)
+ FPLA address decoding for the 8 on card devices - 4 PROM/ROM/RAM sockets, 58167, 9511A/9512, 6840, 1K scratchpad RAM
+ Software re-addressing of the 8 on card devices
  (allows software switching of on board monitors)
+ All FPLA decoded devices can be individually enabled/disabled
+ FPLA decoded devices are available for DMA access
+ Extended addressing for the FPLA decoded devices (can be disabled)
+ Jumper selectable interrupts for the 6840, 58167, and 9511A/9512
+ Any one of 3 memory management techniques can be used
  + Straight Bank Select
  + GIMIX Enhanced DAT w/software write protect (optional)
  + SWTPC compatible DAT (for SBUG-E) (optional)
+ Jumper selectable processor clock speeds (1, 1.5, 2 MHz.)
  (2MHz CPU optional)
+ Separate buffers for the 6809 and the on card devices.
+ NMI input can be jumpered to the bus or to an external connector
+ BA & BS jumper selectable for independent or gated operation
+ User defined latch output
+ Gold MOLEX connectors for trouble free contact
+ SS-50 and SS-50C compatible
+ Full DMA capabilities (works with any of the 6809 DMA methods)
+ Full Slow memory capabilities
+ Fully assembled, tested and burned in

NOTE: The GIMIX 6809+ CPU BOARD does not include a baud rate generator. In systems that require a baud rate generator, it must be provided elsewhere. The GIMIX 6800/6809 mainframe includes a baud rate generator on the motherboard.
INTRODUCTION

The GIMIX 6809+ CPU combines on one board many features that would otherwise require several different boards. The board is in effect 2 separate boards on one printed circuit card.

The CPU section consists of the 6809 processor with the necessary data and address buffers, reset circuitry, etc. plus the extended address/dynamic address translation features. The second section consists of the 8 on card devices (4 PROM/ROM/RAM sockets, 58167 time of day clock, 6840 programmable timer, 9511A or 9512 arithmetic processor, and 1K of scratch pad RAM) with their own data and address buffers, extended address decoder FPLA address decoding and the necessary support circuitry to interface them to the SS50/SS50C bus and the 6809 processor. This division of the board into two separate sections allows the 6809 to operate with minimal loading of its output lines and also permits DMA devices such as disk controllers to access the 8 on board devices directly. The board can be used with any of the standard 6809 DMA techniques.

Two methods of Dynamic Address Translation (DAT) are available as options, the one is SWTP compatible and allows the SWTP SBUG-E monitor to be used without modification, the other is an enhanced DAT that allows faster and easier task/user switching in multi-tasking /multi-user systems. Extended addressing which effectively extends the memory addressing range of the 6809 to 1 MBYTE can be derived from a bank select latch in versions without DAT or from either of the optional DAT methods.

The 4 PROM/ROM/RAM sockets permit a variety of options for on board software. Each of the 4 sockets can be individually jumper programmed to accept from 1 to 8K byte, single or multiple supply voltage, 2708 pinout compatible devices. The FPLA address decoder is preprogrammed for several different combinations of device sizes. Custom programming of the FPLA can permit other combinations of device sizes to be used.

The 58167 time of day clock option provides time, day, date information as well as time, day, date dependent programmable interrupt generation. The battery back up feature maintains accurate time even when the system power is off for extended periods of time.

The 6840 programmable timer option gives the user 3 hardware counter/timers that can be used independently or in combination with each other for timing or counting applications as well as programmable interrupt generation.

The optional 9511A or 9512 arithmetic processor provides fast, efficient mathematical capabilities in hardware. This can be extremely important in real time situations or whenever speed is an important factor.

The 1K scratchpad RAM option provides space for temporary storage, pointers, interrupt vectors, etc. The GIMIXBUG09 monitor uses some of this area for internal use. When the 58167 clock option is installed 1K of CMOS RAM may be installed to provide battery backup of the scratchpad RAM.
The GIMIX 6809 CPU is a very complex and versatile board. Most of the selectable features and options are controlled by DIP-switch and/or solderless programming jumpers. In certain cases options require additional integrated circuits or moving ICs to different sockets. The following sections describe all of the features of the GIMIX 6809 CPU in detail. Diagrams are included to show most of the common jumper configurations and software examples are provided as a guide for using the DAT, clock, arithmetic processor, etc.

NOTE: THIS MANUAL COVERS ALL VERSIONS AND AVAILABLE OPTIONS FOR THE GIMIX 6809 CPU FAMILY. SOME SECTIONS OF THE MANUAL MAY NOT APPLY TO THE PARTICULAR BOARD YOU HAVE. CERTAIN OPTIONS ARE ONLY AVAILABLE WITH A PARTICULAR VERSION OF THE BOARD. SOME OPTIONS ARE ONLY AVAILABLE AT THE TIME OF ORIGINAL PURCHASE AND CANNOT BE ADDED LATER. PLEASE CONSULT THE GIMIX PRICE LIST/ORDER FORM OR CONTACT THE FACTORY FOR FURTHER INFORMATION REGARDING OPTIONS AND AVAILABILITY.

CPU SECTION

RESET/NMI connector (CA-1)

In accordance with the SS59C bus definition the master reset line, normally the input from the front panel reset switch, has been removed from the bus. The master reset connection is available instead at connector CA-1 which is located just above IC U-54 toward the left center of the circuit board. CA-1 also provides an input for an NMI/ABORT switch. The NMI input at CA-1 is only active when the NMI option jumper ( JA-13 ) is in the NMI to reset connector position( See the NMI option jumper section). Both inputs are fully buffered and debounced and only require a single pole normally open switch from the appropriate input to ground. The ground connection is also available at CA-1.

When purchased as part of a complete system the appropriate mating connector is provided pre-wired to the front panel. CPU boards purchased separately come with a connector which can be wired to existing systems as required.

Figure A : Sheet 1, of the switch and jumper configuration drawings shows the pinouts of the RESET/NMI connector (CA-1).

CPU CLOCK SPEED option jumper (JA-3)
The GIMIX 6809 CPU gives the user the option of selecting either a 1, 1.5, or 2 MHz operating speed for the 6809. Most standard versions of the 6809 will function at either 1 or 1.5 MHz. The 2 MHz. 6809 is required for operation at 2 MHz. Note: the actual input to the 6809 is 4 times the operating speed. Thus a 1 MHz. 6809 requires a 4 MHz. input clock frequency etc. CPU speed is selected by the position one of the jumper blocks at Jumper area JA-8. JA-8 also determines the clock speed of the optional 9511A or 9512 arithmetic processor (see the 9511A/9512 section of the manual). Jumper area JA-8 is located to the upper right of the circuit board between IC U-10 and U-11.

Figures A through H: Sheet 3 of the switch and jumper configuration drawings show the pinouts and jumper positions for the CPU and arithmetic processor clock speeds.

NMI option jumper (JA-13)

The GIMIX 6809 CPU gives the user the option of connecting the NMI input of the processor to either the RESET/NMI connector (CA-1) or to one of the lines of the SS50/SS50C bus. These options are selected by the position of the jumper block at Jumper area JA-13. When the NMI to reset connector option is selected only a switch or device connected to the board through the RESET/NMI connector (CA-1) can generate an NMI. When the NMI to bus option is selected any of the on card devices that generate interrupts, as well as any other boards in the system that are connected to the proper bus line, can generate an NMI. In the NMI to bus position the NMI input buffer is connected to the bus line designated NMI on the SS50 bus and BUSY on the SS50C. Jumper area JA-13 is located at the lower left of the circuit board near trimmer resistor R-24.

Figure B: Sheet 1, of the switch and jumper configuration drawings shows the jumper positions for JA-13.

BA/BA-AND-BS option jumper (JA-14)

The 6809 CPU has 2 output lines, BUS AVAILABLE (BA) and BUS STATUS (BS) that are used to by other devices in the system to determine the status of the processor (see the 6809 data sheet for details on decoding BA and BS) Among other things BA and BS are used by DMA devices to determine when the processor has released the bus to them for DMA operations. To provide compatibility with certain existing SS-50 boards the GIMIX 6809 CPU BOARD has provisions for gating BA with BS and placing the resulting logical AND of the 2 signals on the BA line of the SS-50C bus. The BA AND BS position of JA-14 matches the SWTP configuration of BA/BS.

Figure C: sheet 1 of the switch and jumper configuration drawings shows the jumper positions for JA-14
The 6809 CPU is capable of addressing a maximum of 64K of memory space directly. This capability, while sufficient for many applications, is a limiting factor when larger, especially multi-user/multi-tasking, systems are considered. In order to expand the addressing capability of the 6809, the SS50C bus definition includes 4 extra or extended address lines. These additional lines give the bus a total of 20 address lines and allow for up to 1 MBYTE of address space. Since the 6809 generates only 16 address lines, some combination of hardware and software must be used to simulate the extra 4 signals. NOTE: In order to take advantage of extended addressing, the devices to be addressed must be capable of decoding all 20 address lines. Boards such as the GIMIX 32K STATIC RAM board, 8 PORT SERIAL I/O board, and 128K PROM/ROM/RAM board have the capability to decode all 20 address lines and can be used with extended addressing. The GIMIX 6809 CPU also provides extended address decoding for the 8 devices decoded by the FPLA.

The GIMIX 6809 CPU has provisions for three methods of simulating the 4 extra address lines. These methods include STRAIGHT BANK SELECT, which uses a simple latch, and two different methods of Dynamic Address Translation (DAT) which can also generate the 4 extended address signals. Each method has advantages and disadvantages and it is up to the user to determine which method best suits his needs. In addition, some software available from various vendors may require that a particular method be used.

STRAIGHT BANK SELECT

In the straight bank select method of extended addressing, the 1 MBYTE of available address space is divided into 16 sections or banks of 64K each. Any devices in the system that are set up to decode a particular bank address will only appear on the bus when their bank address appears on the extended address lines of the bus. For example, if a multi-user system were being set up in which each of three users was to have 32K of memory dedicated to his use, three 32K memory boards could be installed in the system. Each of the 3 boards could be set to the same base address such as $0000-$7FFF but with each set to a different bank. When a particular user was to have access to his portion of memory, the monitor/supervisor program would place the bank address of that user on the extended address bus and his bank of memory would be enabled. Areas of memory and devices such as I/O, disk controllers etc., that were to be shared by all users, would be set to ignore the extended address lines and would appear at the same address in all banks. While this method is relatively simple and straightforward it has limitations. For example, it requires that each user or task be permanently assigned a particular amount of dedicated memory. If a user needs more memory than he is originally assigned, the hardware must be reconfigured to increase his available memory. If a user does not require all of the memory allocated to him, the unused portion...
is unavailable to other users unless the hardware configuration is changed. Also since it may only be possible to assign an entire board to a particular bank, memory may be wasted because a full 32K must be assigned to a user who only requires 16K.

BANK SELECT LATCH

When using the STRAIGHT BANK SELECT method of extended addressing, the data on the extended address lines is determined by the value stored in the BANK SELECT LATCH.

The bank select latch is a write only device that appears at memory location $FFFF. A processor write to this location stores data in the latch. A read from this location returns data from any other device addressed at this location. Normally a read from this location would return the restart vector stored in a PROM/ROM monitor located from $FFFF down. The least significant 4 bits written to the latch are the bank number $0-$F of the bank to be enabled.

Any software that modifies the contents of the bank select latch must be located in an area of memory that is not affected by switching banks. In most applications bank switching would be done by a monitor/supervisor program located in memory shared by all users and not affected by the extended address lines.

Note: 2 of the remaining bits of the latch are used to control other functions on the board. If either of these functions are in use any programs that modify the bank select latch must not inadvertently change these 2 bits. Since the bank select latch is a write only device the last data written to the latch should be kept in temporary storage for comparison when new data is to be written. Bit 5 of the bank select latch is the software control latch for the FPLA address decoder and bit 4 controls the user defined latch output at solder pad "A". See the appropriate section of the manual for further information on these features.

DYNAMIC ADDRESS TRANSLATION (DAT)

Dynamic address translation is a method of memory management that allows better utilization of the memory resources within a system. It overcomes some of the disadvantages of the straight bank select method of extended addressing, in that it allows available memory to be allocated among users as their requirements vary. It also allows memory boards that can only be addressed as large contiguous blocks to be effectively split into 4K segments that can be addressed as required. Since this relocation of memory is under software control it can be done at any time and does not require hardware changes in the system. DAT breaks the entire memory space into 4K segments, so memory assignment can more closely fit the requirements of each user/task. DAT has several other advantages, for example, the system monitor/supervisor program could test each 4K segment of memory before assigning it to a user or task. If a bad segment of memory were found it could be
eliminated from the table of available memory and a good segment substituted. The monitor/supervisor program could then set a flag or indicator to show that the system requires maintenance.

In order to understand how dynamic address translation works, we must first understand the concept of physical and logical addresses. A physical address, as the name implies, is the address at which a device (memory, I/O, etc.) is set by its hardware to respond. For example, the DIP-switches on a memory board determine its physical address. A logical address is the address that the processor outputs when it attempts to access a particular memory location. For example, when the processor attempts to write data to location $0000 it outputs the logical address $0000 on its address lines. In a system that does not have dynamic address translation, the address lines from the processor are connected directly to the bus and the physical and logical address are always equal. When the processor writes to location $0000 the memory at $0000 responds. Dynamic address translation allows us to assign different logical address to memory that has a particular physical address. For example, memory with a physical address of $0000 might be assigned the logical address $2000. When the processor attempts to access memory at logical address $2000 the memory at physical address $0000 would respond. DAT then is a method of translating the logical address from the processor to a physical address that appears on the bus.

DAT is implemented on the GIMIX 6809 CPU board by inserting a high-speed, random access memory (the DAT RAM) between the upper 4 address lines of the 6809 and their corresponding bus buffers. A second identical RAM is used to generate the 4 bits of the extended address. The 4 address lines from the processor are used to address 1 of 16 locations in the DAT RAM and the data stored in these locations becomes the physical address that appears on the bus. The DAT RAM translates the upper 4 bits of the processor's logical address into the upper 4 bits of a physical address and 4 bits of extended address. Since the DAT only translates the upper 4 bits of address it divides the memory space into 16 4K logical segments. The physical address space consists of 16 4K physical address segments in each of 16 possible banks or a total of 256 possible 4K physical address segments. By placing the proper data in the DAT RAM any of the 256 physical segments can appear at any of the 16 logical address segments. For example, 4K of memory physically addressed at $0000-$0FFF could appear to the processor to be addressed at $C000-$CFFF. A physical address segment could also be made to appear at more than one logical address. For example, a 4K segment physically addressed at $E000-$EFFF could appear at $E000-$EFFF and $C000-$CFFF at the same time.

NOTE: ON POWER UP THE DATA IN THE DAT RAM AND THEREFORE LOGICAL ADDRESSES ARE UNEFIND. ONLY THE UPPER 256 BYTES OF MEMORY $FF00-$FFFF ARE GUARANTEED TO BE AT THEIR PROPER ADDRESS. IN SYSTEMS THAT USE DYNAMIC ADDRESS TRANSLATION THESE BYTES MUST CONTAIN, IN ADDITION TO THE RESET AND INTERRUPT VECTORS SOFTWARE THAT INITIALIZES THE DAT RAMS TO A PREDETERMINED STARTING CONFIGURATION, THIS INITIALIZATION MUST TAKE PLACE BEFORE ANY MEMORY.
ACCESSSES OUTSIDE THIS 256 BYTE AREA ARE MADE.

For the purpose of writing data to the DAT the DAT RAM shares the upper 16 bytes of address space $FFFF-$FFFF with the memory normally at these addresses (usually the system monitor PROM/ROM). The DAT RAM is write only, a write to these locations stores data in the DAT RAM, a read from them returns data from the PROM/ROM monitor. The least significant 4 bits of data written to the DAT are stored in the RAM that translates the upper 4 address lines of the 6809, the upper 4 bits are stored in the RAM that generates the 4 bits of extended address.

Two different methods of dynamic address translation are available as options on the GIMIX 6809 CPU board. One is compatible with the method used on the SWTP MP-09 board and the SWTP SBUG-E monitor ROM. When the board is configured for this DAT method the SWTP SBUG-E ROM can be plugged directly into the GIMIX 6809 CPU board. The GIMIX 6809 can be used as a direct replacement for the SWTP MP-09 board in systems where baud rates are available from a source other than the CPU board. The GIMIX 6809 CPU does not have an on board baud rate generator. The second DAT method is an enhanced version that allows much faster operation when switching tasks.

While dynamic address translation may find some use in smaller systems (64K and under), it will probably be most useful in larger multi-user multi-tasking applications. For example, in the system described in the section on straight bank select, with 3 users in a multi-user application, each user (task) could be assigned the memory required from any of the available memory in the system. As the users memory requirements changed memory could be allocated or deallocated as necessary. With the SWTP compatible DAT switching users (tasks) requires writing a new set of 16 values into the DAT RAM each time the system switches between users (tasks). With the GIMIX enhanced DAT method the DAT values for up to 16 different users (tasks) are written to the DAT RAM and switching between users is done by writing a single byte to a task select register. Each time the system switches users (tasks) only the task select register byte need be written.

SWTP compatible DAT

To use this version of DAT the user must write values to 16 locations in the DAT RAM. Each of the 16 locations corresponds to one of the 16 possible 4K logical address segments. The first location $FFFF0 corresponds to logical address segment $0000-$OFFF the second $FFFF1 to $1000-$1FFF and so on up to the 16th at $FFFFF which corresponds to logical address segment $FO00-$FFFF. The least significant 4 bits of data written are the complement of the upper 4 bits of the desired physical address. The upper 4 bits are the desired bank address. For example if a 4K segment of
GIMIX, INC.  

6809 CPU BOARD  

memory located at physical address $2000 in bank 1 is to appear at logical address $0000 then the value $1D would be written to the DAT RAM at location $FFFO. The 1 indicates that the desired segment is physically located in bank 1 and the D is the complement of the upper 4 bits of its physical address, $2000. The following table shows the 16 logical address segments with their corresponding DAT locations, sample data and the resulting physical addresses:

<table>
<thead>
<tr>
<th>LOGICAL ADDRESS</th>
<th>DAT LOCATION</th>
<th>DATA</th>
<th>PHYSICAL ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0000-$0FFF</td>
<td>$FFFO</td>
<td>$0F</td>
<td>0</td>
</tr>
<tr>
<td>$1000-$1FFF</td>
<td>$FFFF1</td>
<td>$0E</td>
<td>0</td>
</tr>
<tr>
<td>$2000-$2FFF</td>
<td>$FFFF2</td>
<td>$0D</td>
<td>0</td>
</tr>
<tr>
<td>$3000-$3FFF</td>
<td>$FFFF3</td>
<td>$0C</td>
<td>0</td>
</tr>
<tr>
<td>$4000-$4FFF</td>
<td>$FFFF4</td>
<td>$1B</td>
<td>1</td>
</tr>
<tr>
<td>$5000-$5FFF</td>
<td>$FFFF5</td>
<td>$09</td>
<td>0</td>
</tr>
<tr>
<td>$6000-$6FFF</td>
<td>$FFFF6</td>
<td>$0A</td>
<td>0</td>
</tr>
<tr>
<td>$7000-$7FFF</td>
<td>$FFFF7</td>
<td>$08</td>
<td>0</td>
</tr>
<tr>
<td>$8000-$8FFF</td>
<td>$FFFF8</td>
<td>$07</td>
<td>0</td>
</tr>
<tr>
<td>$9000-$9FFF</td>
<td>$FFFF9</td>
<td>$06</td>
<td>0</td>
</tr>
<tr>
<td>$A000-$AFFF</td>
<td>$FFFFA</td>
<td>$05</td>
<td>0</td>
</tr>
<tr>
<td>$B000-$BFFF</td>
<td>$FFFFB</td>
<td>$04</td>
<td>0</td>
</tr>
<tr>
<td>$C000-$CFFF</td>
<td>$FFFFC</td>
<td>$03</td>
<td>0</td>
</tr>
<tr>
<td>$D000-$DFFF</td>
<td>$FFFFD</td>
<td>$02</td>
<td>0</td>
</tr>
<tr>
<td>$E000-$EFFF</td>
<td>$FFFFE</td>
<td>$01</td>
<td>0</td>
</tr>
<tr>
<td>$F000-$FFFF</td>
<td>$FFFF</td>
<td>$00</td>
<td>0</td>
</tr>
</tbody>
</table>

NOTICE THAT IN THE ABOVE TABLE:
1 THE FIRST 4 ENTRIES THE LOGICAL AND PHYSICAL ADDRESSES ARE EQUAL
2 THE FIFTH ENTRY THE PHYSICAL SEGMENT IS IN BANK 1
3 THE SIXTH AND SEVENTH ENTRIES HAVE THEIR PHYSICAL AND LOGICAL ADDRESSES REVERSED.
4 THE REMAINING ENTRIES ALL HAVE PHYSICAL AND LOGICAL ADDRESSES EQUAL

GIMIX enhanced DAT

In this version of DAT the user can write up to 16 sets of 16 values each to the DAT RAM. Each of these sets of values is functionally equivalent to the 16 values written to the SWTP compatible DAT. Each of the 16 sets represents the DAT configuration for one user or task. Once the required values for all the users (tasks) are written to the DAT a single write to the task select register is all that is required to switch between users (tasks). The 16 locations of the DAT RAM appear at the same memory locations as in the SWTP compatible DAT. Before writing to the DAT, the number (0-$F) of the particular user (task) to be written must be stored in the task select register. After the task number is written to the task select register, the 16 DAT values for that user (task) can be written to the DAT. To completely initialize.
the GIMIX enhanced DAT the software must write each of the 16 task numbers to the task select register, in turn, followed by the 16 DAT values for that task. The task select register is located at $FF7F and like the DAT RAM shares its location with other devices. The task select register is a write only device and its logical and physical addresses are always the same.

As in the other version of DAT the least significant 4 bits of data written to the DAT correspond to the most significant 4 bits of the desired physical address, however in the GIMIX Enhanced DAT these bits need NOT BE COMPLEMENTED. They are the true value of the upper 4 bits of the desired physical address. The most significant 4 bits written to the DAT correspond to the bank address of the desired physical segment (bank numbers $0-$FF). The least significant 4 bits written to the task select register determine which of the 16 tasks ($0-$FF) is active.

Any software that modifies the contents of the task select register must reside in memory that will not be affected by the switching of tasks. It is also important to note that 2 of the remaining 4 bits of this register are used to control other functions on the board. If these functions are in use any programs that modify the contents of the task select register must not inadvertently change these bits. Since the task select register is a write only device the last value stored should be maintained in temporary storage for comparison purposes when new data is to be written to the register. Bit 5 of the task select register is the software control latch for the FPLA address decoder and bit 4 controls the user defined latch output at solder pad "A". See the appropriate sections of the manual for further information on these features.

MEMORY MANAGEMENT CONFIGURATION

The GIMIX 6809+ CPU can be configured for any of the three memory management techniques (STRAIGHT BANK SELECT, SWTP compatible DAT, or the GIMIX enhanced DAT) by the installation of the proper integrated circuits at specific locations on the board. If the board is ordered with one of the two DAT configurations, it is shipped with only the proper parts for that DAT configuration installed. If neither DAT is ordered the board is shipped with only the parts for the STRAIGHT BANK SELECT installed. If the user wishes to change the configuration of the board, he can obtain the necessary parts and install them at any time. The following chart shows the parts required and their board locations for each of the three configurations. Refer to the component layout drawing for the locations of the sockets on the board.

<table>
<thead>
<tr>
<th>LOCATION</th>
<th>SWTP COMP. DAT</th>
<th>GIMIX DAT</th>
<th>BANK SELECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>U-21</td>
<td>EMPTY</td>
<td>78L422</td>
<td>EMPTY</td>
</tr>
</tbody>
</table>
SOFTWARE WRITE PROTECT

The GIMIX 6809 CPU gives the user the ability to write protect under software control any of the 4K physical segments of memory space. The software write protect option uses the most significant bit of each location in the DAT RAM and limits the maximum address space to 1/2 MBYTE. This bit which normally controls extended address line A19 is used to inhibit processor writes to selected 4K segments. A jumper option permits setting address line A19 either high or low so that either the lower 8 banks $0-$7 or the upper 8 banks $8-$F can be used. This option can be used only in conjunction with 1 of the 2 DAT methods. To write protect a segment a 1 must be written to the most significant bit of the DAT when the selected segments physical address is written. A 0 written in this position indicates memory that is not write protected. NOTE: When the board is configured for the SWTP compatible DAT this pattern is inverted i.e. a 0 is written to write protect a segment and a 1 is written for a segment that is write enabled. This feature will write protect any type of device including RAM, I/O devices, ETC. However, it only protects these devices from writes by the 6809, it does not protect them from writes by other devices that can take control of the bus such as DMA disk controllers etc.

CAUTION: If the software write protect feature is enabled some existing software for the SWTP compatible DAT may accidentally write protect areas of memory. The software write protect feature can not be used with the SWTP SBUG-E monitor.

Software write protect option Jumper (JA-10)

The software write protect option is enabled by the position of the jumpers at JA-10. This jumper area also selects the upper or lower 1/2 MBYTE of address space when Software write protect is enabled. JA-10 is located near the center of the board above DIP-switch S2.

Figures J through M: sheet 2 of the switch and jumper configuration drawings show the pinouts and jumper positions for JA-10.
USER DEFINED LATCH OUTPUT

The device used for the TASK SELECT REGISTER or the BANK SELECT LATCH depending on which memory management option is installed, can store 6 bits of data. Since only 5 of these bits are used by the board, the sixth bit is available as a user defined output from the board. This output is available for external connection at solder pad "A", located below the battery ( B-1 ), between U-23 and U-24. ( see the component layout drawing) The output is "LS" TTL compatible and is capable of driving an equivalent load. The status of the user defined output is determined by data written to bit 4 of either the TASK SELECT REGISTER or the BANK SELECT LATCH depending on which is installed. If a 1 is written the output at pad "A" will be high, when a 0 is written it will be low. The output of the user defined latch could be used in a variety of ways. For example, it could be used to drive an external indicator ( LED, buzzer, etc. ) to indicate some internal condition of the system. It could also be used, if wired correctly, as a second software control latch input to the FPLA address decoder.

Software that uses the user defined latch must not modify the other bits in the TASK SELECT REGISTER or BANK SELECT LATCH if they are being used. See the appropriate sections of the manual for details on the bit positions in the TASK SELECT REGISTER and BANK SELECT LATCH, and the precautions required for software that writes to them.

ON CARD DEVICE SECTION

FPLA Address Decoding:

Address decoding for the eight on card devices ( 4 PROM/ROM/RAM sockets, 58167 Time of Day Clock, 6240 Programmable Timer, 9511A/9512 Arithmetic Processor and the on board RAM) is controlled by a Field Programmable Logic Array (FPLA). The addresses occupied by each of the 8 devices are fixed by the programming of the FPLA. The FPLA has sixteen input lines and eight output lines and can be programmed for up to forty-eight different input to output combinations. Twelve of the input lines are connected to address lines A4 - A15 thus enabling the FPLA to decode address ranges as small as 16 bytes. The remaining 4 inputs are connected to DIP-switch S2, sections 1 through 4. Sections 1 - 3 directly control their respective FPLA inputs. An ON (closed) switch equals a logical 0 to the FPLA, an OFF (open) switch equals a logical 1 to the FPLA. Section 4 connects the remaining FPLA input to the output of a software controlled latch. The output of this latch is determined by the data stored in bit 5 of either task select register or the bank select register depending on which option is installed. When S2-4 is ON (closed) the data stored in the latch determines the input to the FPLA when S2-4 is OFF(open) the input remains high (logical 1 ). This combination of switch and
latch inputs allows both hardware and software selection of address decoding for the on card devices.

Each of the 8 FPLA outputs connects to the select logic of one of the 8 on card devices. Each device also has a separate disable switch, DIP-switch S1 sections 1 through 8, that can be used to disable that device regardless of the FPLA programming. When a device is disabled by these switches its address space is available for use by other devices.

The FPLA supplied with the GIMIX 6809 CPU BOARD is programmed with several different addressing combinations. Combinations are included for several different existing PROM/ROM monitors. Other configurations are included which may be useful in special applications such as dedicated systems and monitor and system design. An addressing combination is selected by the settings of the FPLA input switches S2 (1-4) and the software control latch if enabled. A table is included with this manual that lists each of the addressing combinations available in the FPLA as well as the required switch and latch settings for each combination. Since one of the inputs to the FPLA can be controlled by software, is it possible to select between two different monitors under software control. It is also possible to install static RAM in one or more of the PROM/ROM/RAM sockets, load a monitor into it from disk and switch to the RAM monitor from the resident PROM/ROM monitor. This feature is very useful when writing and debugging a custom monitor or when it is desirable to easily and rapidly switch between a variety of different monitors. Since the upper 256 bytes of memory $FF00 through $FFFF are shared with the DAT, RAM addressed at these locations can not be written to.

NOTE: The FPLA is a fusible link programmed device and requires special equipment to program. Contact GIMIX, INC. for information on custom configurations to meet your requirements.

FPLA SOFTWARE CONTROL LATCH

When the board is configured for either of the 2 dynamic address translation methods the FPLA latch is controlled by writing to bit 5 of the task select register located at $FF7F. When configured for straight bank select the latch is controlled by writing to bit 5 of the bank select latch at $FFFF. The FPLA control latch shares these locations with the task/bank select functions and the user defined latch output. Programs that write to the FPLA latch must not inadvertently modify the other bits at these locations. Since the latch is a write only device the last value written should be kept in temporary storage for comparison purposes when new data is to be written. See the appropriate sections of the manual for more information on the task/bank select functions and the user defined latch output. Since changing the FPLA latch bit when the latch is enabled, DIP-switch S2 section 4 is ON (closed), may change the address of devices decoded by the FPLA, software that modifies this bit must be located in an area of memory that is not affected by the change.

EXTENDED ADDRESS DECODING:
Extended address decoding is provided to allow the eight FPLA decoded devices to respond to various combinations of the extended address lines as well as the regular address bus. An additional decoding circuit called the Fxxx decoder is also available on the board. The extended address decoder detects a match between the address present on the extended address lines (A16-A19) and the extended address set by sections 5 through 8 of DIP-switch S2 (S2 5-8). The Fxxx decoder detects the presence of an F on the upper 4 lines of the address bus (A12 through A15). Its output is true any time a read or write takes place to the upper 4K of memory ($F000-$FFFF). The outputs of these decoders can be combined in various ways by the arrangement of the programming jumpers at jumper area JA-11. And used to selectively enable the FPLA address decoder.

NOTE: The extended address decoding, with some exceptions as described later, affects all of the devices decoded by the FPLA.

Figures A through G : sheet 2 of the switch and jumper configuration diagrams show the pinouts of JA-11 and the jumper positions for the various combinations.

The following table lists the combinations that can be selected by the jumpers at JA-11.

<table>
<thead>
<tr>
<th>#</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>FPLA always enabled (extended address and Fxxx decoders disabled).</td>
</tr>
<tr>
<td>#2</td>
<td>Only devices addressed by the FPLA between $F000 and $FFFF are enabled regardless of extended address.</td>
</tr>
<tr>
<td>#3</td>
<td>FPLA only enabled when the extended address presented on the bus matches the address set by the extended address switches (S2 5-8).</td>
</tr>
<tr>
<td>#4</td>
<td>Devices addressed by the FPLA between $F000 and $FFFF are always enabled. Devices addressed below $F000 respond only when the extended address matches.</td>
</tr>
<tr>
<td>#5</td>
<td>Only devices addressed by the FPLA between $F000 and $FFFF are enabled only when the extended address matches.</td>
</tr>
<tr>
<td>#6</td>
<td>FPLA always disabled.</td>
</tr>
</tbody>
</table>

In a single user application where extended addressing is not used jumper area JA-11 would normally be set for configuration 1 (FPLA always enabled). Configuration 2 effectively disables all devices addressed by the FPLA below $F000. When configuration 3 is used devices controlled by the FPLA only appear on a single bank as set by the extended address switches. Configuration 4 allows devices addressed by the FPLA between $F000 and $FFFF to appear in all banks while those addressed below $F000 only appear in 1 specific bank. This configuration could be used for example in a bank select system where the system monitor and scratchpad RAM would appear in all banks while the other devices would only appear in a specific bank. Configuration 5 effectively disables all devices addressed by the FPLA below $F000 and only enables devices addressed between $F000 and $FFFF when the extended address matches. The last configuration, 6, permanently disables The FPLA. This configuration would be used if the system monitor was to be located on a separate board and none of the on card devices were to be used.
EXTENDED ADDRESS SELECTION

The extended address for the FPLA decoded devices is determined by the setting of DIP-switch S2 sections 5, 6, 7, and 8 (S2 5-8). S2-5 corresponds to the least significant bit (A16) of the extended address lines and S2-8 to the most significant (A19). These switches must be set to the binary equivalent of the desired extended address. When the board is configured for either STRAIGHT BANK SELECT or GIMIX ENHANCED DAT a switch that is ON (closed) corresponds to a 1 in the appropriate bit position of the extended address, a switch that is OFF (open) corresponds to a 0. When the SWTP compatible DAT is installed the switches are inverted, ON (closed) corresponds to a 0 and OFF (open) to a 1. The following diagram shows examples of extended address settings for S2 5-8 with both GIMIX ENHANCED DAT / STRAIGHT BANK SELECT and the SWTP compatible DAT.

<table>
<thead>
<tr>
<th>S2</th>
<th>GIMIX DAT / SWTP comp.</th>
<th>GIMIX DAT / SWTP comp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>ON</td>
<td>OFF</td>
</tr>
<tr>
<td>6</td>
<td>OFF</td>
<td>ON</td>
</tr>
<tr>
<td>7</td>
<td>OFF</td>
<td>ON</td>
</tr>
<tr>
<td>8</td>
<td>OFF</td>
<td>ON</td>
</tr>
</tbody>
</table>

EXTENDED ADDRESS SET FOR BANK $01

NOTE: The setting of the extended address switches is only necessary when jumper area JA-11 is set for one of the configurations that use extended addressing. Configurations 3, 4, and 5 use the extended address decoding and DIP-switch S2 sections 5 through 8 must be set to the desired address. Configurations 1, 2, and 6 ignore the extended address and the switch settings. See the EXTENDED ADDRESS DECODING section for details of these configurations.

PROM/ROM/RAM SOCKETS:

The board has four 24 pin sockets (U-4, 5, 6, and 7) that can accept most 2708/2716 pinout compatible, 1 to 8K byte PROM/ROM/RAM devices. Either single or multiple supply voltage parts can be used and each of the four sockets can be individually jumper programmed for device size and type. Jumper areas located above each socket (JA-2, 3, 4, and 5) rearrange the pinouts of the sockets as required for the various devices. The device size and address location for each socket is determined by the programming of the FPLA address decoder. (see the FPLA section of the manual and the FPLA DATA sheet for information on device size and address locations). Custom programming of the FPLA allows an almost unlimited number of combinations to be used (please contact GIMIX for information on custom FPLA programming if your application requires combinations not available in the standard FPLA supplied).
The PROM/ROM/RAM jumper configurations drawing shows the
pinouts of the jumper areas and gives examples of jumper positions
for the most common devices that can be used. GIMIX will provide,
on request, information on the suitability and jumper programming
for devices not listed on the drawing.

Each of the 4 PROM/ROM/RAM sockets can be individually enabled
or disabled by DIP-switch S1 sections 1,2,3, and 4. When enabled a
socket occupies the address space determined for it by the FPLA and
when disabled that address space is made available for other
devices in the system. The sockets are enabled when their
associated switch is ON(closed) and disabled when it is OFF(open).
Sections 1,2,3, and 4 correspond to sockets U-4,5,6, and 7
respectively. See figure H: sheet 1 of the switch and jumper
configuration drawings.

The speed requirements for devices installed in the
PROM/ROM/RAM sockets depends on the speed of the CPU. When
operating at 1 MHz. devices with an access time of 615 ns.
or less are required. 1.5 and 2 MHz. operating speeds require 360 ns
or less and 240 ns. or less devices respectively. NOTE: These
figures take into account delays introduced by the Dynamic Address
Translation RAMs. In systems that do not use the DAT the figures
can be increased by approximately 40 ns. For example without DAT a
1MHz. system requires devices with an access time of 655 ns. or
less.

SCRATCH PAD RAM

The board has provisions an optional 1K bytes of scratchpad
RAM. The 2 sockets( U-8 and U-9 ) will accept any 2114 compatible
RAMs including CMOS types and has provisions for battery backup of
this RAM when CMOS devices are used and the time of day clock with
battery backup option is installed.

The addressing of the scratchpad RAM is determined by the FPLA
address decoder (see the FPLA section of the manual). The address
at which the RAM appears depends on the programming of the FPLA and
the settings of the FPLA input switches and software control
latch.

The scratchpad RAM can be disabled by DIP-switch S1 section 8
(S1-8). When this switch is ON(closed) the RAM is enabled as
determined by the FPLA. When S1-8 is OFF(open) the RAM is disabled
and can not appear in the address space. See figure H: sheet 1 of
the switch and jumper configuration drawings.

If 2114 equivalent CMOS parts are installed at U-8 and 9, and
the battery backup clock option is installed, Data is retained in
the scratchpad RAM when the system power is turned off. The
NMOS/CMOS RAM option jumper JA-7 connects the U-8 and 9 to either
the regular +5 volt supply or to the battery backup supply.
CAUTION: JA-7 MUST NOT BE SET TO THE CMOS POSITION UNLESS CMOS
MEMORYS ARE INSTALLED! When the JA-7 is set for CMOS RAM and the
battery on/off jumper (JA-6) is in the on position, sockets U8 and
9 are always powered, even if the board is removed from the system.
Be sure that the battery jumper JA-6 is in the OFF position or that
JA-7 is not jumpered for CMOS RAM before removing or installing
parts at U-8 and 9. Jumper areas JA-6 and 7 are located in the
upper right corner of the board to the left of the battery. See
figures C and F; sheet 1 of the switch and jumper configuration
drawings for the configuration of these jumpers.

To insure data integrity, an unsafe voltage detect circuit
inhibits all write operations to the scratchpad RAM when it senses
the loss of system power. This prevents false writes to the RAM
during the transition from system to battery and battery to system
power, when the battery backup option is used.

9511A/9512 Arithmetic Processor:

Either the 9511A or 9512 are available as options on the GIMIX
6809 CPU board. The 9511A/9512 are extremely fast and powerful
devices for performing mathematical calculations. The 9511A
offers 16 and 32 bit fixed-point and 32 bit floating-point
arithmetic and a variety of transcendental functions. These
functions are: Sine, Cosine, Tangent, Arc Sine, Arc Cosine, Arc
Tangent, Square, Square Root, Common Log, Natural log, Natural
Anti Log, exponentiation, and a 32 bit floating PI. The 9512 offers 32
and 64 bit floating and fixed point addition, subtraction,
multiplication and division but none of the transcendental
functions. The use of these devices can take much of the burden
of mathematical calculations off of the processor. The processor can
load data into the 9511A/9512, issue the necessary command and then
either poll the status bit of the part or perform other tasks while
waiting for the part to generate an interrupt, and then read the
results from the proper register.

The 9511A/9512 occupies 16 bytes of address space, its address
is determined by the FPLA programming and FPLA input switches and
software control latch. The first byte is the data register and
the second byte is the status register. For example, if the
9511A/9512 were addressed from $E200 through $E20F, the data
register would appear at locations $E200, $E202, etc. and the data
register would appear at $E201, $E203, etc. Both devices are stack
oriented and have only two 8 bit wide registers. The data is
written to them 1 byte at a time, a command issued, and the result
is read from them a byte at a time.

CAUTION: IN ORDER TO MEET THE TIMING REQUIREMENTS OF THE
9511A/9512 THE MRDY CIRCUITRY OF THE CPU BOARD IS USED TO STRETCH
THE PROCESSOR CLOCK DURING ACCESSES TO THE DEVICE. BECAUSE OF THE
WAY 9511A/9512 FUNCTIONS IT IS POSSIBLE IF CERTAIN PRECAUTIONS ARE
NOT OBSERVED TO GENERATE A MRDY SIGNAL THAT IS LONGER THAN THE 6809
CAN TOLERATE. THIS COULD CAUSE UNPREDICTABLE RESULTS AND POSSIBLY
CAUSE THE ENTIRE SYSTEM TO "CRASH". TO AVOID THIS, SOFTWARE THAT
READS THE RESULTS OF A 9511A/9512 OPERATION FROM THE DATA REGISTER
MUST NOT DO SO UNTIL THE DEVICE HAS COMPLETED THE OPERATION. THIS
CONDITION CAN ALSO OCCUR IF A SECOND COMMAND IS WRITTEN TO THE
DEVICE BEFORE THE PREVIOUS OPERATION IS COMPLETED.
THE COMPLETION OF AN OPERATION CAN BE DETERMINED BY READING THE STATUS REGISTER OF THE DEVICE OR BY WAITING FOR THE DEVICE TO GENERATE AN END OF OPERATION INTERRUPT BEFORE ATTEMPTING TO READ THE RESULT FROM THE DEVICE. THIS PRECAUTION IS NOT REQUIRED WHEN WRITING DATA TO THE DEVICE OR WHEN READING THE STATUS REGISTER.

See the manufacturers data sheets and the sample program listings, included when the 9511A/9512 are factory installed, for information on data formats and program requirements for these devices.

9511A/9512 clock speed

The clock speed of the 9511A/9512 is totally independent of the 6809 processor clock. They share the same jumper area (JA-8) for clock speed selection but there are no restrictions other than the maximum operating speed of the parts, on 6809 and 9511A/9512 clock speed combinations. For example the 6809 can be operated at 1 MHz, while the 9511A/9512 is running at 3 MHz or the 6809 at 1.5 MHz, while the 9511A/9512 is set for 2 MHz, etc. Clock speeds of 2.3, and 4 MHz, are available at JA-8, clock speed for the device. Various versions of both devices are available with different maximum operating speeds. Jumper area JA-8 should not be set for a speed higher than the rating of the part.

Figures A through H: sheet 3 of the switch and jumper configuration drawings show the pinouts and jumper positions for JA-8.

9511A/9512 interrupt selection

The 9511A/9512 is capable of generating interrupts. The interrupt output of the device can be connected, by jumper area JA-12, to any one of the three 6809 hardware interrupt lines (NMI, FIRQ, and IRQ). NOTE: The NMI line at JA-12 is only active when the NMI option jumper (JA-13) is in the NMI to bus position. (See the NMI option jumper section) An additional jumper area JA-9 must be set to either the 9511A or 9512 position depending on which part is installed. This jumper is required because of a difference in the interrupt outputs between the 2 devices. JA-12 is located in the lower left of the board to the left of U-33.

Figure H: sheet 2 of the switch and jumper configuration drawings shows the pinouts of JA-12. Jumper area JA-9 is located below the 9511A/9512 socket U-3. Figure D: sheet 1 of the switch and jumper configuration drawings shows the jumper positions for JA-9.

9511A/9512 enable/disable switch

The 9511A/9512 can be disabled by DIP-switch S1 section 7 (S1-7). When S1-7 is ON(closed) the device is enabled as determined by FPLA. When S1-7 is OFF(open) the device is disabled and its memory space is available for other devices in the system.
6840 Programmable Timer:

The 6840 Programmable Timer has 3 independent, software programmable timers that can be used for timing, counting, frequency and period measurement, etc. The 3 timers can be used independently or they can be cascaded in various ways depending on the application. The input, output, and gate connections for all 3 timers are available at a jumper area (JA-1). This jumper area can be used with jumper blocks and/or standard wire wrap techniques to interconnect the timers, or with the appropriate connector (not supplied) to connect them to external devices. A 1MHz. clock output is also available at JA-1.

6840 addressing

The 6840 occupies sixteen bytes of address space as determined by the programming in the FPLA and the FPLA input switches and software control latch. The following table shows the byte assignment with byte 1 being the lowest address assigned to the device.

<table>
<thead>
<tr>
<th>Byte</th>
<th>Write</th>
<th>Read</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Write Control Register #1/#3</td>
<td>No Defined Operation</td>
</tr>
<tr>
<td>2</td>
<td>Write Control Register #2</td>
<td>Read Status Register</td>
</tr>
<tr>
<td>3</td>
<td>Write MSB Buffer Register</td>
<td>Read Timer #1 Counter</td>
</tr>
<tr>
<td>4</td>
<td>Write Timer #1 Latches</td>
<td>Read LSB Buffer Register</td>
</tr>
<tr>
<td>5</td>
<td>Write MSB Buffer Register</td>
<td>Read Timer #2 Counter</td>
</tr>
<tr>
<td>6</td>
<td>Write Timer #2 Latches</td>
<td>Read LSB Buffer Register</td>
</tr>
<tr>
<td>7</td>
<td>Write MSB Buffer Register</td>
<td>Read Timer #3 Counter</td>
</tr>
<tr>
<td>8</td>
<td>Write Timer #3 Latches</td>
<td>Read LSB Buffer Register</td>
</tr>
</tbody>
</table>

The remaining 8 bytes are a repeat of the first 8 bytes listed above, i.e. Byte 9 is the same as Byte 1, Byte 10 is the same as Byte 2, etc.

See the 6840 manual included with the board for detailed information on the functions of these bytes and on programming and using the 6840 programmable timer.

6840 interrupts

The interrupt output of the 6840 can be jumpered to any one of the three interrupt lines (NMI, FIRQ, and IRQ) by the programming jumpers at Jumper area JA-12. NOTE: The NMI line at JA-12 is only active when the NMI option jumper JA-13 is in the NMI to bus position. (see the NMI option jumper section) JA-12 is located at the lower left of the board to the left of U-33. Figure H: sheet 2 of the switch and jumper configuration drawings shows the pinouts of JA-12.

6840 enable/disable

The 6840 can be disabled by DIP-switch S1 section 6 (S1-6). When
S1-6 is ON(closed) the 6840 is enabled a determined by the FPLA. When S1-6 is OFF(open) the device is disabled and its memory space is available for other devices in the system.

6840 accuracy

Regardless of whether the 6840 uses the internal clock reference or an external one, the device uses the (E) signal from the 6809 for internal synchronization. Since this (E) signal is stretched by the processor during slow memory accesses using the MRDY line the accuracy of the device may be affected if such accesses (slow memory) are made while the 6840 is counting. Under normal circumstances this effect will be negligible. However, in applications where timing is critical and a great deal of slow memory accesses are made this should be taken into consideration.

58167 Time of Day Clock

The 58167 Time of Day Clock option provides the user with any easy means of keeping time (seconds, minutes, hours, day of week, day of month, month of year) in the system. It also provides programmable interrupt capability. The battery backup feature allows the clock to maintain accurate timekeeping even when the system power is removed. The device has its own built in oscillator and a separate crystal and is independent of processor clock speed and timing.

58167 addressing

The 58167 occupies 32 bytes of address space as determined by the programming of the FPLA, and the FPLA input switches and software control latch. The following table shows the byte assignments for the 58167 with byte 1 being the lowest address assigned to the device.

<table>
<thead>
<tr>
<th>Byte</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Counter - Thousandths of Seconds</td>
</tr>
<tr>
<td>2</td>
<td>Counter - Hundredths and Tenths of Seconds</td>
</tr>
<tr>
<td>3</td>
<td>Counter - Seconds</td>
</tr>
<tr>
<td>4</td>
<td>Counter - Minutes</td>
</tr>
<tr>
<td>5</td>
<td>Counter - Hours</td>
</tr>
<tr>
<td>6</td>
<td>Counter - Day of the Week</td>
</tr>
<tr>
<td>7</td>
<td>Counter - Day of the Month</td>
</tr>
<tr>
<td>8</td>
<td>Counter - Months</td>
</tr>
<tr>
<td>9</td>
<td>Latches - Thousandths of Seconds</td>
</tr>
<tr>
<td>10</td>
<td>Latches - Hundredths and Tenths of Seconds</td>
</tr>
<tr>
<td>11</td>
<td>Latches - Seconds</td>
</tr>
<tr>
<td>12</td>
<td>Latches - Minutes</td>
</tr>
<tr>
<td>13</td>
<td>Latches - Hours</td>
</tr>
<tr>
<td>14</td>
<td>Latches - Day of the Week</td>
</tr>
<tr>
<td>15</td>
<td>Latches - Day of the Month</td>
</tr>
<tr>
<td>16</td>
<td>Latches - Months</td>
</tr>
<tr>
<td>17</td>
<td>Interrupt Status Register</td>
</tr>
</tbody>
</table>
NOTE: Bytes 24 through 31 are undefined and not used. They are decoded by the board and are not available for other devices when the 58167 is enabled.

58167 programming

The data format of the time counters (bytes 1 through 8) is BCD (binary coded decimal), two digits per byte. These counters are used to set the time as well as read time from the device. The remaining bytes use a binary format and are used to program the devices interrupt outputs as well as read status, reset the counters, etc. Basically, setting the clock requires resetting the counters to zero by writing to the counter reset byte (byte 19), storing the time in BCD format in the counters (MINUTE through MONTH), and then writing to the GO byte (byte 22). The 58167 data sheet included when the clock option is purchased contains detailed information on setting the clock and using the programmable interrupts. Sample programs, included with this manual, give examples of setting and reading the clock.

NOTE: THE 58167 HAS A STATUS BIT THAT IS SET WHENEVER THE COUNTERS CHANGE (ROLL OVER) WHILE THE TIME IS BEING READ. THIS BIT SHOULD BE TESTED AFTER A READ FROM THE DEVICE AND THE DATA RE-READ IF THE BIT IS SET. THIS IS TO INSURE THAT THE DATA READ IS VALID. THE STATUS BIT IS CLEARED BY READING THE STATUS.

58167 interrupts

The interrupt output from the 58167 can be connected to any one of the three 6809 hardware interrupt lines by programming jumpers at jumper area JA-12. NOTE: The NMI interrupt line at JA-12 is only active when the NMI option jumper JA-13 is in the NMI to bus position. (see the NMI option jumper section) Figure H: sheet 2 of the switch and jumper configuration drawings shows the pinouts of JA-12.

Battery backup

Included with the 58167 time of day clock option is a battery backup circuit that keeps the clock running when the system power is removed.
This battery circuit can also be used to retain data in the 1K scratchpad RAM if the CMOS RAM option is installed. (see the 1K scratchpad RAM section) The battery backup system consists of a rechargeable nickel-cadmium battery and associated charging and unsafe voltage detect circuitry.

Battery charging

The charging circuit charges the battery whenever system power is applied to the board and the battery on/off jumper JA-6 is in the battery on position. The battery used is designed for continuous charging and can not be overcharged.

CAUTION: When the battery on/off jumper is in the battery on position power is applied to the 58167 socket (U-1) at all times, even when the board is removed from the system. The same is true for the 1K scratchpad RAM sockets U-8 and 9, if the CMOS/NMOS RAM option jumper is in the CMOS RAM position. (see the 1K scratchpad section) Before removing or installing parts in these sockets the battery on/off jumper should be placed in the off position or damage to the parts may result. The battery on/off jumper JA-6 is located to the left of the battery. Figure F: sheet 1 of the switch and jumper configuration drawings shows the jumper positions for JA-6.

Unsafe voltage detector

The unsafe voltage detector monitors the unregulated +8 volt supply from the bus. If the bus voltage falls below a pre-set level the circuit holds the processor in a reset state (reset line low) places the 58167 in its power down mode (PD low) and inhibits writes to the scratchpad RAM. This prevents false accesses to these devices during planned or unplanned power losses. When the voltage returns to a safe level the reset state is released (the processor performs its normal power on reset sequence) and the clock and RAM are restored to normal operation.

Unsafe voltage threshold adjust

The voltage at which the unsafe voltage detector activates is determined by the setting of R-24, which is located near the heatsink in the lower left corner of the board. This adjustment is factory set at a voltage just above the point where the on board voltage regulators fall out of regulation, approximately 7.1 volts and should not need readjustment under normal circumstances. NOTE: In certain SS-50 systems with marginal power supplies, the bus voltage may be too low to allow proper operation of the GIMIX 6809 CPU board. These systems will require modifications to their power supplies to provide sufficient voltage. The bus voltage of the system should be somewhat above the setting of the unsafe voltage threshold to provide for normal voltage conditions.
Normal operation of the GIMIX 6809 CPU board cannot be guaranteed if the unsafe voltage threshold is lowered from the factory setting or where the bus voltage is insufficient to provide proper operation of the circuit.
## GIMIX 6309 MEMORY MAP

<table>
<thead>
<tr>
<th>STARTING ADDRESS</th>
<th>ENDING ADDRESS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F800</td>
<td>$FFFF</td>
<td>GMXBUG-09</td>
</tr>
<tr>
<td>$F400</td>
<td>$F7FF</td>
<td>VIDEO PROM OR USER PROM</td>
</tr>
<tr>
<td>$F000</td>
<td>$F3FF</td>
<td>USER PROM ('U' COMMAND)</td>
</tr>
<tr>
<td>$E800</td>
<td>$FFFF</td>
<td>VIDEO RAM (80 X 24)</td>
</tr>
<tr>
<td>$E400</td>
<td>$E7FF</td>
<td>ON CARD SCRATCH PAD RAM</td>
</tr>
<tr>
<td>$E3FC</td>
<td>$E3FF</td>
<td>VIDEO BOARD REGISTERS (80 X 24)</td>
</tr>
<tr>
<td>$E3F8</td>
<td>$E3FB</td>
<td>UNDEFINED, AVAILABLE TO USER</td>
</tr>
<tr>
<td>$E3B0</td>
<td>$E3F7</td>
<td>RESERVED FOR FUTURE GIMIX USE</td>
</tr>
<tr>
<td>$E3A8</td>
<td>$E3AF</td>
<td>GRAPHICS CARD</td>
</tr>
<tr>
<td>$E240</td>
<td>$E3A7</td>
<td>UNDEFINED, AVAILABLE TO USER</td>
</tr>
<tr>
<td>$E220</td>
<td>$E23F</td>
<td>58167 ON CARD DEVICE</td>
</tr>
<tr>
<td>$E210</td>
<td>$E21F</td>
<td>6840 ON CARD DEVICE</td>
</tr>
<tr>
<td>$E200</td>
<td>$E20F</td>
<td>9511A/9512 ON CARD DEVICE</td>
</tr>
<tr>
<td>$E100</td>
<td>$E1FF</td>
<td>UNDEFINED, AVAILABLE TO USER</td>
</tr>
<tr>
<td>$E070</td>
<td>$E07F</td>
<td>I/O PORT 7</td>
</tr>
<tr>
<td>$E060</td>
<td>$E06F</td>
<td>I/O PORT 6</td>
</tr>
<tr>
<td>$E050</td>
<td>$E05F</td>
<td>I/O PORT 5</td>
</tr>
<tr>
<td>$E040</td>
<td>$E04F</td>
<td>I/O PORT 4</td>
</tr>
<tr>
<td>$E030</td>
<td>$E03F</td>
<td>I/O PORT 3</td>
</tr>
<tr>
<td>$E020</td>
<td>$E02F</td>
<td>I/O PORT 2</td>
</tr>
<tr>
<td>$E010</td>
<td>$E01F</td>
<td>I/O PORT 1</td>
</tr>
<tr>
<td>$E000</td>
<td>$E00F</td>
<td>I/O PORT 0</td>
</tr>
<tr>
<td>$C000</td>
<td>$DFFF</td>
<td>USER RAM (GIMIX FLEX)</td>
</tr>
<tr>
<td>$0000</td>
<td>$BFFF</td>
<td>USER RAM</td>
</tr>
</tbody>
</table>

This memory map is for GMXBUG-09. The four FPLA switches are in the off (open) position and the ports are configured for 16 bytes per I/O address.

Note: This is a general memory map, some devices may not be present in your system.
GENERAL INFORMATION

The FPLA (Field Programmable Logic Array) provides address decoding for the 8 memory mapped devices (scratchpad RAM, arithmetic processor, PTM (6840), time-of-day clock (58167), and PROM/ROM/RAM sockets U-4 through U-7) on the CPU board. Each FPLA is pre-programmed with several different configurations which are selected by setting DIP-switch S2 sections 1 through 4 on the CPU board. Software selection of FPLA configuration is also provided by the FPLA software control latch bit on the CPU board. Each configuration places the 8 devices at specific addresses, as indicated in the FPLA configuration sheet(s). The address configurations are chosen to provide features such as; compatibility with existing hardware/software combinations, software and/or hardware selection between two different PROM/ROM resident system monitors or operating systems, software selection between a PROM/ROM resident monitor and a RAM resident monitor (for monitor development/debugging), etc.

HARDWARE SELECTION BETWEEN TWO MONITORS

When this type of configuration is used, two different PROM/ROM resident monitors can be installed on the CPU board. Depending on the switch settings, the socket(s) for one or the other of the monitors will always be enabled at the appropriate addresses and the other socket(s) will be disabled. To switch to the other monitor, the appropriate switch must be changed and the system reset.

SOFTWARE SELECTION BETWEEN TWO MONITORS

This type of configuration is similar to hardware selection except that the monitors are switched by writing the appropriate value to the FPLA software control latch bit on the CPU board. To use this type of configuration, a special program or operating system utility, that must reside outside the address range affected by the switch, is used to change the control latch bit and jump to the beginning of the new monitor. Since the control latch bit is set to "0" on power-up or reset, this type of configuration always defaults to the monitor selected by "0" on power-up and reset.

SOFTWARE SELECTION BETWEEN PROM AND RAM MONITORS

This type of configuration, like the previous one, switches between two monitors under software control. Unlike the previous configurations however; when the primary monitor (selected by setting the control latch bit "0") is active, the socket(s) for the second monitor are relocated to another area in the address space instead of being disabled. This permits the second monitor to be loaded into RAM and then software switched into use.

MISCELLANEOUS CONFIGURATIONS

These configurations provide various combinations of device size and address for special hardware configurations and user-defined applications.
HARDWARE CONSIDERATIONS

After a configuration is chosen, the FPLA switches, S2-1 through 4, should be set to match the settings shown on the configuration sheet and the desired memory devices installed at the locations indicated. Be sure that the jumper areas above each PROM/ROM/RAM socket (JA-2 through JA-4) are set for the proper device size and type. Any unused sockets and devices (9511/9512, scratchpad RAM, PTM, Time-Of-Day clock) should be disabled by turning OFF (OPEN) the appropriate section of DIP-switch S-1. Some configurations require other special hardware considerations. For example, when software selecting between GMXBUG-09 and OS-9 the 74LS174 latch on the CPU board must be installed at location U-23, even if no DAT is installed on the board. This locates the FPLA SOFTWARE CONTROL LATCH at the proper address for OS-9. NOTE: when the CPU board is configured this way, and no DAT is installed, the extended address decoding should be disabled on all boards in the system.

SOFTWARE CONSIDERATIONS

The program that does the actual switching between two software selected monitors must reside outside the address range affected by the switch. The program should first set the FPLA software control latch to the proper value, "1" or "0" and the jump to the entry point of the new monitor. Some monitors, such as GMXBUG-09, set the FPLA software control latch bit as part of their normal initialization procedure. GMXBUG-09 sets the bit to a "0". If GMXBUG-09 were installed as the second monitor (selected by writing "1" to the latch) it would switch the system back to the primary monitor as soon as its initialization routine set the bit to "0". GMXBUG-09, unless modified by the user, must be used as the primary monitor. The secondary monitor must either set the FPLA software control latch bit to a "1" during its initialization or not modify the bit at all.
FPLA #6 is primarily intended to be used in systems that require hardware and/or software switching between the GMXBUG-09/FLEX™ monitor/operating system combination and the MICROWARE OS-9™ operating system. It also includes several configurations that are useful for custom monitor development and debugging.

GMXBUG-09/OS-9™ SWITCHING

The GMXBUG-09 PROM should be installed at location U-4 on the CPU board. The appropriate BOOT PROM, such as the GIMIX AUTOBOOT or BOOT/VIDEO PROM should be installed at U-5. The OS-9™ PROMS, P-1 and P-2, should be installed at locations U-6 and U-7 respectively. Interrupts on the CPU board, disk controller, and I/O boards must be enabled as described in the installation instructions included with OS-9™ (see the sections below for more information on interrupts). Either hardware or software switching can be selected by setting DIP-switch S-2 as shown in the appropriate section of the FPLA #6 configuration sheet. NOTE: Software selection can only be used in systems that use the GIMIX #58 PIO or #68 DMA disk controllers. The #28 PIO controller cannot be used for software selection because it does not have software interrupt enable/disable.

HARDWARE SELECTION

If hardware selection is chosen, the system will default to one of the two monitors, depending on the setting of S2 section 3. If S2-3 is set OFF (OPEN), GMXBUG-09 will be selected on power-up and reset. IF S2-3 is ON (CLOSED) OS-9™ will be selected on power-up and reset. When switching from OS-9™ to GMXBUG-09/FLEX™ the interrupts required by OS-9™ must be disabled or FLEX™ will not function properly. In most cases the interrupts can be disabled by turning the system off when switching to GMXBUG-09. Turning the system off will not disable interrupts from the TIME-OF-DAY CLOCK (58167) on the CPU board or from a #28 DOUBLE DENSITY PIO disk controller. The interrupt output from the 58167 is disabled automatically by GIMIX AUTOBOOT versions 1.1 and later. If a different boot PROM is being used, the interrupt can be disabled manually by using GMXBUG-09 to write $00 to memory location $E231 before attempting to boot FLEX™. The interrupt output of a #28 disk controller must be disabled by changing the jumper on the controller board before attempting to boot FLEX™. It is not necessary to manually disable the interrupts on the GIMIX #68 DMA disk controllers since they are disabled automatically when the system is turned off. The #58 PIO disk controller version of OS-9™ does not use interrupts from the disk controller.

SOFTWARE SELECTION

If software selection is chosen, and the CPU board does not have a DAT (Dynamic Address Translator) installed, the 74LS174 latch must be moved from location U-43 on the CPU board to location U-23. This places the FPLA software control latch at the correct address for OS-9™. NOTE: when the 74LS174 is installed at U-23 and no DAT is installed, the extended address decoding on all boards in the system must be disabled.

In the software select configuration the system will always default to GMXBUG-09 on power-up and reset. The GIMIX FLEX™ utility "OS-9.CMD" is used to switch from GMXBUG-09/FLEX™ to OS-9™. To switch from FLEX™ to OS-9™, execute the utility "OS-9.CMD" from FLEX™. The program will prompt for the insertion of the OS-9™ disk in the proper drive and a carriage return. After the (cr) is entered, the system will switch monitors and OS-9™ will be initialized.
An OS-9 program, called "GMXBUG", is provided with OS-9 for GIMIX systems. This program is used to switch from OS-9 back to GMXBUG-09. When "GMXBUG" is executed it disables the interrupt output from the 58167 Time-Of-Day clock on the CPU board and clears the FPLA SOFTWARE CONTROL LATCH, causing the system to switch back to GMXBUG-09. NOTE: Interrupts from the I/O board normally used by GMXBUG-09 are disabled when GMXBUG-09 re-initializes after the switch. If OS-9 is using I/O boards, other than the standard GMXBUG ACIA at port #0, their interrupts should be disabled before attempting to re-boot FLEX™. This can be done manually, using GMXBUG-09, or the necessary routines can be added to the "GMXBUG" program. The source code for the program "GMXBUG" is provided to permit user modification if required. Interrupts from the GIMIX #68 DMA disk controller are disabled automatically by the GIMIX bootstrap programs. If a user written boot for the #68 controller is used, it must disable the interrupts from the controller.

CAUTION: The "GMXBUG" switch program does not check the status of OS-9 before switching to GMXBUG-09. It is up to the user to determine that all active processes have been completed and that it is "safe" to terminate OS-9, before "GMXBUG" is executed. Failure to observe this caution may cause loss of data and possibly "damage" the OS-9 disk(s) in use when "GMXBUG" is executed.
### SW_01 Switch S2 Settings

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>Lt</th>
<th>DEVICE</th>
<th>STARTING ADDRESS</th>
<th>SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DC DC DC DC DC SCRATCHPAD RAM (2114)</td>
<td>$E400</td>
<td>1K</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DC DC DC DC ARITHMETIC PROCESSOR (9511)</td>
<td>$E200</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DC DC DC DC DC PROGRAMMABLE TIMER (6840)</td>
<td>$E210</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DC DC DC DC DC TIME OF DAY CLOCK (58167)</td>
<td>$E220</td>
<td>32</td>
</tr>
</tbody>
</table>

### Hardware Select Between Two Monitors (GMXBUG-09 & OS-9)

| GMXBUG-09 | OFF OFF OFF OFF DC | PROM/ROM/RAM SOCKET U-4 | $F800 | 2K |
| BOOT/VIDEO | OFF OFF OFF OFF DC | PROM/ROM/RAM SOCKET U-5 | $F000 | 2K |
| OS-9 (P-1) | OFF OFF ON OFF DC | PROM/ROM/RAM SOCKET U-6 | $F800 | 2K |
| OS-9 (P-2) | OFF OFF ON OFF DC | PROM/ROM/RAM SOCKET U-7 | $F000 | 2K |

### Software Select Between Two Monitors (GMXBUG-09 & OS-9)

| GMXBUG-09 | OFF OFF ON ON 0 | PROM/ROM/RAM SOCKET U-4 | $F800 | 2K |
| BOOT/VIDEO | OFF OFF ON ON 0 | PROM/ROM/RAM SOCKET U-5 | $F000 | 2K |
| OS-9 (P-1) | OFF OFF ON ON 1* | PROM/ROM/RAM SOCKET U-6 | $F800 | 2K |
| OS-9 (P-2) | OFF OFF ON ON 1* | PROM/ROM/RAM SOCKET U-7 | $F000 | 2K |

### 2K Monitor + SWTPc DMAF-2 At $F000

| MONITOR | OFF ON OFF DC | PROM/ROM/RAM SOCKET U-4 | $F800 | 2K |
| USER DEF. | OFF ON OFF DC | PROM/ROM/RAM SOCKET U-5 | $F400 | 1K |
| USER DEF. | OFF ON OFF DC | PROM/ROM/RAM SOCKET U-6 | $E800 | 2K |
| USER DEF. | OFF ON OFF DC | PROM/ROM/RAM SOCKET U-7 | $D800 | 2K |

### Software Select Between Two 3K PROM Monitors

| MONITOR #1 | ON OFF ON ON 0 | PROM/ROM/RAM SOCKET U-4 | $F800 | 2K |
| MONITOR #1 | ON OFF ON ON 0 | PROM/ROM/RAM SOCKET U-5 | $F400 | 1K |
| MONITOR #2 | ON OFF ON ON 1* | PROM/ROM/RAM SOCKET U-6 | $F800 | 2K |
| MONITOR #2 | ON OFF ON ON 1* | PROM/ROM/RAM SOCKET U-7 | $F400 | 1K |

### Software Select Between 4K PROM and 4K RAM Monitors

(Two 2K PROMs And Two 2K RAMs)

| PROM MONITOR | ON ON ON ON 0 | PROM/ROM/RAM SOCKET U-4 | $F800 | 2K |
| PROM MONITOR | ON ON ON ON 0 | PROM/ROM/RAM SOCKET U-5 | $F000 | 2K |
| RAM (LOAD) | ON ON ON ON 0 | PROM/ROM/RAM SOCKET U-6 | $B800 | 2K |
| RAM (LOAD) | ON ON ON ON 0 | PROM/ROM/RAM SOCKET U-7 | $B000 | 2K |
| RAM MONITOR | ON ON ON ON 1* | PROM/ROM/RAM SOCKET U-6 | $F800 | 2K |
| RAM MONITOR | ON ON ON ON 1* | PROM/ROM/RAM SOCKET U-7 | $F000 | 2K |
CONFIGURATION SHEET FOR FPLA #6
6809 CPU CARD FPLA DATA SHEET #FPLA-06

SWITCH S2
SETTINGS
1 2 3 4 Lt      DEVICE

STARTING SIZE
ADDRESS (bytes)

SOFTWARE SELECT BETWEEN 2K PROM AND 2K RAM MONITORS
WITH 2K BOOT PROM AT $F000

| PROM MONITOR | OFF | ON | ON | ON | 0 | PROM/ROM/RAM SOCKET U-4 | $F800 | 2K |
| BOOT/ETC.    | OFF | ON | ON | ON | 0 | PROM/ROM/RAM SOCKET U-5 | $F000 | 2K |
| RAM (LOAD)   | OFF | ON | ON | ON | 0 | PROM/ROM/RAM SOCKET U-6 | $E800 | 2K |

RAM MONITOR
OFF ON ON ON 1* PROM/ROM/RAM SOCKET U-6 $F800 2K

BOOT/ETC. OFF ON ON ON 1* PROM/ROM/RAM SOCKET U-5 $F000 2K

FOUR 1K DEVICES FROM $F000 TO $FFFF

#1
ON OFF OFF DC DC PROM/ROM/RAM SOCKET U-4 $FC00 1K

#2
ON OFF OFF DC DC PROM/ROM/RAM SOCKET U-5 $F800 1K

#3
ON OFF OFF DC DC PROM/ROM/RAM SOCKET U-6 $F400 1K

#4
ON OFF OFF DC DC PROM/ROM/RAM SOCKET U-7 $F000 1K

FOUR 4K DEVICES FROM $8000 TO $FFFF

#1
ON ON OFF DC DC PROM/ROM/RAM SOCKET U-4 $F000 4K

#2
ON ON OFF DC DC PROM/ROM/RAM SOCKET U-5 $D000 4K

#3
ON ON OFF DC DC PROM/ROM/RAM SOCKET U-6 $C000 4K

#4
ON ON OFF DC DC PROM/ROM/RAM SOCKET U-7 $B000 4K

† L = FPLA SOFTWARE CONTROL LATCH BIT (SEE MANUAL)

* THIS BIT CAN BE SET TO A "1" BY WRITING TO THE
FPLA SOFTWARE CONTROL LATCH (SEE MANUAL) OR BY
SETTING DIP-SWITCH S2 SECTION 4 OFF (OPEN). IF
S2-4 IS OFF (OPEN) THE BIT IS FORCED TO A "1"
REGARDLESS OF THE VALUE WRITTEN TO THE LATCH.

DC = DON'T CARE

NOTE: UNUSED DEVICES AND SOCKETS SHOULD BE DISABLED,
USING THE APPROPRIATE SECTION OF SWITCH S-1,
TO PREVENT POSSIBLE ADDRESS CONFLICTS WITH
OTHER PARTS OF THE SYSTEM.
TRIPLE SUPPLY

FIG. A

SINGLE SUPPLY

FIG. B

PARTS WITH -O SUFFIX
REQUIRE WIRE JUMPER 'A'

PARTS WITH -I SUFFIX
REQUIRE WIRE JUMPER 'B'

FIG. C

FIG. D

FIG. E

FIG. F

FIG. G

FIG. H

FIG. I

FIG. J

SPECIAL CONFIGURATION
FOR PART

NO.

GND A11 A10 W A12 A11

GND A11 A10 W A12 A11

PROM/ROM/RAM SOCKET CONFIGURATION
JUMPER AREA (1 PER SOCKET)

PROM/ROM/RAM JUMPER CONFIGURATIONS

GIMIX INC.
1337 W. 37th PLACE, CHICAGO, IL 60609
6-12-80 6809 + C.P.U.

C24-0043-1
FIGURE A: NMI/RESET CONNECTOR

FIGURE B: NMI OPTION JUMPER

FIGURE C: CMOS/NMOS RAM OPTION JUMPER

FIGURE D: 9511A/9512 SELECTION

FIGURE E: BA NORMAL OR GATED TO BS

FIGURE F: BATTERY ON/OFF JUMPER

FIGURE G: DIP SWITCH (52)

FIGURE H: DIP SWITCH (51)
FIGURE A: FPLA ENABLE OPTION JUMPER JA-11.

FIGURE B: FPLA ALWAYS ENABLED.

FIGURE C: FPLA ONLY ENABLED FOR ADDRESSES BETWEEN $F000 AND $FFFF.

FIGURE D: FPLA ONLY ENABLED WITH ON CARD EXTENDED ADDRESS MATCH.

FIGURE E: FPLA ADDRESSES BETWEEN $F000 AND $FFFF ALWAYS ENABLED OTHER ADDRESSES ONLY ENABLED WITH ON CARD EXTENDED ADDRESS MATCH.

FIGURE F: FPLA ONLY ENABLED FOR ADDRESSES BETWEEN $F000 AND $FFFF ONLY WITH ON CARD EXTENDED ADDRESS MATCH.

FIGURE G: FPLA ALWAYS DISABLED.

FIGURE H: INTERRUPT OPTION JUMPER JA-12.

FIGURE I: SAMPLE INTERRUPT JUMPER CONFIGURATION. 6840 GENERATES FIRQ 9511A/9512 GENERATES NONE 58167 GENERATES IRQ

FIGURE J: SOFTWARE WRITE PROTECT OPTION JUMPER.

FIGURE K: NO SOFTWARE WRITE PROTECT.

FIGURE L: SOFTWARE WRITE PROTECT LOWER 1/2 MBYTE.

FIGURE M: SOFTWARE WRITE PROTECT UPPER 1/2 MBYTE.
FIGURE A: CLOCK SPEED OPTION JUMPER FOR CPU AND 9511A/9512.

FIGURE B: 6809 at 1.0 MHz.

FIGURE C: 6809 at 1.5 MHz.

FIGURE D: 6809 at 2.0 MHz.

FIGURE E: 9511A/9512 at 2.0 MHz.

FIGURE F: 9511A/9512 at 3.0 MHz.

FIGURE G: 9511A at 4.0 MHz.

FIGURE H: 6809 at 1MHz and 9511A at 4MHz.

FIGURE I: 6840 OPTION JUMPER JA-1.

FIGURE J: TIMERS CASCADED-1 to 2 to 3.

FIGURE K: TIMERS SET FOR CONTINUOUS OPERATION.
POWER AND GROUND PINOUTS

* NOTES

PIN 16 OF U-3 IS CONNECTED TO +12.
U-4, 5, 6, and 7 ARE CONNECTED TO
+5C, +12, and −5 AS DETERMINED BY
THEIR ASSOCIATED JUMPER AREAS —
JA-2, 3, 4, 5.

JUMPER AREA JA-7 CONNECTS U-8
AND U-9 TO +5A WHEN THE STAN-
DARD RAM IS INSTALLED OR TO +5B
FOR BATTERY BACKUP WHEN THE
CMOS RAM OPTION IS INSTALLED.

<table>
<thead>
<tr>
<th>IC NO.</th>
<th>5A</th>
<th>5B</th>
<th>5C</th>
<th>GND</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>24</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>13</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U3</td>
<td>14</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U4</td>
<td>14</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U5</td>
<td>14</td>
<td>12</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>U6</td>
<td>24</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U7</td>
<td>24</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U8</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U9</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U10</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U11</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U12</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U13</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U14</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U15</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U16</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U17</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U18</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U19</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U20</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U21</td>
<td>22</td>
<td>22</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U22</td>
<td>22</td>
<td>22</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U23</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U24</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U25</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U26</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U27</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U32</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U33</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U34</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U35</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U36</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U37</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U38</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U39</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U40</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U41</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U42</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U43</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U44</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U45</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U46</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U47</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U48</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U49</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U50</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U51</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U52</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U53</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>U54</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td></td>
</tr>
</tbody>
</table>

GIMIX INC.
1337 W. 37th PLACE, CHICAGO, IL, 60609
6-10-80 6809+ C.P.U. SHEET 2 of 2

LOGIC SUPPLY

PROPRIETARY MAT’L., ALL RIGHTS RESERVED ©1980 GIMIX INC.
L24-0043