XMICRO-6502

Technical Manual

V2.4 2020-03-29

Features:

- WDC 65C02 CPU @ 4MHz
- 65C02 WAI instruction support
- Memory bank switcher providing 1MB expanded address range
- Optional bank switching for zero-page/stack
- Individually maskable card interrupt lines with priority and vectoring
- DMA system
- Flexible clock divider with /1, /2, /3 options
- Asymmetrical clock option for longer read/write strobes
- Card ID register for system self-configuration
- Fully open-source

Functional Description:

The XMICRO-6502 is an XMICRO primary-master card based on the 65C02 CPU. It boasts an advanced feature set among 6502 microcomputers, while maintaining full compatibility with the XMICRO bus. Its open design is almost entirely based upon 74-series logic ICs, making it very approachable and ensuring long-term component availability. The WDC 65C02 CPU provides an enhanced instruction set over older NMOS 6502s, and its fully static design allows for a much wider range of clock speeds. The card's onboard interrupt handling circuitry includes prioritized IRQ vectoring for faster response times and easier driver loading, and a DMA circuit allows for high-performance peripherals. A bank switching system provides a dramatically expanded memory space up to 1MB. Optional bank switching of the Zero-page/Stack region also allows for high-performance multitasking.

Tables of Information

Table 1 – System Memory Map

Address Range	Function
$0000-$00FF	6502 Zero-Page Memory
$0100-$01FF	6502 Stack Memory
$0200-$023F	Interrupt Vector Table (Cards 0-7)
$023F-$6FFF	Persistent Bank 00
$7000-$7FFF	I/O (Cards 0-15)
$8000-$FFFF	Bank Area

Table 2 – Card Configuration

Setting	Function
JP1	Clock source select (oscillator or divider)
JP2	Clock divisor (2 or 3)
JP3	Bank switcher enable
JP4	$\overline{\text{IOSEL}}$ disable
JP5	$\overline{\text{NMI}}$ enable

Table 3 – Interrupt Vector Locations

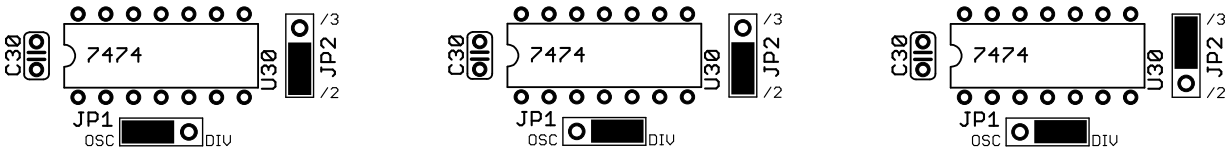
Highest Priority IRQ Signal	Vectored Address
$\overline{\text{IRQ0}}$	$0200
$\overline{\text{IRQ1}}$	$0204
$\overline{\text{IRQ2}}$	$0208
$\overline{\text{IRQ3}}$	$020C
$\overline{\text{IRQ4}}$	$0210
$\overline{\text{IRQ5}}$	$0214
$\overline{\text{IRQ6}}$	$0218
$\overline{\text{IRQ7}}$	$021C
No IRQ Signal	$023C

Table 4 – Onboard Registers

Address	D7	D6	D5	D4	D3	D2	D1	D0
$X00	ZP/Stack Bank			Bank A19	Bank A18	Bank A17	Bank A16	Bank A15
$X01	IRQ7 Mask	IRQ6 Mask	IRQ5 Mask	IRQ4 Mask	IRQ3 Mask	IRQ2 Mask	IRQ1 Mask	IRQ0 Mask
$XFF	Card ID							

CPU Clock

The system clock signal is generated by Oscillator Y1. A clock divider circuit divides the oscillator frequency by two or three by setting JP2. The divide-by-three option provides an asymmetrical clock with a 67% duty cycle to increase $\overline{RD}$/$\overline{WR}$ strobe time for a given frequency. This reduces the timing penalties for higher frequency operation. The divider circuit also provides wait-states by stopping the clock oscillation while $\overline{WAIT}$ is being asserted by the bus. The divider may be either selected or bypassed with JP1, however this also bypasses the wait-state circuit and may cause problems with other cards.



Divide-by-1 Divide-by-2 Divide-by-3

Figure 1 – System Clock Selection

Bank Switcher

The Bank Switcher circuit allows switching between 32 banks of 32K each in the upper half of the 6502's address space. $8000-$FFFF is the "Bank Area" where the currently selected bank can be accessed. The bank is selected by the Bank Register ($X00) bits D<4..0> which correspond to A<19..15> of the bank's physical address.

The lower half of the 6502's address space ($0200-$6FFF) is always mapped to Bank 00. This is persistent space that will not change, so interrupt service routines must be placed in this range to ensure their location is always known. The I/O range ($7000-$7FFF) is also located in persistent space for this reason.

The Zero-Page/Stack range ($0000-$01FF) is persistently located in Bank 00 by default. If bit D7 in the Bank Register is set, the Zero-Page/Stack range will be mapped to the first $200 of the current bank. When this feature is used, care must be taken to prevent interrupts from accessing an unexpected bank.

The bank register is reset to $01 during a system reset, meaning Bank 01 is selected and the ZP/Stack range is persistent. The result is normal 6502 behavior on startup, with memory directly mapped from $0000-$FFFF.

Interrupt Vector Generator

The Interrupt Vector Generator takes the eight card-specific interrupt requests ($\overline{IRQ}$<7..0>) and generates a prioritized interrupt vector. This vector is placed on a 16-bit interrupt vector register, which is only accessible by the CPU itself during a vector pull. Each interrupt request may be individually disabled by setting the corresponding bit of the interrupt mask register ($X01). $\overline{IRQ}$<7..0> originate at the $\overline{IRQX}$ pin of the first eight backplane slots and correspond to their slot number. $\overline{IRQ0}$ is the highest priority interrupt request.

When one or more IRQ lines are asserted, the priority encoder places the value of the highest priority line on D<5..2> of the interrupt vector register's least-significant byte ($FFFE). When the CPU enters its interrupt sequence, it jumps to the location stored in this register and begins execution. Each of the interrupt vector locations contains 4 bytes of memory, and must contain a JMP instruction to an interrupt service routine. Table 3 contains the locations of each interrupt vector address.

The output of the IVG is combined with $\overline{\text{INT}}$ to allow expanded interrupt handling circuitry on large backplanes. Because of this, when the $\overline{\text{INT}}$ signal is asserted the CPU will be sent to the vector at $023C, where a hardware-specific interrupt handler can be placed. The BRK instruction will also send the CPU to the vector at $023C. In systems with no additional interrupt circuitry, $023C will conveniently only be used by the BRK instruction.

DMA System

The onboard DMA (direct memory access) system allows other devices to directly manipulate the system's memory without the intervention of the CPU. Typically this is used to achieve high speed, low-overhead operations which would take significantly more time for the CPU to perform. These operations are asynchronous in nature and do not need to occur at the same frequency as the CPU clock. They are only limited by the timing of the system components being addressed.

An interrupt request is generated for the card when the $\overline{\text{BUSACK}}$ signal is asserted by the DMA circuit, indicating that the CPU has been stopped. This interrupt request is cleared by reading $X01.

These are the steps in a DMA operation:

1. (Optional) A register is set on a secondary master to initiate a DMA operation.
2. The secondary master asserts $\overline{\text{BUSRQ}}$.
3. The XMICRO-6502 asserts an internal $\overline{\text{WAIT}}$ signal to stop the CPU.
4. When safe to do so, the CPU is removed from the bus and the XMICRO-6502 asserts $\overline{\text{BUSACK}}$.
5. The secondary master temporarily controls the bus and performs a DMA operation.
6. The secondary master releases the $\overline{\text{BUSRQ}}$ signal.
7. The XMICRO-6502 places the CPU back on the bus and releases $\overline{\text{BUSACK}}$.
8. (Optional) The CPU responds to the DMA interrupt request generated by this process.

The following signals enter a high-impedance state during a DMA operation:

A<19..0>
D<7..0>
$\overline{\text{RD}}$
$\overline{\text{WR}}$
$\overline{\text{FETCH}}$