W65C02SOC64
6502 Based SoC with MicroModules
Datasheet

Table of Contents

DOCUMENT REVISION HISTORY

Version	Date	Authors	Description
1.0	09/20/2024	Bill Mensch, David Gray	Initial Document Entry

1 INTRODUCTION

The W65C02SOC64 Microcontroller Datasheet is for WDC's customizable educational microcontroller platform. The W65C02SOC64 was designed for professors to adopt to their VLSI/SOC Design courses, self-paced learners, and the basis of commercial products.

This datasheet includes information for the W65C02SOC64 System-on Chip (SOC) and the W65C02SOC64M08SA Intel MAX10M08SA FPGA Microcontroller for emulation of the SOC ASIC. Parts of the datasheet will describe implementation details for use with Efabless' ChipIgnite tapeout using the SkyWater SKY130 PDK.

MyMENSCH™ Rev-C uses a W65C51RTL to drive the CH340 serial-to-USB code port interface for use with WDCTools for both Assembly and C language code development.

The W65C02SOC64 consists of the base microcontroller and micromodules that allow you to design custom modules to expand functionality. The microprocessor unit (MPU) is the W65C02RTL microprocessor. The W6502SOC64 base microcontrollers have interfaces for connected Things for sensing, processing, communicating and actuating (SPCA) are described with the Verilog HDL for use with both FPGAs and ASIC design and manufacturing flow. This concept is furthered with the concept of micromodules. A micromodule is a function or group of functions that expand the functionality of the W65C02SOC64. Examples of micromodules include memory, ADC, DAC, multipliers, dividers, counters, timers, etc.

This product description assumes that the reader is familiar with the W65C02S 8-bit CPU family hardware and programming capabilities. Refer to documentation on the WDC65xx.com website, ***Programming the 65816 Including the 6502, 65C02 and 65802*** Manual,

1.1 Key Features of the W65C02SOC64 Microcontroller

Base Microcontroller Features

- IO Operating Voltage – 3.3V
- Core Operating Voltage – 1.8V
- System Operation Speed – 14.7456 MHz
- W65C02RTL MPU
- W65C51RTL ACIA (x2) – XTLI @ 1.8432 MHz Created with divider logic from XCLK
- W65CGPIO E Port (Used for ACIA_A Handshake, SPI CS, and UART CS)
- SPI Primary
- I2C Primary
- WDC 2K byte Monitor for boot loading and debugging code
- 8K bytes User code SRAM boot loaded from USB or copied from external SPI serial FLASH memory
- 8K bytes of data SRAM

MicroModule Features (28 MMIO Pins Available for all custom microcontrollers)

- W65C22RTL VIA (Uses 20 MMIO Pins)
- W65CGPIO Port A (Uses 8 MMIO Pins)

1.2 W65C02SOC64 Pin Function List using Efabless OpenFrame

Power and Ground Pins (14)

- vdda -User area 0 3.3V supply
- vdda1 -User area 1 3.3V supply
- vdda2 -User area 2 3.3V supply
- vssa -User area 0 analog ground
- vssa1 -User area 1 analog ground
- vssa2 -User area 2 analog ground
- vccd -Common 1.8V supply
- vccd1 -User area 1 1.8V supply
- vccd2 -User area 2 1.8v supply
- vssd -Common digital ground
- vssd1 -User area 1 digital ground
- vssd2 -User area 2 digital ground
- vddio -Common 3.3V ESD supply
- vssio -Common ESD ground

System Functions

- 1x XCLK (System CLK) - *ACIA Clock generated by dividing XCLK
- 1x RESB
- 1x NMIB
- 1x External IRQB
-

Serial Functions

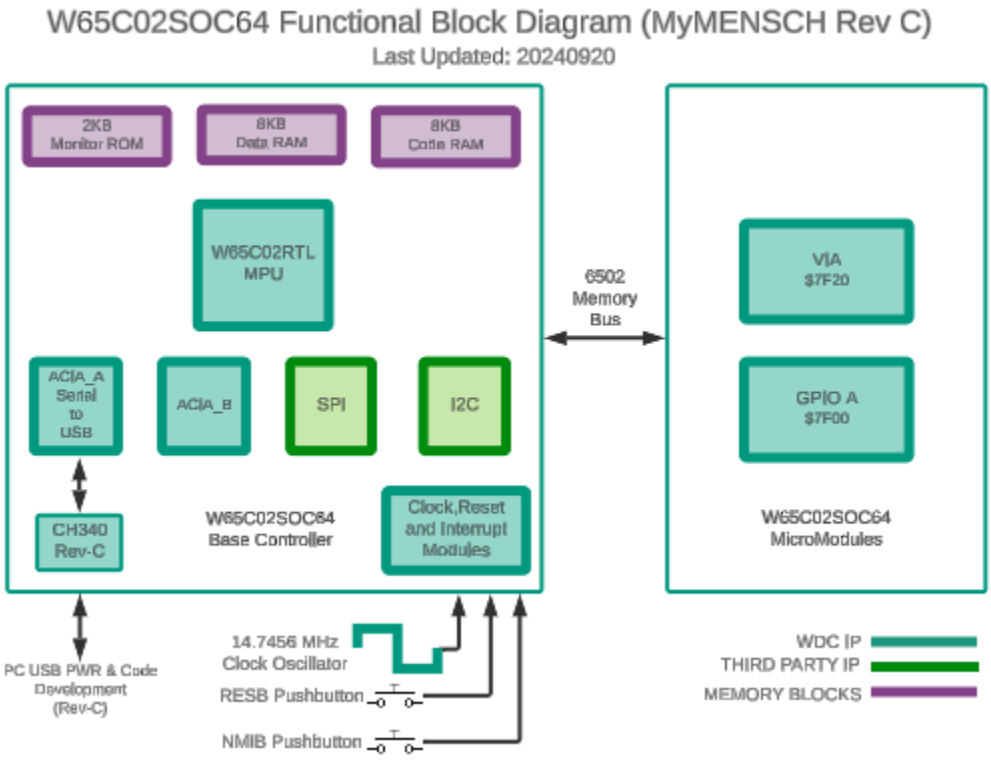
- ACIA_A with Handshake (on GPIOE) (4)
- ACIA_B without Handshake (on GPIOE) (2)
- I2C (2)
- SPI (4)

Micromodule Pins

- 20x VIA_A
- 8x ACIA_A

1.3 Functional Block Diagram

The following block diagram is for the W65C02SOC64.



2 MODULE DESCRIPTIONS

The following are descriptions of the basic modules.

2.1 CLOCK AND RESET MODULE

There is one main system clock (XCLK) aka PHI2. The Serial UART/ACIA will be created by dividing down XCLK. There are no Reset Module Registers and therefore no definitions; just logic to handle the reset functions for the system.

2.2 W65C02RTL Programming Model

Refer to the W65C02S Datasheet for the Microprocessor Programming Model, Status Register Coding and complete information. More information is found in ***Programming the 65816: Including the 6502, 65C02 and 65802*** Manual available through Amazon.

2.3 Priority Interrupt Controller Module Information

The Interrupt Control Module controls the priority and memory map for interrupts. Each interrupt is connected to the Interrupt Control Module for prioritizing.

Interrupt Enable Registers for the various interrupts are the interrupt enable by the various enable bits. Reading the various IER and IFR bits determines the interrupt that occurred. By prioritizing the interrupts one can determine which interrupt occurred in the associated interrupt handler routine. Notice that any of the 8 interrupts for a GPIO 8-bit port will cause a GPIO vectored interrupt to occur.

2.3.1 Priority Encoded Interrupt Vector Module

Vector Address	Label	Function
0xFFFE,F	IRQBRK	BRK – Software Interrupt
0xFFFC,D	IRQRES	RES – “REStart” Interrupt
0xFFFA,B	IRQNMI	Non-Maskable Interrupt/Hardware Breakpoint (HBP)
0xFFF8,9	IRQGPIOE_HS	GPIO Interrupt for UART Handshaking (for all 8 pins)
0xFFF6,7	IRQVIA_A	VIA Interrupt
0xFFF4,5	IRQSPI	SPI Interrupt
0xFFF2,3	IRQI2C	I2C Interrupt
0xFFF0,1	Reserved	Reserved
0xFFEE,F	Reserved	Reserved
0xFFEC,D	IRQGPIOA	GPIOA Interrupt

2.4 Memory Map

Start	End	Size	Description
0xF800	0xFFFF	2048 B	2048 Byte Monitor
0x8000	0x9FFF	8192 B	8K Byte Protected Bootloaded RAM
0x7FA8		1B	Memory Protect
0x7FA0	0x7FA3	4 B	GPIOE
0x7F80	0x7F8F	16 B	I2C
0x7F70	0x7F7F	16 B	SPI
0x7F64	0x7F67	4 B	ACIA_B
0x7F60	0x7F63	4 B	ACIA_A
0x7F30	0x7F5F	48 B	RESERVED
0x7F20	0x7F2F	16 B	VIA
0x7F14	0x7F1F	12 B	RESERVED
0x7F10	0x7F13	4 B	RESERVED
0x7F08	0x7F0B	4 B	RESERVED
0x7F00	0x7F03	4 B	GPIOA
0x0000	0x1FFF	8192 B	8K Byte SRAM

2.5 VIA Port Module

The W65C02SOC64 features one Versatile Interface Adapters (VIA) based on the W65C22S.
See Memory Map for base addresses. See W65C22S Datasheet for full register descriptions.

2.6 GPIO Port Modules

Four GPIO Port Modules are included in this design. These are noted as GPIOA, GPIOB, GPIOC, and GPIO E. All four of the GPIO ports use WDC's 5 Register (8-bit) GPIO module which supports edge sense interrupts and has a PIO Register (PIOx), Data Direction Register (DDRx), Interrupt Flag Register (IFRx), Interrupt Enable Register (IERx), and Edge Sense Register (ESRx). See Memory Map for base addresses.

GPIOA is for GPIO purposes and not used for any special controller purposes

GPIOE is intended to be used as handshake logic for the 2 ACIA modules and Chip Select Outputs for SPI.

2.6.1 GPIO Module Register Descriptions - 5 Register Version

Address = Base + 4		GPIO_ESR: GPIO Edge Sense Register					Reset Value = 0x00	
7:0->	ESR7	ESR6	ESR5	ESR4	ESR3	ESR2	ESR1	ESR0
Bit	Name	Access	Description					
7 – 0	ESR[7:0]	R/W	1 = Positive Edge Sense for PIO7-0					
			0 = Negative Edge Sense for PIO7-0					

Address = Base + 3		GPIO_IER: GPIO Interrupt Enable Register					Reset Value = 0x00	
7:0->	IER7	IER6	IER5	IER4	IER3	IER2	IER1	IER0
Bit	Name	Access	Description					
7 – 0	IER[7:0]	R/W	1 = Enable Interrupt on inputs for PIO7-0					
			0 = Disable Interrupts on inputs for PIO7-0					

Address = Base + 2		GPIO_IFR: GPIO Interrupt Flag Register					Reset Value = 0x00	
7:0->	IFR7	IFR6	IFR5	IFR4	IFR3	IFR2	IFR1	IFR0
Bit	Name	Access	Description					
7 – 0	IFR[7:0]	R/W	1 = Interrupt Occurred on inputs for PIO7-0					
			0 = Interrupts did not occur on inputs for PIO7-0					

Address = Base + 1		GPIO_DDR: GPIO Data Direction Register					Reset Value = 0x00	
7:0->	DDR7	DDR6	DDR5	DDR4	DDR3	DDR2	DDR1	DDR0
Bit	Name	Access	Description					
7 – 0	DDR[7:0]	R/W	1 = PIO data direction set to Output PIO7-0					
			0 = PIO data direction set to Input PIO7-0					

Address = Base		GPIO_DATA: GPIO Data Register					Reset Value = 0x00	
7:0->	PIO7	PIO6	PIO5	PIO4	PIO3	PIO2	PIO1	PIO0
Bit	Name	Access	Description					
7 – 0	PIO[7:0]	R/W	1 = PIO line is logic 1 value read and sets a 1 value on write for PIO7-0					
			0 = PIO line is logic 0 value read and sets a 0 value on write for PIO7-0					

2.7 ACIA Modules

The SoC has two Asynchronous Communications Interface Adapter (ACIA) modules used to transfer information to and from various communications modules such as LoRa, GSM, Bluetooth, Wi-Fi radio modules and UART enabled devices. See the Memory Map for base addresses. The baud rates are derived from 1.8432MHz XTLI input. In depth information for the W65C51 can be found in the datasheet.

The Micro USB connector is dual purpose. It is both the power connector that powers the board and the USB code port. The CH340 chip that interfaces to the connector has been pre-programmed so that when the board is power from a USB power on a computer, the chip will request 500mA of current from the host machine. The board can also be powered by any USB port that supplies 5V DC. Note that the board has 3v3 and 1v8 voltage regulators.

In addition to the power, the USB port serves as an interface to WDC's tool suite for debugging and loading programs into the onboard SRAM.

2.7.1 ACIA Register Descriptions

Address = Base + 3		ACIA_CTRL: ACIA Control Register					Reset Value = 0x00	
7:0->	SBN	WL1	WL0	RSC	SBR3	SBR2	SBR1	SBR0
HWRES	0	0	0	1	0	0	0	0
SWRES	-	-	-	1	-	-	-	-

Bit	Name	Access	Description
7	SBN	R/W	1 = 2 Stop bits, 1 ½ Stop bits for WL = 5, 1 Stop bit for WL = 8 and parity
			0 = 1 Stop bit
6	WL1	R/W	11 = 5 bits
			10 = 6 bits
5	WL0	R/W	01 = 7 bits
			00 = 8 bits
4	RSC	R/W	1 = Baud rate
			0 = RSC clock source
3	SBR3	R/W	1110 = 9600, 1111 = 19200
			1100 = 4800, 1101 = 7200
2	SBR2	R/W	1010 = 2400, 1011 = 3600
			1000 = 1200, 1001 = 1800
1	SBR1	R/W	0110 = 300, 0111 = 600
			0100 = 134.58, 0101 = 150
0	SBR0	R/W	0010 = 75, 0011 = 109.92
			0000 = 115.2K, 0001 = 50

Address = Base + 2	ACIA_CMR: ACIA Command Register						Reset Value = 0x00	
7:0->	PCM1	PCM0	PME	REM	TIC1	TIC0	IRD	DTR
HWRES	0	0	0	0	0	0	0	0
SWRES	-	-	-	0	0	0	0	0

Bit	Name	Access	Description
7	PCM1	R/W	11 = Space parity
			10 = Mark parity
6	PCM0	R/W	01 = Odd parity
			00 = Even parity
5	PME	R/W	1 = Parity enabled
			0 = Parity disabled
4	REM	R/W	1 = Receiver Echo Mode not available
			0 = Receiver Echo Mode not available
3	TIC1	R/W	11 = RTSB = low, Transmitter interrupt disabled, Transmit Break
			10 = RTSB = low, Transmitter interrupt disabled
2	TIC0	R/W	01 = RTSB = low, Transmitter interrupt enabled
			00 = RTSB = high, Transmitter interrupt disabled
1	IRD	R/W	1 = Receiver Interrupt Disabled
			0 = Receiver Interrupt Enabled
0	DTR	R/W	1 = Data Terminal Ready
			0 = Data Terminal Transmitter Not Ready

Address = Base + 1	ACIA_STR: ACIA Status Register						Reset Value = 0x10	
7:0->	IRQ	DSRB	DCDB	TDRE	RDRF	OVRN	FE	PE
HWRES	0	0	0	1	0	0	0	0
SWRES	-	-	-	1	-	-	-	-

Bit	Name	Access	Description
7	IRQ	R/O	1 = Interrupt has occurred
			0 = No Interrupt
6	DSRB	R/O	1 = Not ready and not clear to send data
			0 = Ready and clear to send data
5	DCDB	R/O	1 = DCD Not Detected
			0 = DCD Detected
4	TDRE	R/O	1 = Empty
			0 = Not Empty
3	RDRF	R/O	1 = Full
			0 = Not Full
2	OVRN	R/O	1 = Overrun has occurred
			0 = No overrun
1	FE	R/O	1 = Framing error detected
			0 = No framing error
0	PE	R/O	1 = Parity error detected
			0 = No parity error
Address = Base + 1		W/O	Program Reset aka SWRES

Address = Base	ACIA_DR: ACIA Data Register						Reset Value = 0x00	
7:0->	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0

Bit	Name	Access	Description
7 - 0	DR[7-0]	R/W	R = Read Receiver Data Register
			W = Write Transmitter Data Register

2.8 I2C Interface Module

For the I2C register descriptions and I2C Operation, refer to the "I2C Design Specification".

2.8.1 I2C Status Register Definitions

Address = Base + 5		SR: I2C Status Register					Reset Value = 0x00	
7:0->	RxACK	WCOL	BUSY	RESERVED	WFFULL	WFEMPTY	RFFULL	RFEMPTY

Bit	Name	Access	Description
7	RxACK	R/O	1 = No acknowledge received
			0 = Acknowledge received
6	BUSY	R/W	1 = After start I2C bus busy signal detected
			0 = After stop I2C bus busy signal detected
5	RSRVD	R/O	1 = Never
			0 = Always
4	RSRVD	R/O	1 = Never
			0 = Always
3	RSRVD	R/O	1 = Never
			0 = Always
2	RSRVD	R/O	1 = Never
			0 = Always
1	TIP	R/O	1 = Transfer in progress when transferring data
			0 = When transfer complete
0	IF	R/O	1 = Interrupt is set when one byte is transferred, processor interrupt request if IEN bit is set.
			0 = No interrupt

2.8.2 I2C Command Register Definitions

Address = Base + 4		CR: I2C Command Register					Reset Value = 0x00	
7:0->	STA	STO	RD	WR	ACK	RSVRD	RSVRD	IACK

Bit	Name	Access	Description
7	STA	R/W	1 = Generate start condition
			0 = Do not generate start condition
6	STO	R/W	1 = Generate stop condition
			0 = Do not generate stop condition
5	RD	R/W	1 = Read from slaver
			0 = Do not read from slave
4	WR	R/W	1 = Write slave
			0 = Do not write slave
3	ACK	R/W	1 = NACK
			0 = ACK
2	RSVRD	R/W	1 = Never
			0 = Always
1	RSVRD	R/W	1 = Never
			0 = Always
0	IACK	R/W	1 = Clear a pending interrupt
			0 = Don’t clear a pending interrupt

2.8.3 I2C Receive Register Definitions

Last byte received via I2C.

Address = Base + 3		RXR: I2C Receive Register					Reset Value = 0x00	
7:0->	RXR7	RXR6	RXR5	RXR4	RXR3	RXR2	RXR1	RXR0

Bit	Name	Access	Description
7 - 0	RXR[7-0]	R/O	**R = Read Receiver Data Register**
			W = no operation

2.8.4 I2C Transmit Register Definitions

7:1 RW Next byte to transmit via I2C 0 RW In case of a data transfer this bit represent the data's LSB. In case of a slave address transfer this bit represents the RW bit. '1' for reading from slave '0' for writing to slave

Address = Base + 2		TXR: I2C Transmit Register					Reset Value = 0x00	
7:0->	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0
Bit	**Name**	**Access**	**Description**					
7 - 0	TXR[7-0]	R/W	R = Read Receiver Data Register					
			W = Write Transmitter Data Register					

2.8.5 I2C Control Register

The core responds to new commands only when the 'EN' bit is set. Pending commands are finished. Clear the 'EN' bit only when no transfer is in progress, i.e. after a STOP command, or when the command register has the STO bit set. When halted during a transfer, the core can hang the I2C bus.

Address = Base +1		CTR: I2C Control Register					Reset Value = 0x10	
7:0->	EN	IEN	RSRVD	RSRVD	RSRVD	RSRVD	RSRVD	RSRVD
Bit	**Name**	**Access**	**Description**					
7	EN	R/W	1 = I2C Core enabled					
			0 = I2C Core disabled					
6	IEN	R/W	1 = I2C Core interrupt enabled					
			0 = I2C Core interrupt disabled					
5	RSRVD	R/W	1 = Never					
			0 = Always					
4	RSRVD	R/W	1 = Never					
			0 = Always					
3	RSRVD	R/W	1 = Never					
			0 = Always					
2	RSRVD	R/W	1 = Never					
			0 = Always					
1	RSRVD	R/W	1 = Never					
			0 = Always					
0	RSRVD	R/W	1 = Never					
			0 = Always					

2.8.6 I2C Clock Prescale Register Definitions

For the I2C register descriptions, refer to the "I2C Design Specification".

This register is used to prescale the SCL clock line. Due to the structure of the I2C interface, the core uses a 4*SCL clock internally. The prescale register must be programmed to this 4*SCL bitrate. Change the value of the prescale register only when the 'EN' bit is cleared.
Example: CLK_I = 32MHz, desired SCL = 100 KHz
Prescale = 32MHZ = 80 (dec) = 50 (hex) 4 * 100 KHz
Reset value: 0xFFFF

Address = Base		PRER: I2C Clock Prescale Register					Reset Value = 0xFF	
7:0->	PRER7	PRER6	PRER5	PRER4	PRER3	PRER2	PRER1	PRER0
Bit	**Name**	**Access**	**Description**					
7 - 0	PRER[7-0]	R/W	R = Read Receiver Data Register					
			W = Write Transmitter Data Register					

2.9 SPI Module

The SPI module described in the standard SPI Specification found in this link.

2.9.1 SPI Extension Register

Address = Base + 3		SPER: SPI Extension Register					Reset Value = 0x00	
7:0->	ICNT1	ICNT0	RESERVED	RESERVED	RESERVED	RESERVED	ESPR1	ESPR0

Bit	Name	Access	Description
7	ICNT1	R/W	11 = SPIF is set after every four completed transfers
			10 = SPIF is set after every three completed transfers
6	ICNT0	R/W	01 = SPIF is set after every two completed transfers
			00 = SPIF is set after every completed transfer
5	RESERVED	R/W	1 = Never
			0 = Always
4	RESERVED	R/W	1 = Never
			0 = Always
3	RESERVED	R/W	1 = Never
			0 = Always
2	RESERVED	R/W	1 = Never
			0 = Always
1	ESPR1	R/W	11 = Reserved, do not use
			10 = Add these two bits to the SPI Clock Rate 0=512, 01=1024, 02=2048, 03=4096
0	ESPR0	R/W	01 = Add these two bits to the SPI Clock Rate 0=8, 01=64, 02=128, 03=256
			00 = Add these two bits to the SPI Clock Rate 00=2, 01=4, 02=16, 03=32

2.9.2 SPI Data Register

Address = Base + 2		SPDR: SPI Data Register					Not Initialized on Reset	
7:0->	SPDR7	SPDR6	SPDR5	SPDR4	SPDR3	SPDR2	SPDR1	SPDR0

Bit	Name	Access	Description
7 - 0	SPDR[7-0]	R/W	R = Read SPI Data buffer
			W = Write SPI Data buffer

2.9.3 SPI Status Register

Address = Base + 1		SPSR: SPI Status Register							Reset Value = 0x05
7:0->	SPIF	WCOL	RESERVED	RESERVED	WFFULL	WFEMPTY	RFFULL	RFEMPTY	
Bit	**Name**	**Access**	**Description**						
7	SPIF	R/W	1 = SPI Interrupt Flag is set on completion of a transfer block						
			0 = SPI not interrupting						
6	WCOL	R/W	1 = SPI Core write collision when SPI data register when Write FIFO is full						
			0 = SPI Core disabled						
5	RESERVED	R/O	1 = Never						
			0 = Always						
4	RESERVED	R/O	1 = Never						
			0 = Always						
3	WFFULL	R/O	1 = Write FIFO full						
			0 = Write FIFO not full						
2	WFEMPTY	R/O	1 = Write FIFO empty						
			0 = Write FIFO not empty						
1	RFFULL	R/O	1 = Read FIFO full						
			0 = Read FIFO not full						
0	RFEMPTY	R/O	1 = Read FIFO empty						
			0 = Read FIFO not empty						

2.9.4 SPI Control Register

Address = Base		SPCR: SPI Control Register					Reset Value = 0x10	
7:0->	SPIE	SPE	RESERVED	MSTR	CPOL	CPHA	SPR1	SPR0
Bit	**Name**	**Access**	**Description**					
7	SPIE	R/W	1 = SPI Interrupt Enabled					
			0 = SPI Interrupt Disabled					
6	SPE	R/W	1 = SPI Core enabled					
			0 = SPI Core disabled					
5	RESERVED	R/W						
4	MSTR	R/W	1 = Master					
			0 = Slave					
3	CPOL	R/W	1 = Negative Clock Polarity					
			0 = Positive Clock Polarity					
2	CPHA	R/W	1 = Clock Phase Not Shifted					
			0 = Clock Phase Shifted					
1	SPR1	R/W	These values are used with the ESPR bits to determine the extended clock rate.					
			Refer to the SPI Datasheet for detailed selection information.					
0	SPR0	R/W	These values are used with the ESPR bits to determine the extended clock rate.					
			Refer to the SPI Datasheet for detailed selection information.					

3 Pinouts

3.1 Left IO Connector J3 on MyMENSCH™

The J3 left connector has 46 IO, 2x 3v3 power and 2x VSS pins. Ball assignments labeled NA are Non-Assigned pins.

J3 – Left Expansion Connector					
Pin	***Signal Name***	***FPGA Ball***	***Pin***	***Signal Name***	***FPGA Ball***
1	VSS	-	2	VDD	-
3	I2C_SCL	L4	4	NA	L3
5	I2C_SDA	K6	6	NA	K5
7	NA	H4	8	NA	N2
9	VIA_A_CB2	M4	10	NA	N3
11	VIA_A_CB1	M5	12	NA	N4
13	VIA_A_PB7	L5	14	NA	N5
15	VIA_A_PB6	N7	16	NA	N6
17	VIA_A_PB5	N8	18	NA	M7
19	VIA_A_PB4	M9	20	NA	M8
21	VIA_A_PB3	M10	22	NA	N9
23	VIA_A_PB2	M11	24	NA	N10
25	VIA_A_PB1	N12	26	NA	N11
27	VIA_A_PB0	M13	28	NA	M12
29	VIA_A_PA7	L13	30	NA	L12
31	VIA_A_PA6	K13	32	NA	K12
33	VIA_A_PA5	K8	34	NA	J8
35	VIA_A_PA4	J9	36	NA	L10
37	VIA_A_PA3	K10	38	NA	L11
39	VIA_A_PA2	K11	40	NA	J10
41	VIA_A_PA1	J12	42	NA	H10
43	VIA_A_PA0	K7	44	NA	J13
45	VIA_A_CA1	H13	46	NA	G12
47	VIA_A_CA2	G13	48	NA	F12
49	VDD	-	50	VSS	-

3.2 Right IO Connector J4 on MyMENSCH™

The J4 right connector has 46 IO, 2x 3v3 power and 2x VSS pins. Ball assignments labeled NA are Non-Assigned pins. The ADC inputs are not used for this build and should not be connected to anything.

J4 – Right Expansion Connector					
Pin	***Signal Name***	***FPGA Ball***	***Pin***	***Signal Name***	***FPGA Ball***
1	VDD	-	2	VSS	-
3	NA	H3	4	AGND	(G2)
5	NA	H1	6	ADC_IN0	(F5)
7	NA	H2	8	3v3REF	(F6)
9	DNU/ADC_IN5	F1	10	5vBAT	(G1)
11	DNU/ADC_IN6	E1	12	NA	(E5)
13	DNU/ADC_IN7	C1	14	DNU/ADC_IN1	D1
15	DNU/ADC_IN8	B1	16	DNU/ADC_IN2	C2
17	NA	B2	18	DNU/ADC_IN3	E3
19	NA	A2	20	DNU/ADC_IN4	E4
21	NA	B3	22	XCLK	E6
23	NA	B4	24	UART_CS_E4	A3
25	NA	B5	26	SPI_CS0_E5	A4
27	NA	B6	28	SPI_SDI	A5
29	NA	B7	30	SPI_SDO	A6
31	NA	A7	32	SPI_SCLK	D9
33	NA	A8	34	NA	E8
35	NA	C9	36	NA	F8
37	NA	C10	38	NA	A9
39	NA	B10	40	NA	A10
41	NA	B11	42	TXD_B	A11
43	NA	B12	44	RXD_B	A12
45	NA	B13	46	RTSB_B_E6	C11
47	NA	C13	48	CTSB_B_E2	C12
49	VSS	-	50	VDD	-

3.3 W65C02SOC64 QFN Pinout

Pin	WDC QFN64 Signal	Openframe Signal	Pin	WDC QFN64 Signal	Openframe Signal
1	N/C	vssa2	33	GPIO_A0	mprj_io[2]
2	VIA_CB2	mprj_io[25]	34	GPIO_A1	mprj_io[3]
3	VIA_CB1	mprj_io[26]	35	GPIO_A2	mprj_io[4]
4	VIA_PB7	mprj_io[27]	36	GPIO_A3	mprj_io[5]
5	VIA_PB6	mprj_io[28]	37	GPIO_A4	mprj_io[6]
6	VIA_PB5	mprj_io[29]	38	N/C	vssa1
7	VIA_PB4	mprj_io[30]	39	N/C	vssd1
8	VIA_PB3	mprj_io[31]	40	N/C	vdda1
9		vdda2	41	GPIO_A5	mprj_io[7]
10		vssd2	42	GPIO_A6	mprj_io[8]
11	VIA_PB2	mprj_io[32]	43	GPIO_A7	mprj_io[9]
12	VIA_PB1	mprj_io[33]	44	IRQB_EXT	mprj_io[10]
13	VIA_PB0	mprj_io[34]	45	RTSB_A_E7	mprj_io[11]
14	RXD_B	mprj_io[35]	46	CTSB_A_E3	mprj_io[12]
15	TXD_B	mprj_io[36]	47	N/C	vdda1
16	UART_CS0_E4	mprj_io[37]	48	RXD_A	mprj_io[13]
17	VDD3V3	vddio	49		vccd1
18	VDD1V8	vccd	50	TXD_A	mprj_io[14]
19	N/C	N/C	51	VIA_PA0	mprj_io[14]
20	N/C	vssa	52	N/C	vssa1
21	RESB	resetb	53	VIA_PA1	mprj_io[14]
22	PHI2	mprj_io[38]	54	VIA_PA2	mprj_io[14]
23	VSS	vssd	55	VIA_PA3	mprj_io[14]
24	SPI_CS0_E5	mprj_io[39]	56	VSS	vssio
25	SPI_SCLK	mprj_io[40]	57	VIA_PA4	mprj_io[14]
26	SPI_SDI	mprj_io[41]	58	VIA_PA5	mprj_io[14]
27	SPI_SDO	mprj_io[42]	59	VIA_PA6	mprj_io[14]
28	NMIB	mprj_io[43]	60	VIA_PA7	mprj_io[14]
29	VSS	vssio	61	VIA_CA1	mprj_io[14]
30	N/C	vdda	62	VIA_CA2	mprj_io[14]
31	I2C_SCL	mprj_io[0]	63	N/C	vccd2
32	I2C_SDA	mprj_io[1]	64	VDD3V3	vddio

ANALOG
DIGITAL
Caravel Std vs Openframe
QFN64 Corner
W65C02SOC Base Controller
MicroModule IO Pins

4 FCC Compliance

The Western Design Center, Inc. (WDC) provides the enclosed product under the following conditions: This board is intended for use for Education, Engineering Development or Evaluation Purposes ONLY and is not considered by WDC to be a finished consumer product. This board should be handled with caution using good electronics handling practices. This board is compliant per RoHS/Green directives. It does not fall within the scope of directives such as FCC, CE, and UL. It generates uses and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC rules.

5 Ordering Information

The W65C02SOC64 is ordered through the MPW tapeout service provider. WDC's first trial tape out will be with Efabless and targeting the 130nm SkyWater process.