MDIO Advanced Example

1.0

Features

- Demonstrates the MDIO Interface component in Advanced mode
- Demonstrates how to connect the MDIO Interface component to an MDIO host device
- Shows how to configure the device to automatically process the received MDIO frame
- Configured to run on the PSoC Development Kit (CY8CKIT-001)

General Description

This example project demonstrates the advanced operation of the MDIO Interface component. The advanced mode allows the user to have full customizability of the register spaces through the component customizer and allows the component hardware to take care of the data transfers, memory accesses and frame error handling.

Development Kit Configuration

1. The PSoC 3 or PSoC 5LP Processor Module must be installed on the PSoC Development Kit (DVK)
2. Configure the pin connections as shown on the project schematic
3. Apply power to the PSoC DVK
4. Build the project and program the CY8C3866AXI-040 using a MiniProg3 programmer
5. Observe the frame information on the LCD

Project Configuration

The example project consists of the MDIO Interface and the Character LCD components. The project assumes that an MDIO host device is available and that necessary pin connections can be made to the Development Kit. The design schematic is shown in Figure 1. The blue annotation components in the schematic show the external connections required for correct component operation. This requires a (recommended) 330 Ohm pull-up resistor on the MDIO_S data and MDC_S clock line to a 1.2V supply, and a common ground connection between the host and the interface.

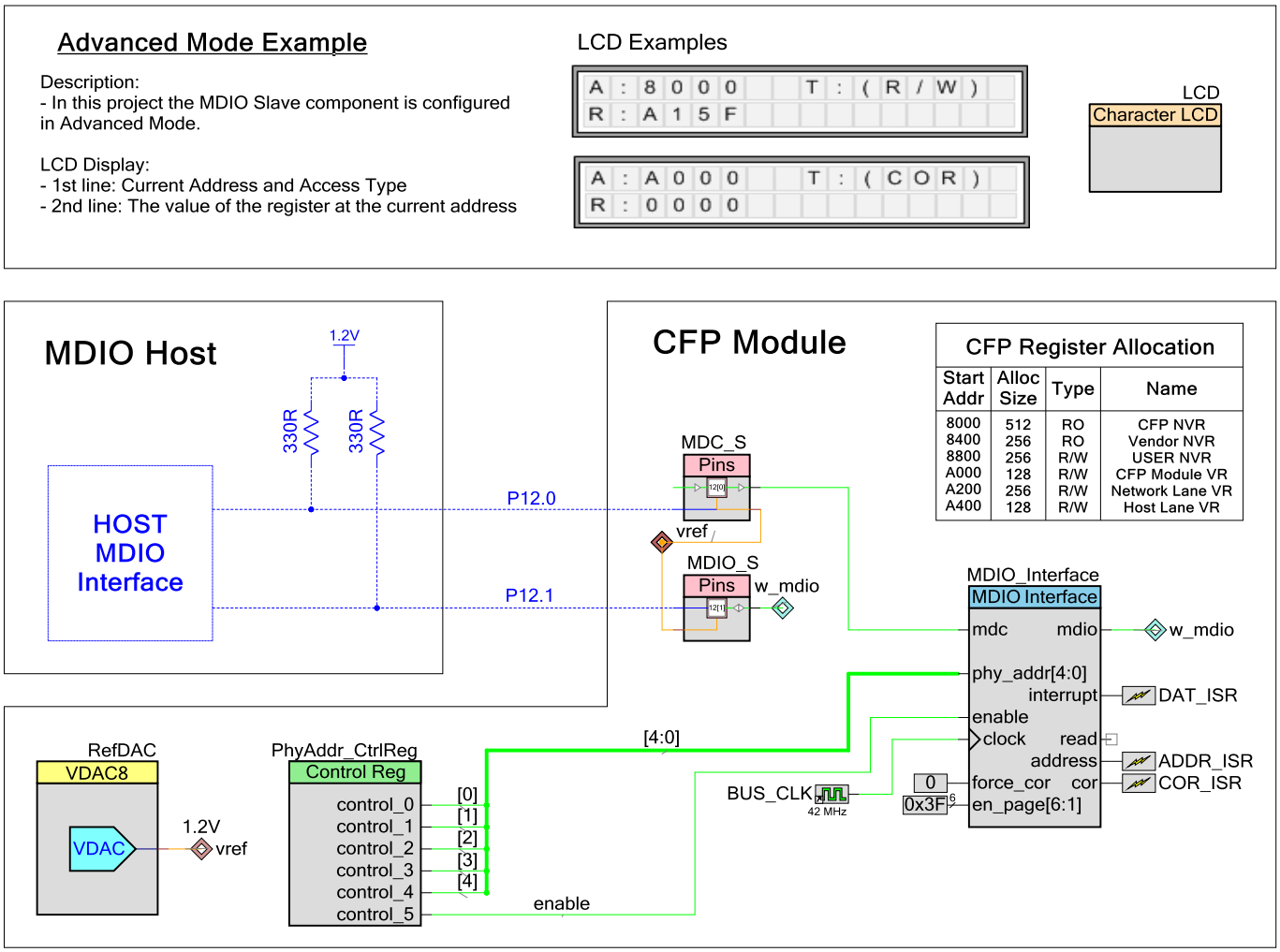

Figure 1. Example project design schematic

Pin configuration

The pins are configured as shown in the table below. Aside from these, the MDIO_S line needs to be connected to a 1.2V supply through an external 330 Ohm resistor, and the ground connection of the Development Kit must be tied to the MDIO Host. A voltage reference signal used for the MDIO I/O pins is generated by a **VDAC** placed in the top design, which is configured to create 1.2 Volts voltage reference.

Pin Name	Port	Description
MDC_S	P12[0]	The MDC clock signal connected to the MDIO host's MDC terminal
MDIO_S	P12[1]	Terminal for the MDIO data signal
LCD:LCDPort[6:0]	P2[6:0]	Connection to the Character LCD on the Dev. Kit

Customizer settings information

The customizer for the MDIO Interface component in Advanced mode requires configuration in the General tab, and the Register Spaces tab.

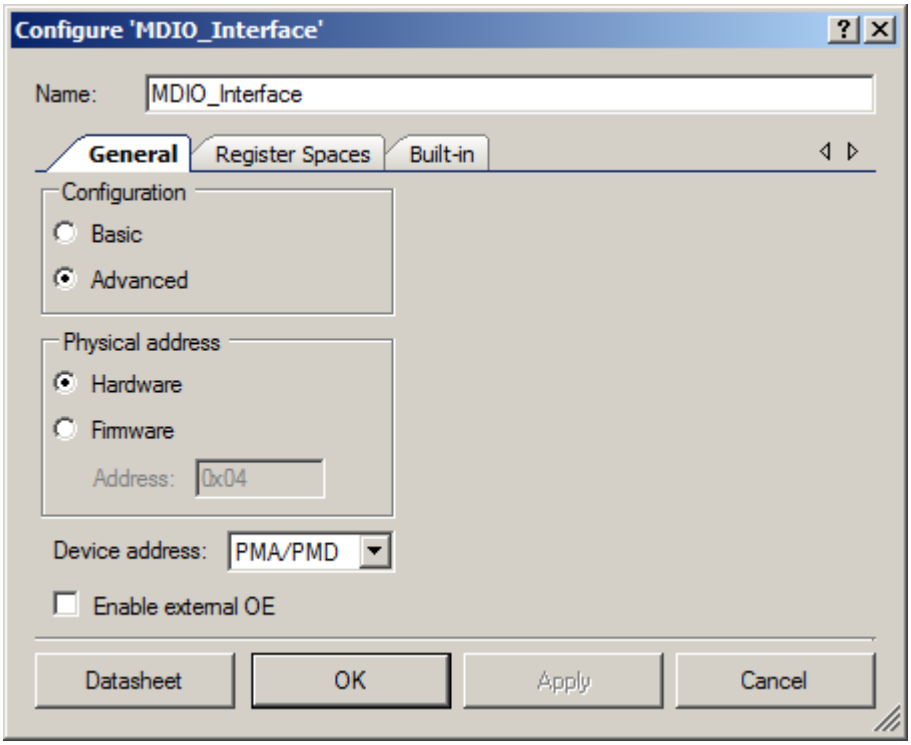

Figure 2. MDIO Interface component configuration - General tab

In the General tab, the component is set to Advanced mode, the physical address is configured to Hardware, and the device address is set to PMA/PMD. The Enable external OE checkbox is unchecked.

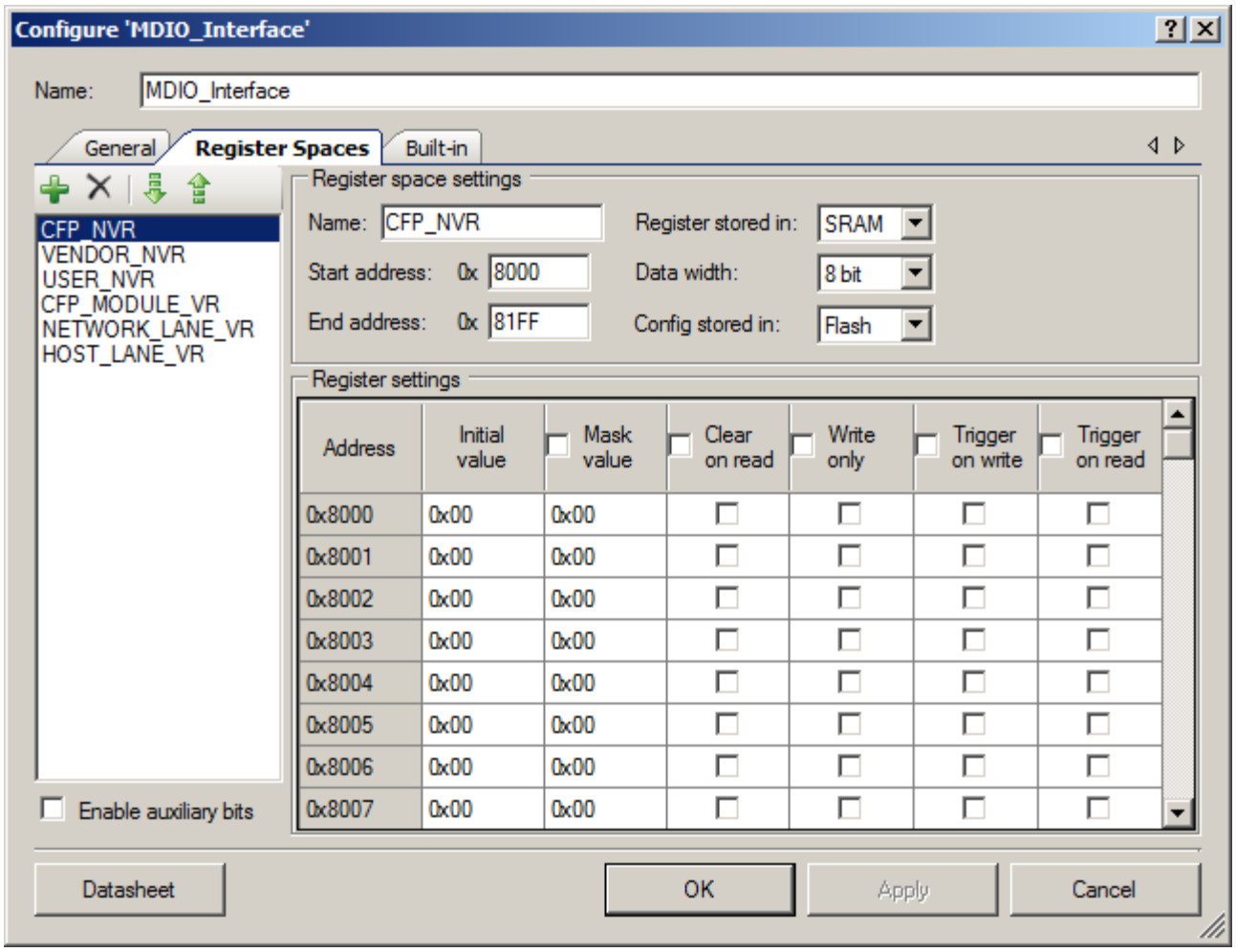

Figure 3. MDIO Interface component configuration – Register Spaces tab

The Register Spaces tab allows the user to configure the register spaces. These include the start and end addresses of the Register Space, read only (configured by setting the Register Mask to be zero), the data width and various other characteristics that are configured for the individual registers. In this example, The Register Spaces are configured as shown on the design page schematic.

Project Description

In the main firmware routine, the MDIO Interface component populates the CFP NVR and Vendor NWR register spaces with random data. When a frame is received from the MDIO host, the frame information and the address access status are checked. The address, data, and the register type information are displayed on the Character LCD.

Expected Results

The LCD is used to display information about the data exchanged between the MDIO Host and MDIO Interface. The Figure 4 shows the meaning of each field in the screen (all values are shown in hexadecimal).

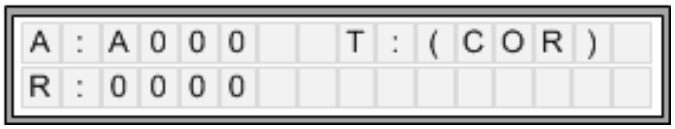

Figure 4. LCD Display Information

The current address being accessed by the MDIO Host is always displayed in the first line of the LCD. The register that belongs to a specific MDIO address can have 4 different access types when accessed by the MDIO Host:

- Read Only (RO)
- Read and Write (R/W)
- Clear on Read (COR)
- Not accessible (N/A)

All the registers located in SRAM are writable by the MDIO Interface.

The second line displays information about the register stored in the memory.

© Cypress Semiconductor Corporation, 2009-2013. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC® is a registered trademark, and PSoC Creator™ and Programmable System-on-Chip™ are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.