Application Report

SPRAAN3–July 2007

TMS320DM643x Pin Multiplexing Utility

Bernard Thompson

ABSTRACT

The DM643x devices use a great deal of internal pin multiplexing to allow the most functionality in the smallest and lowest cost package. The software accompanying this application report allows the pin multiplexing registers of the device to be calculated with ease, as well as showing what peripherals can be used together and what devices of the DM643x family support the peripherals that are selected. This software is useful to anyone creating a system with a DM643x device.

This application report contains project code that can be downloaded from http://www.ti.com/lit/zip/SPRAAN3.

Contents

List of Figures

1 System Requirements

The DM643x pin multiplexing utility was designed and tested on a Windows® XP operating system, however due to the relative simplicity of the utility it can function in other versions of Windows® as well. This utility requires the Microsoft® .net framework version 2.0; this can be downloaded from Microsoft at http://msdn2.microsoft.com/en-us/netframework/aa731542.aspx. It is suggested to use this utility with a screen resolution of 1024×768 or higher.

2 Using the DM643x Pin Multiplexing Utility

This utility is a stand alone executable, there is no installation process required. Once you have opened the utility there is a user interface with various sections covering the multiplexed peripherals of the DM643x devices. Simply select the peripherals necessary to your application. Note that as some peripherals share pins, when selecting one peripheral you may disable another. This way you can determine how many different combinations of peripherals are possible on the DM643x devices. As the peripherals are selected, the utility dynamically updates which DM643x family devices have selected the peripherals and the appropriate PINMUX register values.

Once the selections have been entered, the PINMUX register values are used in the application code to properly configure the device; all the peripherals that were selected are enabled. If the PINMUX register writes are to be done in the C environment then use the register level chip support library (CSL) provided in the peripheral support package. The code would be as shown below:

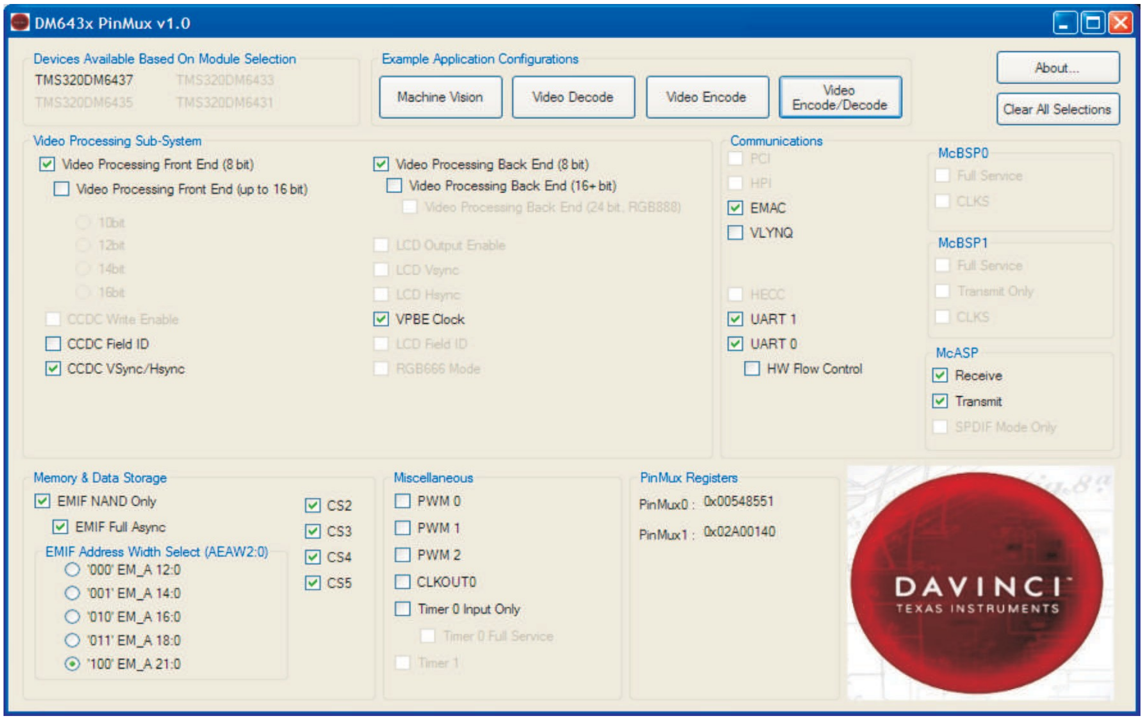
```
sysCtlRegs->PINMUX0 = 0x12345678 //where 0x12345678 is the PINMUX0 value generated
sysCtlRegs->PINMUX1 = 0x12345678 //where 0x12345678 is the PINMUX1 value generated
```

3 Example Pin Multiplexing Configurations

This utility also includes four example application configurations which provide a set of selected peripherals for a particular application. These can be selected by simply clicking on one of the buttons in the example application configurations box. Once an example configuration has been selected it can be modified to suit the needs of a specific application. The video encode/decode example provides a step-by-step guide showing how to use the pin multiplexing utility to create it.

3.1 Video Encode/Decode

The video encode/decode example configuration provides both video capture and display capabilities, which could be used in applications such as digital video recorders where both the capture of the video and display is necessary. As this is using both the video port front end (VPFE) and video port back end (VPBE), this is a good application example for the DM6437 device. This configuration uses basic interfaces for both the VPFE and VPBE as well as providing the full asynchronous external memory interface (EMIF). Additionally, the Ethernet media access controller (EMAC) is enabled for network connectivity, and the multichannel audio serial port (McASP) is enabled fully to allow for audio input and output as can be seen in Figure 1.



Devices Available Based On Module Selection
TMS320DM6437
TMS320DM6433
TMS320DM6435
TMS320DM6431

Video Processing Sub-System
☑ Video Processing Front End (8 bit)
□ Video Processing Front End (up to 16 bit)
○ 10bit
○ 12bit
○ 14bit
○ 16bit
□ CCDC Write Enable
□ CCDC Field ID
☑ CCDC VSync/Hsync

☑ Video Processing Back End (8 bit)
□ Video Processing Back End (16+ bit)
□ Video Processing Back End (24 bit, RGB888)
□ LCD Output Enable
□ LCD Vsync
□ LCD Hsync
☑ VPBE Clock
□ LCD Field ID
□ RGB666 Mode

Communications
□ PCI
□ HPI
☑ EMAC
□ VLYNQ
□ HECC
☑ UART 1
☑ UART 0
□ HW Flow Control

McBSP0
□ Full Service
□ CLKS

McBSP1
□ Full Service
□ Transmit Only
□ CLKS

McASP
☑ Receive
☑ Transmit
□ SPDIF Mode Only

Memory & Data Storage
☑ EMIF NAND Only
☑ EMIF Full Async
EMIF Address Width Select (AEAW2:0)
○ '000' EM_A 12:0
○ '001' EM_A 14:0
○ '010' EM_A 16:0
○ '011' EM_A 18:0
● '100' EM_A 21:0
☑ CS2
☑ CS3
☑ CS4
☑ CS5

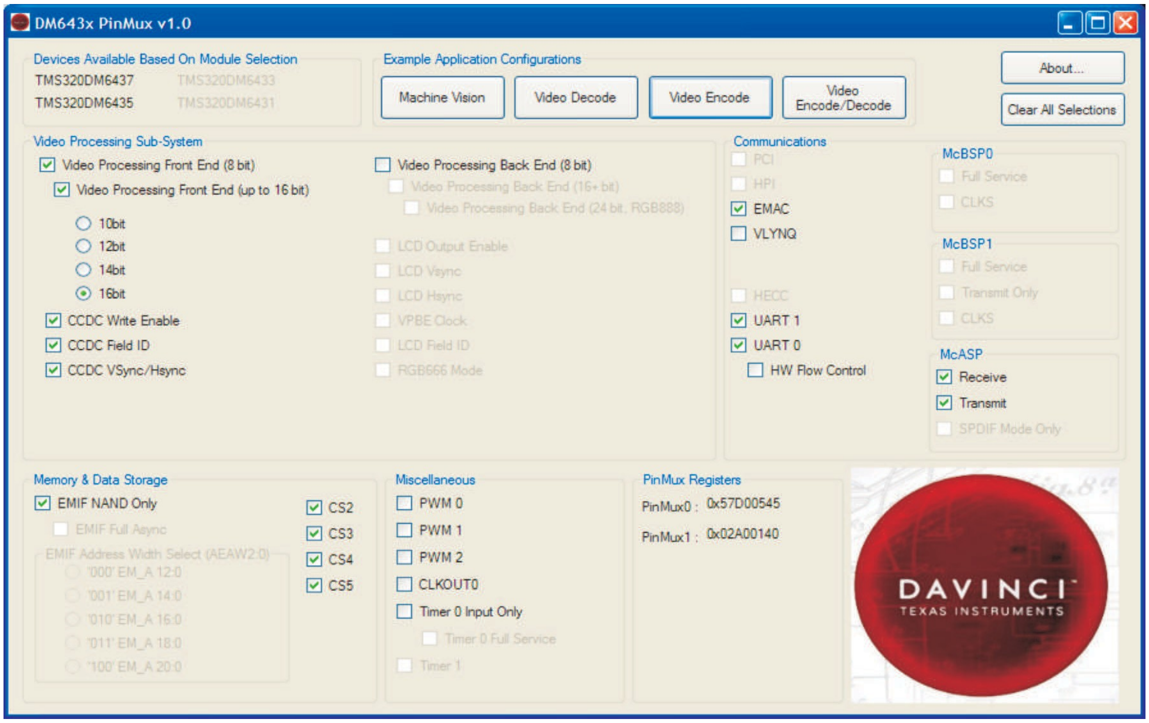
Miscellaneous
□ PWM 0
□ PWM 1
□ PWM 2
□ CLKOUT0
□ Timer 0 Input Only
□ Timer 0 Full Service
□ Timer 1

PinMux Registers
PinMux0 : 0x00548551
PinMux1 : 0x02A00140

Figure 1. Video Encode/Decode Configuration

3.2 Video Encode

The video encode configuration provides a full featured VPFE to allow for a high bandwidth and flexible video capture interface. This would be found in applications such as webcams and video security where capturing high-quality video for storage or transmission is necessary. As this is using just the VPFE and requires additional processing power, it is a good example application for the DM6435 as a lower cost alternative to the DM6437. In this case, the asynchronous EMIF is given up to provide a more versatile VPFE with the write enable signal allowing a versatile interface for communication with imagers. The EMAC is enabled to allow the broadcast of the encoded video signal as well as a full McASP for capturing an audio stream as can be seen in Figure 2.



Devices Available Based On Module Selection

TMS320DM6437 TMS320DM6433
TMS320DM6435 TMS320DM6431

Example Application Configurations

Video Processing Sub-System

- ☑ Video Processing Front End (8 bit)
- ☑ Video Processing Front End (up to 16 bit)
- ○ 10bit
- ○ 12bit
- ○ 14bit
- ● 16bit
- ☑ CCDC Write Enable
- ☑ CCDC Field ID
- ☑ CCDC VSync/Hsync
- □ Video Processing Back End (8 bit)
- □ Video Processing Back End (16+ bit)
- □ Video Processing Back End (24 bit, RGB888)
- □ LCD Output Enable
- □ LCD Vsync
- □ LCD Hsync
- □ VPBE Clock
- □ LCD Field ID
- □ RGB666 Mode

Communications

- □ PCI
- □ HPI
- ☑ EMAC
- □ VLYNQ
- □ HECC
- ☑ UART 1
- ☑ UART 0
- □ HW Flow Control

McBSP0

- □ Full Service
- □ CLKS

McBSP1

- □ Full Service
- □ Transmit Only
- □ CLKS

McASP

- ☑ Receive
- ☑ Transmit
- □ SPDIF Mode Only

Memory & Data Storage

- ☑ EMIF NAND Only
- □ EMIF Full Async
- EMIF Address Width Select (AEAW2:0)
- ○ '000' EM_A 12:0
- ○ '001' EM_A 14:0
- ○ '010' EM_A 16:0
- ○ '011' EM_A 18:0
- ○ '100' EM_A 20:0
- ☑ CS2
- ☑ CS3
- ☑ CS4
- ☑ CS5

Miscellaneous

- □ PWM 0
- □ PWM 1
- □ PWM 2
- □ CLKOUT0
- □ Timer 0 Input Only
- □ Timer 0 Full Service
- □ Timer 1

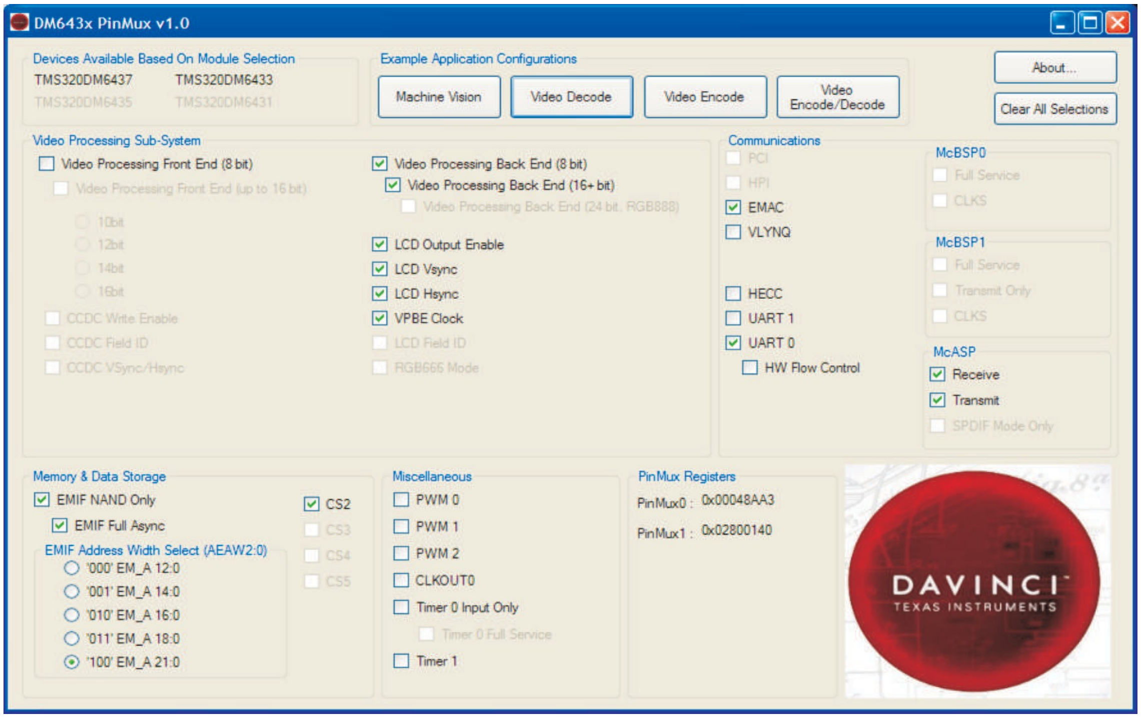
PinMux Registers

PinMux0 : 0x57D00545
PinMux1 : 0x02A00140

Figure 2. Video Encode Configuration

3.3 Video Decode

The video decode configuration uses a full featured VPBE interface for a display only application, something that would be found in a basic media player. As this is a display only system, the DM6433 could be implemented to support the VPBE at the lowest cost. In this configuration, a full asynchronous EMIF is used to provide a more expansive storage capability. The EMAC is enabled to allow for network connectivity providing a potential source for a compressed video stream. The McASP is set up to provide output of an audio stream with the decoded video as can be seen in Figure 3.



Devices Available Based On Module Selection
TMS320DM6437 TMS320DM6433
TMS320DM6435 TMS320DM6431

Video Processing Sub-System
□ Video Processing Front End (8 bit)
□ Video Processing Front End (up to 16 bit)
○ 10bit
○ 12bit
○ 14bit
○ 16bit
□ CCDC Write Enable
□ CCDC Field ID
□ CCDC VSync/Hsync
☑ Video Processing Back End (8 bit)
☑ Video Processing Back End (16+ bit)
□ Video Processing Back End (24 bit, RGB888)
☑ LCD Output Enable
☑ LCD Vsync
☑ LCD Hsync
☑ VPBE Clock
□ LCD Field ID
□ RGB666 Mode

Communications
□ PCI
□ HPI
☑ EMAC
□ VLYNQ
□ HECC
□ UART 1
☑ UART 0
□ HW Flow Control

McBSP0
□ Full Service
□ CLKS

McBSP1
□ Full Service
□ Transmit Only
□ CLKS

McASP
☑ Receive
☑ Transmit
□ SPDIF Mode Only

Memory & Data Storage
☑ EMIF NAND Only
☑ EMIF Full Async
EMIF Address Width Select (AEAW2:0)
○ '000' EM_A 12:0
○ '001' EM_A 14:0
○ '010' EM_A 16:0
○ '011' EM_A 18:0
● '100' EM_A 21:0
☑ CS2
□ CS3
□ CS4
□ CS5

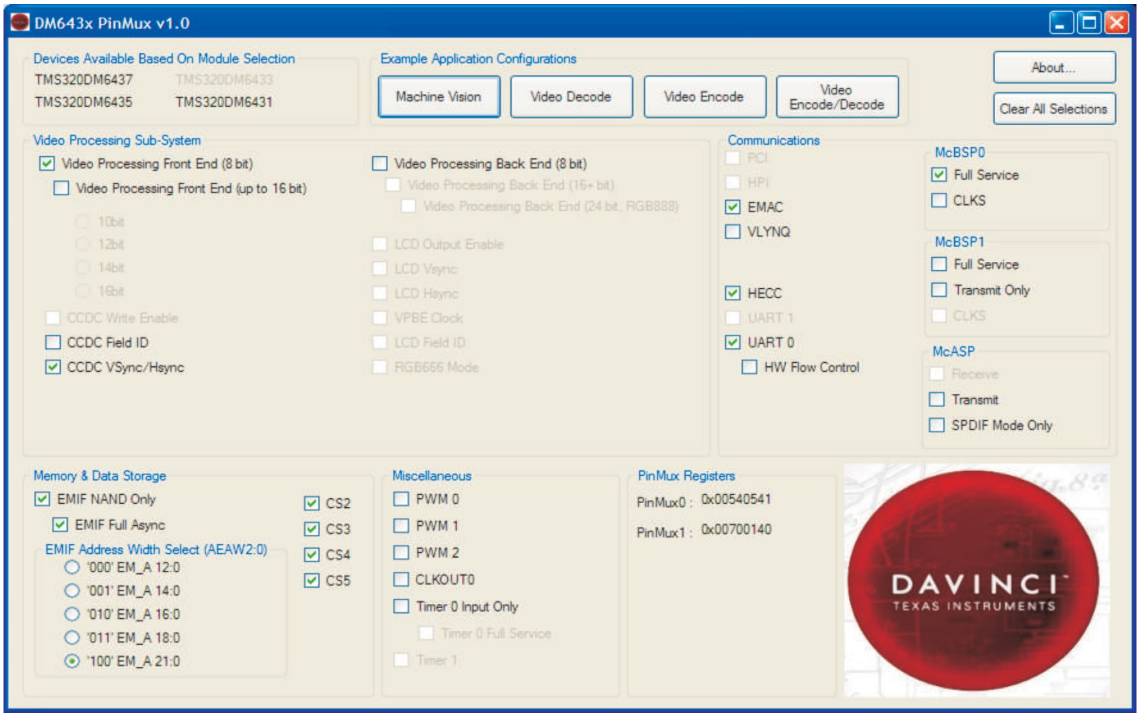
Miscellaneous
□ PWM 0
□ PWM 1
□ PWM 2
□ CLKOUT0
□ Timer 0 Input Only
□ Timer 0 Full Service
□ Timer 1

PinMux Registers
PinMux0 : 0x00048AA3
PinMux1 : 0x02800140

Figure 3. Video Decode Configuration

3.4 Machine Vision

This machine vision configuration is put together to show a potential machine vision application configuration as found in many automotive and industrial applications, see Figure 4. In this case you could utilize the DM6431 device since only basic video capture is necessary. There are also other peripherals such as the high-end CAN controller (HECC) for communication with automotive/industrial CAN networks, as well as a full asynchronous EMIF and multichannel buffered serial port (McBSP) to allow for additional flexibility in interfacing to external devices and memories.



Devices Available Based On Module Selection
TMS320DM6437   TMS320DM6433
TMS320DM6435   TMS320DM6431

Example Application Configurations

Video Processing Sub-System
☑ Video Processing Front End (8 bit)
□ Video Processing Front End (up to 16 bit)
○ 10bit
○ 12bit
○ 14bit
○ 16bit
□ CCDC Write Enable
□ CCDC Field ID
☑ CCDC VSync/Hsync

□ Video Processing Back End (8 bit)
□ Video Processing Back End (16+ bit)
□ Video Processing Back End (24 bit, RGB888)
□ LCD Output Enable
□ LCD Vsync
□ LCD Hsync
□ VPBE Clock
□ LCD Field ID
□ RGB666 Mode

Communications
□ PCI
□ HPI
☑ EMAC
□ VLYNQ
☑ HECC
□ UART 1
☑ UART 0
□ HW Flow Control

McBSP0
☑ Full Service
□ CLKS

McBSP1
□ Full Service
□ Transmit Only
□ CLKS

McASP
□ Receive
□ Transmit
□ SPDIF Mode Only

Memory & Data Storage
☑ EMIF NAND Only
☑ EMIF Full Async
EMIF Address Width Select (AEAW2:0)
○ '000' EM_A 12:0
○ '001' EM_A 14:0
○ '010' EM_A 16:0
○ '011' EM_A 18:0
● '100' EM_A 21:0
☑ CS2
☑ CS3
☑ CS4
☑ CS5

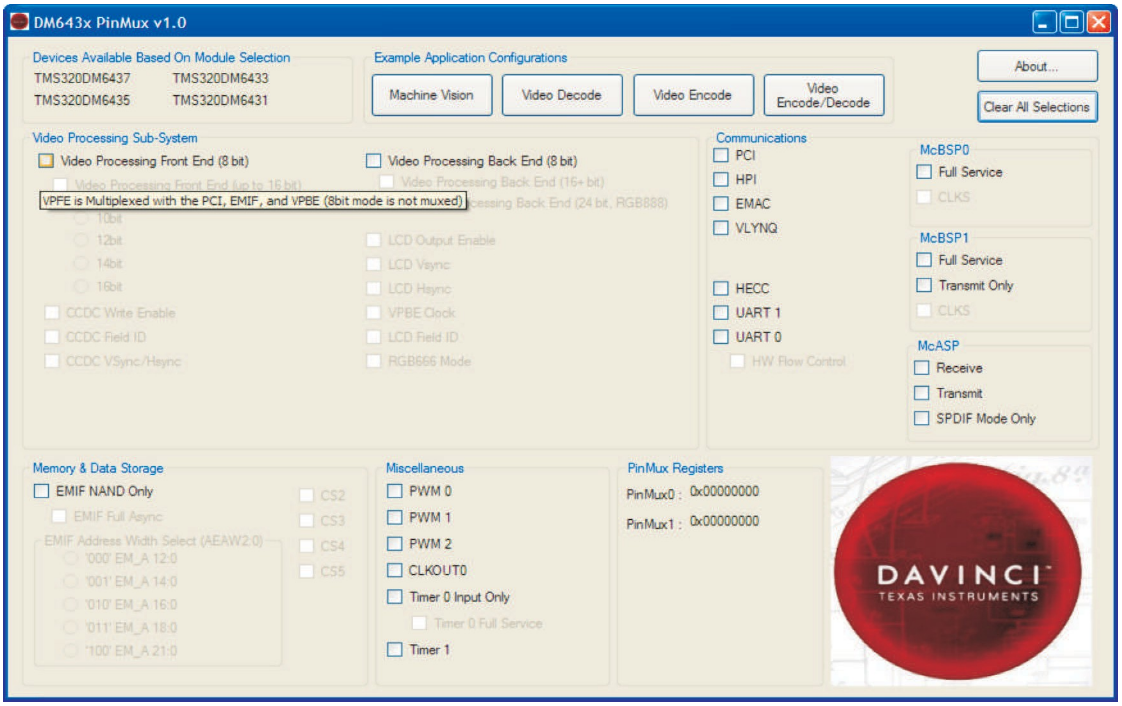
Miscellaneous
□ PWM 0
□ PWM 1
□ PWM 2
□ CLKOUT0
□ Timer 0 Input Only
□ Timer 0 Full Service
□ Timer 1

PinMux Registers
PinMux0 : 0x00540541
PinMux1 : 0x00700140

Figure 4. Machine Vision Configuration

4 Creating the Video Encode/Decode Configuration

This section discusses the creation of the video encode/decode configuration in a step-by-step process starting from a blank screen; this is what is seen when the utility is first started. At any time while using the utility you can determine the peripheral level multiplexing by holding the mouse over a check box; this brings up a tool tip as shown in Figure 5.



VPFE is Multiplexed with the PCI, EMIF, and VPBE (8bit mode is not muxed)

PinMux0 : 0x00000000

PinMux1 : 0x00000000

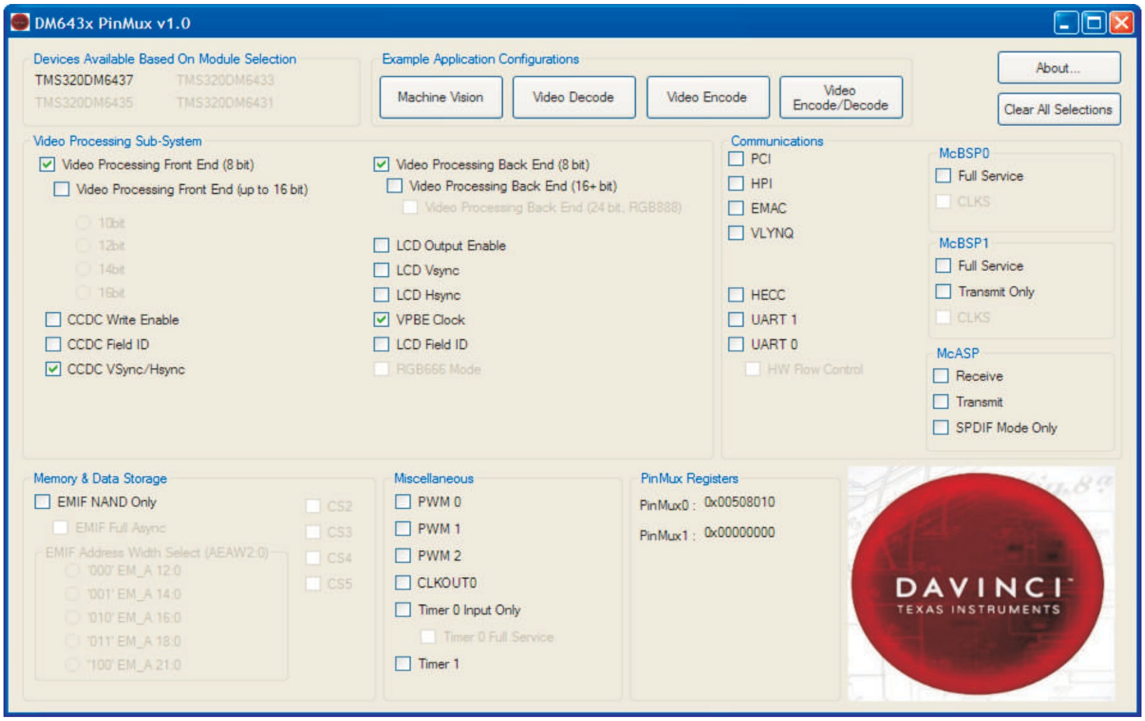
Figure 5. Blank DM643x Pin Multiplexing Utility With Tool Tip

4.1 Selecting the Video Processing Sub-System Peripherals

As this configuration is both capturing/encoding and displaying/decoding video, the first peripherals you can select are the VPFE and the VPBE.

Note: This requires the use of the DM6437 device, as it is the only device choice in the DM6437 family that has both a VPFE and VPBE as seen by the selections in Figure 6.

Additionally, you can check the *CCDC Vsync/Hsync* box as well as the *VPBE Clock* box, as these are only multiplexed with general-purpose input/output (GPIO) and do not have an effect on other peripheral availability.



Devices Available Based On Module Selection
TMS320DM6437 TMS320DM6433
TMS320DM6435 TMS320DM6431

Example Application Configurations

Video Processing Sub-System
☑ Video Processing Front End (8 bit)
□ Video Processing Front End (up to 16 bit)
○ 10bit
○ 12bit
○ 14bit
○ 16bit
□ CCDC Write Enable
□ CCDC Field ID
☑ CCDC VSync/Hsync
☑ Video Processing Back End (8 bit)
□ Video Processing Back End (16+ bit)
□ Video Processing Back End (24 bit, RGB888)
□ LCD Output Enable
□ LCD Vsync
□ LCD Hsync
☑ VPBE Clock
□ LCD Field ID
□ RGB666 Mode

Communications
□ PCI
□ HPI
□ EMAC
□ VLYNQ
□ HECC
□ UART 1
□ UART 0
□ HW Flow Control

McBSP0
□ Full Service
□ CLKS

McBSP1
□ Full Service
□ Transmit Only
□ CLKS

McASP
□ Receive
□ Transmit
□ SPDIF Mode Only

Memory & Data Storage
□ EMIF NAND Only
□ EMIF Full Async
EMIF Address Width Select (AEAW2:0)
○ '000' EM_A 12:0
○ '001' EM_A 14:0
○ '010' EM_A 16:0
○ '011' EM_A 18:0
○ '100' EM_A 21:0
□ CS2
□ CS3
□ CS4
□ CS5

Miscellaneous
□ PWM 0
□ PWM 1
□ PWM 2
□ CLKOUT0
□ Timer 0 Input Only
□ Timer 0 Full Service
□ Timer 1

PinMux Registers
PinMux0 : 0x00508010
PinMux1 : 0x00000000

Figure 6. Selecting VPFE and VPBE

4.2 Selecting the Communication Peripherals

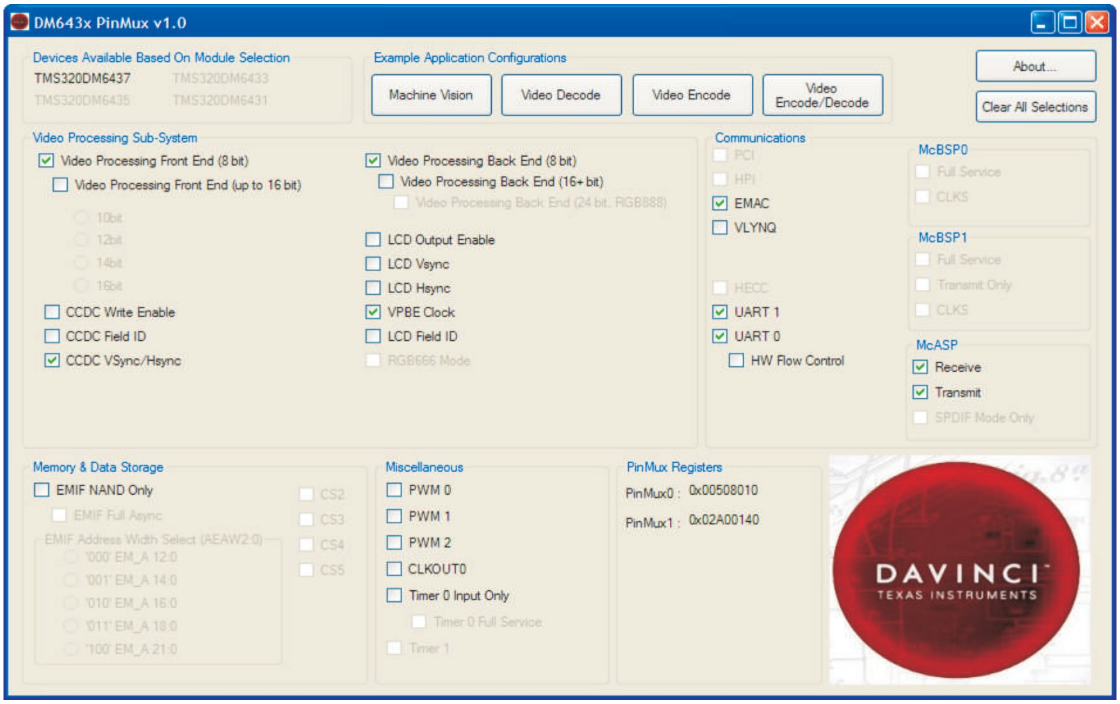
Next, take a closer look at the communications section of the utility; you can select peripherals that can be used to communicate with other devices or data converters. Here, you can select the EMAC, as it is commonly used in such applications.

> **Note:** This causes the peripheral component interconnect (PCI) and host port interface (HPI) to be grayed out as they are multiplexed with the EMAC and cannot be used simultaneously as seen in Figure 7.

You can also select the two universal asynchronous receiver/transmitter (UART 1 and UART 0) peripherals to allow for simple serial communication with the DM6437 or for the DM6437 to communicate with external UART based peripherals. The selection of the UART 1 grays out the HECC due to multiplexing. UART 0 is only multiplexed with GPIO so its selection does not cause the loss of any other peripherals.

To complete the communications section, the receive and transmit capabilities of the McASP are selected. This action grays out the McBSP ports due to multiplexing as well as the *SPDIF Mode Only* check box of the McASP.

The SPDIF only mode is simply a subset of the transmit capability of the McASP that allows the use of McBSP 1 in a transmit configuration simultaneously; you can still use the SPDIF format if the McASP is multiplexed for full transmit capability.



DM643x PinMux v1.0

Devices Available Based On Module Selection
TMS320DM6437 TMS320DM6433
TMS320DM6435 TMS320DM6431

Example Application Configurations
Machine Vision | Video Decode | Video Encode | Video Encode/Decode

About... | Clear All Selections

Video Processing Sub-System
☑ Video Processing Front End (8 bit)
□ Video Processing Front End (up to 16 bit)
○ 10bit
○ 12bit
○ 14bit
○ 16bit
□ CCDC Write Enable
□ CCDC Field ID
☑ CCDC VSync/Hsync
☑ Video Processing Back End (8 bit)
□ Video Processing Back End (16+ bit)
□ Video Processing Back End (24 bit, RGB888)
□ LCD Output Enable
□ LCD Vsync
□ LCD Hsync
☑ VPBE Clock
□ LCD Field ID
□ RGB666 Mode

Communications
□ PCI
□ HPI
☑ EMAC
□ VLYNQ
□ HECC
☑ UART 1
☑ UART 0
□ HW Flow Control

McBSP0
□ Full Service
□ CLKS

McBSP1
□ Full Service
□ Transmit Only
□ CLKS

McASP
☑ Receive
☑ Transmit
□ SPDIF Mode Only

Memory & Data Storage
□ EMIF NAND Only
□ EMIF Full Async
EMIF Address Width Select (AEAW2:0)
○ '000' EM_A 12:0
○ '001' EM_A 14:0
○ '010' EM_A 16:0
○ '011' EM_A 18:0
○ '100' EM_A 21:0
□ CS2
□ CS3
□ CS4
□ CS5

Miscellaneous
□ PWM 0
□ PWM 1
□ PWM 2
□ CLKOUT0
□ Timer 0 Input Only
□ Timer 0 Full Service
□ Timer 1

PinMux Registers
PinMux0 : 0x00508010
PinMux1 : 0x02A00140

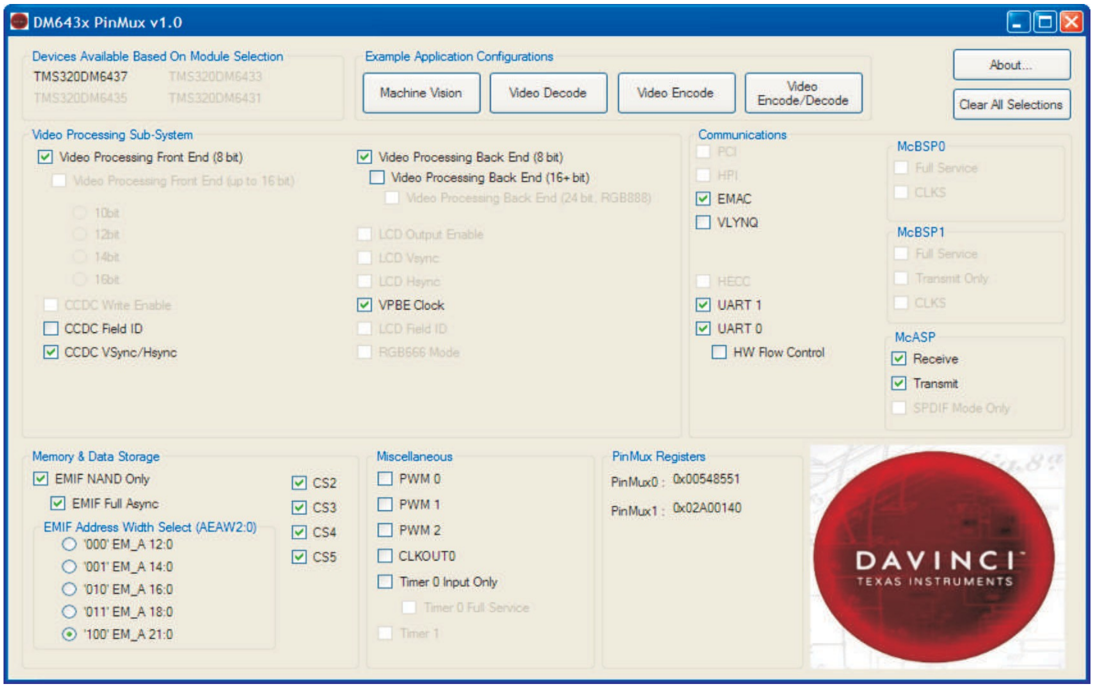
Figure 7. Selecting the Communication Peripherals

4.3 Selecting the Memory and Data Storage Peripherals

The next section to look at on the pin multiplexing utility is the memory and data storage section, this covers the EMIF of the device which is used to connect to any number of parallel asynchronous devices such as NAND or NOR flash memory as seen in Figure 8. Here, you must first select the *EMIF NAND Only* box to enable the most basic capability of the EMIF, NAND mode, which is an asynchronous interface that operates without separate address lines. Enabling this prevents the use of some of the wider bit width video interface modes.

To gain a more versatile EMIF interface, the *EMIF Full Async* box must be checked; this box allows for the interfacing of most any parallel asynchronous device such as a NOR flash by enabling the address bus of the EMIF as seen in Figure 8. Enabling the full asynchronous capability of the EMIF limits the video interface by taking away the CCDC write enable and the LCD field ID as well as preventing the use of the PCI. This application is likely to use simpler video interfaces such as the BT.656 standard; therefore it should not matter.

From here the final selections to the EMIF can be made, adding in the chip select lines that are needed as well as selecting how wide the address bus can be. The address bus is multiplexed with the VPFE, the wider the address bus, the less wide the VPFE can be and vice-versa. In this case since only an 8 bit VPFE is being used, a full 21:0 bit EMIF address width can be selected. The CS2 line can be used without any other peripherals being limited as it is only multiplexed with GPIO. The other CS lines CS3, CS4, and CS5 are multiplexed with the VPBE lines LCD output enable, LCD Vsync, and LCD Hsync, respectively.



DM643x PinMux v1.0

Devices Available Based On Module Selection
TMS320DM6437
TMS320DM6433
TMS320DM6435
TMS320DM6431

Example Application Configurations

Video Processing Sub-System
☑ Video Processing Front End (8 bit)
□ Video Processing Front End (up to 16 bit)
○ 10bit
○ 12bit
○ 14bit
○ 16bit
□ CCDC Write Enable
□ CCDC Field ID
☑ CCDC VSync/Hsync
☑ Video Processing Back End (8 bit)
□ Video Processing Back End (16+ bit)
□ Video Processing Back End (24 bit, RGB888)
□ LCD Output Enable
□ LCD Vsync
□ LCD Hsync
☑ VPBE Clock
□ LCD Field ID
□ RGB666 Mode

Communications
□ PCI
□ HPI
☑ EMAC
□ VLYNQ
□ HECC
☑ UART 1
☑ UART 0
□ HW Flow Control

McBSP0
□ Full Service
□ CLKS

McBSP1
□ Full Service
□ Transmit Only
□ CLKS

McASP
☑ Receive
☑ Transmit
□ SPDIF Mode Only

Memory & Data Storage
☑ EMIF NAND Only
☑ EMIF Full Async
EMIF Address Width Select (AEAW2:0)
○ '000' EM_A 12:0
○ '001' EM_A 14:0
○ '010' EM_A 16:0
○ '011' EM_A 18:0
● '100' EM_A 21:0
☑ CS2
☑ CS3
☑ CS4
☑ CS5

Miscellaneous
□ PWM 0
□ PWM 1
□ PWM 2
□ CLKOUT0
□ Timer 0 Input Only
□ Timer 0 Full Service
□ Timer 1

PinMux Registers
PinMux0 : 0x00548551
PinMux1 : 0x02A00140

Figure 8. Selecting the Memory and Data Storage

With the completion of the memory and data storage section, the configuration now fully resembles the built in video encode/decode example application configuration. From here, you could change any of the selections to better suit a particular application such as adding in pulse width modulation (PWM) outputs or deselecting unnecessary peripherals.

5 References

- *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345)
- *TMS320DM6435 Digital Media Processor Data Manual* (SPRS344)
- *TMS320DM6433 Digital Media Processor Data Manual* (SPRS343)
- *TMS320DM6431 Digital Media Processor Data Manual* (SPRS342)

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265