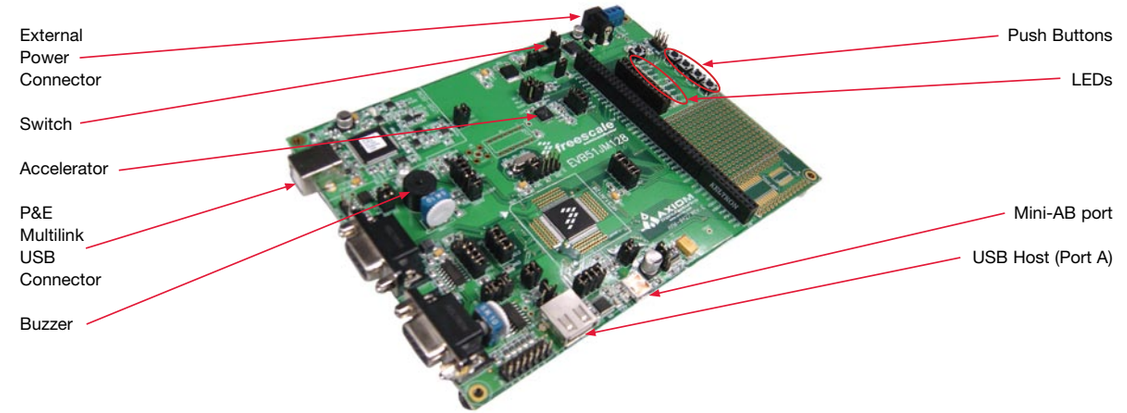


Figure 1. EVB51JM128 board

Learn More: For more information about Freescale products, please visit **www.freescale.com/flexis**.

Agile Number: 926-77155 / REV A
Document Number: EVB51JM128LAB / REV 0

freescale semiconductor

MCU Port Connector Pinout

The following is the pinout for the MCU port connector on the EVB51JM128 board Rev E.

VDD	1	2	IRQ/TPMCLK
VSS	3	4	RESET
PTE0/TxD1	5	6	BKGD/MS
PTE1/RxD1	7	8	VUSB33
PTG0/KBIP0	9	10	PTB0/MISO2/ADP0
PTG1/KBIP1	11	12	PTB1/MOSI2/ADP1
PTE2/TPM1CH0	13	14	PTB2/SPSCK2/ADP2
PTE3/TPM1CH1	15	16	PTB3/SS2/ADP3
PTE5/MOSI1	17	18	PTB4/KBIP4/ADP4
PTE4/MISO1	19	20	PTB5/KBIP5/ADP5
PTE6/SPSCK1	21	22	PTB6/ADP6
PTE7/SS1	23	24	PTB7/ADP7
PTF0/TPM1CH2	25	26	PTC0/SCL
PTF1/TPM1CH3	27	28	PTC1/SDA
PTF2/TPM1CH4	29	30	PTG2/KBIP6
PTF3/TPM1CH5	31	32	PTG3/KBIP7
VREFH	33	34	PTF4/TPM2CH0
VREFL	35	36	PTF5/TPM2CH1
PTD0/ADP8/ACMP+	37	38	PTC5/RxD2
PTD1/ADP9/ACMP–	39	40	PTC3/TxD2

PTD2/KBIP2/ACMP0	41	42	PTG4/XTAL
PTD3/KBIP3/ADP10	43	44	PTG5/EXTAL
PTD4/ADP11	45	46	PTA0/ RGPIO0
PTD5	47	48	PTA1/ RGPIO1
PTD6	49	50	PTA2/ RGPIO2
PTD7	51	52	PTA3/ RGPIO3
PTC2	53	54	PTA4/ RGPIO4
PTC4	55	56	PTA5/ RGPIO5
PTC6/RXCAN	57	58	PTF6
VSSAD	59	60	PTF7/ TXCAN
PTC7	61	62	PTH0/SDA2
VDDAD	63	64	PTH1/SCL2
VDD	65	66	PTH2/RGPIO8
VSS	67	68	PTH3/RGPIO9
USBDN	69	70	PTH4/RGPIO10
USBDP	71	72	PTJ0/RGPIO11
PTA6/RGPIO6	73	74	PTJ1/RGPIO12
PTA7/RGPIO7	75	76	PTJ2/RGPIO13
PTG6	77	78	PTJ3/RGPIO14
PTG7	79	80	PTJ4/RGPIO15

Default Jumper Settings

The following is a list of default jumper settings for EVB51JM128 board Rev E. The settings listed indicate the ON (or installed) position.

Jumper	Installed settings
PWR_SEL	2 & 3 (BDM)
VDD_SEL	1 & 2 (+5V)
VDD	1 & 2
VSSA	1 & 2
VREFL	1 & 2
VREFH	1 & 2
VDDA	1 & 2
OSC_SEL	1 & 3, 2 & 4
ACC_OUT	3 & 4, 5 & 6, 7 & 8
COM_EN	5 & 6, 7 & 8, 9 & 10, 11 & 12
VX_EN	1 & 2
LCD_EN	1 & 2, 3 & 4, 5 & 6, 7 & 8
JP1	1 & 2, 3 & 4

Jumper	Installed settings
SEL1	2 & 3
SEL2	2 & 3
SLEEP	1 & 2
I²C_PULL_EN	ALL ON
USER1	ALL ON
USER2	1 & 2, 3 & 4, 5 & 6, 7 & 8, 11 & 12
PD_EN	OPEN
USPD	OPEN
VBSEL	2 & 3 (P)
OTG_EN	1 & 2, 3 & 4, 5 & 6, 7 & 8
HOST_EN	1 & 2, 3 & 4
CAN_EN	1 & 2, 3 & 4

About EVB51JM128 Labs

EVB51JM128 is a cost-effective kit that enables quick evaluation of the MCF51JM128 microcontroller. These labs will show you the various USB examples included with the complimentary USB-LITE by CMX stack for MCF51JM128. Start each lab with the board powered ON and both provided USB cables connected between host PC and board.

Easy-to-use USB

LAB 1 HID Class Mouse

This lab will pick up with the HID class mouse example.

1. Open CodeWarrior for Microcontrollers. From Windows start menu, you can locate it using the “Programs>Freescale CodeWarrior>CW for Microcontroller V6.x>CodeWarrior IDE.exe” path.
2. From CodeWarrior startup dialog, click on “Start Using CodeWarrior.”
3. Open the HID demo example project by pressing “CTRL” + “O” keys simultaneously. Follow C:\CMXUSB_LITE_V1\usb-peripheral\projects\CodeWarrior\mcf51xx\hid-demo and select “hid-demo.mcp” and click “Open.”
4. With board turned off and both cables (A to B, A to mini-AB, and RS-232 serial communication cable) attached between host and board.
5. Turn the PWRSW switch to the ON position. The +5V and +3V LED will illuminate.
6. Change the target to “EVB Flash,” then compile and program the MCF51JM128 microcontroller with hid class demo example by clicking on “Debug” button, launching debugger.
7. From Connection Manager menu, select “EVB51JM128 on USB1” port and click on “Connect (Reset).”
8. From Erase and Program Flash menu, click on “Yes” to allow the debugger to mass erase the microcontroller’s on-chip flash memory and program it with the new application.
9. Click on the “Start/Continue (F5)” button in the debugger to run application. Close debugger.
10. Your computer will recognize the MCF51JM128 as an HID mouse device and begin installation (no user interaction needed). When hardware installation is complete, the MCF51JM128 will control your mouse cursor by moving it back and forth.

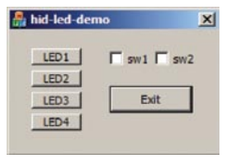
LAB 2 HID Class Keyboard Device

1. Make sure that the HID class mouse example is running.
2. Press and hold button labeled “PTG1” while you reset the microcontroller by pressing on button labeled “RESET.”
3. Release button labeled “PTG1.”
4. Your computer will recognize the MCF51JM128 as an HID keyboard device and begin installation (no user interaction needed). When hardware installation is complete, the MCF51JM128 will now behave as a keyboard.
5. Click button labeled “PTG1” for Page Up key action.
6. Click button labeled “PTG2” for Page Down key action.
7. Click Caps Lock and/or Num Lock key on your keyboard and a “PTE3” and “PTE2” LEDs will turn on/off respectively.

LAB 3 HID Class Generic Device

This lab will pick up with the HID class mouse example which is described by Lab 1 in this document. The HID class generic device example can be entered by following the steps listed below. This example will use a virtual C++ graphical user interface (GUI) to read data from and write data to the MCF51JM128 over USB interface.

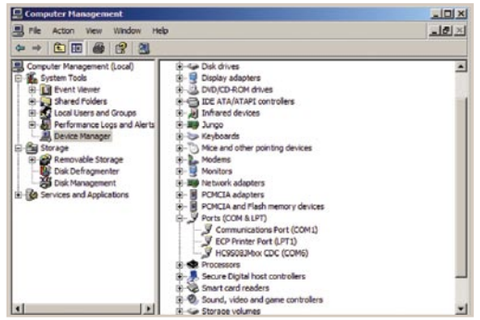
1. Make sure the HID class mouse example is running.
2. Press and hold button labeled “PTG2,” while you reset the microcontroller by pressing on button labeled “RESET.”
3. Release button labeled “PTG2.”
4. Your computer will recognize the MCF51JM128 as an HID generic device (a.k.a. HID led demo) and begin installation (no user interaction needed). When hardware installation is complete, the MCF51JM128 will now behave as a generic USB device.
5. Open “hid-led-demo.exe” GUI on your computer by double-clicking on the file within the C:\CMXUSB_LITE_V1\usb-peripheral\pc-side\hid-led-demo\Debug directory.
6. Click on virtual LED1, LED2, LED3 and LED4 buttons on computer GUI to turn on/off the LEDs tied to “PTF0,” “PTF1,” “PTE2” and “PTE3” on board.
7. Press buttons labeled “PTG1” and “PTG2” on board to turn on/off SW1 and SW2 indicators on computer GUI.
8. When done, click “Exit” button on GUI.



LAB 4 CDC to/from UART

This lab will create a bridge between CDC and UART. The CDC to/from UART example will demonstrate how to transfer commands/data between the CDC virtual COM port and true serial COM port using two independent terminal windows.

1. Open CodeWarrior for Microcontrollers. From Windows start menu, you can locate it using the “Programs>Freescale CodeWarrior>CW for Microcontroller V6.x>CodeWarrior IDE.exe” path.
2. From CodeWarrior startup dialog, click on “Start Using CodeWarrior”.
3. Open the CDC terminal example project by pressing “CTRL” + “O” keys simultaneously. Follow C:\CMXUSB_LITE_V1\usb-peripheral\projects\CodeWarrior\mcf51xx\cdc-demo and select “cdc-demo.mcp” and click “Open.”
4. With board turned off and both cables (A to B, A to mini-AB, and RS-232 serial communication cable) attached between host and board.
5. Turn the PWRSW switch to the ON position. The +5V and +3V LED will illuminate.
6. Then compile and program the MCF51JM128 microcontroller with CDC class terminal example by clicking on “Debug” button, launching debugger.
7. From Connection Manager menu, select “EVB51JM128 on USB1” port and click on “Connect (Reset).”
8. From Erase and Program Flash menu, click on “Yes” to allow the debugger to mass erase the microcontroller’s on-chip flash memory and program it with the new application.
9. Click on the “Start/Continue (F5)” button in the debugger to run application. Close debugger.
10. Your computer will recognize the MCF51JM128 as a CDC device and begin installation. Some interaction is needed to provide the USB driver. When prompted by computer choose to specify location and provide the “mcf51xx.inf” driver located in C:\CMXUSB_LITE_V1\usb-peripheral\src\mcf51xx\cdc-demo. When hardware installation is complete, the MCF51JM128 will now be recognized as a CDC device with an assigned virtual COM port.
11. Verify the assigned virtual COM port for your board in the device manager under “Ports (COM & LTP).”
12. Launch two Hyperterminal window utilities. Hyperterminal can be found under Start>Programs>Accessories>Communications.
13. In terminal window 1, set port assigned to your board per device manager.
14. Set baud rate to 9600 (8 Data bits, none Parity, 1 Stop bit, none Flow control).
15. Open COM port.
16. Set port to COM1.
17. Set baud rate to 9600.
18. Click on “Open Serial Port” button.
19. Type “Hello World!!” or any other characters in CDC side and see them appear on UART side. The same will work in the opposite direction.
20. Close COM ports and terminal windows.



LAB 5 Host-HID Class

This lab will show how to connect with one USB mouse or USB keyboard following the steps listed below.

1. Change the factory setting on board. Change VBSEL jumper to 1 & 2 (HOST mode), also connect PD_EN jumpers 1 & 2 (DPPD) and 3 & 4 (DNPD).
2. Open CodeWarrior for Microcontrollers. From Windows start menu, you can locate it using the “Programs>Freescale CodeWarrior>CW for Microcontroller V6.x>CodeWarrior IDE.exe” path.
3. From CodeWarrior startup dialog, click on “Start Using CodeWarrior.”
4. Open the Host-HID example project by pressing “CTRL” + “O” keys simultaneously. Follow C:\CMXUSB_LITE_V1\usb-host\projects\CodeWarrior-6.x\hid\ and select “host-hid-demo.mcp” and click “Open.”
5. With board turned off and both cables (A to B, RS-232 serial cable) attached between host and board.
6. Turn the PWRSW switch to the ON position. The +5V and +3V LED will illuminate.
7. Change the target to “EVB Flash,” compile and program the MCF51JM128 microcontroller with host-hid-demo example by clicking on “Debug” button, launching debugger.
8. From Connection Manager menu, select “EVB51JM128 on USB1” port and click on “Connect (Reset).”
9. From Erase and Program Flash menu, click on “Yes” to allow the debugger to mass erase the microcontroller’s on-chip flash memory and program it with the new application.
10. Close debugger.
11. Launch Hyperterminal window utilities. Hyperterminal can be found under Start>Programs>Accessories>Communications.
12. In terminal window, set port to COM1.
13. Set baud rate to 9600.
14. Click on “Open Serial Port” button.
15. Reset board.
16. The below information will be shown on the terminal window.

    Host HID Demo

    Waiting for device
17. Connect USB mouse or USB keyboard to USB “A” Port (J3 on the board). In terminal you will see the transmitted information from device to host.
18. Disconnect device from board when completed.

LAB 6 Host-mass-storage Class

This lab will show how to connect with one USB Thumb Disk following the steps listed below.

1. Change the factory setting on board. Change VBSEL jumper to 1 & 2 (HOST mode), also connect PD_EN jumper 1 & 2 (DPPD) and 3 & 4 (DNPD).
2. Open CodeWarrior for Microcontrollers. From Windows start menu, you can locate it using the “Programs>Freescale CodeWarrior>CW for Microcontroller V6.x>CodeWarrior IDE.exe” path.
3. From CodeWarrior startup dialog, click on “Start Using CodeWarrior.”
4. Open the Host-HID example project by pressing “CTRL” + “O” keys simultaneously. Follow C:\CMXUSB_LITE_V1\usb-host\projects\CodeWarrior-6.x\mass-storage\ and select “mass-storage-demo.mcp” and click “Open.”
5. With board turned off and both cables (A to B, RS-232 serial cable) attached between host and board.
6. Turn the PWRSW switch to the ON position. The +5V and +3V LED will illuminate.
7. Change the target to “EVB Flash,” then compile and program the MCF51JM128 microcontroller with mass-storage example by clicking on “Debug” button, launching debugger.
8. From Connection Manager menu, select “EVB51JM128 on USB1” port and click on “Connect (Reset).”
9. From Erase and Program Flash menu, click on “Yes” to allow the debugger to mass erase the microcontroller’s on-chip flash memory and program it with the new application.
10. Close debugger.
11. Launch Hyperterminal window utilities. Hyperterminal can be found under Start>Programs>Accessories>Communications.
12. In terminal window, set port to COM1.
13. Set baud rate to 9600.
14. Click on “Open Serial Port” button.
15. Reset board.
16. The below information will be shown on the terminal window.

    Host HID Demo

    Waiting for device
17. Connect USB Disk to the USB “A” Port (J3 on the board). In terminal type “help” to see available commands.
18. Disconnect device from board when completed.