



**ZMOTIONL300ZCOG**

**Z8F6481 ZMOTION®  
Library Development Kit**

**User Manual**

UM028901-0318



**Warning:** DO NOT USE THIS PRODUCT IN LIFE SUPPORT SYSTEMS.

---

## **LIFE SUPPORT POLICY**

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

## **AS USED HEREIN**

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

## **DOCUMENT DISCLAIMER**

©2018 Zilog, Inc All rights reserved. Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

ZMOTION® and Z8 Encore! XP® are trademarks or registered trademarks of Zilog, Inc., an IXYS Company. All other product or service names are the property of their respective owners.

# Revision History

Each instance in the following Revision History table reflects a change to this document from its previous version. For more details, refer to the corresponding pages or appropriate links listed in the table below.

<b>Date</b>	<b>Revision Level</b>	<b>Description</b>	<b>Page Number</b>
Mar 2018	01	Original Issue.	All

## Overview

Zilog’s Z8F6481 ZMOTION Library Development Kit provides a general-purpose platform for evaluating the capabilities and operation of the ZMOTION Engine Library using the Z8F6481 Series of microcontrollers. This kit is equipped with two Z8F6481 series MCUs and comes complete with various lenses and PIR sensors to test the ZMOTION Engine Library for multiple applications.

Zilog’s ZMOTION Engine Library provides an integrated and flexible solution for Passive Infra-Red (PIR)-based motion detection applications. The Software library is comprised of the PIR sensor signal processing algorithms for motion detection, transient and noise detection, white light detection, and several other motion-related functions and is integrated with the user’s application code.

An Application Programming Interface (API) allows the application code to configure, control, and monitor the Library in real time. API configuration parameters allow the Engine operation to be optimized for the particular lens and pyro electric sensor being used in the application. This allows the designer to create their own application-specific software while taking advantage of Zilog’s ZMOTION Motion Detection technology.

For more information about the operation of the ZMOTION Library, refer to [ZMOTION Engine Library for the F6481 Series \(UM0275\)](#).

## Kit Contents

The Z8F6481 ZMOTION Engine Library Development Kit includes the components listed in Table 1.

**Table 1. Kit Contents**

Item	Description	Quantity
1	Z8F6481 ZMOTION Library Development Board	1
2	USB SmartCable	1
3	Selection of Lenses: ZAA09GIT1, ZCM077GIV3, ZCM077GIV5, ZCWM05GIV1, ZNCL11, ZNCL926, ZNCL10IL, ZNCL3B, ZNCL10R, ZNCL10S	10
4	Selection of PIR Sensors: ZRE200GE (installed), ZSBG323671 (Dual Element), ZSBG446671 (Quad Element)	3
5	USB Cable – type A/B	1
6	Lens Holders – Circular for ZCM077GIV3 and ZCM077GIV5 lenses Rectangular for ZAA09GIT1 lens	2

Figure 1 shows the contents of the ZMOTION Detection and Control Development Kit.



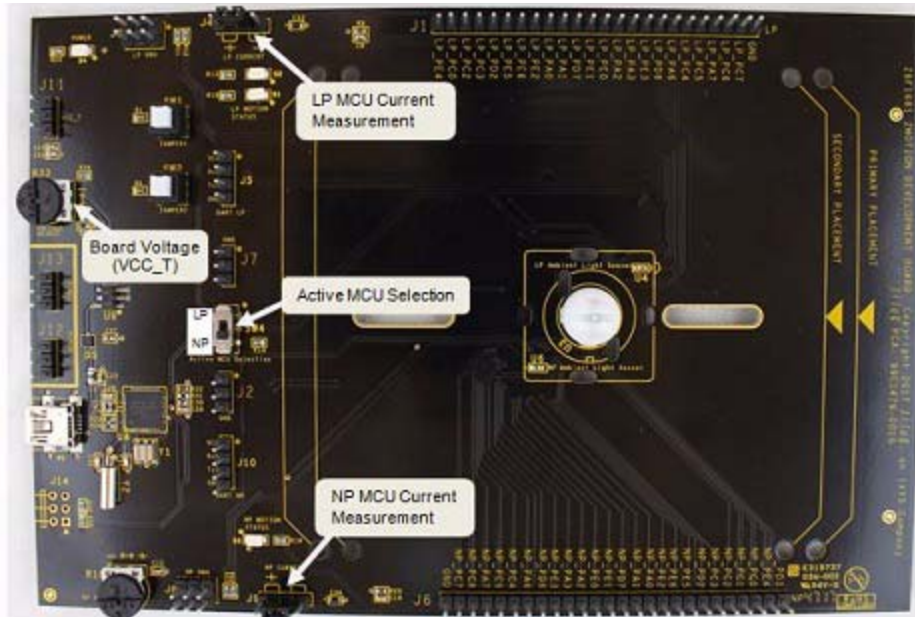


Figure 2. Development Board (Top Side)

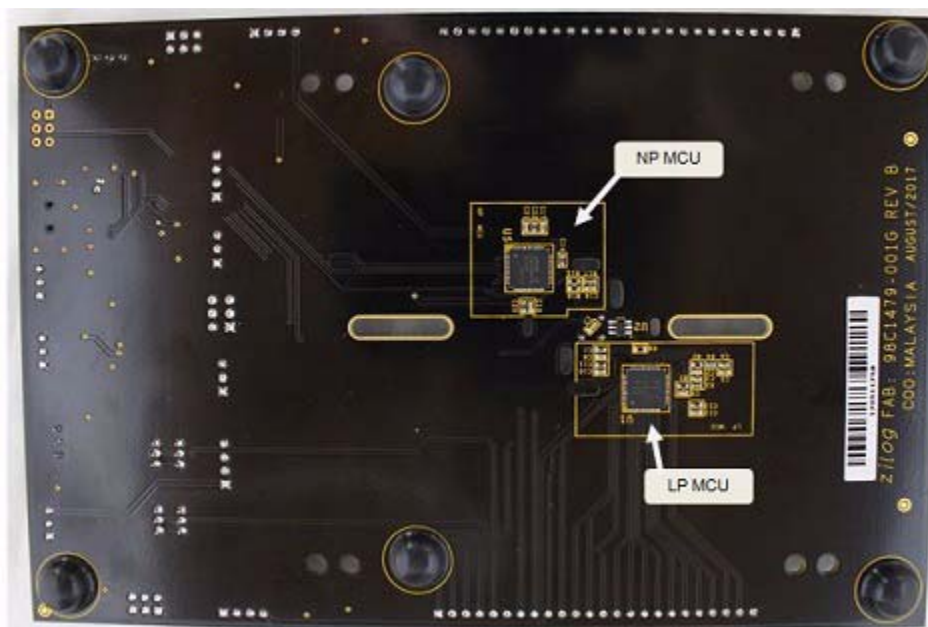


Figure 3. Development Board (Bottom Side)

## PIR Sensors

The three PIR sensors listed in Table 2 are included with the kit. The ZRE200GE sensor is soldered on the board.

**Table 2. PIR Sensors Included with Kit**

Part Number	Description
ZRE200GE	Basic Dual Element (installed)
ZSBG323671	Premium Dual Element with EMI filtering
ZSBG446671	Quad Element with EMI filtering

To change the PIR sensor, carefully unsolder the installed sensor and solder in the new sensor. Avoid extended duration of heat on the sensors' pins, because excessive heat may cause deterioration of the sensor (for example, avoid duration beyond 5 seconds at 350°C).

The pyroelectric sensors are easy to identify by observing the window size (the board comes with the ZRE200GE installed).



**Figure 4. PIR Sensors Included with Kit**

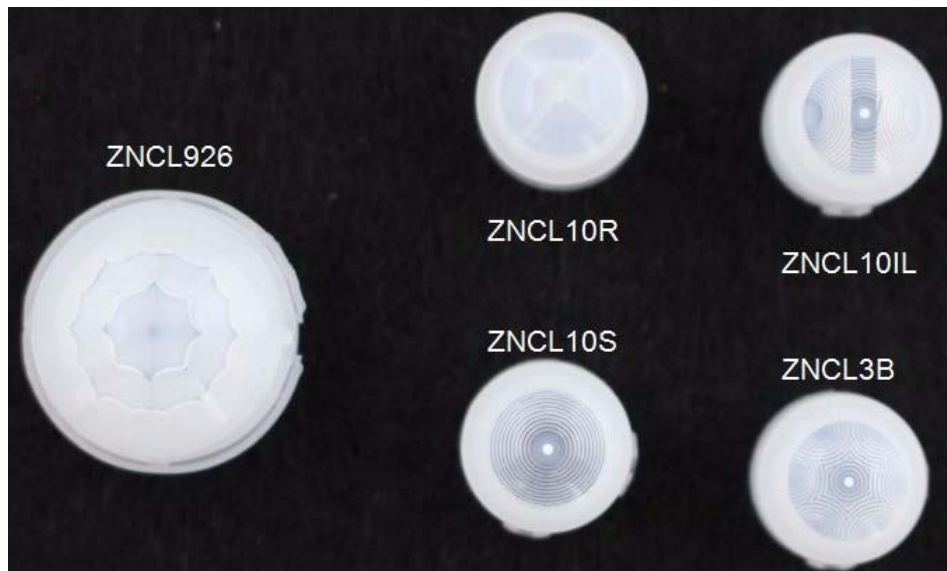
## Lenses

The Development Kit includes ten lenses. Four mounting options are supported – Clip on, PCB Mount, Circular Holder, and Rectangular Holder. Table 3 describes the lenses that are included with the Kit.

**Table 3. Lenses Included with Kit**

Part Number	Description	Mounting
ZNCL10IL	9mm; 80°x30°; 7m range (installed)	Clip on
ZNCL3B	9mm; 60°x60°; 3m range	Clip on
ZNCL10R	9mm; 360°; 5m range	Clip on
ZNCL10S	9mm; 15°x15°; 12m range	Clip on
ZNCL926	15mm; 360°; 5m range	Clip on
ZNCL11	Flat; 104° x37°; 4m range	PCB Mount
ZCWM05GIV1	Round; 24mm; 360°/180°; 9m range	PCB Mount
ZCM077GIV3	Ceiling Mount; 360°; 3.7m height	Circular Holder
ZCM077GIV5	Ceiling Mount; 360°; 12m height	Circular Holder
ZAA09GIT1	35.6mm x 50mm Flat; 22.9mm focal; 25m range	Rectangular Holder

Figure 5 displays the clip-on lenses.



**Figure 5. Clip-on Lenses**

Figures 6 and 7 show PCB mounted lenses.





**Figure 6. ZCWM05GIV1 - PCB Mount**



**Figure 7. ZNCL11 - PCB Mount**

Figures 8 and 9 show lenses with circular holder mounting.

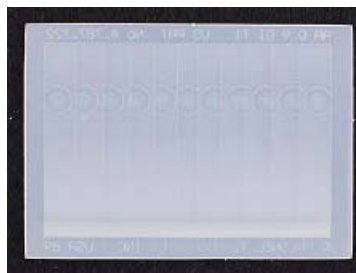


**Figure 8. ZCM077GIV5 - Circular Holder Mount**



**Figure 9. ZCM077GIV3 - Circular Holder Mount**

Figure 10 shows a rectangular holder mounted lens.



**Figure 10. ZAA09GIT1 - Rectangular Holder Mount**

## Z8F6481 ZMOTION Library Development Board

This section describes the components of the ZMOTION Library Development Board.

### Features

- Two Z8F6481QN024XK2247 ZMOTION MCUs
  - Normal Power configuration
  - Low Power configuration
- USB Serial Interface
- Tamper switches
- Ambient Light Sensors
- Status LEDs
- Variable power supply 2.0V to 3.3V

Figure 11 shows a block diagram of the Z8F6481 ZMOTION Development Board. Refer to Appendix A for the complete schematic diagrams.

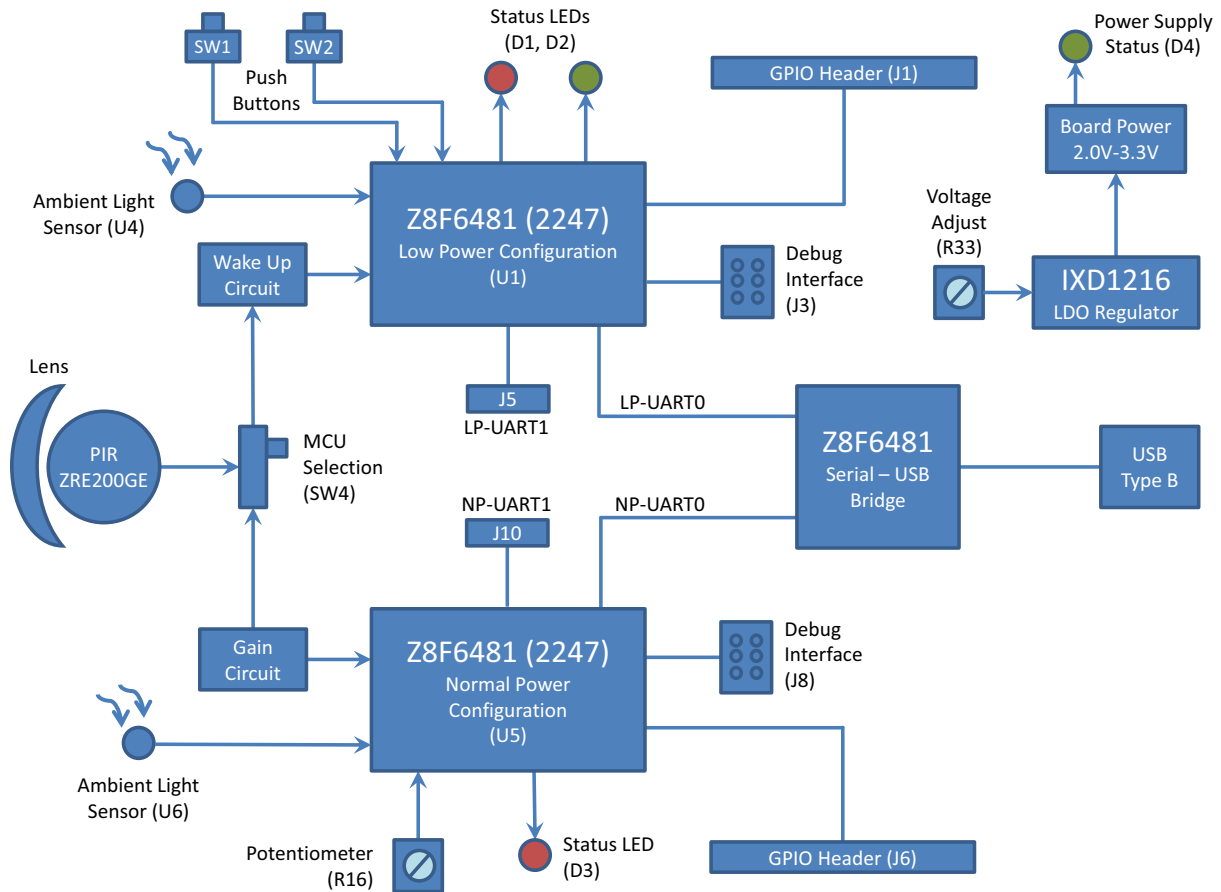


Figure 11. Z8F6481 ZMOTION Library Development Board Block Diagram

## The Z8F6481 (2247) ZMOTION MCU

The Z8F6481 (2247) suffix is a specific version of the Z8F6482 MCU series that supports the ZMOTION Library. Five devices with the 2247 suffix are available, as listed in Table 4.

**Table 4. Z8F6481 Devices**

MCU Part Number	Package	Memory
Z8F1681QK024XK2247	32-Pin QFN	16KB Flash, 2KB RAM
Z8F1681QN024XK2247	44-Pin QFN	16KB Flash, 2KB RAM
Z8F1681AN024XK2247	44-Pin LQFP	16KB Flash, 2KB RAM
Z8F6481QN024XK2247	44-Pin QFN	64KB Flash, 3.75KB RAM
Z8F6481AN024XK2247	44-Pin LQFP	64KB Flash, 3.75KB RAM

► **Note:** \*The Z8F6481QN024XK (44-Pin QFN) device is included with the development kit.

### Features

The Z8F6481 (2247) ZMOTION MCU has the following features:

- 24MHz eZ8 CPU core
- 16KB or 64KB Flash memory with in-circuit programming capability
- 2KB or 3.75KB internal RAM
- 128 Bytes Non-Volatile Data Storage (NVDS) on 16KB devices
- Internal 32.768KHz precision oscillator with DCO/FLL
- Support for external crystal oscillators
- Up to 17-channel, 12/14-bit Analog-to-Digital Converter (ADC)
- 12-bit Digital-to-Analog Converter (DAC)
- Two on-chip analog comparators (32-pin package contain only one)
- Two on-chip, low-power operational amplifiers (32-pin package contain only one)
- On-chip temperature sensor
- 128-bit Advanced Encryption Standard (AES) encryption/decryption hardware accelerator
- 8-channel event system provides communication between peripherals for autonomous triggering

- Four-channel DMA controller
- Full-speed Universal Serial Bus (USB 2.0) device supporting eight endpoints with integrated USB PHY
- Two full-duplex 9-bit UARTs supporting Local Interconnect Network (LIN), Digital Addressable Lighting Interface (DALI) protocols and DMX (32-pin package contain only one)
- Enhanced Serial Peripheral Interface (ESPI) controller
- I2C controller which supports Master/Slave modes
- Real-Time Clock (RTC) supporting both Counter and Clock modes
- Three enhanced 16-bit timers with Capture, Compare, and PWM capability
- Two additional basic 16-bit timers with interrupts (shared as UART Baud Rate Generator)
- 16-bit multi-channel timer which supports four Capture/Compare/PWM modules (not available on 32-pin package)
- Watchdog Timer (WDT) with independent oscillator
- 26 to 67 General-Purpose Input/Output (GPIO) pins, depending upon package
- On-Chip Debugger (OCD)
- Power-On Reset (POR), Voltage Brown-Out (VBO) and Low-Voltage Detection (LVD)
- Wide 1.8V-3.6V operating voltage range
- 32- and 44-pin packages
- -40°C to +85°C (extended) operating temperature range

## Getting Started

Download and install the ZMOTIONL300ZCOG\_1.0 installation file from the [Zilog web-site](#). This will create a folder that includes the VCOM USB driver, reference schematics, sample applications, ZMOTION Engine Library, and related documentation.

After installation the following folders will be created:

```
Zilog\ZMOTIONL300ZCOG_1.0
  Documentation
  Drivers
  Z8F6481 ZMOTION Library
    Reference Schematics
    Sample Applications
    ZMOTION Engine Library
```

The development board comes with two Z8F6481QN024XK2247 MCUs that are preprogrammed with sample applications to demonstrate basic motion detection.

- The LP MCU is preprogrammed with the ZM\_LP\_Basic application.
- The NP MCU is preprogrammed with the ZM\_NP\_Basic application.

---

► **Note:** The 2247 suffix is a specific version of the Z8F6481 MCU series that supports the ZMOTION Library. Devices without the 2247 suffix do not support the ZMOTION Library.

---

The MCUs share the same PIR sensor, so only one device may be used at a time. SW4 (Active MCU Selection) is used to select the active MCU. The unselected MCU is held in Reset.

- The LP MCU is in the Low Power configuration
- The NP MCU is in the Normal Power configuration

For a demonstration with the preprogrammed application, perform the following steps:

1. Install the VCOM USB driver included with the ZMOTIONL300ZCOG\_1.0 installation file by right-clicking on the .exe file and selecting **Run as administrator**. This driver is located in the Drivers folder.
  - Zilog\_VCOM\_Install.x64.exe: 64-bit Windows
  - Zilog\_VCOM\_Install.x86.exe: 32-bit Windows
  - The Device Driver will be installed as (Zilog USB Serial Device) in Device Manager under Ports (COM & LPT)

2. Set SW4 to select the active MCU configuration (LP or NP).
3. Install the ZNCL10IL lens on the PIR sensor.
4. Ensure the shunts are installed across pins 2 and 3 on J4 (LP MCU) and J9 (NP MCU).
5. With the USB A (male) to Mini-B cable, connect Port P1 on the Development Board to a USB port on the development PC or other suitable power source to apply power to the Development Board.
  - Alternatively, power can be supplied from a bench type power supply using J12 and J13. An input of 5V to 12V is supported.

When power is first applied, the status LED will blink until the PIR sensor is stable. When the PIR sensor is stable, the status LED will turn off and blink when motion is detected.

If the USB interface is connected to a PC, a terminal emulation program like RealTerm or CoolTerm can be used to observe the motion status events. Set the serial interface to 57600bps, N, 8, 1.

Refer to the [Sample Applications](#) section on page 13 to learn more about specific MCU operations.

## MCU Current Measurement

The MCU current consumption can be easily measured by removing the shunt on J4 (LP Current) or J9 (NP Current) and connecting an ammeter on the respective positive (+) and negative (–) header pins.

## Operating Voltage Adjustment

The MCU operating voltage can be adjusted from 2.0V to 3.0V by the R33 potentiometer. Turn the potentiometer clockwise to decrease the voltage. The operating voltage can be measured from the J11 header (labeled VCC\_T).

## Sample Applications

Table 5 lists four sample application projects that are included with the ZMOTIONL300ZCOG\_1.0 installation file. Contact [Zilog Technical Support](#) for additional sample projects.

The LP and NP MCUs on the Development Board come preloaded with the ZM\_LP\_Basic and ZM\_NP\_Basic projects. To load a different project, refer to the [Using ZDS-II to Download and Debug a Project](#) section on page 16.

**Table 5. Sample Applications**

Project	Folder	MCU Configuration
ZM_LP_Basic	ZM_Z8F1681_LP_Basic	Low Power
ZM_LP_Command	ZM_Z8F1681_LP_Command	Low Power
ZM_NP_Basic	ZM_Z8F1681_NP_Basic	Normal Power
ZM_NP_Command	ZM_Z8F1681_NP_Command	Normal Power

The Z8 Encore! family of MCUs supports three operating modes:

- Run – In this mode, the CPU is running and all peripherals/clock sources can be enabled (normal running mode).
- Halt – In this mode, the CPU is not running and all peripherals/clock sources can be enabled. Exit via interrupt.
- Stop – In this mode, the CPU is not running and some peripherals/clock sources can be enabled. Exit via Stop Mode Recovery (SMR).

Stop mode provides the lowest current consumption and is used in the ZM\_LP\_Basic and ZM\_LP\_Command projects. The ZM\_NP\_Basic and ZM\_NP\_Command projects use Halt mode to reduce current consumption.

However, the Stop mode operation is not supported while executing code with the debugger; only Halt and Run modes are supported. To use the debugger with the ZM\_LP\_Basic and ZM\_LP\_Command projects, the following configuration definition is included in the main.h file that enables the use of Halt mode in place of Stop mode.

```
#define USE_STOP_MODE
```

Set this definition to 0 and rebuild the project before using the debugger to run the application. Set this definition to 1 when running the application without the debugger. Current consumption will be higher when using Halt mode.



## ZM\_LP\_Basic

This project uses the Low Power MCU configuration to provide motion detection with ZMOTION Digital PIR validation.

The MCU stays in Low Power Stop mode until an Analog PIR event is detected. The MCU then wakes up and runs the ZMOTION Engine on the PIR signal to validate the motion event.

When the motion event is validated, the status LED is flashed and a message is sent via the VCOM USB serial interface (connected to UART0 on the LP MCU).

Typical current consumption is <13uA @ 3.0V.

Sliding SW4 to the LP position causes the NP MCU to be held in reset and releases the LP MCU to run. The following status messages will be displayed on the terminal:

```
ZM_LP_Basic
Version: 1.0
ZMOTION: Enabled

Waiting for PIR Stability...
```

The MCU normally stays in Stop mode to conserve power. Each time it wakes up, a status message is sent to indicate the reason for the wake up. Table 6 lists these status message descriptions.

**Table 6. Status Messages**

Status Code	Description
*08.	Indicates that the MCU woke up as part of the power-on stability timer.
*04.	Indicates that the MCU woke up to enable motion detection.
*02.	Indicates that the MCU woke up from an analog motion event. It will run the ZMOTION Engine to digitally validate the motion event.
PIR Stable	PIR sensor is stable – Power on stability delay complete.
Motion Detected	The ZMOTION Engine validated the motion event.

---

► **Note:** In Table 6, \* indicates MCU wake up and . indicates MCU going to Stop mode. When Motion Detected status is sent, an ASCII BELL character (0x07) is also sent. Some terminal applications can interpret this and provide an audible indication of motion detection.

---

## ZM\_LP\_Command

This project uses the Low Power MCU configuration to provide motion detection with ZMOTION Digital PIR validation and a serial command interface.

The MCU stays in Low Power Stop mode until an Analog PIR event is detected. The MCU then wakes up and runs the ZMOTION Engine on the PIR signal to validate the motion event.

When the motion event is validated, the status LED is flashed, motion trigger output is pulsed, and a message can be sent via Serial interface.

Typical current consumption is <15uA @ 3.0V.

The serial command interface has the following features that help adjust system operations and performance:

- Pet Immunity mode
- Programmable Digital and Analog sensitivity
- Programmable LED, trigger and motion delay time
- Ambient light levels and threshold
- Pass-Through and Hyper-Sense (Micro-Motion) modes
- Temperature Compensation – adjusts Analog PIR thresholds based on ambient temperature
- Temperature Change Protection – Ignores motion events generated by temperature changes
- Vdd Shift Protection – Ignores motion events generated by shifts in supply voltage

Refer to the Serial Command Functions LP file included with the project for a full list of functions and features.

## ZM\_NP\_Basic

This project demonstrates the use of the ZMOTION Library and API to detect motion.

When motion is detected, an LED is turned on, output is driven active, and a message is sent via the VCOM USB serial interface (connected to UART0 on the NP MCU).

Typical current consumption is <1mA @ 3.0V.

Move SW4 from the LP position to the NP position. The LP MCU will be held in reset. The NP MCU will start running and display the following messages:

```
Z8F1681 ZMOTION Library Sample Program
ZM_NP_Basic
Version: 1.0

Waiting for PIR Sensor Stability
.....
PIR Stable
```

Each time the MCU wakes up from a motion event it will display the following status message:

```
001 Motion Detected  
002 Motion Detected
```

001 and 002 represent the total count of motion events detected since reset.

## ZM\_NP\_Command

This project demonstrates the use of the ZMOTION Library and API to detect motion with ambient light detection and a serial command interface. When a motion event is detected, the status LED is flashed, motion trigger output is pulsed, and a message can be sent via Serial interface.

Typical current consumption is <1mA @ 3.0V.

The serial command interface includes the following features that help adjust system operations and performance:

- Pet Immunity mode
- Programmable Digital sensitivity
- Programmable LED, trigger and motion delay time
- Ambient light levels and threshold
- Pass-Through and Hyper-Sense (Micro-Motion) modes
- Temperature Compensation – adjusts Analog PIR thresholds based on ambient temperature
- Temperature Change Protection – Ignores motion events generated by temperature changes
- Vdd Shift Protection – Ignores motion events generated by shifts in supply voltage

Refer to the *Serial Command Functions NP* file included with the project for a full list of functions and features.

## Using ZDS-II to Download and Debug a Project

Zilog Developer Studio II (ZDSII) Integrated Development Environment is a complete standalone system that provides a state-of-the-art development environment. Based on standard Windows user interfaces, ZDSII integrates a language-sensitive editor, project manager, C-compiler, assembler, linker, librarian, and source-level symbolic debugger to provide a development solution specifically tailored to the Z8 Encore! Family of micro-controllers.

1. Download and install ZDSII using the procedure outlined in [Appendix B. IDE- Zilog Developer Studio II](#).
2. If not already installed, install the VCOM USB driver included with the ZMOTIONL300ZCOG\_1.0 installation file by right-clicking on the .exe file and selecting **Run as administrator**. This driver is located in the Drivers folder.
  - Zilog\_VCOM\_Install.x64.exe: 64-bit Windows
  - Zilog\_VCOM\_Install.x86.exe: 32-bit Windows
  - The Device Driver will be installed as (Zilog USB Serial Device) in Device Manager under Ports (COM & LPT)

## Connecting the ZMOTIONL300 Development Board to your PC

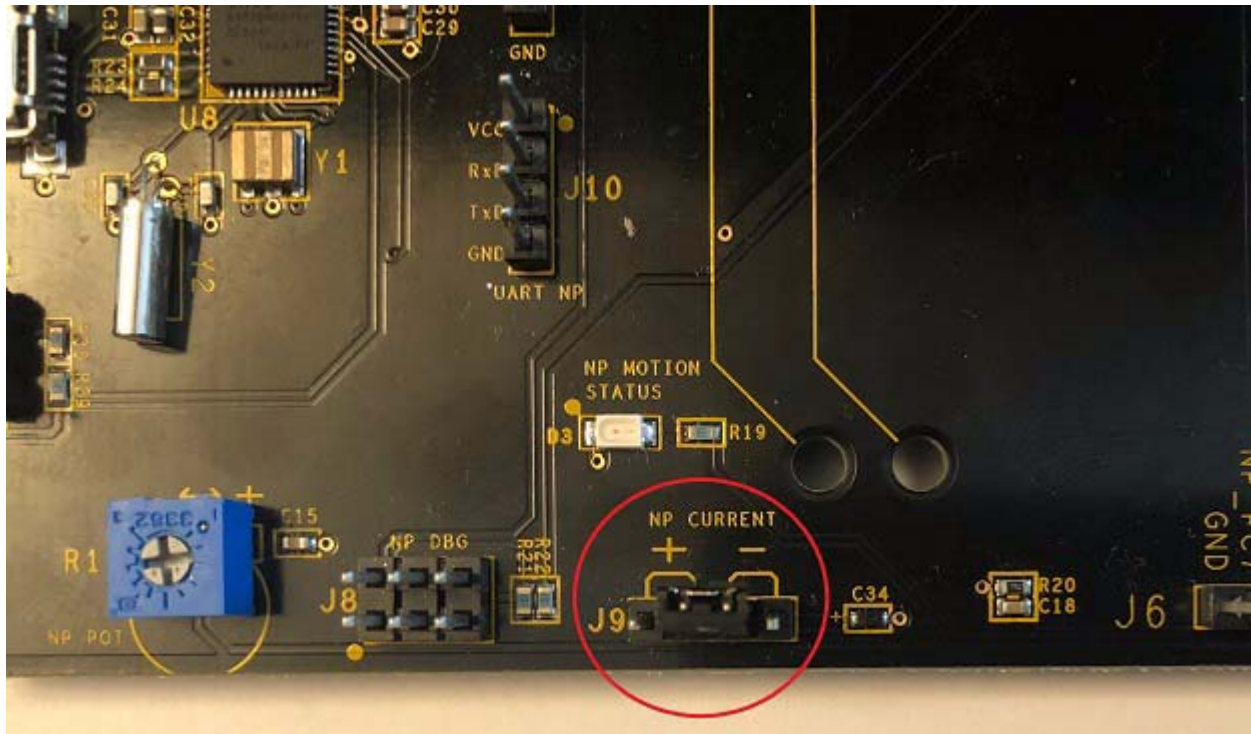
Observe the following procedure to connect the ZMOTIONL300 Development Board to your PC. We will use the ZM\_NP\_Command project for this example.



**Caution:** Disconnect or turn off the power to the ZMOTIONL300 Development Board before connecting or disconnecting the USB SmartCable.

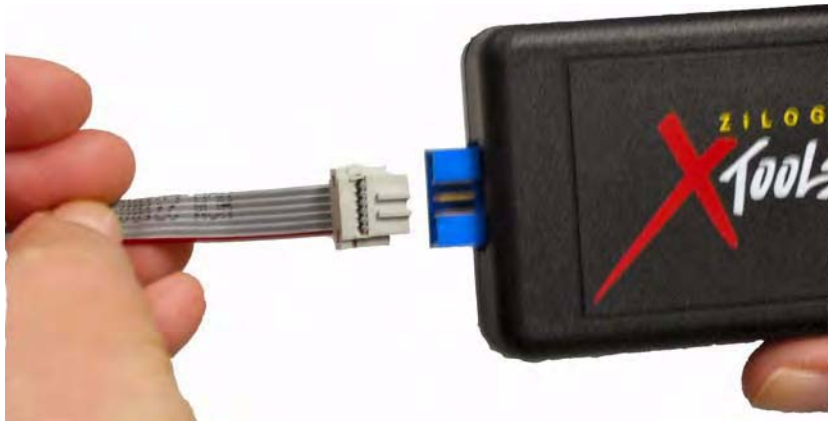
---

1. Ensure that the NP Current Jumper is installed on J9. It should be installed across the centre two pins, as shown in Figure 12.



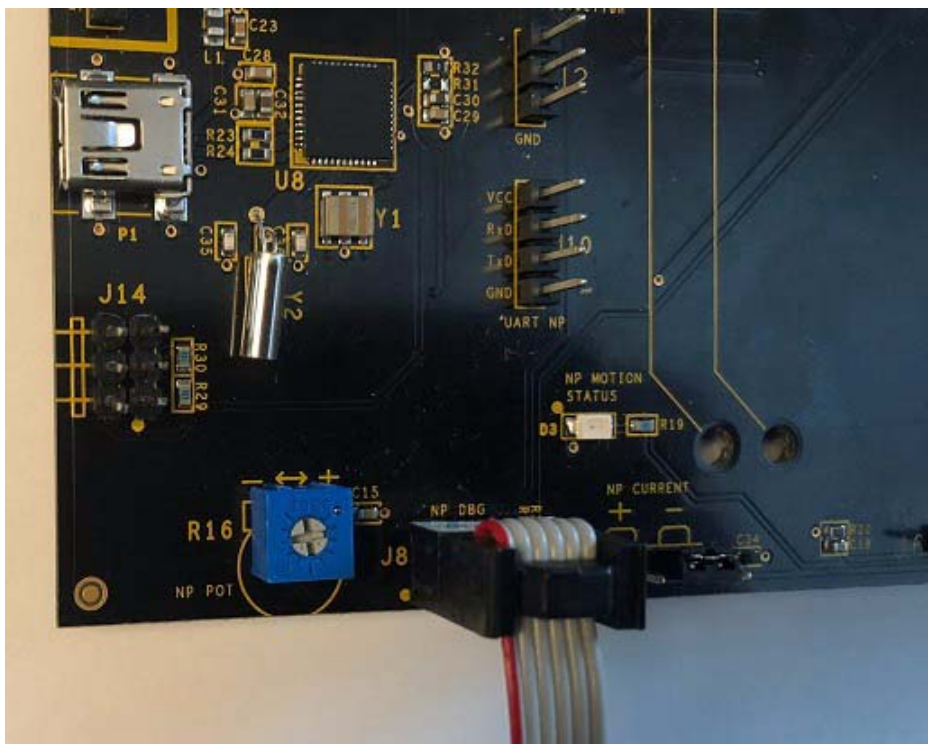
**Figure 12. NP Current Jumper Installed on J9**

2. Place switch SW4 into the NP position.
3. Connect one end of the 6-circuit ribbon cable provided in your Kit to the USB Smart-Cable unit, ensuring that the ribbon's male connector is aligned correctly with the female connector on the unit, as indicated by the red stripe in Figure 13.



**Figure 13. Connecting the 6-Circuit Ribbon Cable to the USB SmartCable**

4. Connect the other end of the ribbon cable to NP Debug Connector J8 on the Development Board. Ensure that Pin 1 on the ribbon cable is aligned with Pin 1 on the target connector, as shown in Figure 14.



**Figure 14. Connecting Other End of the Ribbon Cable to NP Debug Connector J8**

5. With the USB A (male) to Mini-B cable, connect Port P1 on the ZMOTIONL300 Development Board to a USB port on the development PC to apply power and provide a serial interface for the Development Board. See Figure 15.

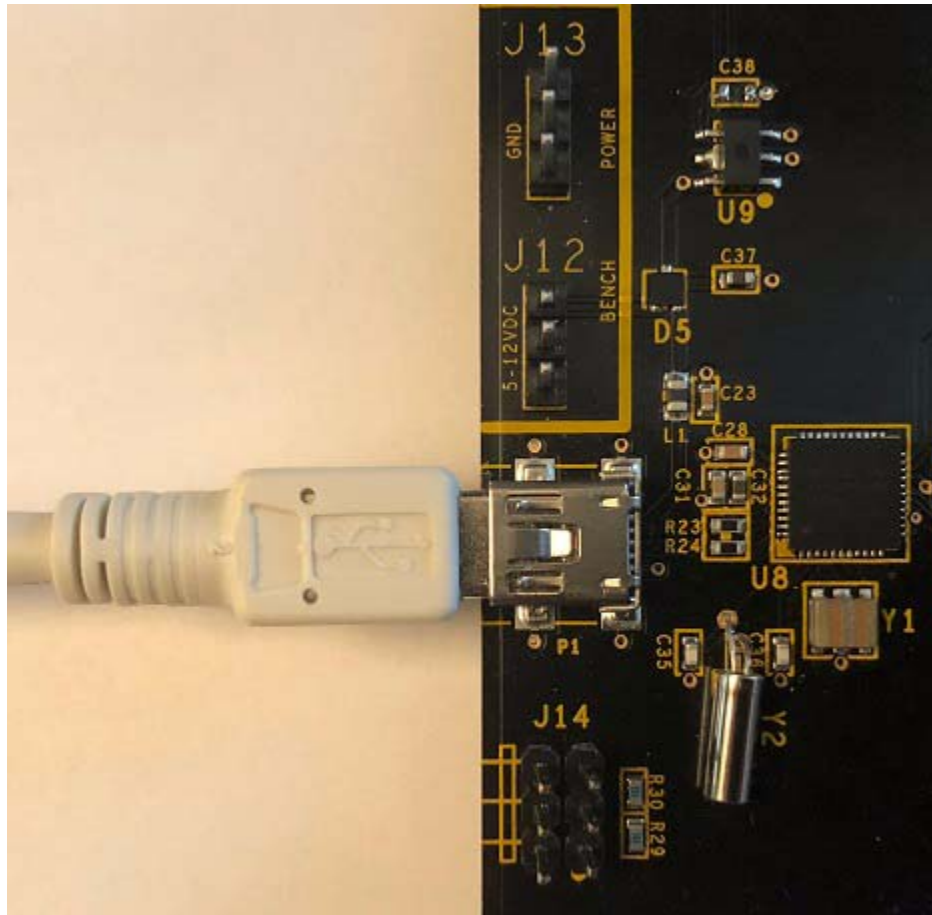


Figure 15. Connecting Port P1 to a USB Port on Development PC

6. Open a serial terminal application like CoolTerm or RealTerm and connect to the Zilog USB Serial Device at 57,600 bps, No Parity, 1 Stop bit. Open **Device Manager** and look under **Ports (COM & LPT)** to identify the COM port assigned to the Zilog USB Serial Device connected to P1 on the ZMOTIONL300 Development Board.

7. Start ZDSII and open the NP Serial Command project (ZM\_NP\_Command). This can be done by starting ZDSII and opening the project by clicking **File** → **Open Project** from the menu or by navigating to the folder in Windows File Explorer and double-clicking the project file located in:

```
ZMOTIONL300ZCOG_1.0\Z8F641 ZMOTION Library\Sample
Applications\ZMOTION_NP_Command_R1_0
```

Two projects are provided – one includes a Bootloader while the other does not. The projects share the same source code:

```
ZM_NP_Command.zdsproj - Does not include Bootloader
```

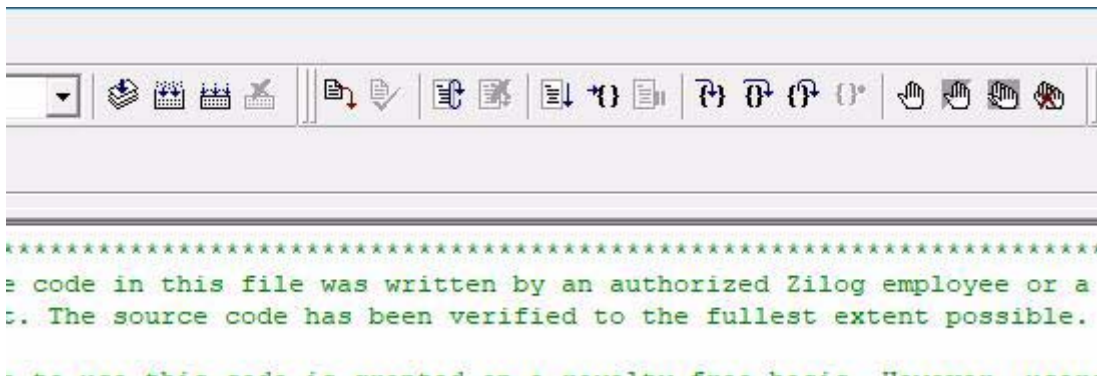
```
ZM_NP_Command_BL.zdsproj - Includes Bootloader
```

---

► **Note:** When using the project with Bootloader, care must be taken not to set Breakpoints in the application code area when starting the application. Setting a Breakpoint modifies the Flash memory location to a Break instruction which will cause the Bootloader’s checksum test performed after reset to fail.

---

8. If the Debug icons are not displayed, as in Figure 16, right-click in the empty grey area under the icons and select **Debug** and **Workbook Mode**.



**Figure 16. Debug Icons**

9. Hover the mouse over each icon to display its function. Click the **Rebuild All** icon. ZDSII will build the project using the current versions of all source files and generate the necessary output files. If there were **no** errors detected during the build, a message similar to the one in Figure 18 will be displayed.



```

x
Space          Base          Top          Size          Used          Unused
-----
EDATA          E:0100        E:048E      E00H          38FH          A71H
              (            (            3584          911          2673)
RDATA          R:30          R:8B        D0H           5CH           74H
              (            (            208           92           116)
ROM            C:0000        C:3550      10000H        3551H        CAAFH
              (            (            65536         13649        51887)

OUTPUT CHECKSUM
=====
ZM_NP_Command.hex  9B1B
ZM_NP_Command.lod  9B1B

0 warning(s)
0 error(s)
Build succeeded.

<
|< < > >| \ Build \ Debug \ Find in Files 1 \ Find in Files 2 \ Messages \ Command /
For Help, press F1

```

**Figure 17. Build Succeeded Message**

10. Click the **Go** icon. ZDSII will connect to the target MCU through the USB SmartCable, download the project to its Flash memory, reset the MCU, and start running the application. A message similar to the one shown in Figure 18 is displayed.

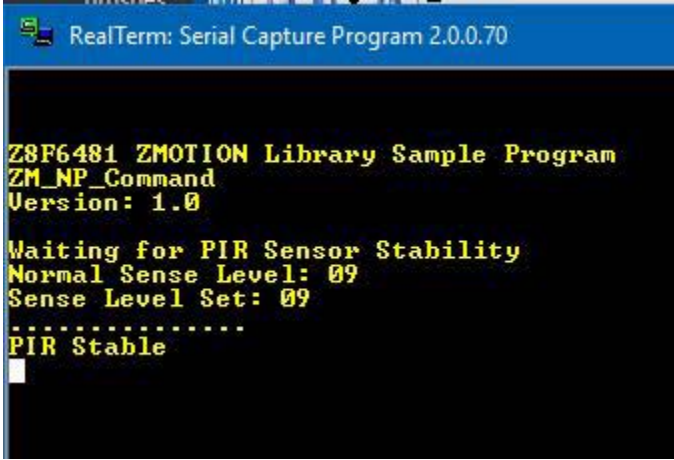
```

Connected to target Z8F64820100KITG
Starting debug session [project:ZM_NP_Command, configuration:ZM_NP_COMMAND]...
Cpu Z8F6481XN Rev C, USB Smart Cable Firmware Version is 1.60, SN #081112-000c
Loading file C:\Zilog\ZMOTIONL300ZCOG_1.0\Z8F641 ZMOTION Library\Sample Applications\ZMOTION_NP_C
Loading file C:\Zilog\ZMOTIONL300ZCOG_1.0\Z8F641 ZMOTION Library\Sample Applications\ZMOTION_NP_C
Checksum(NOT including pad bytes): 0x9B1B
Target is in HALT/SLEEP mode!

```

**Figure 18. Message Displayed upon Connecting to Target MCU**

11. The red LED (D3) should start flashing 1 second on, 1 second off while the application waits for the PIR sensor to stabilize.
12. If a serial terminal application is running and connected to the Zilog USB Serial Device, the messages shown in Figure 19 will be displayed.



```
RealTerm: Serial Capture Program 2.0.0.70

Z8F6481 ZMOTION Library Sample Program
ZM_NP_Command
Version: 1.0

Waiting for PIR Sensor Stability
Normal Sense Level: 09
Sense Level Set: 09
.....
PIR Stable
█
```

Figure 19. Messages Displayed on RealTerm Serial Capture Program

13. Each time motion is detected, *ZN* will be displayed.
14. The *ZM\_NP\_Command* project supports both UARTs provided on the Z8F6481 MCU (Refer to the schematic in Appendix A). The UART0 interface becomes the Zilog USB Serial Device through the USB Serial Bridge IC (U8) and is referred to as the Dev Port. The Dev Port provides operating status information in an ASCII format to be easily viewed on a terminal. UART1 is connected to J10, providing a direct hardware (TTL) connection to the UART interface on the MCU and is referred to as the Host Port. The Host Port is intended for connection to another MCU (Host MCU) and provides a serial command interface to adjust operating parameters like motion sensitivity, output timing, and various operating modes. Refer to the *Serial Command Functions NP.pdf* document included with the *ZM\_NP\_Command* project for more details on the supported commands and interface options. By default, no information is transmitted on this interface unless requested by the Host.
15. To demonstrate the Host Port operation, use a USB-to-TTL adapter connected to J10 and send hex values from a PC terminal application. Sending 0x17, 0x01 from the Host causes the application to start sending 0x90 to the Host every time a motion event is detected. Sending 0x17, 0x00 causes the application to stop sending 0x90 to the Host every time a motion event is detected (this is the default setting). Refer to the *Serial Command Functions NP.pdf* document included with the *ZM\_NP\_Command* project for more details on the Host command interface.

## Documentation

Additional information can be found in the following documents. These are available free for download from the [Zilog website](#). They are also included with the ZDS II installation or the ZMOTIONL300ZCOG\_1.0 installation.

**Table 7. Z8F6481 ZMOTION Library Development Kit Documentation**

Documents Included in the ZDSII Z8 Encore! Installation		
Description	Document ID	Installed Location
F6482 Series Product Specification	<a href="#">PS0294</a>	Documentation\Chip_Documentation
F6482 Series Product Brief	<a href="#">PB0246</a>	Documentation\Chip_Documentation
eZ8 CPU User Manual	<a href="#">UM0128</a>	Documentation\Chip_Documentation
Z8 Encore! XP F6482 Series API Programmer's Reference Manual	<a href="#">RM0064</a>	Documentation\Technical_Articles
Z8 Encore! Design for Debug	<a href="#">TN0036</a>	Documentation\Technical_Articles
Ethernet SmartCable User Manual	<a href="#">UM0207</a>	Documentation\Tools_Documentation
Opto-Isolated USB SmartCable User Manual	<a href="#">UM0195</a>	Documentation\Tools_Documentation
USB SmartCable User Manual	<a href="#">UM0181</a>	Documentation\Tools_Documentation
Documents included in the ZMOTIONL300ZCOG_1.0 Installation		
Description	Document ID	Installed Location
Z16F81 ZMOTION Development Kit Quick Start Guide	<a href="#">QS0095</a>	Documentation
Z8F6481 ZMOTION Library Development Kit User Manual	<a href="#">UM0289</a>	Documentation
ZMOTION Engine Library for the F6482 Series User Manual	<a href="#">UM0275</a>	Documentation
ZMOTION Product Brief	<a href="#">PB0258</a>	Documentation

# Appendix A. Schematic Diagrams

Figures 20 through 22 show the schematic diagrams for the Z8F6481 ZMOTION Library Development Board.

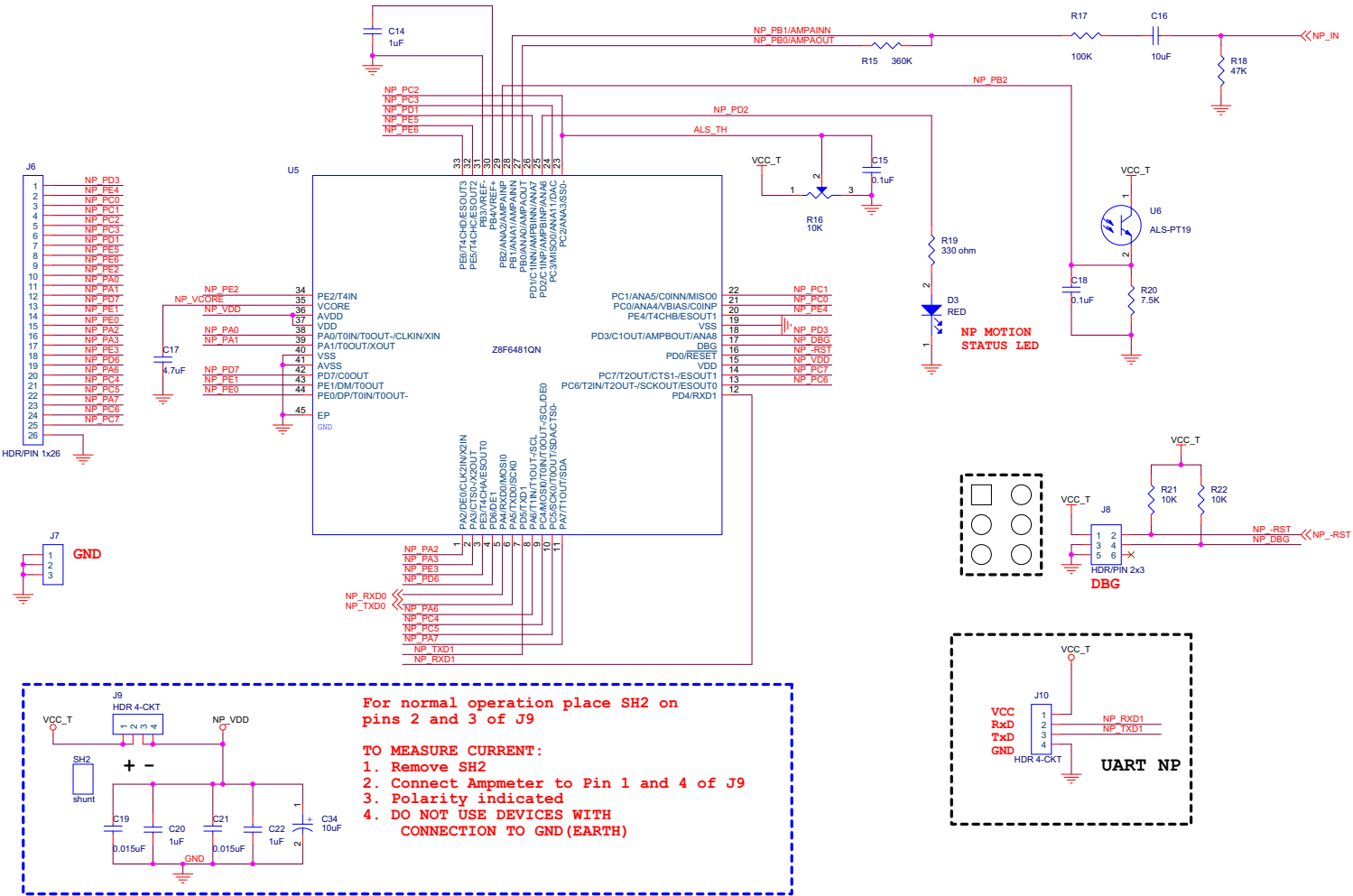


Figure 20. Schematic Diagram (1 of 3)

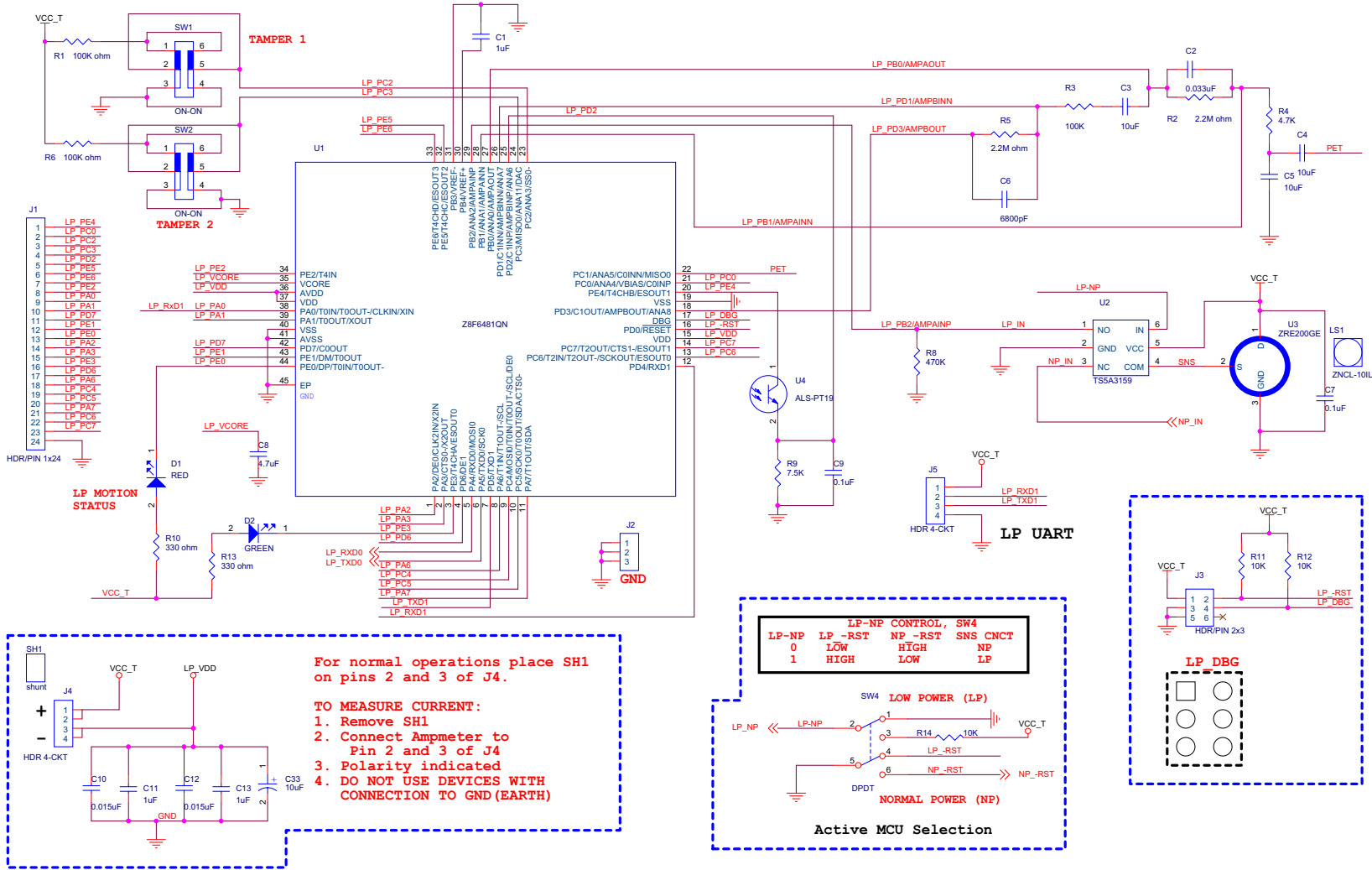


Figure 21. Schematic Diagram (2 of 3)



## Appendix B. IDE-Zilog Developer Studio II

Observe the following steps to download and install the ZDSII software package.

---

► **Note:** If you have already installed ZDS II – Z8 Encore! Version 5.2.2 or later and have downloaded the sample applications and documentation by following the procedure on the paper insert in your kit (FL0188), you're ready for [Appendix C. Installing the USB SmartCable Driver](#).

---

1. Prior to connecting the F6482 Series Development Board to your development PC, download ZDS II for Z8 Encore! v5. 2.2 (or later) from the [Zilog website](#).
  - Select the most recent version of ZDS II – Z8 Encore!
2. When your download is complete, double-click the installation file named ZDS2\_Z8Encore!\_<version>.exe, and follow the on-screen instructions.

### Supported Operating Systems

ZDS-II is supported for the following operating systems:

- Microsoft Windows 10 (32-bit/64-bit)
- Microsoft Windows 7 (32-bit/64-bit)
- Microsoft Windows Vista (32-bit/64-bit)
- Microsoft Windows XP

## Appendix C. Installing the USB SmartCable Driver

The USB SmartCable can be installed on PCs that run on Windows 7 (32- and 64-bit), Windows Vista (32- and 64-bit) and Windows XP operating systems. The procedures in this section will guide you through the USB SmartCable installation process.

### Windows 7 32/64 Systems

Observe the following steps to install the USB SmartCable on a Windows 7 system.

1. Connect the USB SmartCable to a USB port on your development PC. When the PC detects the new hardware, it will display the Installing device driver software dialog.
2. Windows automatically searches for the driver; this process can take a few moments. Because there is no option to terminate this search process, wait for the search to complete. If the driver was previously installed, Windows will automatically install the USB SmartCable driver. In this event, skip ahead to Step 9. If Windows cannot find the driver, close the search dialog and proceed to the next step.
3. In the Search programs and files field in the Windows Start menu, enter Device Manager. The Device Manager will appear in a list of search results.
4. From this list of results, click Device Manager to open the Device Manager dialog, which presents a list of devices that operate on your PC. Find and toggle **Other devices** to view a sublist of additional devices, and right-click your mouse on USB SmartCable.
5. In the submenu that appears, click **Update Driver Software...**
6. In the Update Driver Software – USB SmartCable dialog that appears, click the **Browse my computer for driver software** option.
7. Click the **Browse...** button to browse to one of the following driver directories, depending on the configuration of your PC.  
On 32-bit Windows 7 systems, navigate to:  
<ZDS II Installation Directory>\device drivers\USB\x32  
On 64-bit Windows 7 systems, navigate to:  
<ZDS II Installation Directory>\device drivers\USB\x64
8. Click **Next** to install the driver. On 32-bit Windows systems, a security dialog will appear; select Install this driver software anyway.
9. After the Wizard finishes the installation, click **Close**.



## Windows Vista 32/64 Systems

Observe the following steps to install the USB SmartCable on a Windows Vista system.

1. Connect the USB SmartCable to a USB port on the development PC.
2. After the PC detects the new hardware, it will display the Found New Hardware Wizard dialog box. Click **Locate and install driver software (recommended)**.
3. Depending on your development PC's User Account Control settings, Windows may ask for permission to continue the installation. Click **Continue**.
4. When the *Insert the Disc* dialog appears, select **I don't have the disc. Show me other options**. Click the **Next** button to display the *Windows couldn't find driver* dialog.
5. Select **Browse my computer for driver software (advanced)** to display the Browse For Driver dialog, which prompts you to key in or browse for the location of the driver's .inf file. Depending on the type of computer you use (32-bit or 64-bit), use the **Browse...** button to navigate to one of the following paths, then click the **Next** button.

On 32-bit Vista systems, navigate to:

<ZDSII Installation>\device drivers\USB\x32

On 64-bit Vista systems, navigate to:

<ZDSII Installation>\device drivers\USB\x64

6. When the Windows Security dialog prompts you whether to install or not to install, click **Install this driver software anyway** and wait until the installation is completed (Windows may prompt you more than once).
7. When the software has been installed successfully, click **Close**.

## Windows XP-32(SP3) Systems

Observe the following steps to install the USB SmartCable on a Windows XP system.

1. Connect the USB SmartCable to a USB port on the development PC. When the PC detects the new hardware, it will display the Found New Hardware Wizard dialog.
2. In the Wizard, select **Install from a list or specific location (Advanced)**, then click **Next**.

---

► **Note:** If the Windows Hardware Installation dialog appears, click **Continue Anyway**.

---

3. In the **Please choose your search and installations** dialog, select **Search for the best driver in these locations and include this location in search**.
4. Use the **Browse...** button to navigate to one of the following paths:  
<ZDSII Installation>\device drivers\USB\x32
5. Click **Next** to locate the appropriate driver.
6. Click **Next** a second time, then click **Finish** to complete the installation.