Reference Design

ZMOTION Intelligent Lighting Control with DALI, DSI, and 1–10V

RD003801-0418

Overview

This reference design demonstrates the use of Zilog's Z8F6482 MCU in intelligent lighting control applications, including lighting control via DALI, DSI, and 1–10V interface/s, motion sensing with ZMOTION, ambient light detection, and multiple communication interface/s.

The Intelligent Lighting Control reference design board is programmed with the RD0038-SC01 firmware and is ready to use.

Note: The source code file associated with this reference design, RD0038-SC01.zip, is available free for download from the Zilog website.

Warning: THIS INTELLIGENT LIGHTING CONTROL BOARD USES HIGH VOLTAGE (100–240VAC) FOR SOME OF THE LIGHTING CONFIGURATIONS. DISCONNECT ALL POWER SUPPLIES PRIOR TO HANDLING THE BOARD.

Features

Key features of this ZMOTION Intelligent Lighting Control reference design include:

- Lighting Control Interfaces: DALI, DSI, 1–10V
- Lighting Control Modes: Occupancy, Vacancy, and Occupancy Pass-Through
- Lighting Control Profile: Dual or Multi-State (ON, Pre-OFF, OFF)
- Switched AC Control
- PIR Motion Detection with ZMOTION Digital Processing
- Ambient Light Sensor for Light Gating
- Local Configuration via DIP Switches
- IR interface for remote configuration
- Additional remote configuration and control on various communication interfaces – UART, I2C, USB
- Powered from a 5VDC wall supply

Potential Applications

This reference design can be used to develop lighting applications for a variety of spaces such as:

- Offices
- Conference and meeting rooms
- Classrooms and lecture halls
- Dressing/fitting/locker rooms
- Lobbies and hallways
- Pantries
- Rest rooms
- Gymnasiums
- Warehouses

Theory of Operation

The market for intelligent lighting control applications is well established and shows steady growth. Traditionally, fluorescent and LED ballasts have incorporated dimming capability using DALI, DSI, and 1–10V interfaces. More recently, these interfaces have been added to sensing technologies such as motion/occupancy detectors and ambient light detection to create a truly intelligent lighting control system. This reference design provides an Intelligent Lighting Control platform which can be used to develop and demonstrate multiple lighting control interfaces.

Figure 1 shows the Intelligent Lighting Control Board with the Z8F6482 MCU.

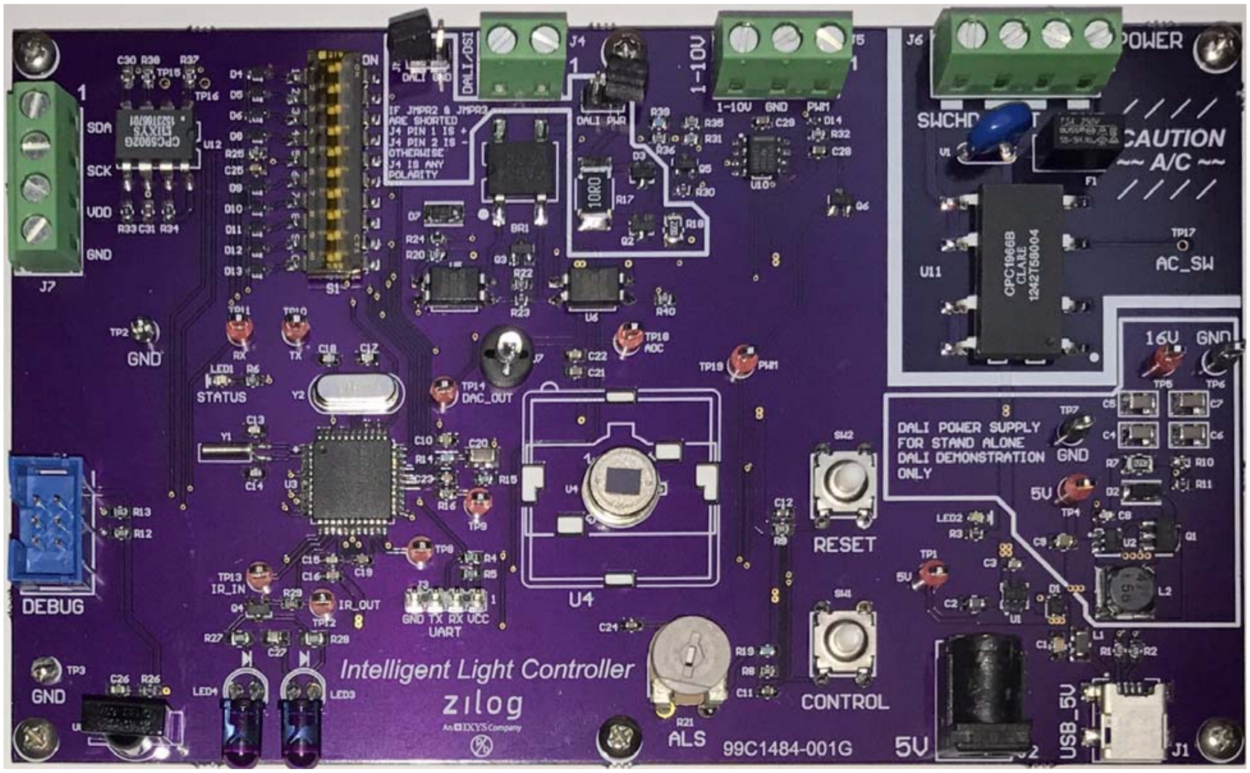


Figure 1. Intelligent Lighting Control Reference Design Board

The reference design hardware supports multiple software configurations and PIR motion detection with digital validation and ZMOTION. Control interfaces such as IR, UART, USB, and isolated I2C are also included. The software and hardware are designed modularly to allow customers to easily include or exclude functions as desired. This reference design is a good starting point for designing an end product.

System Parameters (NVDS)

The overall system behavior depends on the parameters stored in the Z8F6482 MCU's on-chip Nonvolatile Data Storage (NVDS). These parameters can be changed through any of the five configuration interfaces, namely, DIP Switch, IR, RS232, I2C, and USB. Table 1 lists the parameters used in this application. Address 0 is used to identify whether the NVDS contains valid values. This feature is especially useful when running the application for the first time.

Table 1. NVDS Parameters

Addr	Parameter	Type	Value	Remarks
0		Byte	00–FEh	NVDS contain valid values
			FFh	NVDS does not have valid values
1	ucVARS_CtrlMode	Byte	00h	Occupancy Mode
			01h	Occupancy Pass-through Mode
			02h	Vacancy Mode
			03h	Motion Off
2	bVARS_SenseHarsh	Bool	FALSE	Motion Sensitivity = Normal
			TRUE	Motion Sensitivity = Harsh
3	bVARS_HyperSense	Bool	FALSE	Disable Motion HyperSense
			TRUE	Enable Motion HyperSense
4	ulVARS_OutDlyTm	Byte	01–0Fh	1–15 minutes Output Delay Time
5	ucVARS_CtrlIf	Byte	00h	Use 1V–10V Interface
			01h	Use DSI Interface
			02h	Use DALI Interface
			03h	Use Other Interface
6	ucVARS_OnLevel	Byte	00–64h	0%–100% Light Output ON Level
7	ucVARS_PreOffLevel	Byte	00–64h	0%–100% Light Output PRE-OFF Level
8	ucVARS_OffLevel	Byte	00–64h	0%–100% Light Output OFF Level
9	tVARS_LastDip	2-Byte Structure	0000–03FFh	Last known DIP Values

Lighting Control Mode

Two standard lighting control modes are implemented – Occupancy Mode and Vacancy Mode. Two additional lighting control modes are added – Occupancy Pass-through Mode and Motion Off Mode. These additional modes are derivatives of the standard lighting control mode. The control mode in effect is defined by the `ucVARS_CtrlMode` parameter. These modes determine how certain events affect the ON and OFF state of the light/s, including:

- Motion event
- External switch
- Any of the remote control interfaces (i.e., USB, IR, I2C)

The light luminance level is determined by the current lighting profile setting. For more information, refer to the Lighting Control Profile section on page 8.

Another parameter, `ulVARS_OutDlyTm`, which is the amount of time that the light should stay ON from the time of the last event, affects the operation of the lighting control mode.

Occupancy Mode

In Occupancy mode, the light is automatically turned to ON state when motion is detected, and remains ON until the output delay timer, `ulLYT_OutDlyTmr`, expires. For each motion event, this delay timer is reset to `ulVARS_OutDlyTm`.

Occupancy Pass-Through

Occupancy Pass-through mode is similar to Occupancy mode, with an additional feature that allows a person to simply pass by. At the first motion event, a pass-through timer is loaded with 20 seconds and the output delay timer is loaded with 10 seconds. While the pass-through timer is counting, each motion event reloads the output delay timer with 10 seconds. The pass-through timer is not reset upon each motion event. After the pass-through timer expires (20 seconds), normal Occupancy Mode operation is enabled and subsequent motion events cause the output delay timer to be loaded with the `ulVARS_OutDlyTm` value.

Vacancy Mode

In Vacancy mode, initial motion events do not cause the light to turn to ON state. ON state is turned on only when a command is received (through the external switch or any of the remote control interfaces), and remains ON until the specified output delay time expires. This output delay time is reloaded at each motion event.

Motion Off

When the Motion Off mode is selected, motion events do not cause changes to the current light state. The light is turned to ON state only when a command is received (via the external switch or any of the remote control interfaces), and remains ON until the specified output delay time expires. This output delay time is reloaded upon each command received from any of the remote control interfaces. Only the external switch can turn the light OFF.

Lighting Control Interface

The application supports digital and analog lighting control interfaces including DALI, DSI, and 1–10V interfaces. Ballasts typically control fluorescent, HID, and LED lighting which can be connected as single ballast or a network of ballasts. The active interface is defined by the `ucVARS_CtrlIf` parameter stored in NVDS.

1–10V Interface

1–10V is a current sink interface as described in IEC 60929, Annex E. This interface relies on a variable voltage called *control voltage*, to control ballast power. The standard requires that the ballast provide maximum light level when the control voltage is greater than or equal to 10 volts. At a control voltage of less than or equal to 1 volt, the ballast provides minimum light level.

Digital Serial Interface

Digital Serial Interface (DSI) is a unidirectional, non-addressable lighting communication protocol. It is a simple Manchester-encoded serial interface that provides the same level of control as 1–10V. DSI operates at 1200 bps, 1 start bit, 8-bit data, and 2 stop bits. This is the same encoding method as used by DALI. Figure 2 shows a DSI data format.

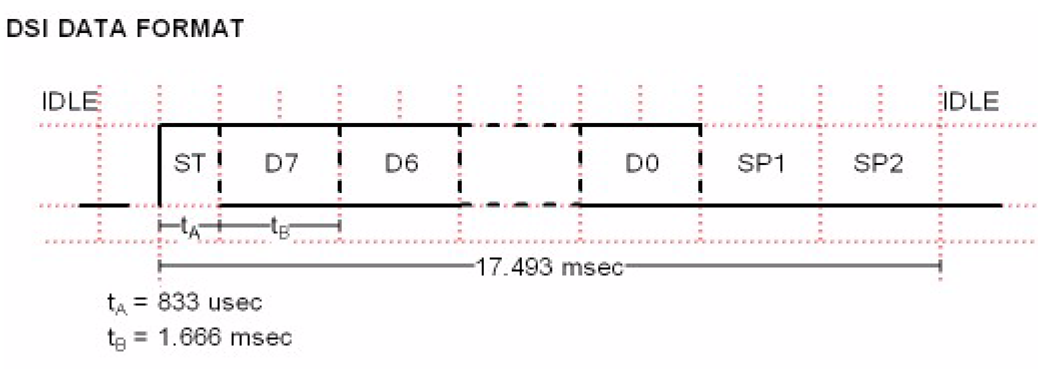


Figure 2. DSI Data Format

As shown in Figure 2, DSI operates in a very different fashion than common UART communication – the idle line and the start and stop bit polarity is low. When Manchester coding is applied, a low-to-high transition on the data line signifies a logic low state; while a high-to-low transition indicates a logic high state. Manchester coding is not applied on the stop bits. The Z8F6482 MCU UART peripherals support this encoding format.

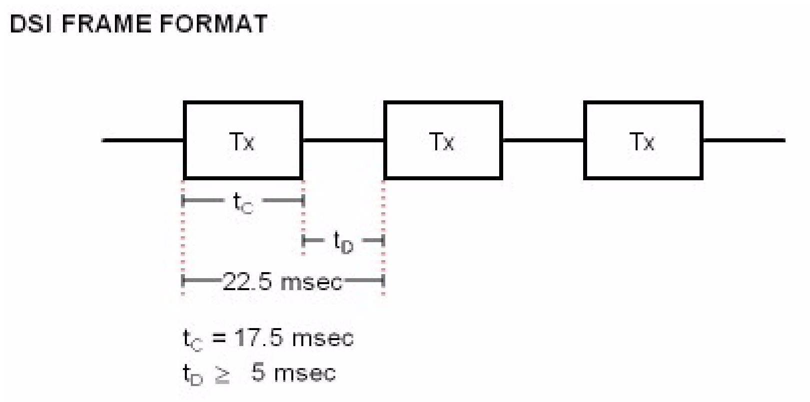


Figure 3. DSI Command Frame

A DSI command is composed of a 1-byte data representing the desired light output level, `00h–FFh`. No response is required from the ballast/s. An idle time greater than or equal to 5 msec is required between each command frame.

Digital Addressable Lighting Interface

Digital Addressable Lighting Interface (DALI) is a bidirectional, addressable lighting communication protocol created as a common platform for DALI-enabled equipment from different manufacturers. The DALI standard is defined in IEC 60929, Annex E (now IEC 62386). DALI operates at 1200 bps, with logic low being represented by a high-to-low transition. It consists of forward frames (command) and backward frames (response), with addressing support. A forward frame consists of 1 start bit, 16 data bits, and 2 stop bits. A backward frame consists of 1 start bit, 8 data bits, and 2 stop bits. An idle time of greater than 9.17 msec is required in between frames. Figures 4 and 5 illustrate a DALI data format and command frame.

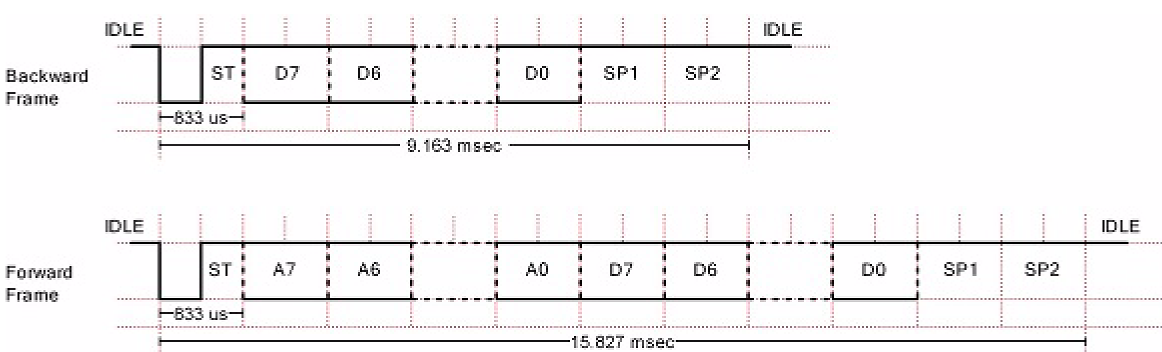


Figure 4. DALI Data Format

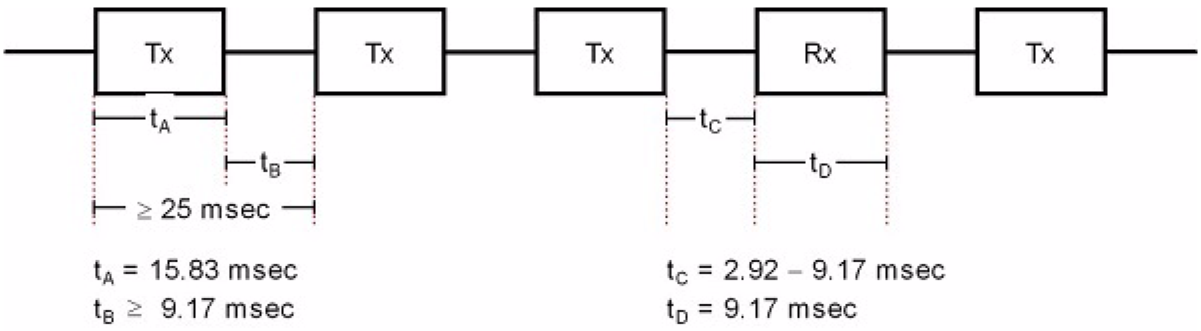


Figure 5. DALI Command Frame

DALI commands are divided into five classifications. This reference design supports all commands except application extended commands. Appendix B. DALI Commands on page 34 shows a list of supported commands.

- Arc Power Control Commands – commands to controls the light output level
- Configuration Commands – commands to set up ballast behavior according to group or scene
- Query Commands – commands to inquire about ballast or light information
- Application Extended Commands – commands specific to device type
- Special Commands – commands to set up ballast addressing

Lighting Control Profile

The lighting control profile determines the light output level at a given state. Three states are provided – ON, PRE-OFF, and OFF. The PRE-OFF state defines the light output level prior to being turned OFF. Lighting levels for each state are defined in NVDS, which are `ucVARS_OnLevel`, `ucVARS_PreOffLevel`, and `ucVARS_OffLevel`. Two Lighting Control Profiles are provided – Dual State and Multi-State.

Dual State Lighting Control Profile: ON, OFF

The dual state lighting control profile employs two light output level states – ON state and OFF state. When the light is turned ON, it stays ON while motion is detected prior to the output delay timer expiry. After the delay timer has expired, the output is driven to OFF state.

Multi-State Lighting Control Profile: ON, PRE-OFF, OFF

The multi-state lighting control profile has three light output level states – ON state, PRE-OFF state, and OFF state. When the light is turned ON, it stays ON while motion is detected prior to the output delay timer expiry. After the delay timer has reached 90%, the light is driven to PRE-OFF state. The output delay timer continues counting until it expires, which will drive the light to OFF state.

Light Gating

When gated ambient light control is enabled, the ambient light sensor, together with the ambient light pot, is used to identify the necessary light output level. The ambient light sensor is used to measure the current lighting condition while the ambient light pot is used to set the light gate set point.

Light gating is only effective from the OFF to ON state transition of light output during Occupancy and Occupancy Pass-through modes. When motion is detected, the application reads the current lighting condition and compares it against the light gate set point. If the current lighting condition is less than the light gate set point, the light will switch from OFF state to ON state. Otherwise, the light will remain in OFF state.

Motion Detection Modes

The following two motion detection parameters can be configured in this application – Sensitivity and HyperSense. Active modes `bVARS_SenseHarsh` and `bVARS_HyperSense` respectively, are defined in NVDS.

Sensitivity

If motion sensitivity is set to Normal, ZMOTION uses the default lens settings for motion detection. If set to Harsh, de-sensitized range and sensitivity settings are used.

HyperSense

If motion hypersense is enabled, the settings are adjusted to increase sensitivity to micro-motion events for a specified hypersense timeout period. This mode is re-triggered upon each motion event and times out when 50% of the delay time has expired. When the hypersense time expires, the settings are returned to their original positions.

Configuration Interfaces

The system configuration can be controlled by five user interfaces: DIP Switches, IR, UART, I2C, and USB. Each of these interfaces can alter parameters stored in NVDS, thereby changing the way the system behaves. Some interfaces offer additional features, as discussed in this section.

DIP Switch

The DIP switch is used to identify the initial and default parameter values. If the values stored in NVDS are invalid (address 0 = 0xFF), the application uses the corresponding DIP switch settings as the active parameter value. Table 2 shows the DIP switch settings.

Table 2. DIP Switch Configuration

SW #	Description	Remarks		
1, 2	Lighting Control Mode	DIP2	DIP1	
		0	0	Occupancy Mode
		0	1	Occupancy Pass-through Mode
		1	0	Vacancy Mode
		1	1	Motion Off
3, 4	Lighting Control Profile	DIP4	DIP3	
		0	0	0% OFF, 100% ON
		0	1	10% OFF, 100% ON
		1	0	0% OFF, 50% PRE-OFF, 100% ON
		1	1	10% OFF, 50% PRE-OFF, 100% ON
5	Light Gating	0		Disabled
		1		Enabled
6	Motion Sensitivity	0		Normal Mode
		1		Harsh Mode
7	Motion Hyper-Sense	0		Disabled
		1		Enabled
8	Output Delay Time	0		1 Minute
		1		5 Minutes
9, 10	Output Control Interface	DIP10	DIP9	
		0	0	1–10V
		0	1	DSI
		1	0	DALI
		1	1	Other Interface

IR Interface Commands

The IR interface enables the user to control and configure the application remotely using an IR remote control. This application implements the Samsung AA59-00472A IR protocol. Table 3 describes the associated functions of each remote control button.

Table 3. Remote Control Configuration

Button	Description
Volume Up	Increases/decreases the output delay time by 1 minute. Pressing this button automatically saves the new value of `ulVARS_OutDlyTm` in NVDS.
Volume Down	
Channel Up	Increases/decreases the current light output level. Pressing this button causes the light output mode to act as if it is in the ON state.
Channel Down	
Power	Immediately forces the light output to OFF state.
0	Restores the current DIP switch settings as the default values in NVDS.
1	Saves the current light output level as ON output level, `ucVARS_OnLevel`, in NVDS.
2	Saves the current light output level as PRE-OFF output level, `ucVARS_PreOffLevel`, in NVDS.

UART Interface Commands

The UART interface provides an asynchronous serial interface that can be used for command execution. This reference design provides a driver module using UART1 (57600-8-N-1) to enable users to easily add their custom commands.

Table 4. Serial Commands

Command	Description
1	Set Sensitivity level
3	Set motion trigger output time
4	Set motion delay time
5	Display Events On/Off (Toggle)
6	Display Event Log
7	Reset Event Log
8	Display API Registers
9	Get Current Ambient Light Level
S	Display Application Settings

I2C Interface Commands

The I2C interface can be used to execute commands and to communicate with other modules over short distances. This reference design provides an I2C slave implementation where users can easily add their custom commands.

Table 5. I2C Commands

Register Address	Description
1	Sensitivity Level
2	Motion Trigger Out Time (MSB)
3	Motion Trigger Out Time (LSB)
4	Motion Delay Time (MSB)
5	Motion Delay Time (LSB)

The Slave address is defined by `I2C_SLAVE_ADD` (default = 0x50) found in `i2c.h`. The data transfer format for reading and writing is shown in Figure 6.

Figure 6. I2C Data Transfer

USB Interface Commands

The USB interface allows the user to control and configure the application via PC. Additional functions such as monitoring motion events, DALI configuration and setup, and RTC setup are also included through the USB interface.

Table 6. USB Commands

Command	Parameters	Description
A	<isGroupAddr><Addr><Value>	Execute DALI Direct Arc Power Control Command
B	<isGroupAddr><Addr><CmdNum>	Execute DALI Indirect Arc Power Control Command
C	<isGroupAddr><Addr><CmdNum>	Execute DALI Configuration Command
D	<Addr><CmdNum>	Execute DALI Query Command
E	<CmdNum><Data>	Execute DALI Special Command
F	<StartAddr><InitOptions>	Execute DALI Assign Short Address Command
G	<StartAddr><InitOptions>	Change DALI Short Address
O	--	returns <dow><mm><dd><yy> in hex
o	<mmddyyyy>	Set RTC Date
P	--	returns <hh><mm><ss> in hex
p	<hhmmss>	Set RTC Time
Q	--	Read RTC Alarm
q	<isDom><DomDow><hhmm>	Set RTC Alarm

Note: All parameters must be in hex string format.

Status LED

The status LED can be used as a feedback mechanism to determine the state of the application. This reference design uses the LED as feedback for every IR command received and to indicate motion events. Table 7 lists the LED behavior when certain events occur.

Table 7. Status LED Activity

LED Behavior	Description
ON for 300 msec	Motion detected
ON for 100 msec	IR request received
Blink X times at 200 msec interval	Output delay time adjusted to X minutes

Hardware

The hardware for this reference design is designed modularly to allow customers to easily add or remove components as required for their end product.

The Z8F6482 Series MCUs include a set of analog, digital, and serial interface peripherals, which make them an excellent choice for intelligent lighting control applications. The UART peripherals support the DALI protocol which greatly simplifies DALI and DSI communication. The DAC and ADC peripherals require minimal external circuitry to enable the 1–10V interface of the application. The on-chip op-amp combined with the ZMOTION Library provides all the tools necessary for reliable PIR motion sensing.

The block diagram in Figure 7 shows the Intelligent Lighting Control design based on the Z8F6482 MCU.

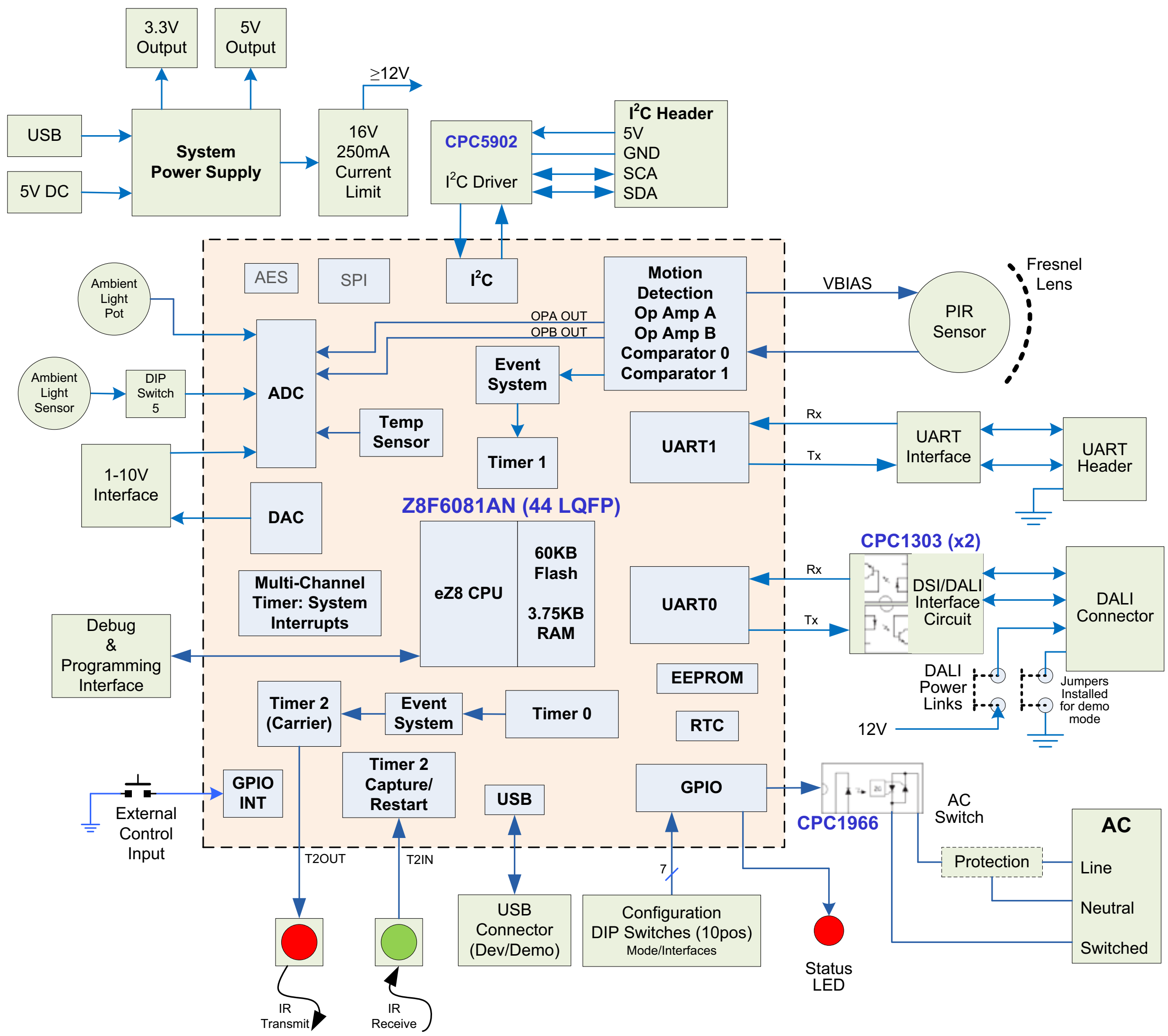


Figure 7. Hardware Block Diagram

Power Supply

The power supply circuit shown in Figure 8 uses two Zilog Power Management ICs, IXD1209 and IXD2121. The main supply voltage of 5V is stepped down to 3.3V using the IXD1209, and functions as the supply for the MCU, Pyrosensor, IR, Switches, and Op-amp. Additionally, the 5V is stepped up by the IXD2121 to 16V to supply the DALI/DSI interface.

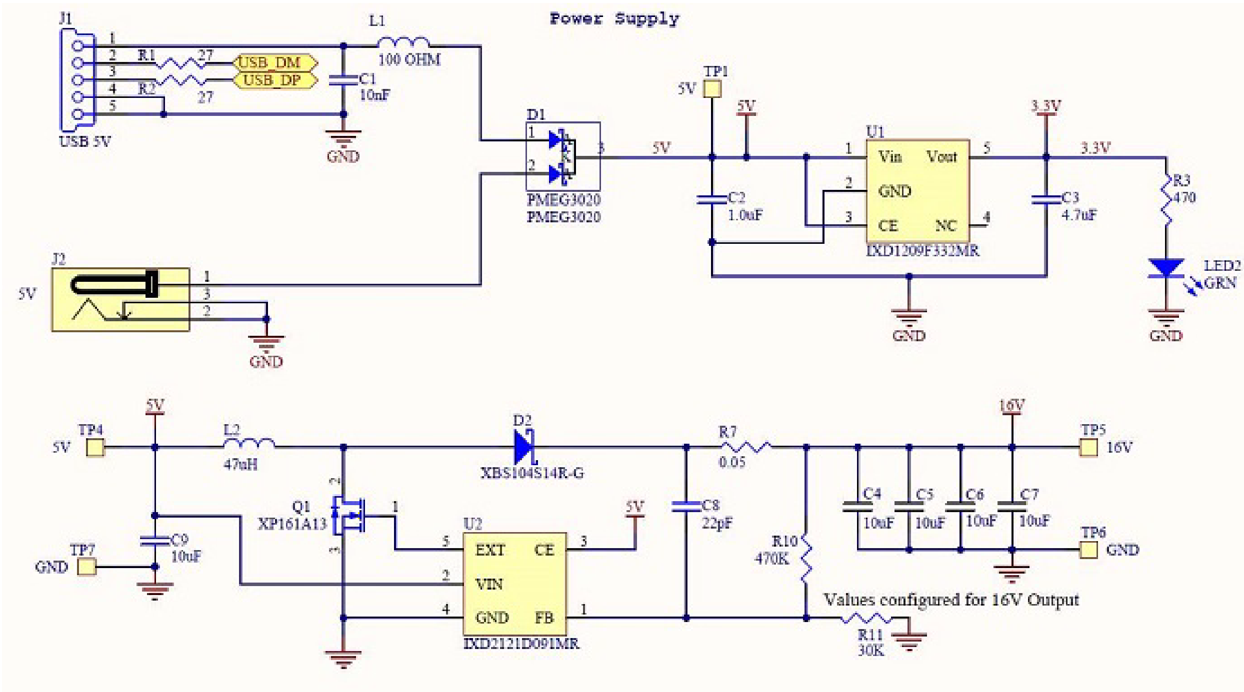


Figure 8. Power Supply Circuit

DSI/DALI Interface Circuit

Figure 9 shows the DALI/DSI circuit. Part of the DALI/DSI interface is the current limiter, which is used when the DALI/DSI line has no power supply, typically during standalone demonstration. To use this circuit, JMPR2 and JMPR3 should be ON.

A bridge-type rectifier diode is used so that the DALI/DSI ballast line can be connected to J4 in any polarity. However, if the current limiter is used, J4 pin 1 is the (+) terminal and J4 pin 2 is the (–) terminal for the ballast.

Two single unidirectional input opto-couplers (IXYS CPC 1303) are used to isolate the MCU from the DALI/DSI ballast.

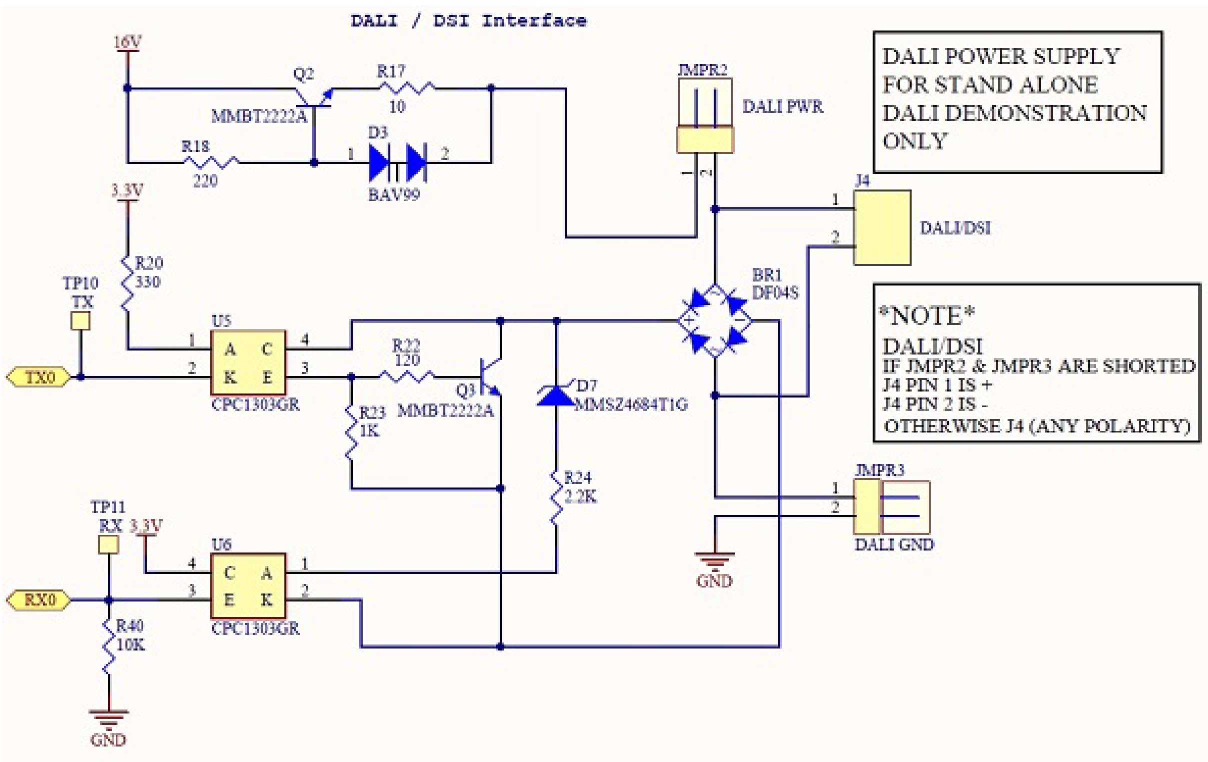


Figure 9. DSI/DALI Circuit

1–10V Interface Circuit

The 1–10V interface is a current sink circuit, as described in IEC 60929, Annex E. The standard requires that the ballast provides full light output when the control voltage is 10V or above. If the control voltage is at or below 1V, the ballast provides its minimum light level.

The voltage level corresponds to the dim level. For example, if the interface circuit pulls down the ballast voltage to 6V, this indicates that the dim level is 60%.

The 1–10V interface circuit in Figure 10 consists of an op-amp, a 2N3904 transistor (current sink), and a voltage divider circuit for ADC feedback.

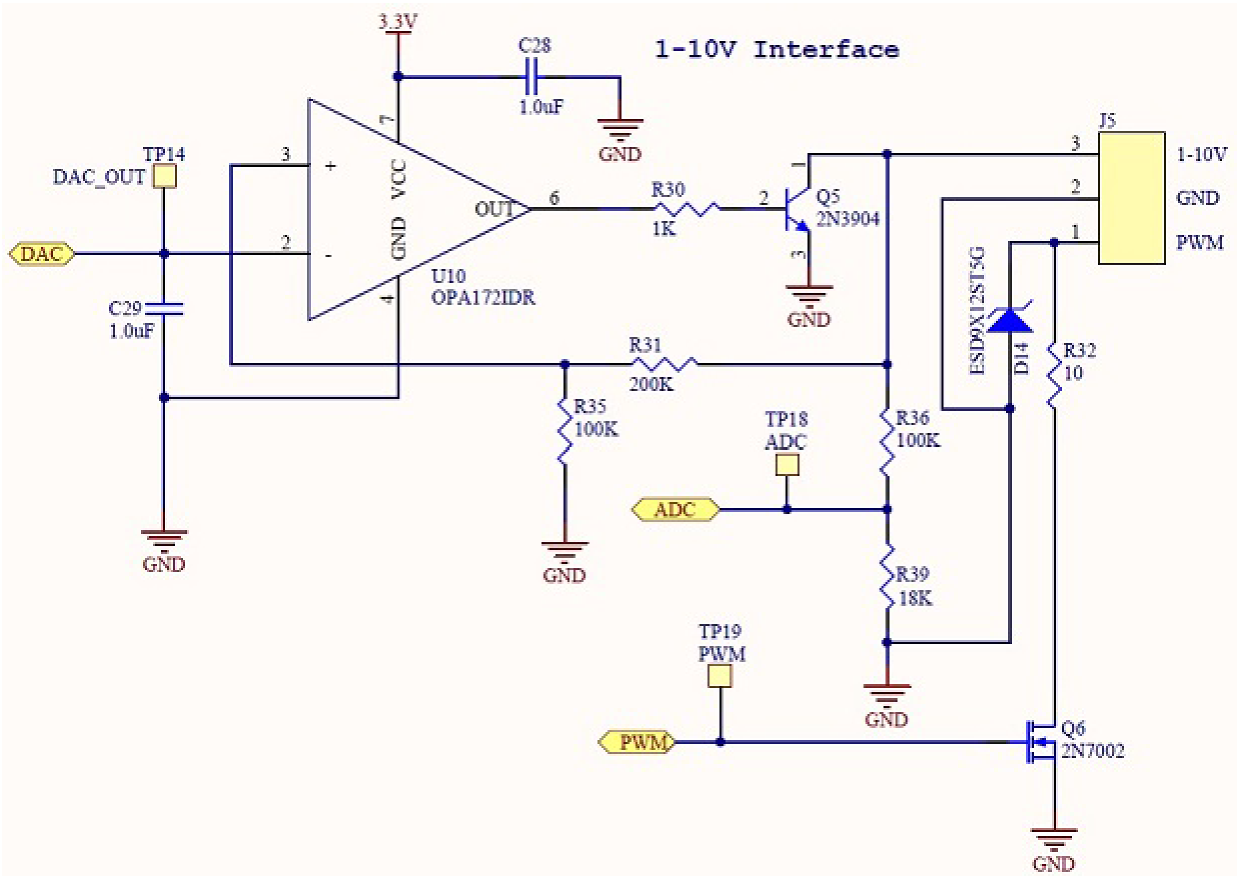


Figure 10. 1–10V Circuit

AC Power Switch Circuit

The AC switch is implemented using a CPC1966B AC Power Switch to control up to a 3A load at 240VAC. It is controlled directly with a GPIO from the MCU and is used in the 1–10V interface to switch off the output completely. Figure 11 shows the AC switch circuit.

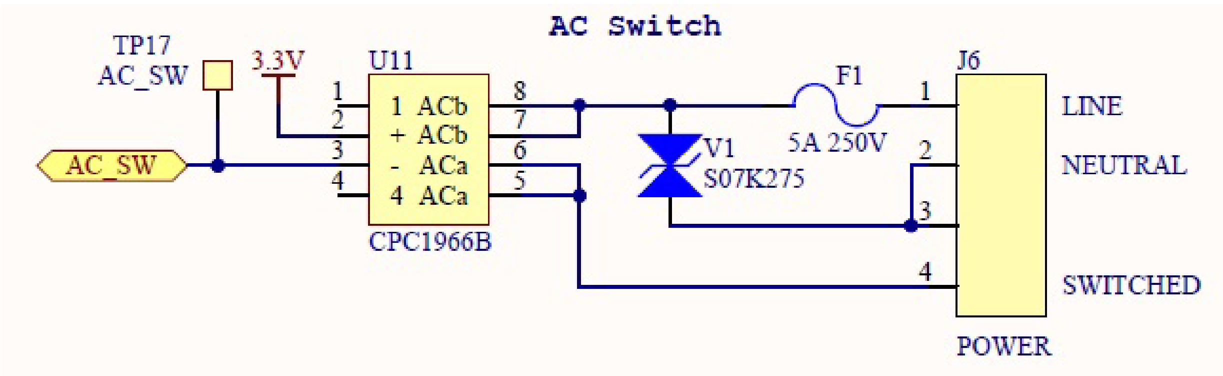


Figure 11. AC Switch Circuit

I2C Driver Circuit

This reference design provides an optically isolated I2C interface using IXYS' CPC5902, which can be utilized when interfacing the I2C device for transmitting DALI/DSI commands, lighting configurations, and dim functions.

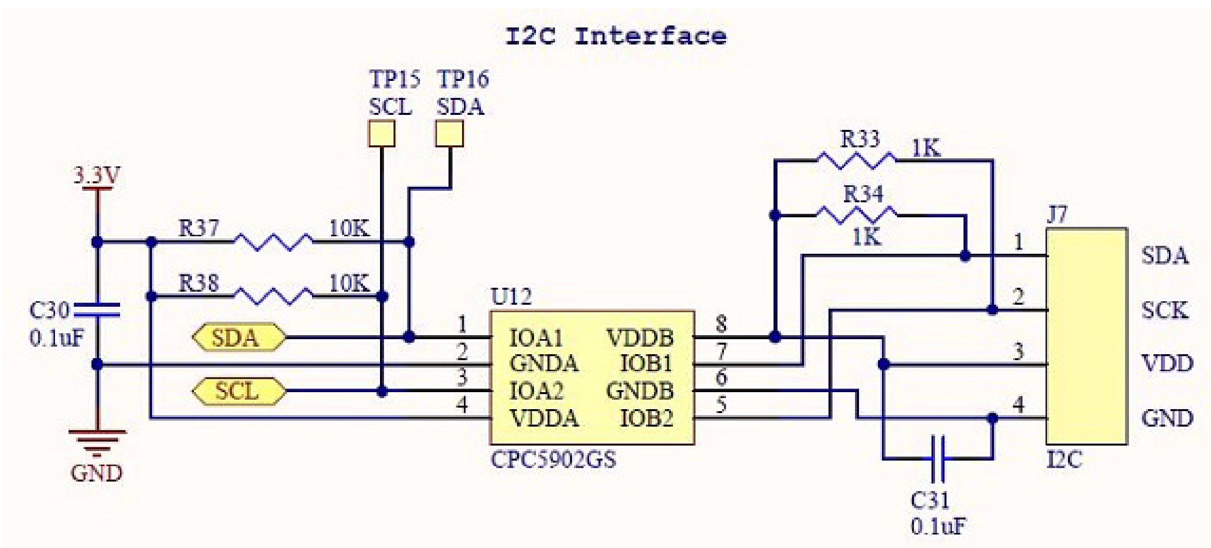


Figure 12. I2C Circuit

Firmware

The firmware is divided into four blocks:

Control Block. The control block handles lighting behavior based on input from the light trigger block and system parameters.

Light Trigger. The light trigger block is a collection of input sources that directly affects current light state and intensity.

Output Interface. The output interface block directly communicates with the ballast/s to control light output state as specified by the control block.

Configuration Block. The system behavior is defined and/or modified by the configuration block.

Figure 13 shows a graphical illustration of the firmware division.

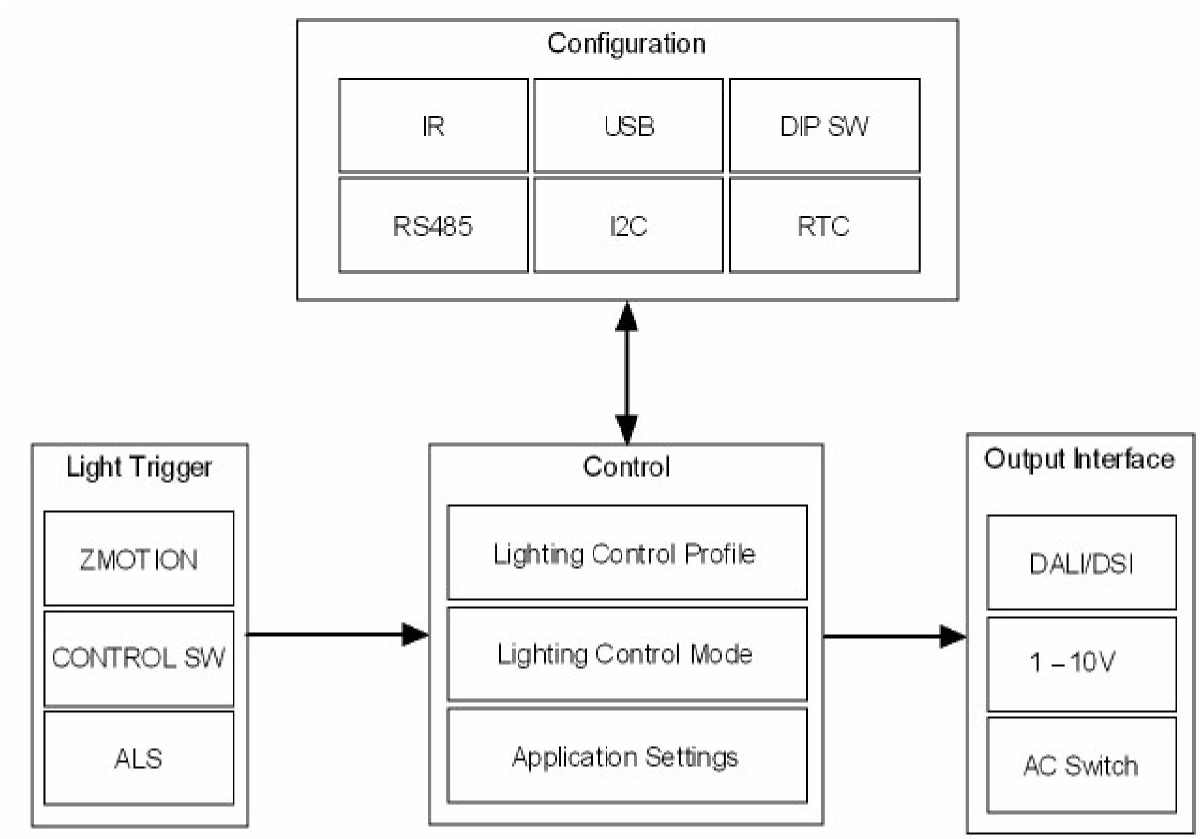


Figure 13. Firmware Block Diagram

Light Trigger Block

The light trigger block is composed of the external switch, ambient light sensor, and motion sensing modules. These modules monitor changes in the external environment and notify the control block of any changes detected.

Motion detection is performed by checking the `ZM_STATUS0` register of the ZMOTION library. For more information, refer to the ZMOTION Engine Library for the F6482 Series (UM0275) user manual. Ambient light level detection is performed by feeding the sensor and pot inputs into the F6482 comparator. When the ambient light level (sensor level) is above the threshold (pot level), the module allows light state changes from OFF to ON state. The control switch is a pushbutton switch connected to an interrupt input to Port D3 of the F6482. Depending on the lighting control mode in effect, the control switch causes the light state to change from OFF to ON state.

Table 8 lists the APIs that affect the light trigger block.

Table 8. Light Trigger API List

API Name	Source File	Description
isrCtrlSwitch	main.c	Interrupt routine to detect external switch events
ProcessMotionEvent	main.c	Performs tasks related to a motion event
AmbientLightSetup	AmbientLight.c	Set up ambient light sensor input on comparator 1
RequestAmbientThreshold	AmbientLight.c	Set up to request ADC sample from the ambient light threshold potentiometer
ProcessLightTH	AmbientLight.c	Debounce and update the ambient light threshold level

Output Block

The output block controls the actual light level output via different lighting control interfaces. It is composed of the DALI, DSI, 1–10V, and AC switch modules.

The DALI protocol is supported by the Z8F6482 UART peripheral, and is compliant with IEC60929-2006. The UART hardware provides character transfers to support the DALI protocol, including biphase encoding and decoding, message formation, and slave address extraction for matching with the comparison address. As such, implementing a DALI interface by software becomes as simple as using the regular UART. This reference design uses UART0 for implementing the DALI protocol.

The DSI protocol works in a similar fashion as the DALI protocol. Therefore, the UART peripheral can also be used to implement the DSI protocol.

The 1–10V interface utilizes the F6482 DAC and ADC peripheral for controlling light output level. The ADC serves as a feedback mechanism to maintain light output level stability.

Table 9 lists the APIs under the control block.

Table 9. Output Block API List

API Name	Source File	Description
ANA_Init	analog.c	Initializes MCU pins for 1–10V interface.
ANA_TurnOnSwitch	analog.c	Turns ON the AC switch.
ANA_TurnOffSwitch	analog.c	Turns OFF the AC switch.
ANA_IsSwitchOn	analog.c	Checks if AC switch is turned ON.
ANA_BeginCalibration	analog.c	Initiates calibration of PWM/DAC output constants.
ANA_CreateCurve	analog.c	Creates the necessary PWM/DAC output value/s to achieve a good light curve when switch light states.
ANA_SetLightLevel	analog.c	Sets light level by initiating the dimming process based on the requested new light output level.
ANA_ProcessEvents	analog.c	Handles 1–10V events – actual execution of dimming or calibration process occurs here.
DSI_Init	dsi.c	Initializes UART0 for DSI operation.
DSI_SendCommand	dsi.c	Sets light level by sending a command to the ballast.
isr_DsiTimer	dsi.c	Clears or sets UART0 break signal.
DALI_Init	dali.c	Initializes UART0 for DALI operation.
DALI_DisablePorts	dali.c	Disables the DALI interface.
DALI_TxIsr	dali.c	Transmits DALI data byte/s.
DALI_RxIsr	dali.c	Reads received data from the DALI bus.
DALI_SendCommand	dali.c	Sends a DALI command frame via UART0.
DALI_DirectArcPowerCtrlCmd	dali.c	Sets ballast light level directly.
DALI_IndirectArcPowerCtrlCmd	dali.c	Executes DALI command 0–31.
DALI_ConfigurationCmd	dali.c	Executes DALI command 32–28.
DALI_QueryCmd	dali.c	Executes DALI command 144–196.
DALI_SpecialCmd	dali.c	Executes DALI command 256–272.
DALI_SelectBallast	dali.c	Searches and selects a DALI ballast.
DALI_AssignShortAddr	dali.c	Assigns successive short addresses to ballasts connected to the DALI network.
DALI_ChangeAddr	dali.c	Changes the address of a ballast with existing address.
DALI_GetGroup	dali.c	Reads the group/s the specified ballast address belongs to.
DALI_AddToGroup	dali.c	Adds ballast with the specified address to the specified group.
DALI_RemoveFromGroup	dali.c	Removes ballast with the specified address from the specified group.
DALI_GetSceneLevel	dali.c	Reads light level specified for the specified scene.
DALI_AddScene	dali.c	Assigns the specified light level as scene level for the specified ballast address.
DALI_RemoveScene	dali.c	Removes specified ballast address from the specified scene.

Configuration Block

The configuration block is composed of the DIP Switch, IR, USB, RTC, I2C, and RS232 modules. These modules are composed of commands that directly affect the system parameters defined in NVDS. Some commands may also serve as a light trigger. Table 10 lists the APIs under the configuration block:

Table 10. Configuration Block APIs

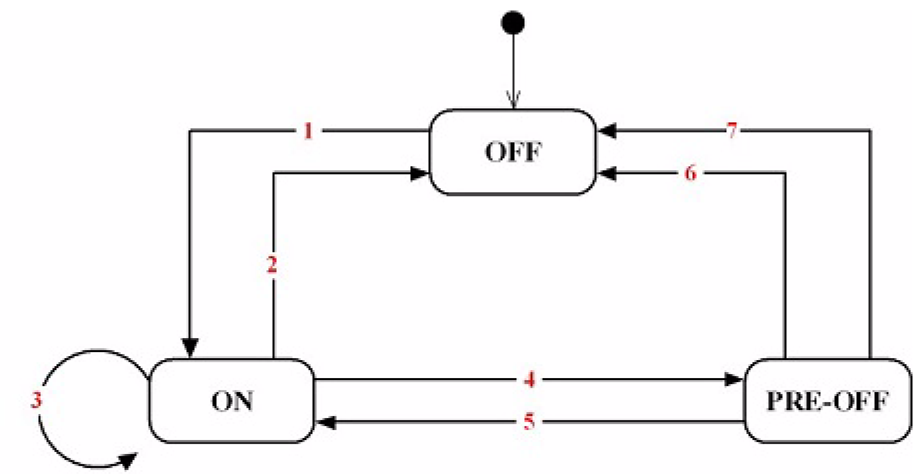
API Name	Source File	Description
VARS_Init	vars.c	Initializes NVDS and loads application settings.
VARS_CheckDip	vars.c	Checks DIP switch and applies necessary changes.
DIP_GetSettings	vars.c	Reads DIP switches.
VARS_LoadDipSettings	vars.c	Loads application settings from the DIP switches.
VARS_LoadNvdsSettings	vars.c	Loads application settings from the NVDS.
VARS_UpdateNvds	vars.c	Updates NVDS values.
VARS_SetMotionSensitivity	vars.c	Sets ZMOTION parameters to enable/disable harsh sensitivity.
VARS_SetMotionHyperSense	vars.c	Sets ZMOTION parameters to enable/disable hypersense.
VARS_SetLightingControlProfile	vars.c	Sets lighting control profile.
VARS_UpdateChar	vars.c	Updates 8-bit data in NVDS.
VARS_UpdateInt	vars.c	Updates 16-bit data in NVDS.
VARS_UpdateLong	vars.c	Updates 32-bit data in NVDS.
IR_InitRx	ir.c	Initializes TIMER2 for IR receive mode.
IR_InitTx	ir.c	Initializes TIMER0 for IR transmit mode.
IR_Init	ir.c	Initializes IR Rx and Tx.
IR_TxKey	ir.c	Initiates data transmission via IR.
Timer0_Isr	ir.c	Handles TIMER0 reload event for transmitting IR data bits.
Timer2_Isr	ir.c	Handles TIMER2 interrupt. Transmit Mode: Returns to receive mode. Receive Mode: Reads the capture value.
IR_ProcessEvents	ir.c	Performs Manchester format decoding of the captured value during receive mode. Also executes IR command when a valid key is received.
CheckCommand	SerialConfig.c	Checks for any received serial data and executes its corresponding command.
RTC_Init	rtc.c	Initializes the RTC.
RTC_SetTime	rtc.c	Sets RTC time registers.
RTC_SetDate	rtc.c	Sets RTC date registers.
RTCReadTime	rtc.c	Reads current RTC time.

Table 10. Configuration Block APIs (Continued)

API Name	Source File	Description
RTCReadDate	rtc.c	Reads current RTC date.
RTC_SetAlarm	rtc.c	Sets RTC alarm registers.
USB_CheckCommand	usbConfig.c	Checks for any received USB data and executes its corresponding command.

Control Block

The control block is the main application layer responsible for controlling system behavior based on various system inputs. This block primarily integrates the operations presented in the Lighting Control Mode, Lighting Control Profile, and Light Gating sections. Figure 14 shows the state diagram of the control block.



1	Passthrough ‖ Occupancy Mode: Motion detected [PASSTHRU] DLYTMR = 10 sec; PASSTMR = 20 sec [OCCUPANCY] DLYTMR = Delay Time Motion Off ‖ Vacancy Mode: Control switch pressed DLYTMR = Delay Time
2	Vacancy Mode: Control switch pressed ‖ (PASSTMR == 0 && DLYTMR == 0)
3	Motion detected [PASSHTRU] (PASSTMR > 0) DLYTMR = 10 sec (PASSTMR == 0) DLYTMR = Delay Time [ELSE] DLYTMR = Delay Time
4	Passthrough Mode: PASSTMR == 0 && DLYTMR <= 10% Delay Time Else: DLYTMR <= 10% Delay Time
5	Motion detected DLYTMR = Delay Time
6	Control switch pressed
7	DLYTMR == 0

Figure 14. Control Block State Diagram

Table 11 lists APIs for the control block.

Table 11. Control Block APIs

API Name	Source File	Description
LYTCTRL_Init	lytctrl.c	Initializes ballasts to OFF state.
LYTCTRL_IncOnLevel LYTCTRL_DecOnLevel	lytctrl.c	Increments/decrements current light level. This routine forces the current light output to ON as if motion is detected in Occupancy mode and light gating is disabled.
LYTCTRL_SaveOnLevel	lytctrl.c	Saves the current light level as ON level to NVDS.
LYTCTRL_SavePreOffLevel	lytctrl.c	Saves the current light level as PRE-OFF level to NVDS.
LYTCTRL_IncDelayTime LYTCTRL_DecDelayTime	lytctrl.c	Increments/decrements output delay time and updates NVDS. This routine forces the current light output to ON as if motion is detected in Occupancy mode and light gating is disabled.
LYTCTRL_ToggleState	lytctrl.c	If light output is in OFF or PRE-OFF state, forces the current light output to ON as if motion is detected in Occupancy mode and light gating is disabled. If light output is in ON state, forces the current light output to OFF as if the delay timer in Occupancy mode has expired.
LYTCTRL_GoToOnState LYTCTRL_GoToPreOffState LYTCTRL_GoToOffState	lytctrl.c	Switches current light state to ON/PRE-OFF/OFF state.
LYTCTRL_ProcessEvents	lytctrl.c	Updates current light status based on recent events including motion, delay timeout, 90% delay timeout, external switch, and pass-through timeout.

Testing Procedure

The following ballasts were used to test this reference design:

- Tridonic PCA TC ECO Xitec II Ballast with Biax T/E ECO 42W CFL for DALI and DSI
- Pacific 1–10V LED Driver with High Power LED for 1–10V

To test this reference design, connect the DALI/DSI/1–10V ballast to its designated connectors, as shown in Figures 15 through 17.

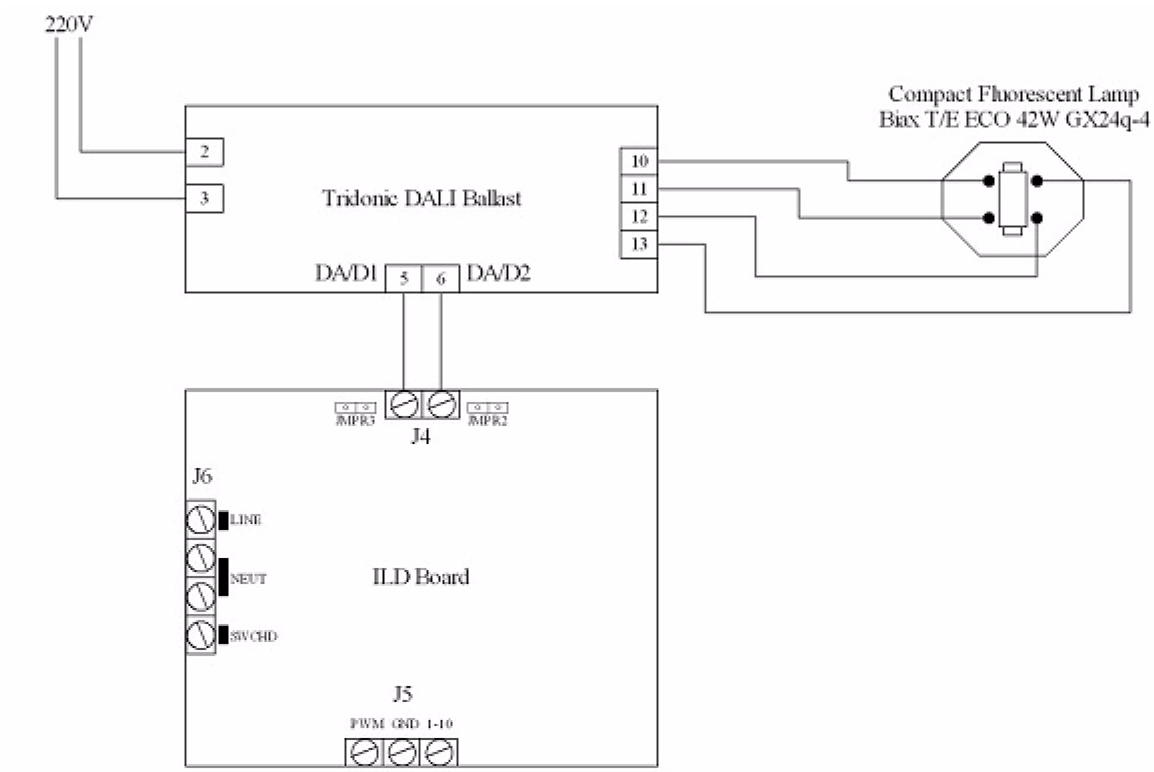


Figure 15. Intelligent Lighting Control Reference Design Test Setup Using DALI Ballast

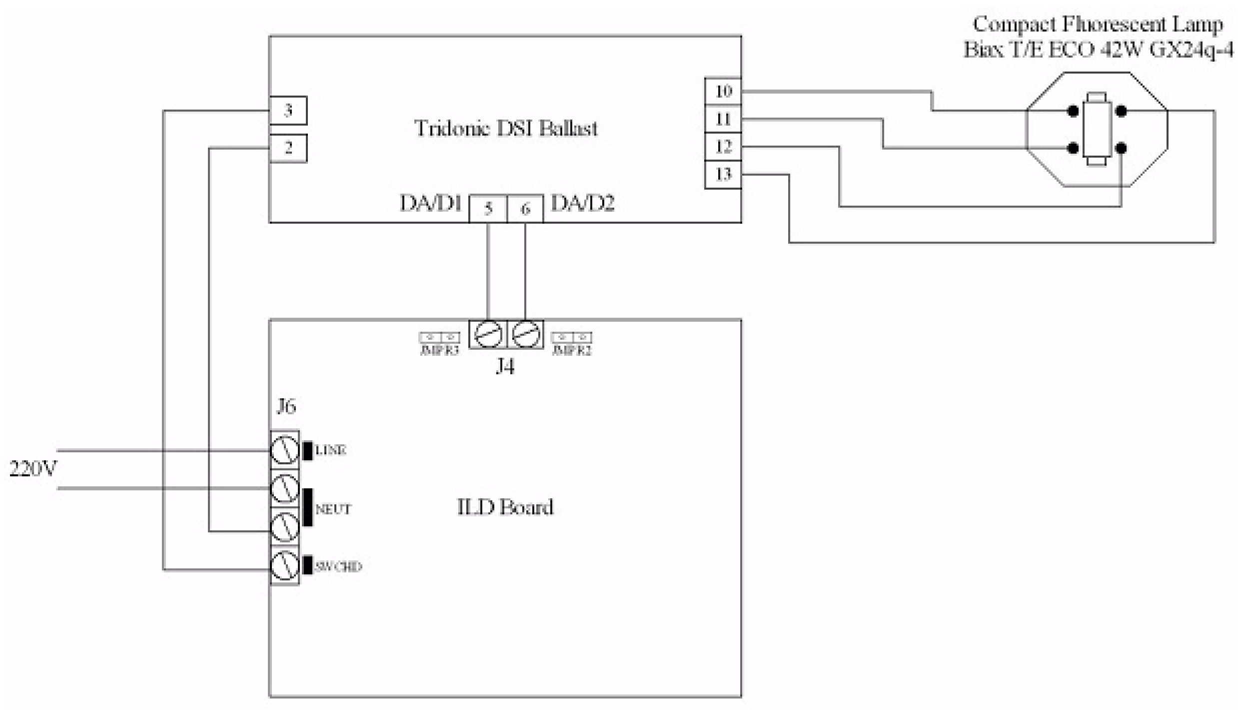


Figure 16. Intelligent Lighting Control Reference Design Test Setup Using DSI Ballast

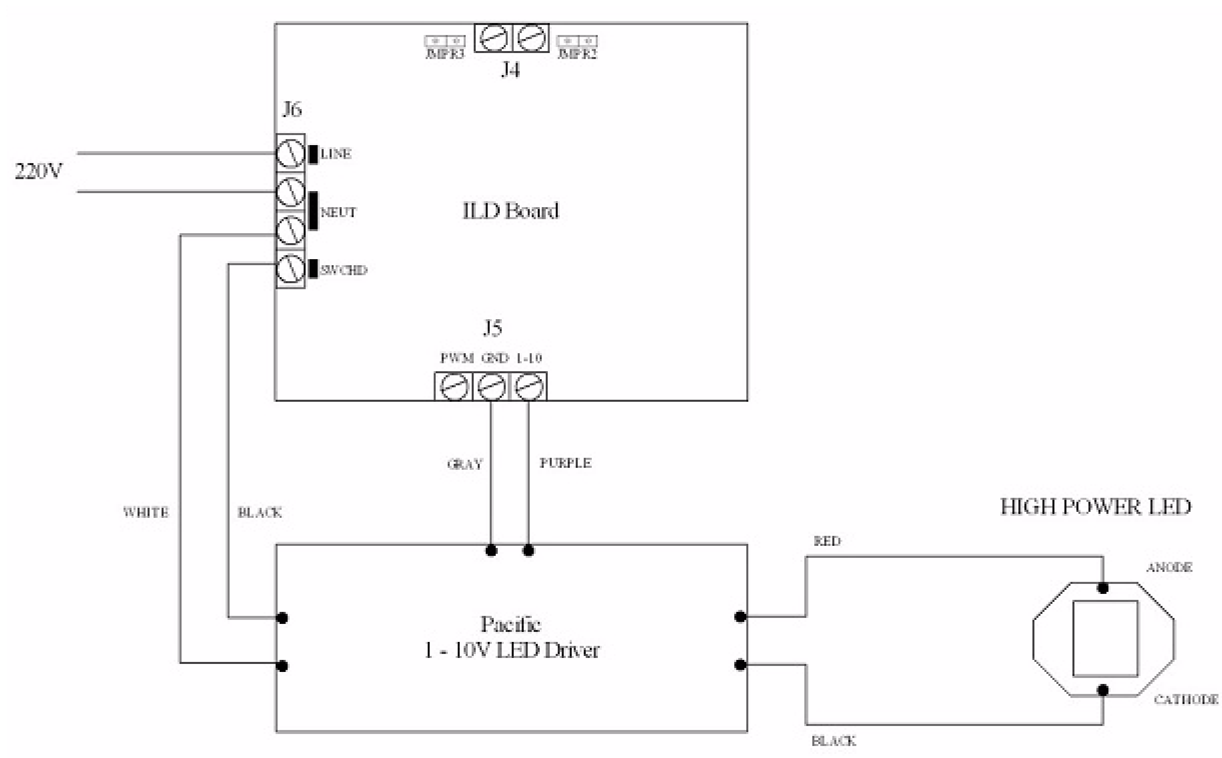


Figure 17. Intelligent Lighting Control Reference Design Test Setup Using 1–10V Ballast

Observe the following procedure to test the Intelligent Lighting Control reference design:

1. Connect the USB SmartCable to the DBG terminal of the board.
2. Power the board using a 5V DC supply. The power LED indicator (LED2) turns on when power is applied to the board.
3. Download and extract RD0038-SC01.zip from zilog.com.
4. Open the ZDS II–Z8 Encore v5.2.2 (or later) IDE. From the **File** menu, select **Open Project...**.
5. Navigate to the extracted folder, select the **RD0038-SC01.zdsproj** project and click **Open** to open the project.
6. From the **Build** menu, select **Rebuild All** to rebuild the project.

> **Note:** The application timing in the supplied code is reduced such that 1 minute = 10 seconds. This value can be changed in `systick.h` via the `APPTIME_DEMO_MODE` preprocessor directive.

7. From the **Debug** menu, select **Download Code** to load the program to the MCU.

8. After programming is complete, from the **Debug** menu, select **Stop Debugging**.
9. Remove power from the board.
10. Disconnect the 6-pin ribbon cable from the DBG connector on the board.
11. Set DIP1–DIP8 to occupancy mode, 0% OFF state, 100% ON state, light gating disabled, normal motion sensitivity, disabled motion hyper-sense, 1 minute output delay time. See Figure 18.

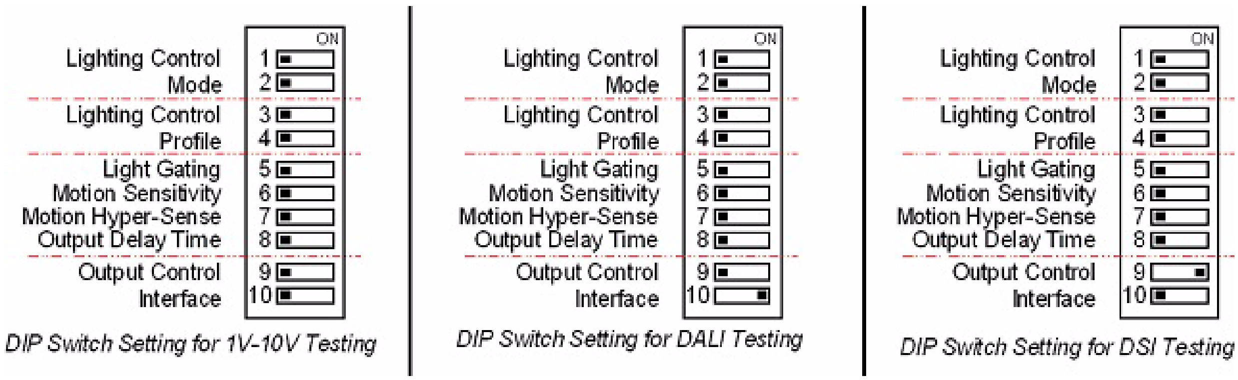


Figure 18. DIP Switch Setting for Testing

12. Set DIP9 and DIP10 to select the desired output control interface (1–10V, DALI, or DSI). See Figure 18.

Note: If using DALI bus power supplied with the board, place a jumper at JMPR3 (DALI GND) and another jumper at JMPR2 (DALI PWR).

Warning: THIS INTELLIGENT LIGHTING CONTROL BOARD USES HIGH VOLTAGE (100–240VAC) FOR SOME OF THE LIGHTING CONFIGURATIONS. DISCONNECT ALL POWER SUPPLIES PRIOR TO HANDLING THE BOARD.

13. Power the ballast (220V for DALI/DSI and 100–240V for 1–10V).
14. Power the board using a 5VDC supply.
15. Wait for the STATUS LED (LED1) to stop blinking. This will indicate that PIR stabilization has completed.
16. Upon each motion event, the STATUS LED will blink for 1 second.

Electrical Specifications

Table 12 lists the electrical specifications of the Intelligent Lighting Control reference design.

Table 12. Electrical Specification

Parameter	Min	Typical	Max	Units	Notes
Board Supply Voltages		5		VDC	
AC Switch Load (V)			240	VAC	
AC Switch Load (I)			3	A	
DALI Bus Power			250	mA	Current Limit
I2C Interface	2.7		5.5	V	

Packaging

Figures 19 and 20 display the assembly diagram of the Intelligent Lighting Control reference design.

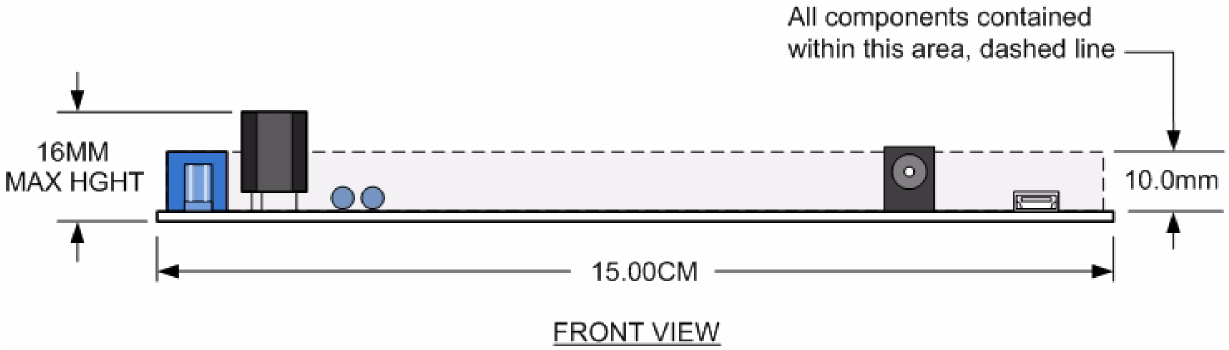


Figure 19. Assembly Diagram (Profile)

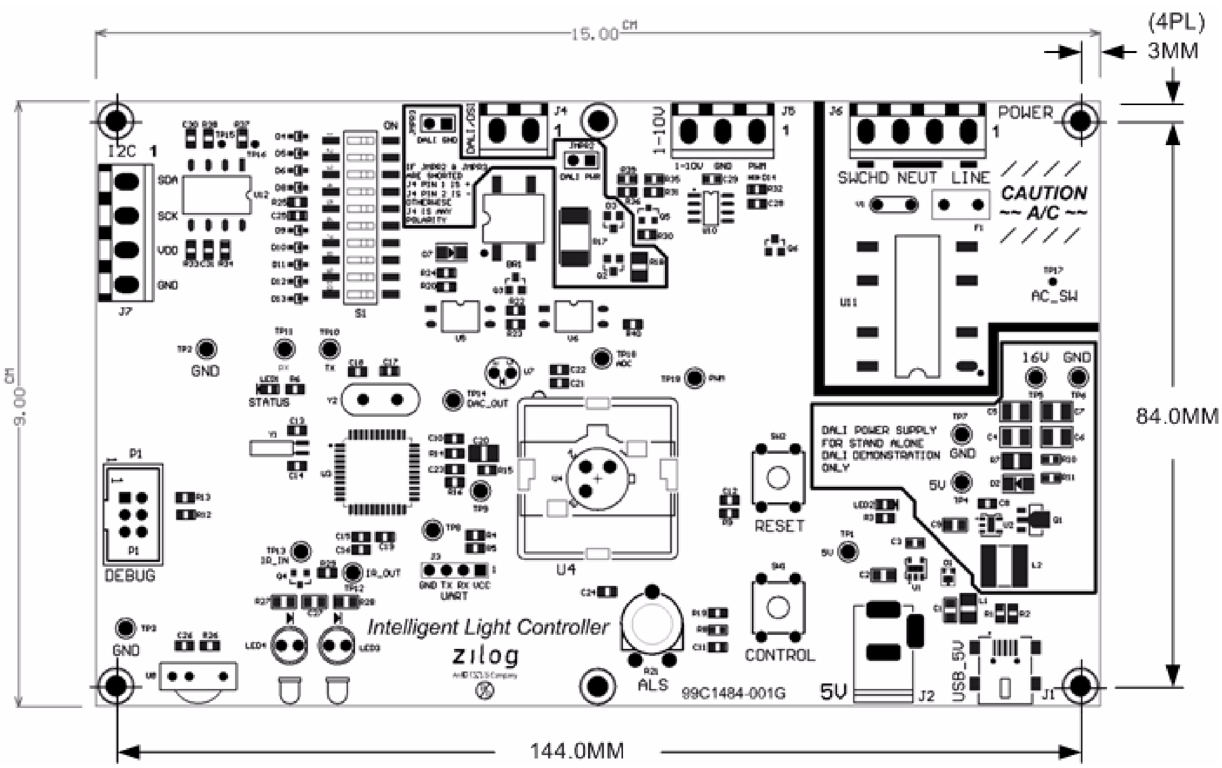


Figure 20. Assembly Diagram (Top View)

Ordering Information

The products associated with this ZMOTION Intelligent Lighting Control reference design are available as a kit and can be ordered from the Zilog Store using the part number listed in Table 13.

Table 13. Ordering Information

Part Number	Description	Store Product ID
ZMOTIONL200ZRDG	ZMOTION Intelligent Lighting Control Reference Design Kit	RD10046

Kit Contents

The ZMOTION Intelligent Lighting Control Reference Design Kit contains the following items:

- Intelligent Lighting Control Board
- USB SmartCable
- USB cable (A male to Mini-B male)

Related Documentation

The documents associated with the ZMOTION Intelligent Lighting Control reference design are listed in Table 14. Each of these documents can be obtained from the Zilog website by clicking the link associated with its document number.

Table 14. Related Documentation

Document Number	Description
RD0038	This ZMOTION Intelligent Lighting Control Reference Design document
UM0275	ZMOTION Engine Library for the F6482 Series User Manual
CPC1303	Optocoupler with Single-Transistor Output Datasheet
CPC5902	Optically Isolated I2C Bus Repeater Datasheet
CPC1966B	Optically Isolated AC Switches Datasheet
PS0338	High Speed LDO Regulator with ON/OFF Control (IXD1209) Datasheet
PS0361	Step-Up DC/DC Controller (IXD2121) Datasheet
AN0408	IR Decoder using the Timer Capture Restart Mode of Z8 Encore! XP Series MCUs Application Note
AN0411	Z8F6482 MCU USB Peripheral Application Note

zilog
Embedded in Life
An IXYS Company

Appendix A. Schematic Diagrams

Figures 21 and 22 show the schematic diagrams for the ZMOTION Intelligent Lighting Control reference design.

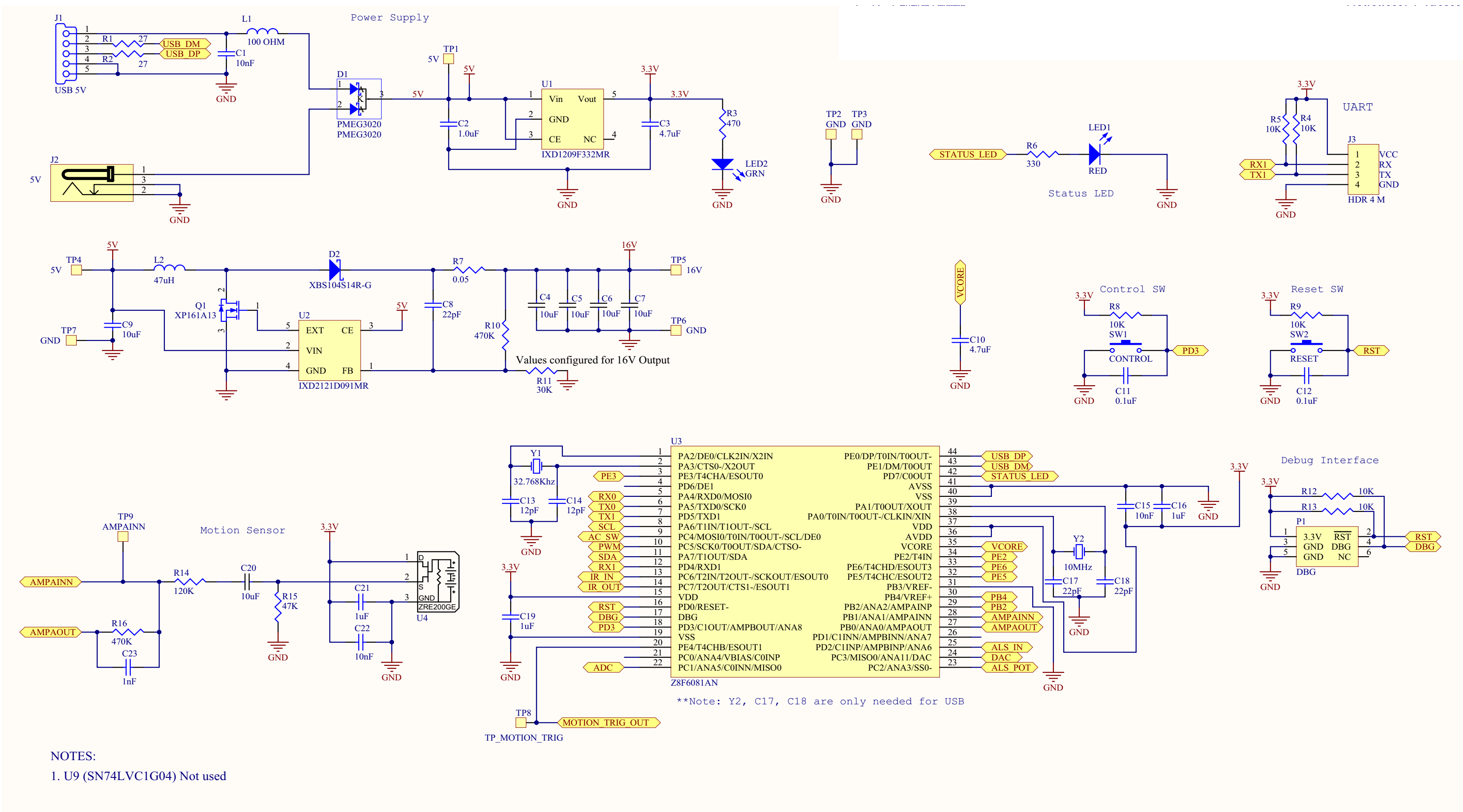


Figure 21. Schematic Diagram, #1 of 2

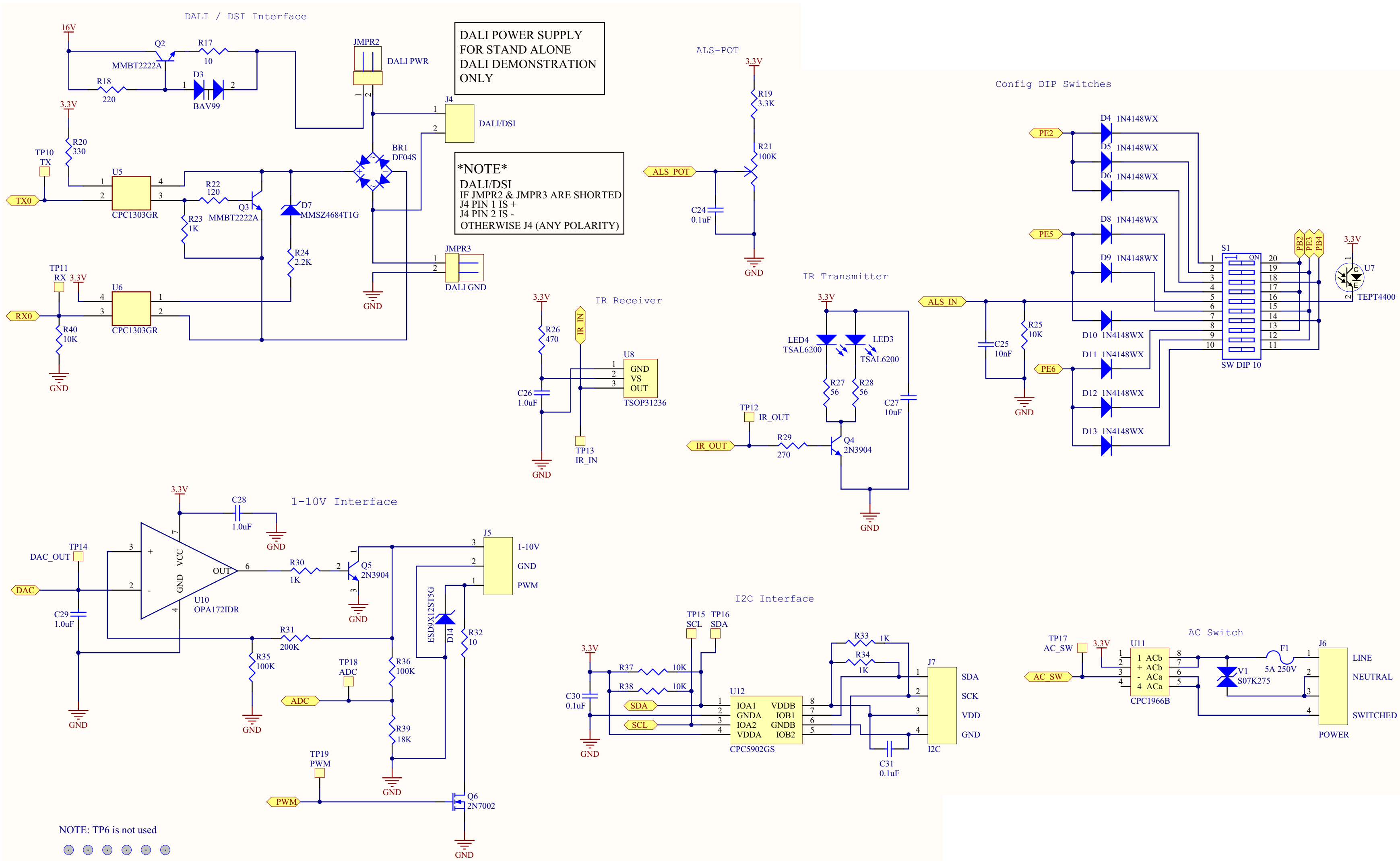


Figure 22. Schematic Diagram, #2 of 2

Appendix B. DALI Commands

Table 15 lists the DALI commands supported in this reference design.

Table 15. DALI Commands

Number	Command Name
	Direct Arc Power Control
0	OFF
1	UP
2	DOWN
3	STEP UP
4	STEP DOWN
5	RECALL MAX LEVEL
6	RECALL MIN LEVEL
7	STEP DOWN AND OFF
8	ON AND STEP UP
16–31	GO TO SCENE 0–15
32	RESET
33	STORE ACTUAL LEVEL IN DTR
42	STORE DTR AS MAX LEVEL
43	STORE DTR AS MIN LEVEL
44	STORE DTR AS SYSFAIL LEVEL
45	STORE DTR AS POWER ON LEVEL
46	STORE DTR AS FADE TIME
47	STORE DTR AS FADE RATE
64–79	STORE DTR AS SCENE 0-15
80–95	REMOVE FROM SCENE 0-15
96–111	ADD TO GROUP 0–15
112–127	REMOVE FROM GROUP 0–15
128	STORE DTR AS SHORT ADDRESS
144	QUERY STATUS
145	QUERY BALLAST
146	QUERY LAMP FAILURE
147	QUERY LAMP POWER ON
148	QUERY LIMIT ERROR
149	QUERY RESET STATE
150	QUERY MISSING SHORT ADDRESS
151	QUERY VERSION INFO
152	QUERY CONTENT DTR

Table 15. DALI Commands (Continued)

Number	Command Name
153	QUERY DEVICE TYPE
154	QUERY PHYSICAL MIN LEVEL
155	QUERY POWER FAILURE
160	QUERY ACTUAL LEVEL
161	QUERY MAX LEVEL
162	QUERY MIN LEVEL
163	QUERY POWER ON LEVEL
164	QUERY SYSTEM FAILURE LEVEL
165	QUERY FADE TIME/RATE
176–191	QUERY SCENE 0-15 LEVEL
192	QUERY GROUP 0–7
193	QUERY GROUP 8–15
194	QUERY RANDOM ADDRESS (H)
195	QUERY RANDOM ADDRESS (M)
196	QUERY RANDOM ADDRESS (L)
256	TERMINATE
257	DATA TRANSFER REGISTER
258	INITIALISE
259	RANDOMISE
260	COMPARE
261	WITHDRAW
264	SEARCH ADDRESS (H)
265	SEARCH ADDRESS (M)
266	SEARCH ADDRESS (L)
267	PROGRAM SHORT ADDRESS
268	VERIFY SHORT ADDRESS
269	QUERY SHORT ADDRESS

Customer Support

To share comments, get your technical questions answered, or report issues you may be experiencing with our products, please visit Zilog's Technical Support page at http://support.zilog.com.

To learn more about this product, find additional documentation, or to discover other facets about Zilog product offerings, please visit the Zilog Knowledge Base at http://zilog.com/kb or consider participating in the Zilog Forum at http://zilog.com/forum.

This publication is subject to replacement by a later edition. To determine whether a later edition exists, please visit the Zilog website at http://www.zilog.com.

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

HIGH VOLTAGE (100–240 VAC) PRESENT ON THIS BOARD. DISCONNECT ALL POWER SUPPLIES PRIOR TO HANDLING.

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