Design and Realization of Airfield Lighting Terminal Controller

Hongliang Wang¹,*, HongFei Wang², b, Huazhu Wu¹, *

¹Institute of Electrical Information, Dalian Jiaotong University, Dalian 116028, China
²Institute of Energy and Power Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, China

*Corresponding author: dlwhzh @163.com, *976033186@qq.com,
b1017308757@qq.com

Abstract. Aiming at the characteristics of airfield lighting and the military control requirements of the airport lighting for airfield warhead lighting, this paper elaborates the hardware circuit, software development and design process of the navigation light terminal control node through the research of wireless communication technology and LED light source driving technology. The hardware adopts CC2530 as the microprocessor to add the Radiofrequency (RF) amplification chip CC2591 and design the navigation light terminal control nodes. The Full Function Device (FFD) has the routing function, which not only can realize the network self-healing function, but also can further expand the network coverage. Terminal control nodes software program development is based on the Z-stack protocol stack for light-driven control program and navigation light fault diagnosis program design.

1. Introduction

The traditional navigation light system adjusts the brightness of a single light and switch control through a dimmer, which is a single lamp without any control function itself. In addition, most of the domestic airport navigation light fault diagnosis using manual inspection method with slow, low efficiency, fault maintenance timely can not be guaranteed [1]. Through the control of wireless sensor network achieve flexible control airport navigation lights and fault inspection. It greatly improve the airfield lighting operational efficiency to ensure the reliability and stability.

Airfield lighting wireless ad-hoc network control system consists of three parts: Airfield lighting terminal control nodes, coordinator nodes and monitoring center. The terminal node controller is responsible for the status data acquisition of navigation lights, at the same time, the analysis and execution of control commands, including the detection of the temperature, illuminance and fault of the airfield lighting, the driving and guidance control of navigation lights. The system uses a mesh topology network increasing the coverage of the network and the configuration of each terminal node has routing capabilities. Airfield lighting system uses LED light instead of the traditional navigation light source and equipped with independent power supply, which can reduce system power consumption, low power consumption, networking flexibility, network self-healing and other functions.
2. Terminal Control Node
Terminal control nodes with network routing forwarding function, in order to achieve multi-hop information forwarding and network self-healing function [2]. The terminal node is generally far away from the network coordinator. As a network edge of branch device, the terminal node does not have the function of being a routing node and a parent node, and is mainly responsible for data acquisition and control of the monitoring object. The terminal node adopts CC2530 as the controller, which is responsible for the status and driving adjustment of the airfield lights, and the terminal of the airfield lights can automatically and periodically detect the illumination, fault information, lighting temperature and other information of the airfield lights and upload to the coordination device.

3. The Hardware Design of Terminal Control Node
Terminal control nodes of airfield lighting adopts the chip of TI Company CC2530F256 which is used for 2.4GHz RF communication microprocessors. CC2530 integrates 8051 MCU core functions and RF transceiver functions. Its integrated design can simultaneously satisfy the system nodes RF communication control and ZigBee wireless networking communication functions [3]. Due to the large number of airfield lightings control terminals and limited RAM memory of the CC2530 RF chip, the coordinator nodes can not satisfy the storage and information efficient processing. Therefore, the system coordinator nodes adopts the STM32F103RBT6 based on the ARM Cortex-M3 core master as coordinator nodes. Airfield terminal control nodes hardware architecture shown in Fig.1.

![Structure diagram of the terminal control node hardware](image)

Airfield lighting terminal control node consists of two parts: the control circuit and drive circuit. The control circuit is based on the design of the CC2530F256 microprocessor, an external temperature sensor, illumination sensor and relay, to achieve monitoring and control of the airfield lights. Drive circuit by the LM3464 driver chip with peripheral circuits and lamp power supply components to driver the LED navigation light smartly, PWM dimming control, voltage current detection and fault feedback. LED light board temperature detection by DS18b20 temperature sensor, in order to achieve over-temperature protection of navigation lights and provide data reference. Illumination sensor TSL2561 can provide information about the illumination of the airfield lights, which in turn can adjust the brightness of the navigation lights reasonably [4]. CC2530 microprocessor through control the relay, and then turn the lights to achieve the control switch. The sensor transmits the collected information of the navigation light temperature and illuminance to the CC2530 microprocessor, and
the microprocessor processes the data and sends the data to the coordinator nodes through the ZigBee wireless network. Driving circuit through the microprocessor send the PWM signal to achieve the LED airfield lighting dimming control. CC2530 microprocessor chips by detecting the LED driver current voltage and fault signals determine whether the LED navigation lights fault. The use of TI's RF power amplifier chips can increase the distance between nodes, not only to meet the system's communication distance requirements, but to reduce the delay in the transmission of data.

4. The Software Design Terminal Control Node

IAR Embedded Workbench are used CC2530F256 microprocessor system software development tools (abbreviation EW). Z-Stack is loaded into a project based on the IAR development environment and fully supports the IEEE802.15.4/ZigBee CC2530 system-on-chip solution. Powerful IAR Embedded Workbench providing compiled download capabilities, which can be combined with the programmer to track debug single step, monitor on-chip registers and Flash data. Z-Stack protocol stack is based on the architecture of the operating system. It adopts the event polling mechanism. When all layers initialized, the system enters low-power operation mode. If an event occurs, the system wakes up and enters the interrupt processing event, then ends the interrupt event continuing to enter the low-power mode of operation [5]. When multiple events occurred simultaneously, which determine the priority firstly and then deal with the incident. The main work of the Z-Stack protocol stack includes system initialization, OSAL startup and task polling. OSAL operating system is mainly responsible for system initialization tasks and event task processing.

The system is initialized first after the system powered on. The system hardware and software are initialized through the execution of an initialization function in ZMain.c under the ZMain folder. The ZMain folder includes off total interrupt function osal_int_disable (INTS_ALL), initialize onboard hardware functions HAL_BOARD_INIT(), initialize I/O port functions InitBoard(OB_COLD), initialize HAL layer driver function HalDriverInit(), initialize non-volatile memory functions sal_nv_init(NULL), initialize MAC layer functions ZMacInit(), assign address function zmain_ext_addr(), initialize operating system functions osal_init_system().

The operating system through the osal_init_system() function into an infinite loop, constantly polling system tasks. If there is an incident, then deal with the incident. If multiple events happened simultaneously, the system will process the events according to the priority of the events, and continue polling if there is no event. The osal_run_system (void) operating system function firstly calls osalTimeUpdate() and Hal_ProcessPoll() to update the operating system and polling the serial port and the timer if there is an event to deal with. Then use the do-while loop to poll the system event task storage functions tasksEvents[idx], each task occupying the corresponding data unit independently. [idx] from low to high indicates that the event priority from high to low reading the highest priority event. When out of the loop, check the event is legal. To call the tasksArr [] function to make the appropriate treatment of the incident if it is legal. When into the critical area, tasksArr [idx] stored in the task is protected until the completion of event processing, function tasksArr [] for storing event handlers. After the event completed, the user exits the critical area, emptying the task event storage area, and writing unprocessed events to the highest priority for processing. Based on this polling mechanism, you can complete the processing of all events in the system operation.

4.1. Driver Control Program Design of Airfield Lighting

The navigation light terminal node parses the received control command data through the command parsing function Sample_command_analysis. Firstly, the sender's PAN ID is checked to determine the data whether is a data packet of the system network. If not, the data will be discarded and not processed; If it is the data of the system network, then it is detected whether the destination address of the data packet is the 16-bit network address of the local node; And if not, the packet will be discarded; If yes, it is determined whether the data is a control command; If it is a control command, trigger the corresponding event handler to complete the corresponding control tasks. Navigation lights control tasks through the Sample_led_control function to complete the appropriate control functions,
including navigation lights on or off, PWM-driven dimming control and plane taxiway lights control. The implementation process command analysis and control shown in Fig. 2.

Fig 2. Command parsing and control execution flow

If the result of the command parsing is the LED navigation light brightness adjustment task, then the corresponding processing event will be triggered. According to the control requirements of the corresponding output duty cycle PWM signal to adjust the brightness of the navigation lights and meet the system control requirements. If it is the LED navigation light sliding guide control task, set the sliding guide control indicator LED_GUIDE_F is 1, start the navigation light slide guide control operation, LED navigation lights to achieve sliding guide control. If it is the LED navigation lights switch control task, then set the switch control logo LED_ON_F. If you open the LED airfield lighting, light LED_ON_F will set to 1, then the microprocessor I/O port output low trigger relay pull to achieve the function of turning on the LED airfield light, otherwise turn off the navigation light.

4.2. Fault Diagnosis Program Design of Airfield Lighting

Terminal control node program design is based on the Z-Stack protocol stack developed. First of all, the task is initialized. The task function of the terminal control node is initialized in osalInitTasks () function and the task function is added to the osalInitTasks () function. The task functions include data acquisition, upload function command parsing and sending functions, then add the event handler in the corresponding task. A task in the Z-Stack stack can include multiple events through the event handler to deal with different events accordingly. Events in the Z-Stack stack include system events (SYS_EVENT_MSG) and user-defined events. According to the system event (SYS_EVENT_MSG) determines which customized event occurs and calls the corresponding event handler to process the event. Any customized event is added in the tasksArr [] function, and an event handler can be called in osal_start_system () to respond to the event if an event occurs. When adding any event function, the corresponding event handler add to the SampleApp_ProcessEvent function. The sequence of events in tasksEvents [] and tasksArr [] must be corresponding. The Nth event handler in tasksArr [] corresponds to tasksEvents [] in the Nth event of the task.
Data acquisition is used to initialize the acquisition event through the data acquisition function Sample_collect_data. The data acquisition function Sample_collect_data is a periodic task. Data collection to 3S for the data acquisition cycle, collection illumination, temperature and fault information through the data send function Sample_send_data complete the data package upload task. During the acquisition cycle, illumination, temperature and status information are collected about navigation lights. After the acquisition cycle, the collected information will be A/D converted. Then determine whether the number of acquisitions has reached 5 times. If the specified number of acquisitions is reached, the data transmission event will be triggered. In order to avoid multiple terminal nodes sending the collected information to the coordinator cause communication congestion, the terminal node packages sends the data to the cluster head node in the area firstly. After the cluster head node integrates the information of the cluster members, coordinator node uploaded to the monitoring center or continue to data acquisition operations. During the data sending process, the data sending status is monitored. If the data sending fails, the number of failed sending of the statistics data is counted. If the number of data failed to send less than 3 times, continuing to send the data package or return to data collection.

5. Conclusion
With the rapid development of LED light technology, LED light will be more and more applied to navigation light system, which will greatly reduce system power consumption, combined with wireless communication technology and achieve "perception" for thousands of navigation lights. The navigation light terminal control node uses CC2530 as a microprocessor and adds a RF amplification chip CC2591. Based on this, STM32 is used as the main controller of the coordinator node to improve the data processing speed and network control performance of the coordinator. The system has the advantages of low power consumption, flexible networking and high efficiency, which can meet the control and monitoring requirements of the airport's navigation lighting [6]. In addition, with the continuous efforts more and more enterprises and R&D institutions, it will promote the realization of low-power green airport and "smart airport".

References
[1] Wang Bingyuan, Gao Jie. Airfield lighting wireless monitoring system node design, J.Measurement and Control Technology. 2014 (8): 91-94.
[2] Tian Hongxian,Guo Wei. Research on intelligent street light routing protocol based on ad hoc network, J. Measurement and Control Technology. 2015 (3): 93-96.
[3] Information on http://www. ti. com/lprf
[4] Liu Shan, Gong Jie and Bao Jianyu. LED driver design of constant illumination dimming, J. Microcontroller and Embedded System Applications. 2014 (10): 61-63.
[5] Drew Gislason. Zigbee wireless networking[M], Oxford: Newnes, 2008: 47-50.
[6] Xu Pujun. Based on wireless sensor network airfield lighting monitoring system, D. Hefei: Hefei University of Technology, 2010: 1-3.