Proximity Sensor-based Remote Control
Remote Control Demonstrator Board

by: **Peter Moravcik**
Rožnov Czech System Center
Czech Republic

The Proximity Sensor-based Remote Control is an example of advance universal remote control, implementing wireless 2.4 GHz connectivity, proximity sensors, and an accelerometer all in a single application. This application note describes the system solution, hardware, and software designs.

The remote control presents several Freescale devices and technologies: ZigBee® (SMAC) capability, accelerometer, and proximity capacitive sensing. The board is placed in a small, handy plastic enclosure and is powered by two AAA batteries.

Integrated into one package, the basic component of this board is the 8-bit MC13213 MCU with a HCS08 core and RF 802.15.4 modem, all integrated into one package.

The remote controller includes a low-power 3-axis accelerometer. The design is universal for the assembly of the analog MMA7340L or the digital MMA7456L version of the sensor. Sensors are modifiable to detect 2g/4g/8g sensitivities.

A contact-less user control interface is realized by a 10-element electrode keyboard handled by Touch Sensing Software and implemented in the MC9S08QG8. An example of implementing an E-Field Lighting Controller with Wireless Connectivity is shown.

Table of Contents

freescale™
semiconductor

1 Application Description

A block diagram of the Proximity Sensor-based Remote Control application is shown in Figure 1. The full Proximity Sensor-based Remote Control consists of the Main Control Board and a small touch pad board, which are both connected together by a flat 5 double wire cable. The goal of the Proximity Sensor-based Remote Control design is to provide a small, handy portable device with the capability of demonstrating Freescale's ZigBee$^{®}$ (SMAC) RF communication devices, Capacitive Touch Sensing Software, and 3-axis accelerometer functions. The whole device and the shape of the PCBs are designed for integration into a small plastic enclosure from the Ruwido company. The Proximity Sensor-based Remote Control incorporates an internal 10-element touch pad keyboard on the top side of the PCB. The device uses the front face from this plastic enclosure as the covering material for these electrodes.There are three main components on the back side of the PCB, connected together via the I^{2}C serial interface on the board. The application is powered from two 1.5 V AAA batteries and uses power down modes when the device is not used by the user.

First, the **MC13213** is an 8-bit MCU device with a HCS08 core and a RF 802.15.4 modem integrated together on the chip. This device reads either the touches recognized by the **MC9S08QG8 MCU** or the 3-axis angle information generated by the accelerometer via the I^{2}C bus, and it performs the relevant function. The device will indicate internal states by turning on a relevant LED on the board, manage power down modes, or it may transmit data by a RF communication modem with a controlled device.

Second, the 3-axis accelerometer is implemented on the board. There is the possibility of assembling either the digital **MAA7456L** or the analog **MAA7340L** version of this sensor. The digital version of the sensor communicates with the MCU via the I^{2}C bus. The analog version just provides an analog level for each axis, which is converted by the MCU ADC module. Sensors are modifiable to detect 2g/4g/8g sensitivities.The PCB is universal for both versions of the sensor.

Third, there is a Capacitive Touch Sensing Software function in the **MC9S08QG8 MCU**, which is based on the original Proximity Software Library provided by Freescale. This Touch Sensing software has been especially adapted for this kind of application and for plastic materials. The MC9S08QG08 is connected by the I^{2}C bus for sending information to the main MC13213.
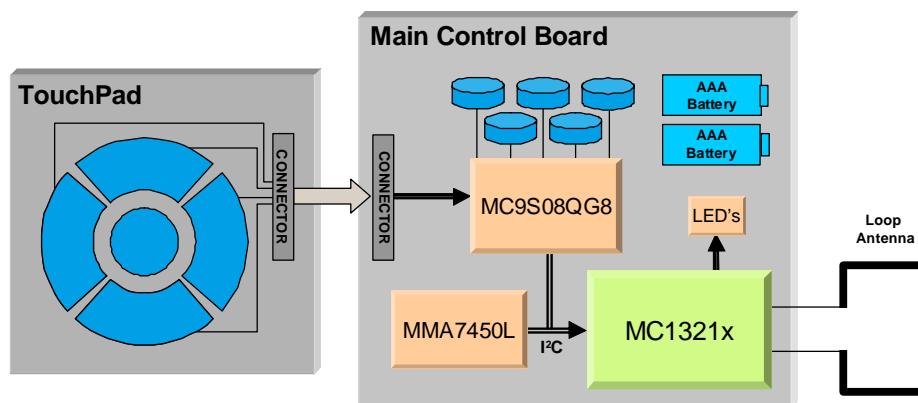

Figure 1. Block Diagram of Proximity Sensor-based Remote Control Application

AN3783, Rev 0

1.1   Features

The main features of the Proximity Sensor-based Remote Control include:

- Proximity sensor based wireless remote controller
- The application includes:
    — MC13213 (HCS08 MCU with 802.15.4 RF modem)
    — MMA7340L/MMA7456L accelerometer
    — MC9S08QG8 for proximity sensing
- The board is suitable for implementation in Ruwido's small Z-series plastic enclosure
- The board is optimized to manage a 10-element touch pad
- Two versions of contact-less electrodes may be accessible:
    – 5 on-board E-field electrodes with a defined purpose
    – a touch pad with 4 E-Field electrodes in a rotary configuration and one central pad
- 3-Axis accelerometer information, used for custom regulation, through tilting
- Internal status indication using six built-in LED's
- Power to the Proximity Sensor-based Remote Control Board from the two AAA batteries
- RF communication using an 802.15.4 modem by SMAC protocol level software
- Software offers the following functions for remote control of the E-Field Lighting Controller with Wireless Connectivity demonstrator and for all individual parts of this demo:
    – On/Off lights by the central pad on the touch pad board
    – Fluent regulation of light intensity either by clockwise/counter-clockwise finger movement on the rotary electrodes or by tilting the accelerometer
    – Selection of the controlled device by the five pads situated on the Main Control Board (selected devices are indicated by the five LED's situated in the center of the board)
    – On/Off Accelerometer function
- BDM connector for device programming and creating the user's own custom algorithm
- Power down modes when the device is not in use (for lower power consumption)
- Most components are on the backside of the PCB. The keypad side is relatively flat for mounting and covering by flat material

Figure 2. System Overview

AN3783, Rev 0

2 Featured Products

In this chapter, the basic features of the main components of the Proximity Sensor-based Remote Control are described. For its proper operation, there are also other parts on the back side of the board. These components include: capacitors, resistors, inductors, connectors, LEDs and a crystal.

2.1 Three-axis Digital Acceleration Sensor MMA745xL

The MMA745xL is a three-axis Low g Digital Output Acceleration Sensor with digital output (I^2C/SPI), a fast response time, low current consumption, low voltage operation, a standby mode in a small profile package, and selectable sensitivity. The flexible g-select feature allows the use of this accelerometer in a design with ranges of 2g, 4g and 8g. By providing both I^2C and SPI, the MMA745xL has a direct interface to the main system processor, allowing communication flexibility.
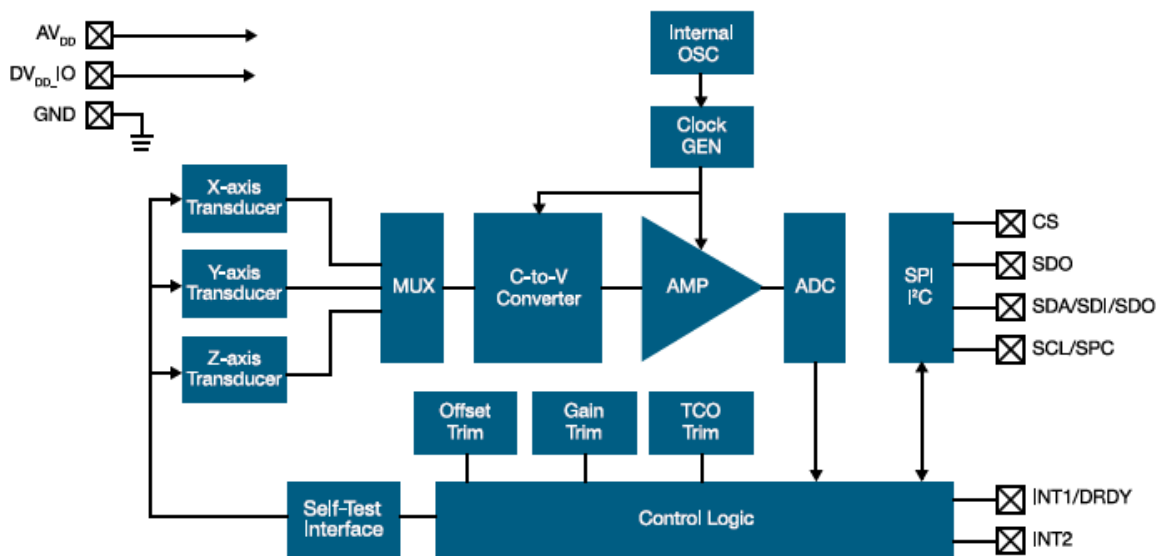
.

Figure 3. Functional Block Diagram of MMA745xL

Features:

- Digital Output (I^2C/SPI) - 10-Bit at 8g Mode
- 3 mm x 5 mm x 1.0 mm LGA-14 Package
- Low Current Consumption: 400 µA
- Self Test for X and Z-Axis
- Low Voltage Operation: 2.4 V - 3.6 V
- Customer Assigned Registers for Offset Calibration
- Programmable Threshold Interrupt Output

- Level/Pulse Detection for Motion Recognition (Shock, Vibration, Freefall)
- Click Detection for Single or Double Click Recognition
- High Sensitivity (64 LSB/g @ 2g and @ 8g in 10-Bit Mode)
- Selectable Sensitivity (±2g, ±4g, ±8g)
- Robust Design, High Shocks Survivability (10,000g)
- RoHS Compliant (Environmentally Preferred Product)

Typical Applications:

- Cell phones: motion dialing, text scrolling, e-compass, portrait/landscape, image stability
- Laptop PC: free-fall detection, anti-theft, event recorder
- Portable media players or HDD: free-fall detection
- GPS navigation: dead reckoning, e-compass tilt compensation
- Portable media players: free-fall detection
- Gaming: tilt and motion sensing, event recorder
- Digital camera and digital video camera: image stability, portrait/landscape
- Robotics: motion sensing for industrial applications

2.2   Microcontroller MC13213

The MC1321x family is Freescale's second-generation ZigBee® platform incorporating a low power 2.4 GHz radio frequency transceiver and an 8-bit microcontroller HCS08 into a 9x9x1 mm 71-pin LGA package.

The MC1321x solution can be used for wireless applications from simple proprietary point-to-point connectivity to a complete ZigBee® mesh network. The MC1321x contains an RF transceiver, which is an IEEE 802.15.4-compliant radio operating in the 2.4 GHz ISM frequency band. The MC1321x also contains a microcontroller based on the HCS08 Family of Microcontroller Units (MCU), and it can provide up to 60 KB of flash memory and 4KB of RAM.
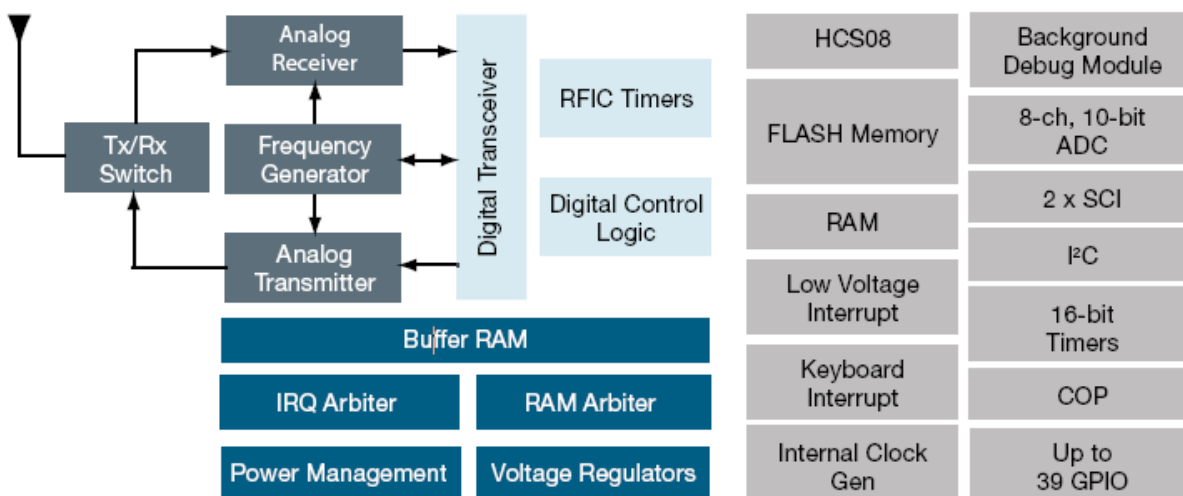

Figure 4. MC13213 Block Diagram

AN3783, Rev 0

Microcontroller Features

- 40 MHz HCS08 low-voltage, low-power core
- Up to 60K flash memory with block protection and security, and 4K RAM
  - MC13211: 16KB Flash, 1KB RAM
  - MC13212: 32KB Flash, 2KB RAM
  - MC13213: 60KB Flash, 4KB RAM
  - MC13214: 60KB Flash, 4KB RAM with ZigBee® Z-Stack™
- Seven addressing modes for the CPU
- Multiple 16-bit timers
- Operational in the 2.4 GHz worldwide unlicensed frequency band
- 2 V to 3.4 V operating voltage with on-chip Voltage regulator
- -40°C to +85°C temperature range
- Low external component count
- Requires a single 16 MHz crystal
- Auto-trim feature for crystal accuracy
- Supports SMAC, IEEE 802.15.4 MAC and ZigBee® software
- 9 x 9 x 1 mm 64-pin LGA package
- 8-bit port keyboard interrupt (KBI)
- 8-channel, 10-bit analog-to-digital converter (ADC)
- Two independent serial communication interfaces (SCI) supporting up to 115.2 kBaud
- Inter-integrated circuit (I^2C) with 100 kbps maximum bus loading
- Internal clock generator (ICG) at 100 kHz or 16 MHz (includes internal ref. generator)
- In-circuit Flash programming available via on-chip background debug module (BDM)
- Programmable low-voltage interrupt (LVI)
- Common on-chip processor (COP) watchdog timer

RF Modem Features

- Fully compliant IEEE 802.15.4 transceiver supports 250 kbps O-QPSK data in 5.0 MHz channels and full spread-spectrum encode and decode
- Operates on one of 16 selectable channels in the 2.4 GHz band
- 0 dBm nominal output power, programmable from -27 dBm to +4 dBm
- Receive sensitivity of < -92 dBm (typical) at one percent PER
- Integrated transmit/receive switch
- Has two PAs that can be programmed for single-ended or full differential operation
- Supports external low-noise amplifier (LNA) and/or PA
- Three low-power modes for increased battery life
- Programmable frequency clock output for use by MCU
- On-board trim capability for the 16 MHz crystal reference oscillator
- Four internal timer comparators available to supplement the MCU timer resources
- Supports streaming and post data processing modes
- Seven general purpose input outputs (GPIOs) to supplement the MCU GPIO

2.3   Microcontroller MC9S08QG8

The MC9S08QG8/4 extends the advantages of Freescale's HCS08 core to low pin count, small-package 8-bit microcontrollers. QG devices are low voltage with on-chip in-circuit flash memory programmable down to 1.8 V, and they afford the standard features of all HCS08 MCUs, including wait mode and multiple stop modes. The functionality is completed with strong analog capabilities, a complete set of serial modules, a temperature sensor, and robust memory options.

Figure 5. MC9S08QG8 Block Diagram

Microcontroller Features

- 20-MHz HCS08 CPU (central processor unit)
- 8 Kbytes flash, 512 bytes RAM
- 12 general-purpose input/output (I/O) pins,
- **ADC:** 8-channel, 10-bit analog-to-digital converter with automatic compare function
- **ACMP:** Analogue comparator module with option to compare to internal reference
- **SCI:** Serial communications interface module with the option for 13-bit break capabilities
- **SPI:** Serial peripheral interface module
- **I^{2}C:** Inter-integrated circuit bus module
- **TPM**: 2-channel timer/pulse-width modulator
- **MTIM**: 8-bit modulo timer module with an 8-bit pre-scaler
- **KBI**: 8-pin keyboard interrupt module with software selectable polarity

3   Hardware Description

A block diagram of the solution is shown in Figure 1. All of the components necessary for a proper operation are on a small board size, 50 x 40 mm, with a specific shape. The board is also connected via a 5 double wire cable with the touch pad board providing 5 electrode pads. This form of construction was chosen for a proper implementation in Ruwido's plastic enclosure. Please note, that the 2.0 V - 3.0 V voltage necessary for board operations is generated by two AAA batteries.

Table 1 provides a description of the components of the Proximity Sensor-based Remote Control Figure 6 and Figure 7 show the locations of components on the board. A schematic is shown in Figure 8.
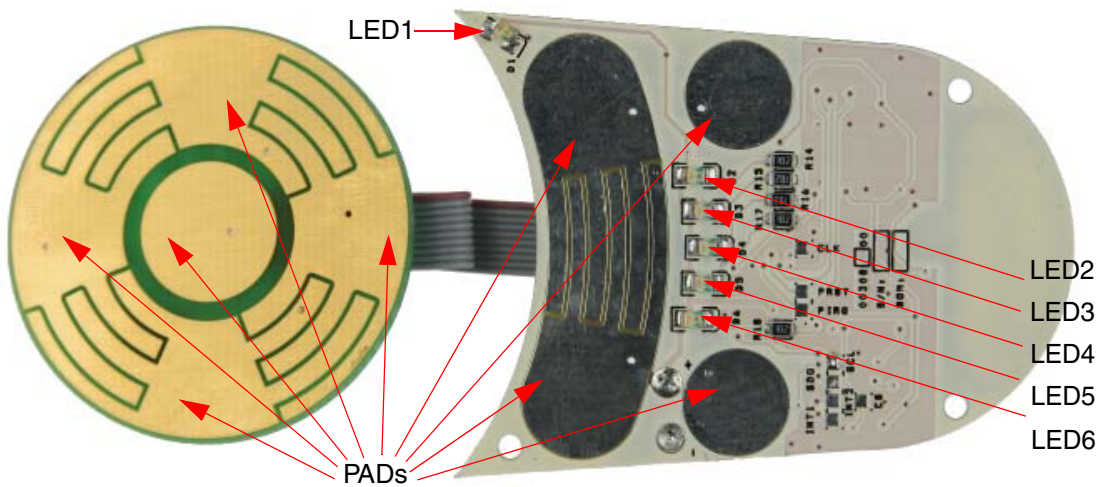

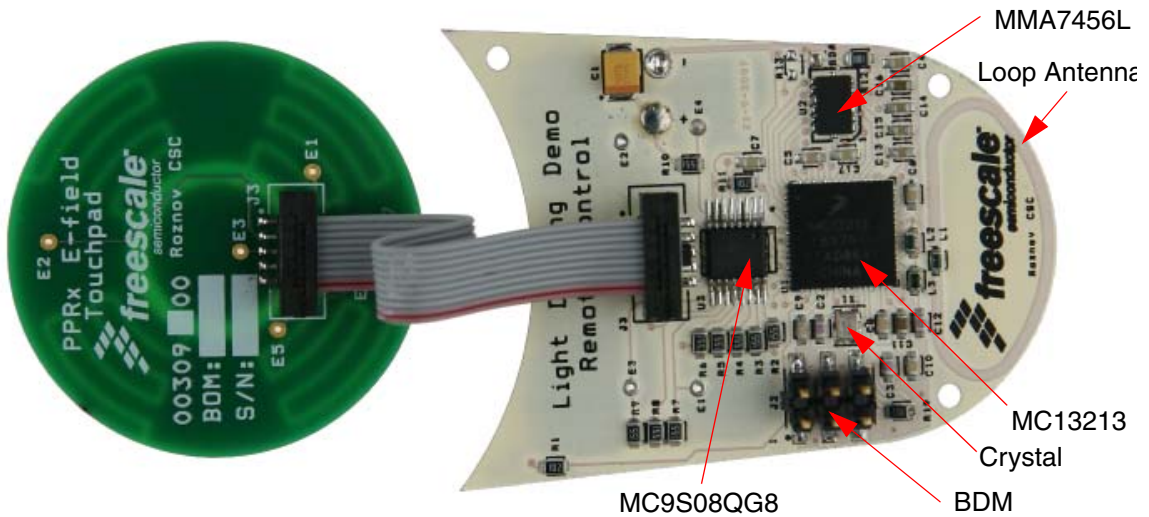**Figure 6. Proximity Sensor-based Remote Control Boards View (TOP Side)**

Figure 7. Proximity Sensor-based Remote Control Boards View (BOTTOM Side)

AN3783, Rev 0

Table 1. Components on the Main Control Board

Reference(s)	Value	Package	Manufacturer	Order code	Qt	Description
C1	220 μF/4V	case A	any acceptable		1	Tantalum polarized filter capacitors
C2, C3, C12	6.8 pF	0603	any acceptable		3	Filter Capacitors for crystal connection and Loop Antenna
C4, C5, C6, C7 C8, C9, C10, C17	100 nF	0805	any acceptable		8	Ceramic filter capacitor
C13, C14, C15	10 nF	0805	any acceptable		3	Ceramic filter capacitor
C11,C16	1 μF	0805	any acceptable		2	Ceramic filter capacitor
D1, D2, D3, D4, D5, D6	LXT0805GW	SMD	Citizen	CL-170G-CD-T	6	Green LEDs for optical indication of device internal status
J3	FTS-105-01-F-DV	10x1, 25 mm	Samtec	FTS-105-01-F-DV	1	Connector for touch pad to Control Board
J2	HDR 2X3	2x3x2, 54 mm	Harwin	M20-9980345	1	Connector for Background Debug Mode on MC13213 MCU
L1	4.7 nH	0603	TDK	MLG1608B4N75	1	Ferrite multi-layer chip inductors
L2,L3	22 nH	0603	TDK	MLG1608B22NJ	2	Ferrite multi-layer chip inductors
R1, R11, R12, R14, R15, R16, R17, R18, R19	1 kΩ	0805	any acceptable		9	Resistors for various purposes. Attenuators for power LED
R2, R3, R4, R5, R6, R7, R8. R9, R10	1.5 MΩ	0805	any acceptable		9	General pull-up resistors for Touch Sensing Electrodes
U1	MC13213	LGA	Freescale	MC13213	1	HCS08 Main control MCU with 802.15.4 RF modem
U2	MMA7456L	LGA CASE 1935-01	Freescale	MMA7456L	1	3-axis digital accelerometer
U3	MC9S08QG8	SOIC8	Freescale	MC9S08QG8CDNE	1	HCS08 MCU with Touch Sensing Software inside
X1	NX2520SA/16 MHz	2520 SMD	NDK	NX2520SA-16.0MH	1	Crystal resonator for MCU

PROXIMITY SENSOR

J3
CON10

R10 R9 R8 R7 R6 R5 R4 R3 R2
1M5 1M5 1M5 1M5 1M5 1M5 1M5 1M5 1M5

VDD

PRST — PPR_RST
PIRQ — PPR_IRQ

VDD

R11 R12 R19
1K 1K 1K

U3
E1 1 RN/E1   RST 1   PPR_RST
E2 15 RNE/E2   IRQ 2   PPR_IRQ
E3 14 RE/E3   Vdd 3
E4 13 RSE/E4   Vss 4
E5 12 RS/E5   SCL 5   SCL/SPC
E6 11 RSW/E6   SDA 6   SDA
E7 10 RW/E7   AD0 7
E8 9 RNW/E8   SOUNDER 8

MC9S08QG8

E1 1
E2 1 E7
E3 1 E8
E4 1 E9

GND

TRIAXIAL G SENSOR

U2
MERCURY/ION

g-select 10 g-select, NC   Xout,GND 2   X-axis
INT1 11   NC, NC   Yout,NC 3   Y-axis
8   NC, INT1/DRDY   Zout,Addr0 4 Z-axis
SDA 13   0g - detect,INT2 9   0g_detect
R13 0
SDA 13   Self_test,SDA/SDI/SDO
SDO 12   NC, SDO   VSS,GND 6   VDD_mcu
SCL/SPC 14   NC,SCL/SPC   VDD,AVdd 5   DVdd_mcu
SLEEP 7   Sleep, CS   NC, DVdd_IO 1

SCL — SCL/SPC
SDA — SDA
SDO — SDO

X-axis Y-axis Z-axis
C13 C14 C15
10nF 10nF 10nF
GND GND GND

INT2 — 0g_detect
INT1 — INT1
CS — SLEEP

BDM

GND VDD

RESET

J2
BDM

PPR_IRQ
BDM

MCU + RF

U1
LED6 62 PTA0/KBD0   PTE0/TxD1 20 LED1
PPR_IRQ 63 PTA1/KBD1 PTE1/RxD1 21
SDO 64 PTA2/KBD2   PTE2/CEBi 68
g-select 1 PTA3/KBD3 PTE3/MISOo 67
0g_detect 2 PTA4/KBD4 PTE4/MOSli 66
INT1 3 PTA5/KBD5   PTE5/SPICLK 65
4 PTA6/KBD6
5 PTA7/KBD7   PTB0/AD0 52 Z-axis
                     PTB1/AD1 53 Y-axis
PPR_RST 12 PTC0/TxD2 PTB2/AD2 54 X-axis
SDA 13 PTC1/RxD2   PTB3/AD3 55
SCL/SPC 14 PTC2/SDA PTB4/AD4 56 DVdd_mcu
15 PTC3/SCL   PTB5/AD5 57 SLEEP
LED2 16 PTC4   PTB6/AD6 58
LED3 17 PTC5   PTB7/AD7 59
LED4 18 PTC6
LED5 19 PTC7   GPIO1 44
                     GPIO2 43
70 PTD1/RXTXENi   GPIO3 42
47 PTD2   GPIO4 41
71 PTD3/RSTBi   GPIO5 24
48 PTD4   GPIO6 25
49 PTD5   GPIO7 26
GND 50 PTD6
VDD_mcu 51 PTD7   ATNBi 46
                     IRQ/IRQo 69 GND
BDM 7 PTG0/BKGD/MS
8 PTG1/XTAL   PAO_M 39
9 PTG2/EXTAL   PAO_P 38
MCU_CLK 1 10 CLKOo   TINJ_M 37
RESET 11 RESET   RFIN_P 36 Antenna
                     RFIN_M 35
60 VREFH   CT_Bias 34   L1 4.7nH
61 VREFL GND   SM 40
45 VDD   VBATT 32 L3 22nH L2 22nH
6 VDDAD   VDDINT 23
XTAL2 28 XTAL2   VDDA 33
XTAL1 27 XTAL1   VDDLO1 30
                     VDDLO2 29   C12 6.8PF
C5 C6 C7   VDDD 22
100nF 100nF 100nF   VDDVCO 31
72 Exposed Pad
MC13211

C8 C9 C10 C11
100nF 100nF 100nF 1uF
GND GND GND GND GND

POWER SOURCE

VDD

BATT1
2x Battery AAA

C1 220uF/4V

GND GND

LED

VDD VDD VDD VDD VDD VDD
D1 D2 D3 D4 D5 D6
LED LED LED LED LED LED
R1 R14 R15 R16 R17 R18
1K 1K 1K 1K 1K 1K
LED1 LED2 LED3 LED4 LED5 LED6

SENSOR SOURCE CAPACITORS

VDD_mcu   DVdd_mcu
C4 C16 C17
100nF 1uF 100nF
GND GND GND

CLOCK SOURCE

XTAL1   C2 6.8PF
XTAL2   C3 6.8PF
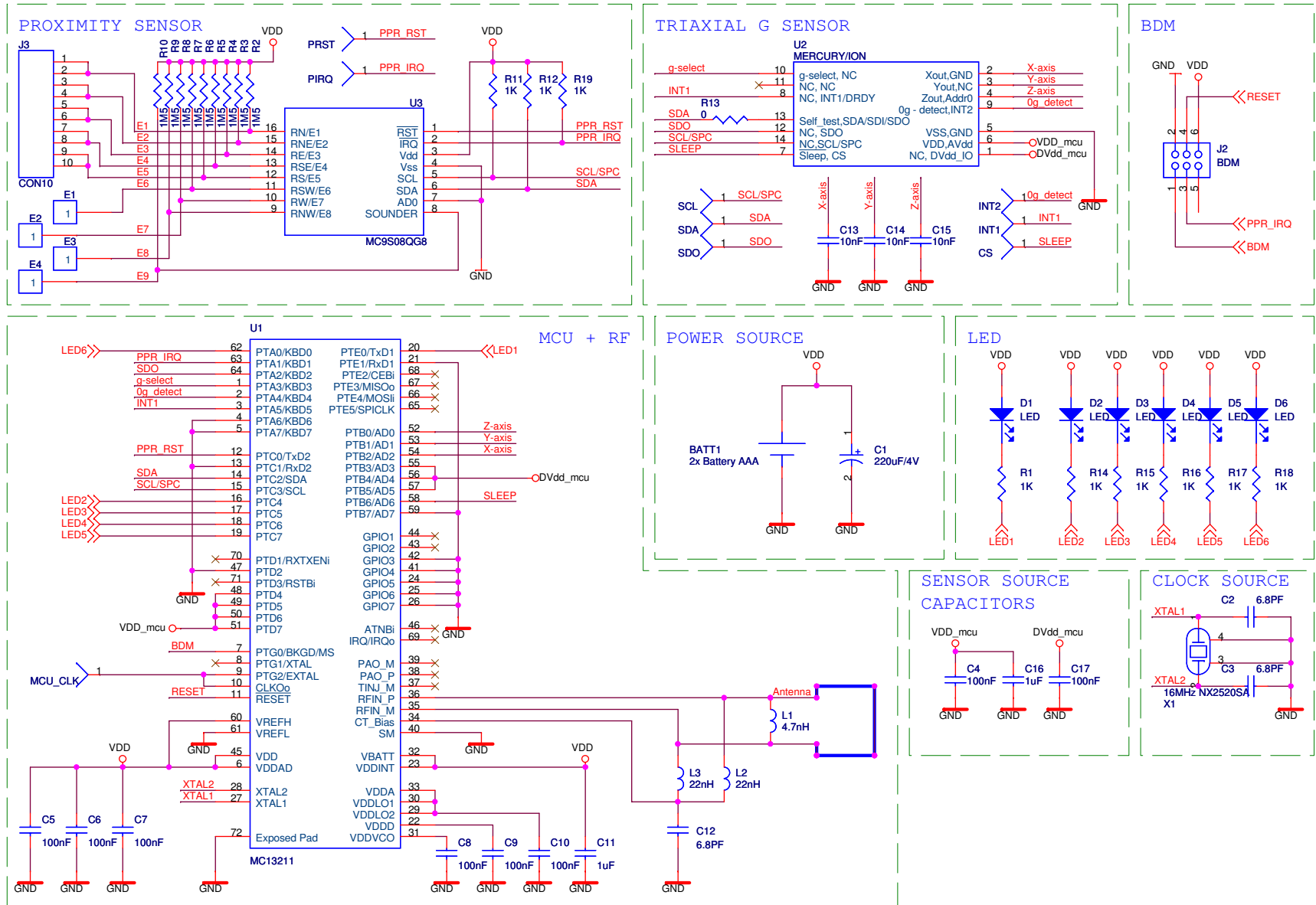16MHz NX2520SA
X1
GND

Figure 8. Main Control Board Schematic
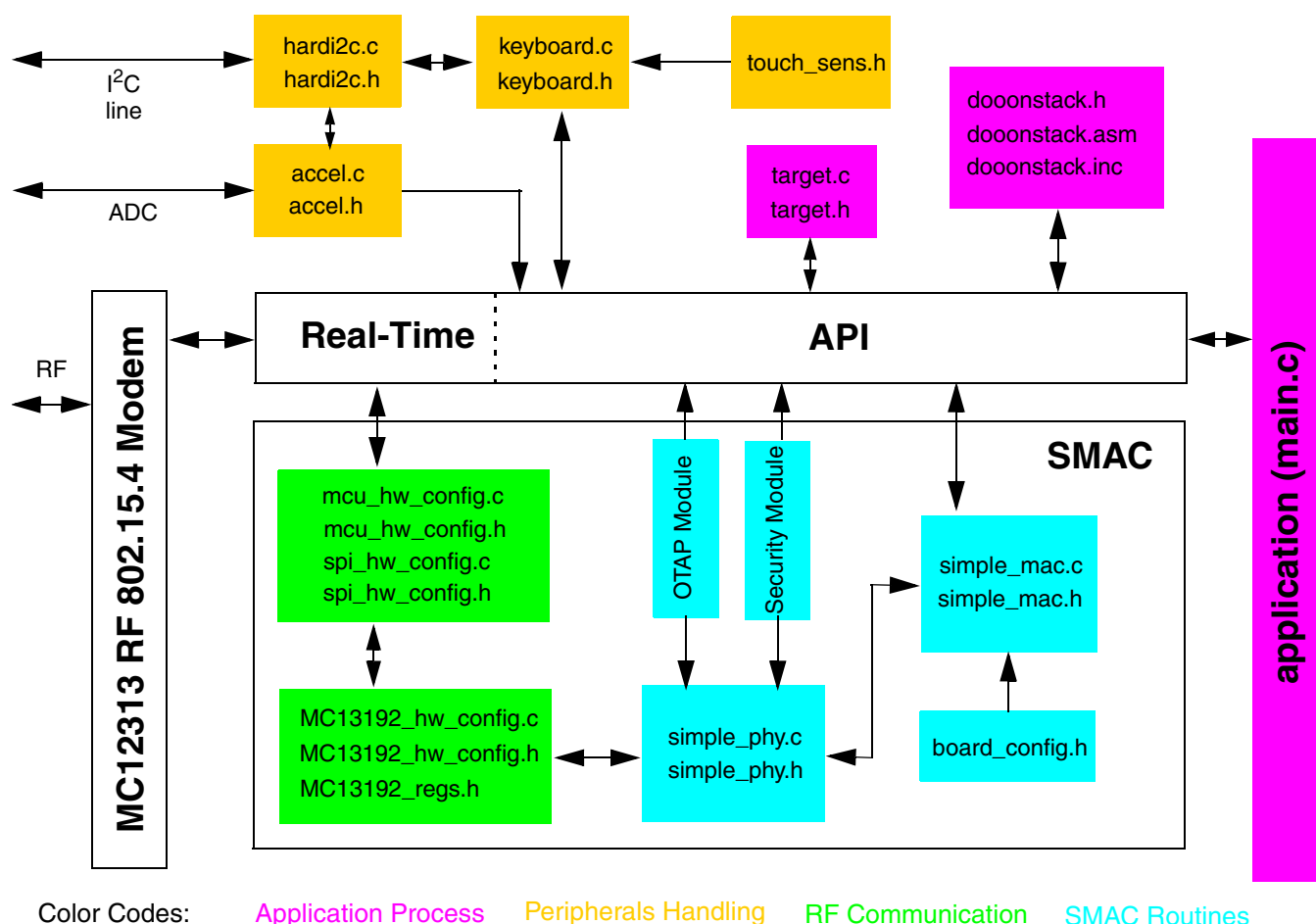
4 Software Description

Figure 9. MC13213 Firmware Overview

4.1 MC13213 Firmware

In Figure 9 there is a complete block diagram of the Proximity Sensor-based Remote Control (MC13213) firmware. This source code is written in the C programming language in the Dev Tech CodeWarrior® Development Studio. The MC13213 firmware code contains the following files:

- **Application process:**
  — *main.c -* main program loop.
  — *target.c (target.h) -* hardware (board) specific routines. Mainly related to initialization.
  — *doonstack.c (doonstack.asm) -* routines for real-time flashing to program memory of microcontroller. This process is usually used in MCU's without an EEPROM for memorizing variables after a device shutdown.

- **Peripherals Handling:**
  - keyboard.c (keyboard.h) - keyboard application command implementation.
  - touch_sens.h - The header file contains definitions for Touch Sensing MCU use.
  - hardi2c.c (hardi2c.h) - I2C routines between the MCU and the MC9S08QG8, or between the MCU and the MMA7456L.
  - accel.c(accel.h) - accelerometer application command implementation.
- **RF communication:**
  - *mcu_hw_config.c (mcu_hw_config.h)* - Header that includes the declaration of the methods used to configure the hardware and the functions to manage the clock, COP, GPIO, SPI and IRQ.
  - *mcu_spi_config.c (mcu_spi_config.h)* - This is the SMAC C source driver file for the driver control interface to the SPI.
  - *MC13192_hw_config.c (MC13192_hw_config.h)* - The main routine is the one that initializes the MC13192. There are also the Disable/Restore MC13192 interrupts commands included.
  - *MC13192_regs.h* - Definition of hardware registers.
- **SMAC routines:**
  - *simple_mac.c (simple_mac.h)* - This is the SMAC C source media (i.e. MAC) layer file for the HC(S)08 MCU and the MC13192 transceiver. The SMAC MAC is the highest layer of C code for the SMAC.
  - *simple_phy.c (simple_phy.h)* - This is the SMAC C source physical layer file for the HC(S)08 MCU and the MC13192 transceiver. The SMAC phy is the second lowest layer of C code.
  - *board_config.h* - contains the definition of the hardware board used for the BeeKit™ tool.

The MC13213 microcontroller offers many features that simplify the Proximity Sensor-based Remote Control design. Table 2 describes the individual available blocks and their usage in the introduced system.

Table 2. MC13213 Module Usage

Module available on MC13213	Used	Purpose
802.15.4 RF modem	yes	Communication between the Remote Control and controlled device
ADC	yes	Reading the 3-axis information from the MMA7340L analog g-sensor
SCI	no	
SPI	yes	Internal comm. used by the SMAC between HCS08 and RF modem
I^{2}C	yes	Communication between the MCU and MC9S08QG8 or MMA745xL
TPM1	yes	Common Delays and time measurement
TPM2	no	
KBI	yes	Interrupt for MCU wake-up when touch is detected by the TSS
IRQ	no	

AN3783, Rev 0

Table 3 shows how much memory is needed to run the Proximity Sensor-based Remote Control Board using the MC13213 microcontroller. A significant part of the microcontroller memory is still available for other tasks.

Table 3. Memory Usage

Memory	Available (MC13213)	Used
FLASH	60912 Bytes	15302 Bytes
RAM	4096 Bytes	1140 Bytes

4.1.1 I^{2}C Communication Driver

The file **hardi2c.c** (**hardi2c.h**) contains I^{2}C routines between the MCU MC13213 and the MC9S08QG8. There are several routines for direct access to the microcontroller I^{2}C registers, but the most important of these routines are *HW_IIC_WriteValue* and *HW_IIC_ReadArray*.

- *HW_IIC_WriteValue* - this function has three input parameters: the I^{2}C address of the destination device, the address of the device register to write to, and the value to write to this register.

- *HW_IIC_ReadArray* - this function has four input parameters: the I^{2}C address of the destination device, the address of the required destination register to read, the address of the pointer to store data, and the number of the byte to read from this required data register (auto-increment loop).

The main API functions for the I^{2}C driver **hardi2c.c** are in the files **keyboard.h** and **accel.h**. Their implementations are in the files **keyboard.c** and **accel.c**. These files include the following functions (API keyboard interface and API accelerometer interface):

- *GetTPadFirst* - This function reads directly the number of the first touched pad from the MC9S08QG8 via I^{2}C. This function must be called continuously in the main loop; otherwise, any touched pads may be lost. This function does not have an input parameter, only an output parameter. Function *GetTPadFirst* returns one byte of information about the touched pad. If there is no touched pad, this function returns 0.

- *GetTPadSecond* - This function reads directly the number of the second touched pad from the MC9S08QG8 via I^{2}C. This function is identical to *GetTPadFirst* function, but it returns the value of the second touched pad at the same moment as the first touched pad.

- *TSS_init* - This function initializes the Touch Sensing Software in MC9S08QG8 via I^{2}C. These setting parameters are represented as 8-bit values, which are written to register variables in the MC9S08QG8. The following registers are defined by this functions: Configuration Register, Fault Register, Sensitivity Registers, Sleep Period Registers. This function doesn't have an output parameter. It is called from the main program during the initialization sequence.

- *Init_SensorIon* - This function initializes the MMA7456L tri-axial digital accelerometer via I^{2}C if the digital sensor is assembled. This function doesn't have an output parameter. It is called from the main program during the initialization sequence.

- *Read_AccelerometerIon* - This function reads directly the status of internal register values from the MMA7456L accelerometer via I^{2}C. This function must be called continuously in the main loop; otherwise, any accelerometer data may be lost. This Function returns the actual bytes read in the *p_buffer* array for each of the 3 axes separately.

- *Init_SensorMercury* - This function initializes the MMA7340L tri-axial analog accelerometer by correctly setting the external pins of this sensor. The function also prepares the ADC for functionality. It is called from the main program during the initialization sequence.

- *Read_AccelerometerMercury* - This function reads directly the output information from the analog outputs of the MMA7340L by the MCU internal ADC convertor. The status of the internal register values from the MMA7340L accelerometer is via I^{2}C. This function must be called continuously in the main loop; otherwise, any accelerometer data may be lost. This function returns values read in the *p_buffer* array for each of the 3 axes separately.

4.1.2   RF Communication Driver

The application uses a simple protocol built on top of the SMAC driver for an RF transfer of information between the Proximity Sensor-based Remote Control and the controlled devices. The protocol used is regularly unidirectional, allowing the setup of communication between one Proximity Sensor-based Remote Control with four Intelligent Outlets and one Intelligent Switch. However, it is easily possible to realize various combinations of control topology. A device is defined by a unique network identifier for determining the destination. Also, it doesn't matter if one packet is not received, because the dimming value is memorized in the controlled device (not by the sender), and the transmitted message changes this value. All data is transferred in the packet structure, as shown in Figure 10.

MC1321x 802.15.4 MODEM Packet Structure
SFD (Start of Frame Delimiter)
FLI (Frame Length Indicator)
FCS (Frame Check Sequence)

Preamble	SFD	FLI	Payload Data	FCS

SMAC Packet Structure

Packet Control Field	SMAC Payload

Packet Structure

Network Number	Destination Address	Source Address	Type	Value
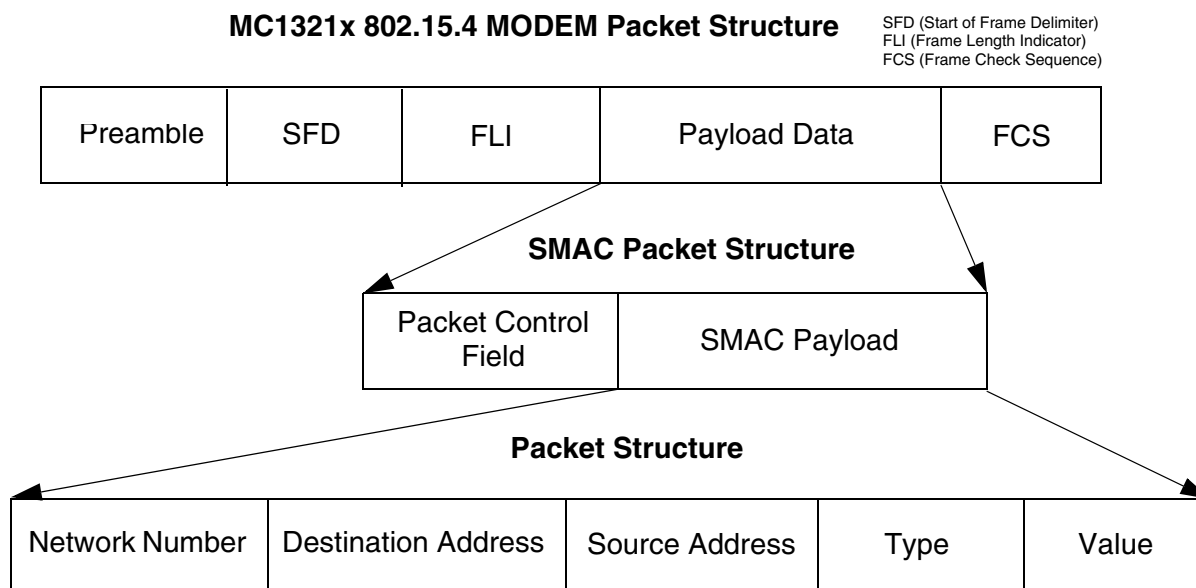
Figure 10. Packet Structure

The Proximity Sensor-based Remote Control packet becomes a payload data for the SMAC standard packet and contains the following fields:

- **Network Number** - is the number defined in the software initialization field and is dedicated for membership definition of devices in one network group. If an equal number is not received, the packet is ignored. It is an *unsigned int* type of variable.
- **Destination Address** - is the number defined in the software initialization field of each Intelligent Outlet and is selected by the Intelligent Switch. It is dedicated for defining the destination Intelligent Outlet. If a number not equal to the device number is received, the packet is ignored. It is an *unsigned char* type of variable.
- **Source Address** - is the number defined in the software initialization field of the Intelligent Switch and is dedicated for defining the correct control source. If the right number is not received, the packet is ignored. It is an *unsigned char* type of variable.
- **Type** - defines the respective type of packet command, which will execute this packet. It is an *unsigned char* type of variable.
- **Value** - user data used for transmitting the defining step value in remote changes to the dimming value. It is an *unsigned int* type of variable.

Table 4. Commands List

Command	Code	Direction	Function
LIGHT_UP	'u'	PSRC to IS or IO	Increasing the light intensity by **Value**
LIGHT_DOWN	'd'	PSRC to IS or IO	Decreasing the light intensity by **Value**
LIGHT_SWITCH	's'	PSRC to IS or IO	Defines the press of the ON/OFF pad

Only the simple commands shown in Table 4 are transmitted with a value of the changes. The RF communication is realized by a combination of Doze mode in the MODEM, STOP3 modes in the MCU, wait delays and delayed windows for data transfer. The PSRC sends commands by 27 repeated packets with 1 ms wide delays. See Figure 11 for a timing diagram.
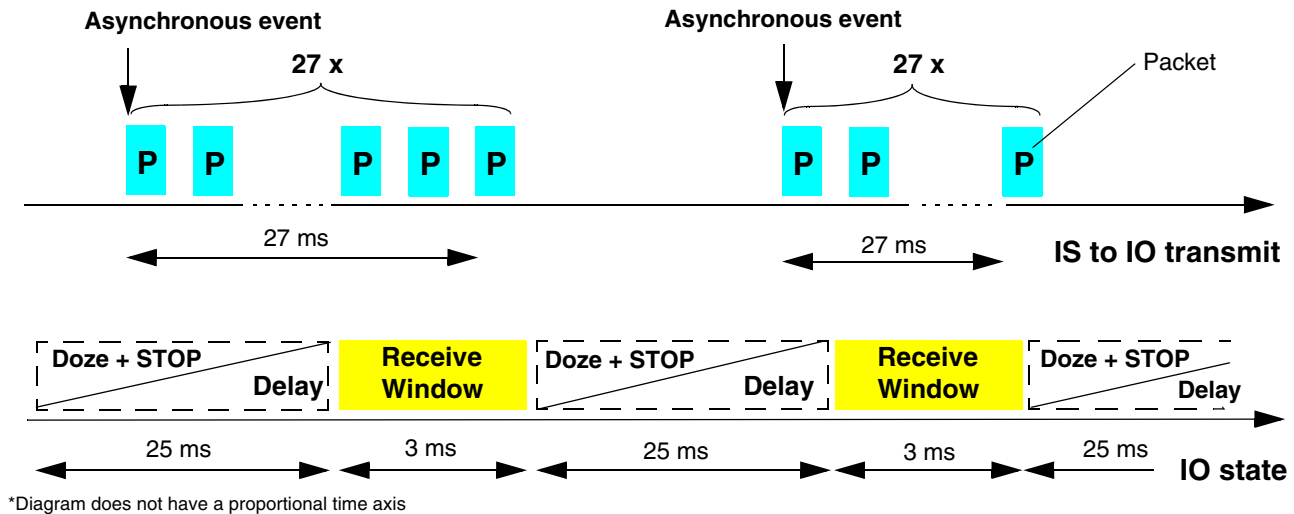
Figure 11. Communication Process

There are several routines provided by the SMAC library situated in the *simple_mac.c* file, which are used in the MC13213 firmware for realization of RF communication:

- *MLMESetMC13192ClockRate* - Set the clock speed for the MC13192 modem
- *MLMEMC13192PAOutputAdjust* - Set the power setting of the antenna signal
- *lMLMESetChannelRequest* - Set the channel used for communication
- *MLMESetMC13192TmrPrescale* - Set the prescaler for definition of the clock
- *MCPSDataRequest(&tx_packet)* - Send the data packet by the RF modem
- *MLMEDozeRequest* - Put the RF modem into the Doze state for a defined time

4.1.3 Main Program Loop

The main program loop is realized in file **main.c**. At the beginning of this file there is initialization which includes these parts: hardware of the MCU and watchdog, I^2C (for MC9S08QG8) driver and Timers settings. Figure 12 on the following page, describes the simplified functions of the system.

AN3783, Rev 0
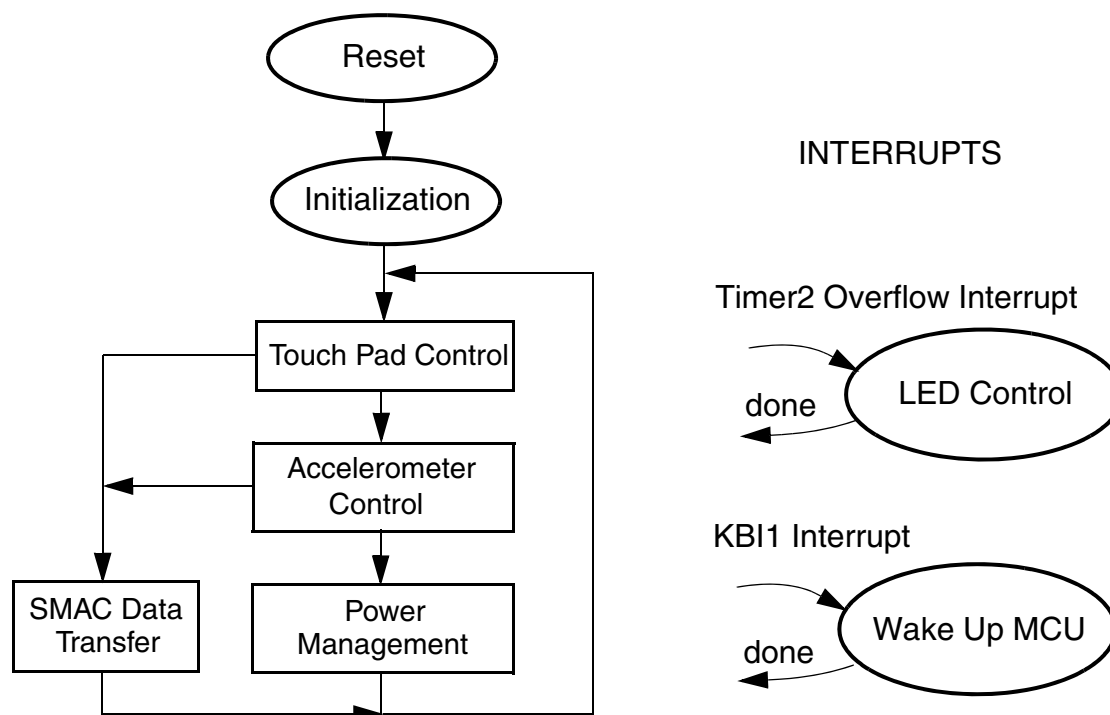
Figure 12. State Diagram - General Overview

In the main program loop these actions are periodically realized:

- Touch Sensing Software (TSS) re-initialization at the start of the system
- Periodically read any data from the TSS on touched pads by using the functions *GetTPadFirst* and *GetTPadSecond*
- Perform RF communication with the controlled device according to the data received from the TSS
- LED control
- Enabling Low Power Consumption mode according to user control of the device, and the MCU goes into STOP1 mode. If an IRQ signal is generated by the TSS after detecting a Finger Touch, then the MC13213 is woken up from STOP1 mode.

Apart from these periodical actions, two independent HW interrupts can occur in real time: the first from Timer2 for LED indication management, the second for IRQ detection from MC9S08QG8 by the KBI1 module.

4.2   MC9S08QG8 Firmware

The MC9S08QG8 MCU contains proprietary Capacitive Touch Sensing Software which detects contact-less finger touch on the dielectric material by the sensing electrodes. The measurement principle is based on the original Proximity Software Library for driving the GPIO pins, and it has been updated for this kind of application. The MC9S08QG8 MCU uses the I^2C bus for sending data on the detection status to the main control MC13213 MCU.

5 Usage of Remote Control for Light Dimming Applications

The Device is intended for the remote control of E-Field Lighting Controller with Wireless Connectivity demonstrator parts as shown in Figure 13.This device can be easily customized for any other purpose by software redesign, and it can be reprogrammed by a BDM connector.
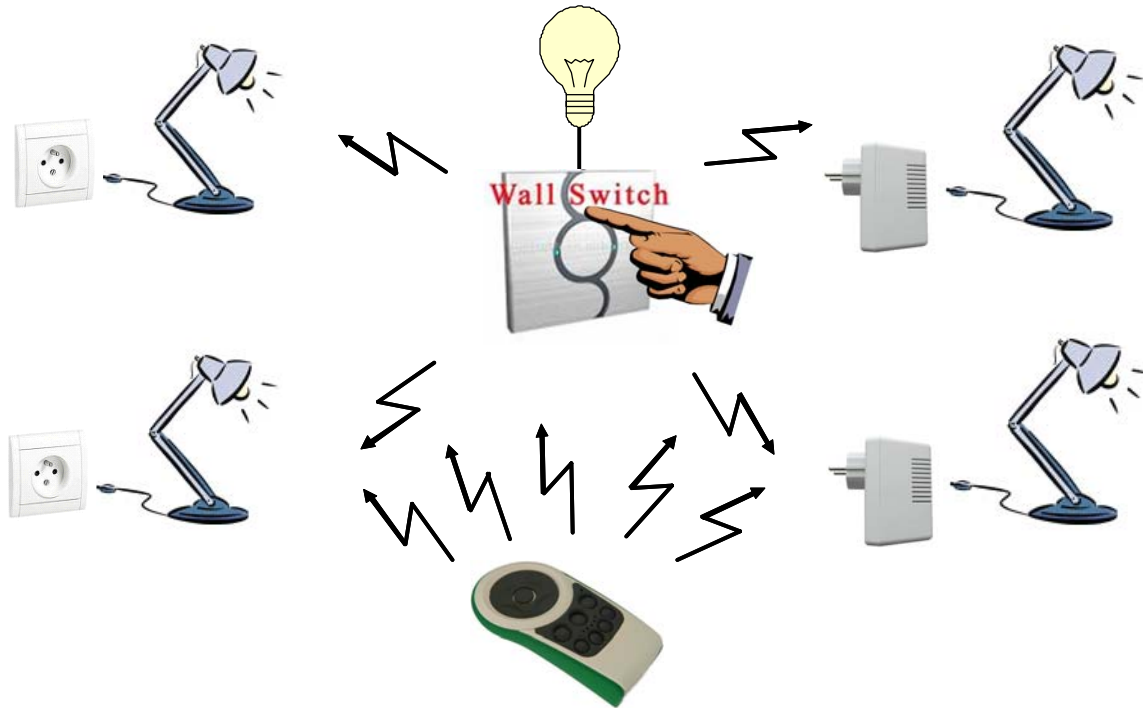
Figure 13. System Functional Diagram

The application can be controlled by two main interfaces provided by the Remote Control, the touch pad, or the Accelerometer. The touch pad interface offers these following functions:

- On/Off lights by the central pad on the touch pad board
- Fluent regulation of light intensity by clockwise or counter-clockwise finger movement on the rotary electrodes
- Selection of a controlled device by the five pads situated on the Main Control Board (selected devices are indicated by five LED's situated in the center of the board)
- On/Off Accelerometer function by one long press of the Left/Right button on the rotary dial

The touch pad functions are described in more detail in Figure 14.
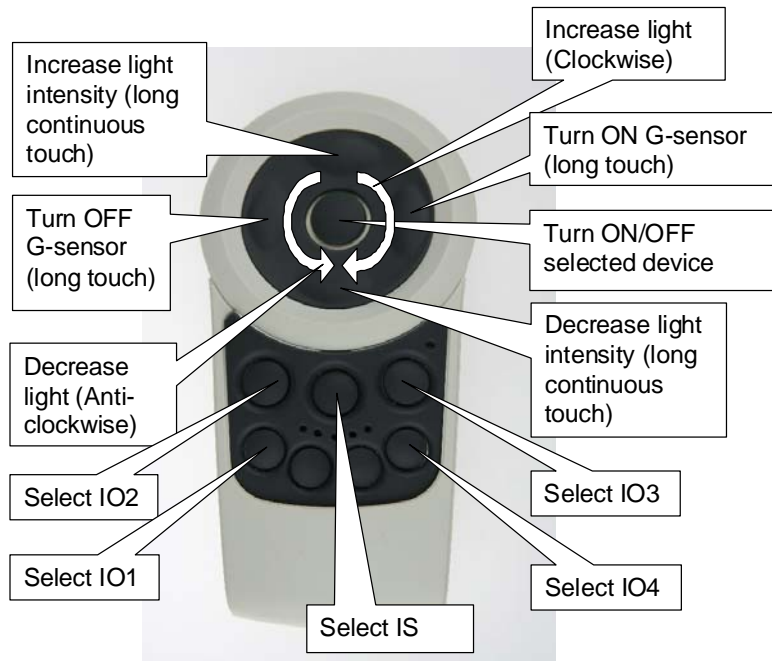
Figure 14. Application Control by the Touch Pad

The application can also be controlled by tilting the remote control enclosure, measured by the accelerometer. Light intensity is simply changed by a slow clockwise or counter-clockwise rotation of the device (See Figure 15). The angle is not equal to the value of the light intensity. The step of light intensity changing depends on the remote control's angle change in the specific sampling frequency, not on the angle value. A very fast tilting causes a loss of samples, so a slower rotation is more precise.

Figure 15. Tilting Control Description

The actual status of the device is indicated by the LED situated on the front face of the touch pad (See Figure 6). The signification of the LED states is written in the Table 5.

Table 5. LED Signification

LED	Function
LED1	Permanently Lit - Normal MODE Fast Blinking - G-Sensor MODE Slow Blinking - Standby MODE
LED2	Intelligent Outlet 4 Selected
LED3	Intelligent Outlet 3 Selected
LED4	Intelligent Switch Selected
LED5	Intelligent Outlet 2 Selected
LED6	Intelligent Outlet 1 Selected

5.1 References

E-Field Lighting Controller with Wireless Connectivity, DRM096, RDM, Freescale Semiconductor, Rev.1.4, 11/2007

MC13211/212/213/214 ZigBee®- Compliant Platform -2.4 GHz Low Power Transceiver for the IEEE® 802.15.4 Standard plus Microcontroller, MC1321xRM, RDM, Freescale Semiconductor, Rev.1.1, 10/2006

±3g, ±11g Three Axis Low-g Micromachined Accelerometer, MMA7340L, Data Sheet, Freescale Semiconductor, Rev.2, 08/2007

±2g/±4g/±8g Three Axis Low-g Digital Output Accelerometer, MMA7455L, Data Sheet, Freescale Semiconductor, Rev.4, 08/2008

MC9S08QG8, Data Sheet, Freescale Semiconductor, Rev.4, 02/2008

Simple Media Access Controller, SMACRM, RDM, Freescale Semiconductor, Rev.1.5, 03/2008

Enabling an MCU for Touch Sensing Proximity Sensor Software, AN3579, Application Note, Freescale Semiconductor, Rev.0, 11/2007

6   Appendix A - Remote Control Application PCB
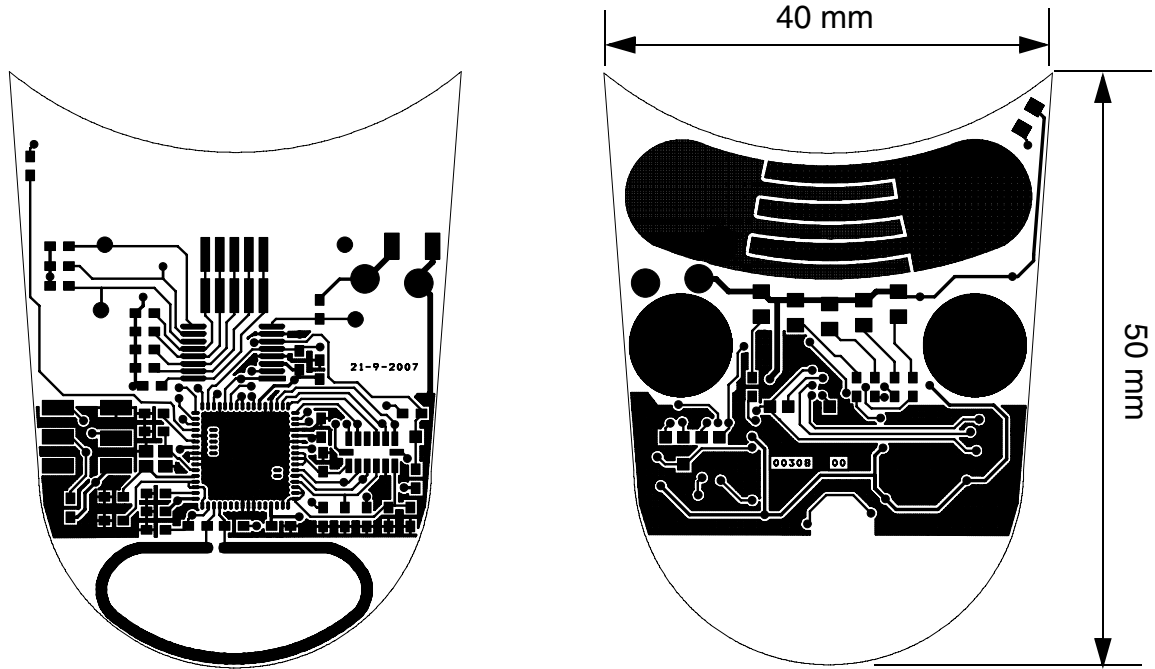
Figure 16. Top and Bottom Layer of the Main Control Board (not to scale)
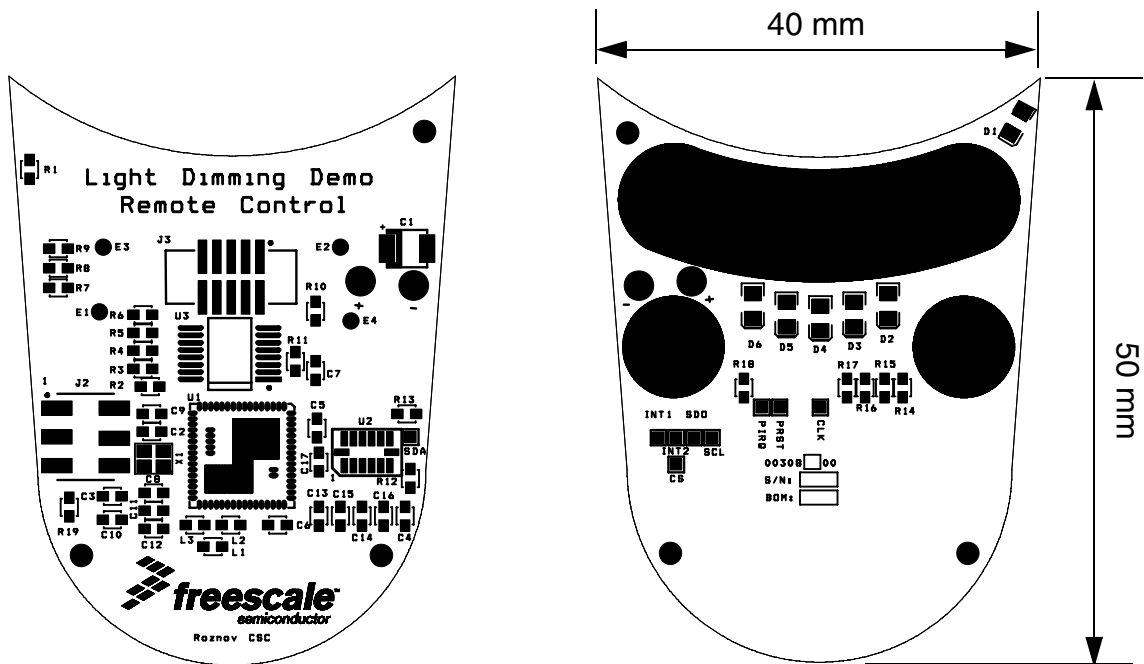
Figure 17. Component Placement On the Top and Bottom Side of the Main Control Board (not to scale)

AN3783, Rev 0
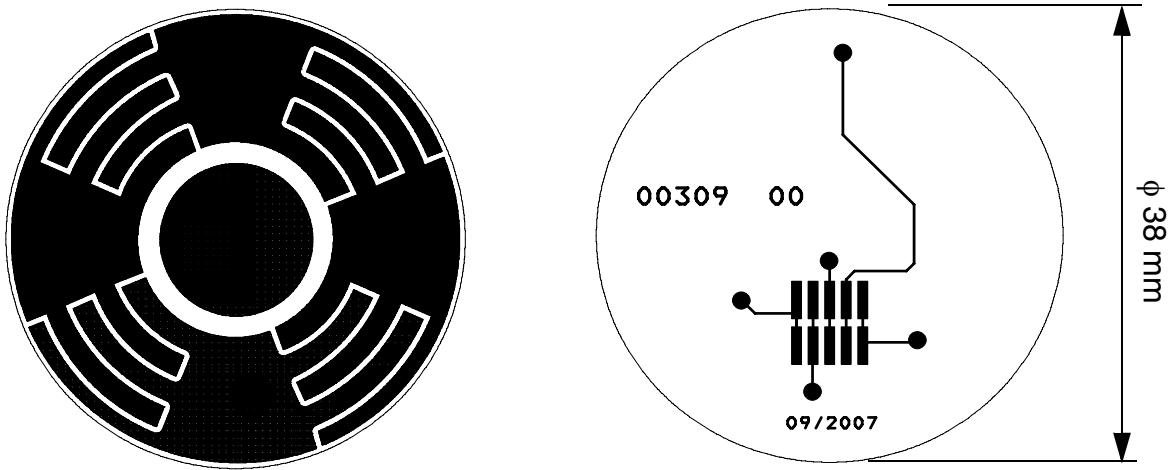
Figure 18. Top and Bottom Layer of the Touch Pad Board (not to scale)
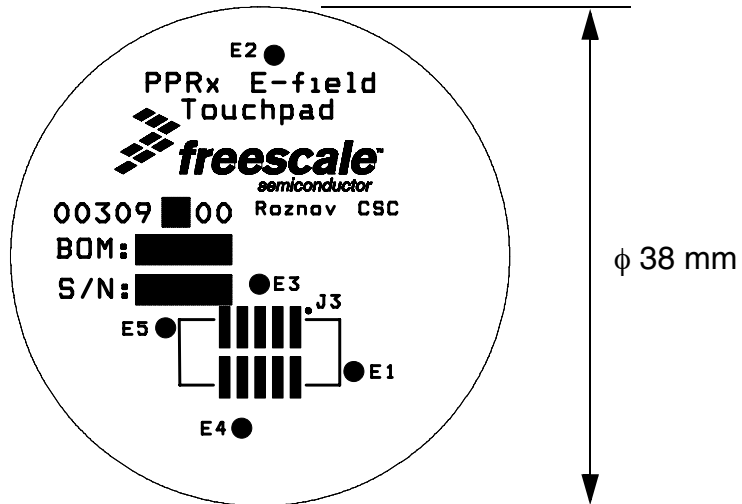
Figure 19. Component Placement on the Bottom Side of the Main Control Board (not to scale)

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 010 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

freescale™
semiconductor