Appendix A

Appendix code for the transmitter side:

```
$regfile = "m16def.dat"
$crystal = 8000000
$baud = 2000
$hwstack = 32
$swstack = 10
$framesize = 40
Config Lcd = 16 * 2
Config Lcdpin = Pin , Db4 = Portc.4 , Db5 = Portc.5 , Db6 = Portc.6 ,
Db7 = Portc.7 , E = Portc.2 , Rs = Portc.0
Config Adc = Single , Prescaler = Auto
Config Portb.0 = Input
Config Portb.1 = Input
Dim C As Integer
Dim W As Word
Dim Temp As Word
Dim V As Single

'Program
Cursor Off
Locate 1 , 1
Lcd "sw 1="
Locate 2 , 1
Lcd "sw 2="
```

```
Do
If Pinb.0 = 0 And Pinb.1 = 0 Then
Printbin 1
Elseif Pinb.0 = 0 And Pinb.1 = 1 Then
Printbin 2
Elseif Pinb.0 = 1 And Pinb.1 = 1 Then
Printbin 3
Elseif Pinb.0 = 1 And Pinb.1 = 0 Then
Printbin 4

End If
If Pinb.0 = 1 Then
Locate 1 , 6
Lcd "on "
Else
Locate 1 , 6
Lcd "off"
End If
If Pinb.1 = 1 Then
Locate 2 , 6
Lcd "on "
Else
Locate 2 , 6
Lcd "off"
End If
Loop
```

Appendix B

code for receiver side:

```
$regfile = "m16def.dat"
$crystal = 4000000
$baud = 2000
$hwstack = 32
$swstack = 10
$framesize = 40
Config Lcd = 16 * 2
Config Lcdpin = Pin , Db4 = Portc.4 , Db5 = Portc.5 , Db6 =
Portc.6 , Db7 = Portc.7 , E = Portc.2 , Rs = Portc.0
Config Adc = Single , Prescaler = Auto
Config Submode = New
Config Timer0 = Counter , Edge = Rising , Prescale = 1024
Config Portb.0 = Input
Config Portb.1 = Input
Config Portb.2 = Input
Config Portb.3 = Input
Config Portb.7 = Output
Sub Frequency1()
Config Timer1 = Pwm , Pwm = 9 , Compare A Pwm = Clear Down
, Prescale = 1024
'4 hz
End Sub
Sub Frequency2()
Config Timer1 = Pwm , Pwm = 8 , Compare A Pwm = Clear Down
, Prescale = 1024
```

```
'8 hz
End Sub
Sub Frequency3()
Config Timer1 = Pwm , Pwm = 9 , Compare A Pwm = Clear Down
, Prescale = 256
'16 hz
End Sub
Sub Frequency4()
Config Timer1 = Pwm , Pwm = 8 , Compare A Pwm = Clear Down
, Prescale = 256
'30 hz
End Sub
Sub Frequency5()
Config Timer1 = Pwm , Pwm = 9 , Compare A Pwm = Clear Down
, Prescale = 64
'61 hz
End Sub
Sub Frequency6()
Config Timer1 = Pwm , Pwm = 8 , Compare A Pwm = Clear Down
, Prescale = 64
'123 hz
End Sub

Sub Frequency7()
Config Timer1 = Pwm , Pwm = 10 , Compare A Pwm = Clear
Down , Prescale = 8
'244 hz
End Sub
Dim Moode As Word
```

```
Dim X As Word
Dim Y As Word
Dim F As Word
Dim Z As Word
Dim M As Word
Dim Ppwm As Word
Moode = 0
Ppwm = 23
Y = 0
Z = 0
M = 0
Portb.7 = 0
Locate 1 , 1
Lcd "Enter frequency"
Do
X = Waitkey()
If Pinb.0 = 0 Then
If Moode < 9 Then
Moode = Moode + 1
Else
Moode = 0
End If
End If
If Moode = 0 Then
Locate 2 , 1
Lcd "4 HZ          "
If Pinb.1 = 0 Then
If X = 2 Or X = 3 Or X = 4 Then
Frequency1
```

```
F = 1
Pwm1a = Ppwm
End If
Moode = 8
End If
End If
If Moode = 1 Then
Locate 2 , 1
Lcd "8 HZ         "
If Pinb.1 = 0 Then
If X = 2 Or X = 3 Or X = 4 Then
Frequency2
Pwm1a = Ppwm
F = 2
End If
Moode = 8
End If
End If
If Moode = 2 Then
Locate 2 , 1
Lcd "16 HZ        "
If Pinb.1 = 0 Then
If X = 2 Or X = 3 Or X = 4 Then
Frequency3
Pwm1a = Ppwm
F = 3
End If
Moode = 8
End If
```

```
End If
If Moode = 3 Then
Locate 2 , 1
Lcd "30 HZ        "
If Pinb.1 = 4 Then
If X = 2 Or X = 3 Or X = 4 Then
Frequency4
Pwm1a = Ppwm
F = 4
End If
Moode = 8
End If
End If
If Moode = 5 Then
Locate 2 , 1
Lcd "60 HZ        "
If Pinb.1 = 0 Then
If X = 2 Or X = 3 Or X = 4 Then
Frequency5
Pwm1a = Ppwm
F = 5
End If
Moode = 8
End If
End If
If Moode = 6 Then
Locate 2 , 1
Lcd "123 HZ       "
If Pinb.1 = 0 Then
```

```
If X = 2 Or X = 3 Or X = 4 Then
Frequency6
Pwm1a = Ppwm
F = 6
End If
Moode = 8
End If
End If
If Moode = 7 Then
Locate 2 , 1
Lcd "244 HZ       "
If Pinb.1 = 0 Then
If X = 2 Or X = 3 Or X = 4 Then
Frequency7
Pwm1a = Ppwm
F = 7
End If
Moode = 8
End If
End If
If Pinb.2 = 0 Then
Pwm1a = Pwm1a - 10
End If
If Pinb.3 = 0 Then
Pwm1a = Pwm1a + 10
End If
If Pwm1a <> 0 Then
Ppwm = Pwm1a
End If
```

```
If Moode = 8 Then
Locate 2 , 1
Lcd "Pwm="
Lcd Pwm1a
Lcd "b1++ b2--"
 If X = 1 And X <> Z Then
Pwm1a = 0
Locate 1 , 1
Lcd "stand mode     "
Elseif X <> Z Then
 Locate 1 , 1
Lcd "walk  mode     "
Select Case F
Case 1 : Frequency1
Case 2 : Frequency2
Case 3 : Frequency3
Case 4 : Frequency4
Case 5 : Frequency5
Case 6 : Frequency6
Case 7 : Frequency7
End Select
Pwm1a = Ppwm
End If
End If
If X = 1 Or X = 3 And M = 0 Then
Y = 0
Z = X
M = 1
End If
```

```
If Y >= 0 And Y < 20 The
If X <> Z Then
Y = 0
M = 0
End If
End If

If Y = 20 Then
Y = 0
M = 0
Pwm1a = 0
 Locate 1 , 1
 Lcd "sleep mode     "
If Pinb.7 = 1 Then
Portb.7 = 0
Else
Portb.7 = 1
End If
End If
Y = Y + 1
Waitms 130
Loop
```

Appendix C

AT mega 16

Features:

• High-performance, Low-power Atmel® AVR® 8-bit Microcontroller

• Advanced RISC Architecture

– 131 Powerful Instructions – Most Single-clock Cycle Execution

– 32 x 8 General Purpose Working Registers

– Fully Static Operation

– Up to 16 MIPS Throughput at 16 MHz

– On-chip 2-cycle Multiplier

• High Endurance Non-volatile Memory segments

– 16 Kbytes of In-System Self-programmable Flash program memory

– 512 Bytes EEPROM

– 1 Kbyte Internal SRAM

– Write/Erase Cycles: 10,000 Flash/100,000 EEPROM

– Data retention: 20 years at 85°C/100 years at 25°C(1)

– Optional Boot Code Section with Independent Lock Bits

In-System Programming by On-chip Boot Program

True Read-While-Write Operation

– Programming Lock for Software Security

• JTAG (IEEE std. 1149.1 Compliant) Interface

– Boundary-scan Capabilities According to the JTAG Standard

– Extensive On-chip Debug Support

– Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface

• Peripheral Features

– Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes

– One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and CaptureMode

– Real Time Counter with Separate Oscillator

– Four PWM Channels

– 8-channel, 10-bit ADC

8 Single-ended Channels

7 Differential Channels in TQFP Package Only

2 Differential Channels with Programmable Gain at 1x, 10x, or 200x

– Byte-oriented Two-wire Serial Interface

– Programmable Serial USART

– Master/Slave SPI Serial Interface

– Programmable Watchdog Timer with Separate On-chip Oscillator

– On-chip Analog Comparator

• Special Microcontroller Features

– Power-on Reset and Programmable Brown-out Detection

– Internal Calibrated RC Oscillator

– External and Internal Interrupt Sources

– Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standbyand Extended Standby

• I/O and Packages

– 32 Programmable I/O Lines

– 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF

• Operating Voltages

– 2.7V - 5.5V for ATmega16L

– 4.5V - 5.5V for ATmega16

• Speed Grades

– 0 - 8 MHz for ATmega16L

– 0 - 16 MHz for ATmega16

• Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L

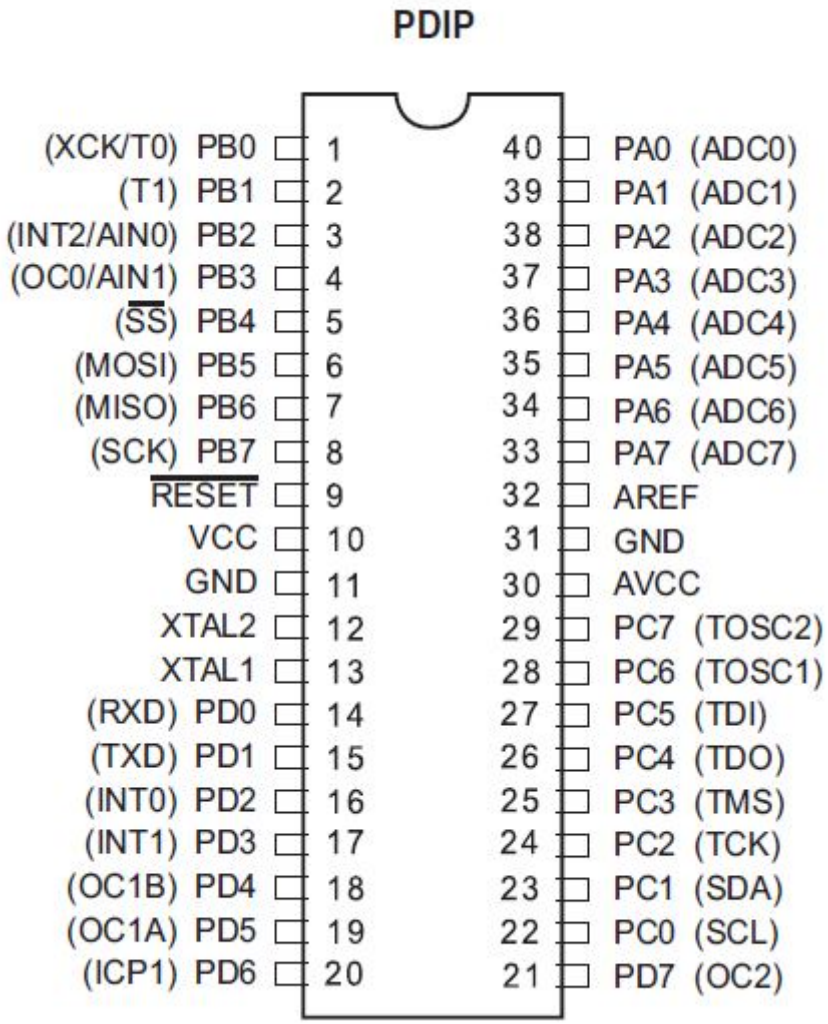
– Active: 1.1 mA

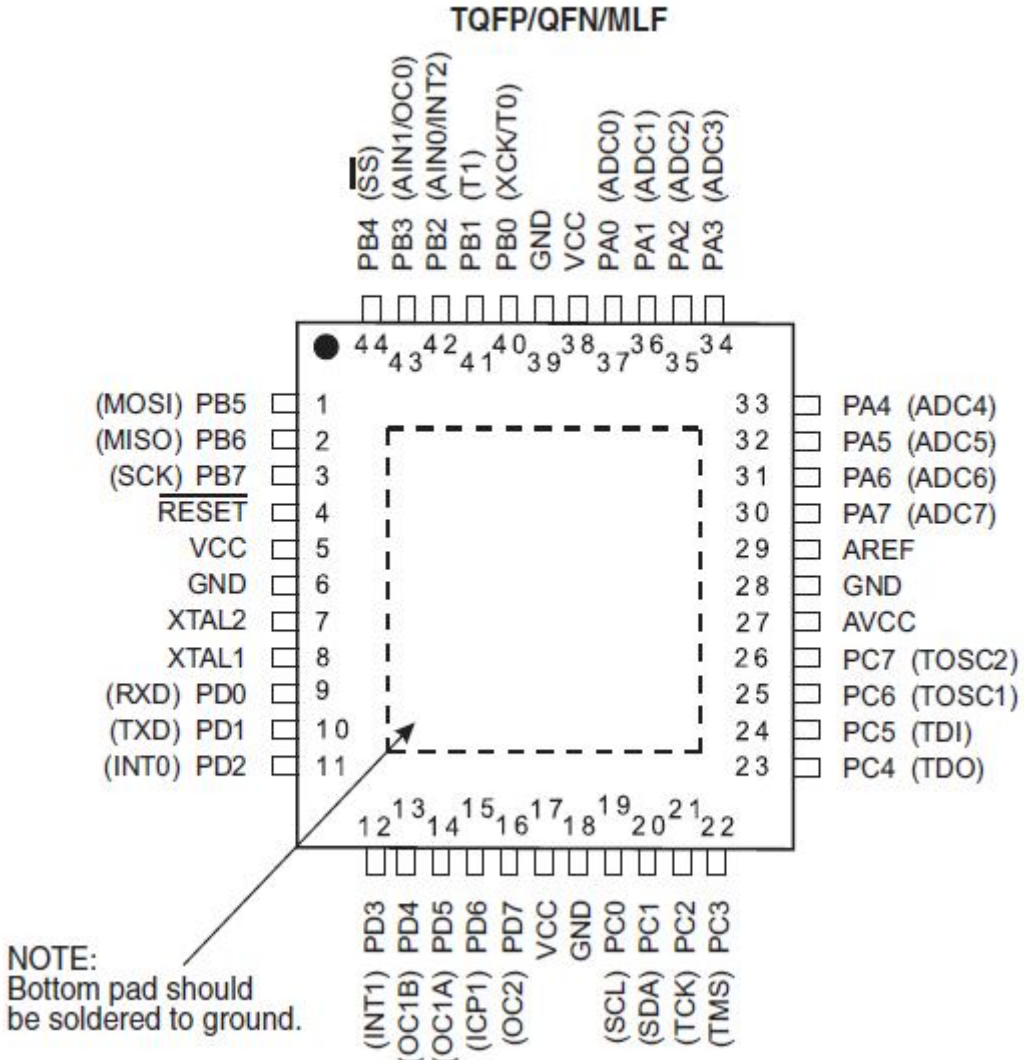
– Idle Mode: 0.35 mA

– Power-down Mode: < 1 μA

PinConfigurations

Figure 1. Pinout ATmega16



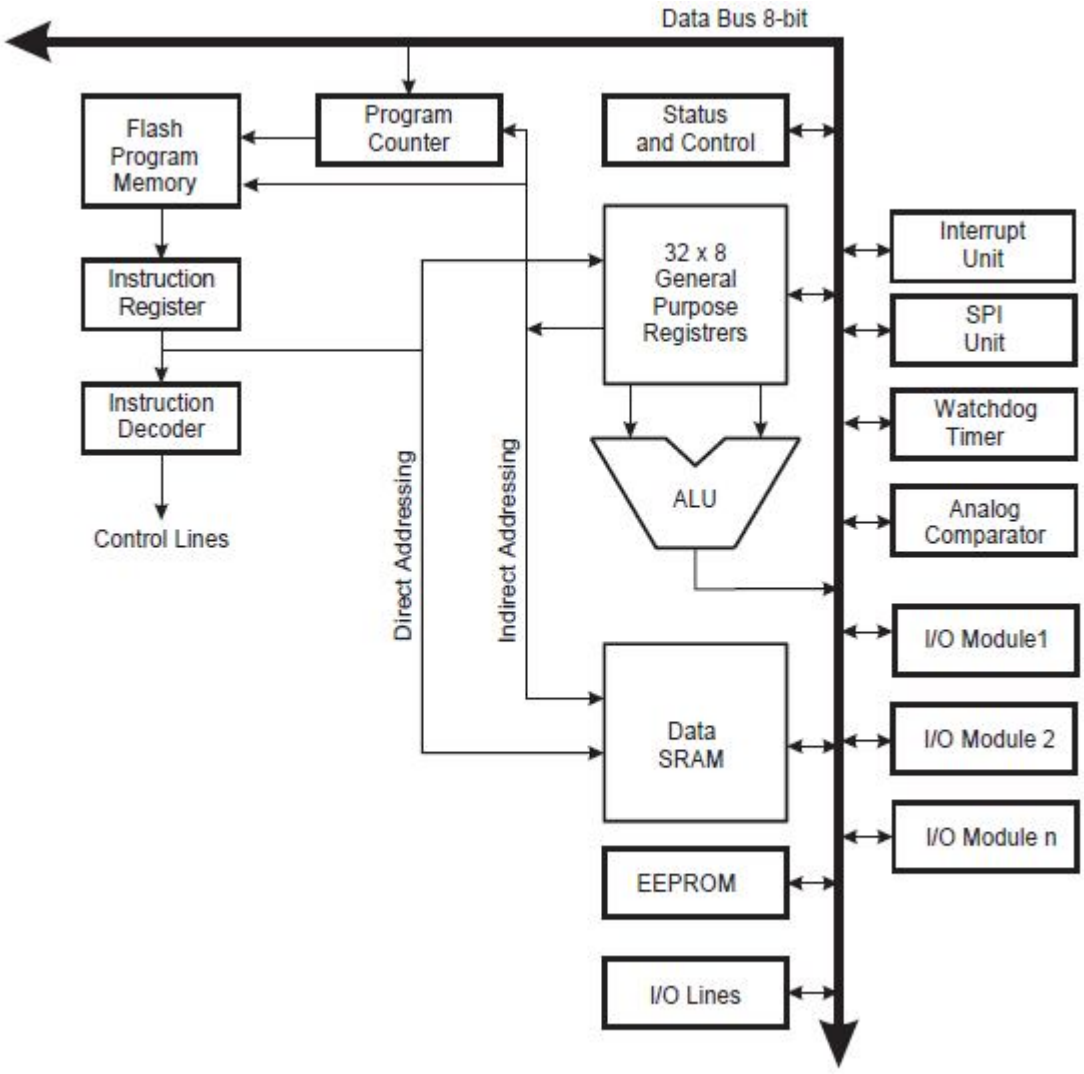


AVR CPU Core

Introduction This section discusses the AVR core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access memories,perform calculations, control peripherals, and handle interrupts.

Architectural Overview

Figure 3. Block Diagram of the AVR MCU Architecture



In order to maximize performance and parallelism, the AVR uses a Harvard architecture – withseparate memories and buses for program and data. Instructions in the program memory areexecuted with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executedin every clock cycle. The program memory is In-System Reprogrammable Flash memory.

The fast-access Register File contains 32 × 8-bit general purpose working registers with a singleclock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typicalALU operation, two operands are output from the Register File, the operation is executed,and the result is stored back in the Register File – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for DataSpace addressing – enabling efficient address calculations. One of the these address pointerscan also be used as an address pointer for look up tables in Flash Program memory. Theseadded function registers are the 16-bit X-register, Y-register, and Z-register, described later inthis section.

The ALU supports arithmetic and logic operations between registers or between a constant anda register. Single register operations can also be executed in the ALU. After an arithmetic operation,the Status Register is updated to reflect information about the result of the operation.

Program flow is provided by conditional and unconditional jump and call instructions, able todirectly address the whole address space. Most AVR instructions have a single 16-bit word format.

Every program memory address contains a 16-bit or 32-bit instruction.

Program Flash memory space is divided in two sections, the Boot program section and theApplication Program section. Both sections have dedicated Lock bits for write and read/writeprotection. The SPM instruction that writes into the Application Flash memory section mustreside in the Boot Program section.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on theStack. The Stack is effectively allocated in the general data SRAM, and consequently the Stacksize is only limited by the total SRAM size and the usage of the SRAM. All user programs mustinitialize the SP in the reset routine (before subroutines or interrupts

are executed). The StackPointer SP is read/write accessible in the I/O space. The data SRAM can easily be accessedthrough the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps.

A flexible interrupt module has its control registers in the I/O space with an additional globalinterrupt enable bit in the Status Register. All interrupts have a separate interrupt vector in theinterrupt vector table. The interrupts have priority in accordance with their interrupt vector position. The lower the interrupt vector address, the higher the priority.

Appendix D

Zigbee (cc2520)

APPLICATIONS:

•IEEE 802.15.4 systems
•ZigBee® systems
•Industrial monitoring and control
•Home and building automation
•Automatic Meter Reading
•Low-power wireless sensor networks
•Set-top boxes and remote controls
•Consumer electronics

KEY FEATURES

•State-of-the-art selectivity/co-existence
Adjacent channel rejection: 49 dB
Alternate channel rejection: 54 dB
•Excellent link budget (103dB)
400 m Line-of-sight range
•Extended temp range (-40 to +125°C)
•Wide supply range: 1.8 V – 3.8 V
•Extensive IEEE 802.15.4 MAC hardware support to offload the microcontroller
•AES-128 security module
•CC2420 interface compatibility mode

Low Power

•RX (receiving frame, -50 dBm) 18.5 mA
•TX 33.6 mA @ +5 dBm
•TX 25.8 mA @ 0 dBm
•<1μA in power down

General

•Clock output for single crystal systems
•RoHS compliant 5 x 5 mm QFN28 (RHD) package

Radio

•IEEE 802.15.4 compliant DSSS baseband modem with 250 kbps data rate
•Excellent receiver sensitivity (-98 dBm)
•Programmable output power up to +5 dBm
•RF frequency range 2394-2507 MHz
•Suitable for systems targeting compliance with worldwide radio frequency

regulations: ETSI EN 300 328 and EN
300 440 class 2 (Europe), FCC CFR47 Part
15 (US) and ARIB STD-T66 (Japan)

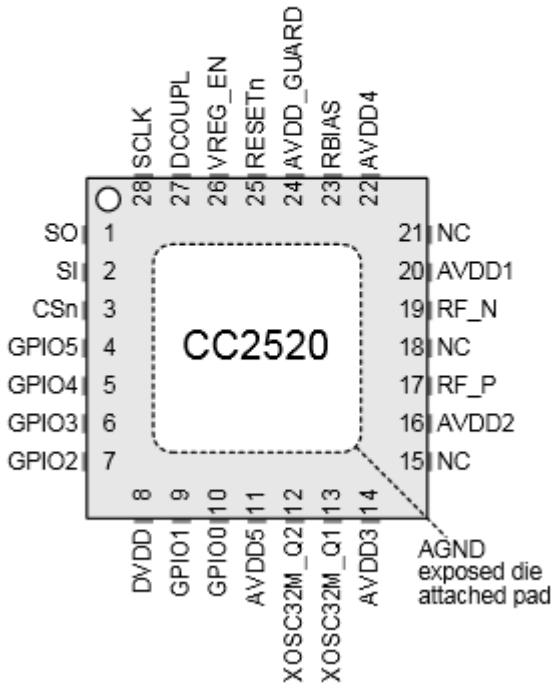
Microcontroller Support

•Digital RSSI/LQI support
•Automatic clear channel assessment for CSMA/CA
•Automatic CRC
•768 bytes RAM for flexible buffering and security processing
•Fully supported MAC security
•4 wire SPI
•6 configurable IO pins
•Interrupt generator
•Frame filtering and processing engine
•Random number generator

Development Tools

•Reference design
•IEEE 802.15.4 MAC software
•ZigBee® stack software
•Fully equipped development kit
•Packet sniffer support in hardware

QFN28 (RHD) PACKAGE
TOP VIEW



Feature:

2394-2507MHz transceiver
•DSSS transceiver
•250kbps data rate, 2 MChip/s chip rate
•O-QPSK with half sine pulse shaping modulation
•Very low current consumption
RX (receiving frame, -50 dBm): 18.5 mA
RX (waiting for frame): 22.3 mA
TX (+5 dBm output power): 33.6 mA
TX (0 dBm output power): 25.8 mA
•Three flexible power modes for reduced power consumption
•Low power fully static CMOS design
•Very good sensitivity (-98dBm)
•High adjacent channel rejection (49 dB)
•High alternate channel rejection (54 dB)
•On chip VCO, LNA, PA and filters.
•Low supply voltage (1.8 - 3.8 V)
•Programmable output power up to +5 dBm
•I/Q direct conversion transceiver
Small Size
•QFN 28 (RHD) package, 5 x 5 mm
•Very few external components
ominimized number of passives
oOnly reference crystal needed
•Clock output for other ICs to limit the
number of crystals needed in a system
•No external filters needed.
Easy and Flexible User Interface
•4-wire SPI
•Serial clock up to 8 MHz
•6 GPIO pins with full flexibility
•Interrupt generator
•Full control of automatic responses to different events
•Embedded packet sniffer mode
•CC2420 compatibility mode
Data Processing Unit For Advanced Data Handling
•Spacious (768 byte) on-chip RAM a
llows powerful on-chip frame processing
•128 byte transmit data FIFO
•128 byte receive data FIFO
•Full read and write access to RAM
•128 bit AES
IEEE 802.15.4 MAC Hardware Support
•Automatic preamble generator

•Synchronization word insertion and detection
•CRC-16 computation and verification over the MAC payload
•Frame filtering
•Automatic ACK and setting of the pending-bit
•Clear Channel Assessment (CCA)
•Energy detection / RSSI
•Link Quality Indication (LQI)
•Fully automatic MAC security (CTR, CBC-MAC, CCM)