# 2.4GHz RF transceiver with embedded 8051 compatible microcontroller and 9 input, 10 bit ADC

# nRF24E1

## FEATURES

- nRF2401 2.4GHz RF transceiver
- 8051 compatible microcontroller
- compatible with nRF24E2
- 9 input 10 bit ADC 100kSPS
- Single 1.9V to 3.6V supply
- Internal voltage regulators
- 2 µA standby with wakeup on timer or external pin
- Internal VDD monitoring
- Supplied in 36 pin QFN (6x6mm) package
- 0.18µm CMOS technology
- Mask programmable version available
- Low Bill of Material
- Ease of design

## APPLICATIONS

- Wireless gamepads
- Wireless headsets
- Wireless keyboards
- Wireless mouse
- Wireless toys
- Intelligent sports equipment
- Industrial sensors
- PC peripherals
- Phone peripherals
- Tags
- Alarms
- Remote control

## TABLE OF CONTENT

# 1 GENERAL DESCRIPTION

The nRF24E1 is a nRF2401 2.4GHz radio transceiver with an embedded 8051 compatible microcontroller and a 10-bit 9 input 100 kSPS AD converter. The circuit is supplied by only one voltage in range 1.9V to 3.6V. The nRF24E1 supports the proprietary and innovative modes of the nRF2401 such as ShockBurst™ and DuoCeiver™.

nRF24E1 is a superset of the nRF24E2 chip, which means that it contains all functions of nRF24E2, and that it is fully program compatible with nRF24E2.

## 1.1 Quick Reference Data

| Parameter | Value | Unit |
|---|---|---|
| Minimum supply voltage | 1.9 | V |
| Temperature range | -40 to +85 | ° C |
| Maximum RF output power | 0 | dBm |
| RF receiver sensitivity | -90 | dBm |
| Maximum RF burst data rate | 1000 | kbps |
| Supply current for microcontroller @ 16MHz @3V | 3 | mA |
| Supply current for ADC @100 kSPS | 0.9 | mA |
| Supply current for RF transmit @ -5dBm output power | 10.5 | mA |
| Supply current for RF receive @1000 kbps | 19 | mA |
| Supply current in Power Down mode | 2 | µA |
| max CPU clock frequency | 20 | MHz |
| max AD conversion rate | 100 | kSPS |
| ADC Differential nonlinearity (DNL) | ±0.5 | LSB |
| ADC Integral nonlinearity (INL) | ±0.75 | LSB |
| ADC Spurious free dynamic range (SFDR) | 65 | dB |
| Package | 36 pin QFN 6x6 | |

Table 1-1 : nRF24E1 quick reference data

| Type Number | Description | Version |
|---|---|---|
| NRF24E1G | 36 pin QFN 6x6, RoHS & SS-00259 compliant | B |
| NRF24E1-EVKIT | Evaluation kit | 1.0 |

Table 1-2 : nRF24E1 ordering information

## 1.2 Block Diagram

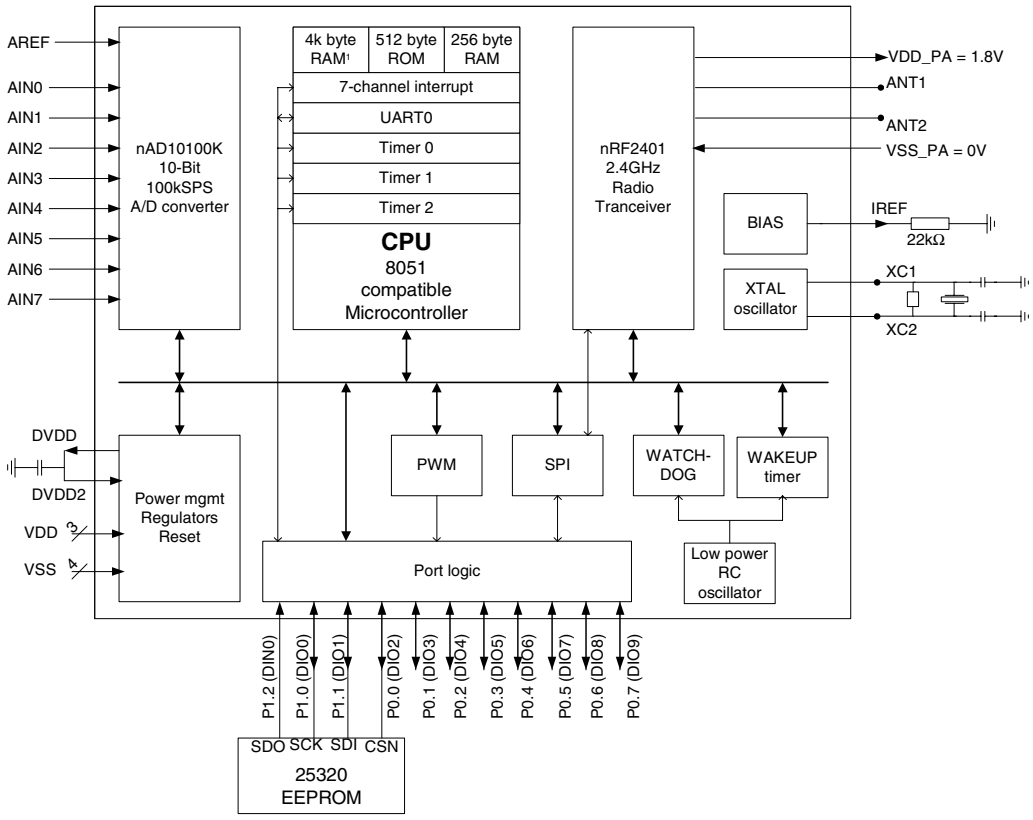
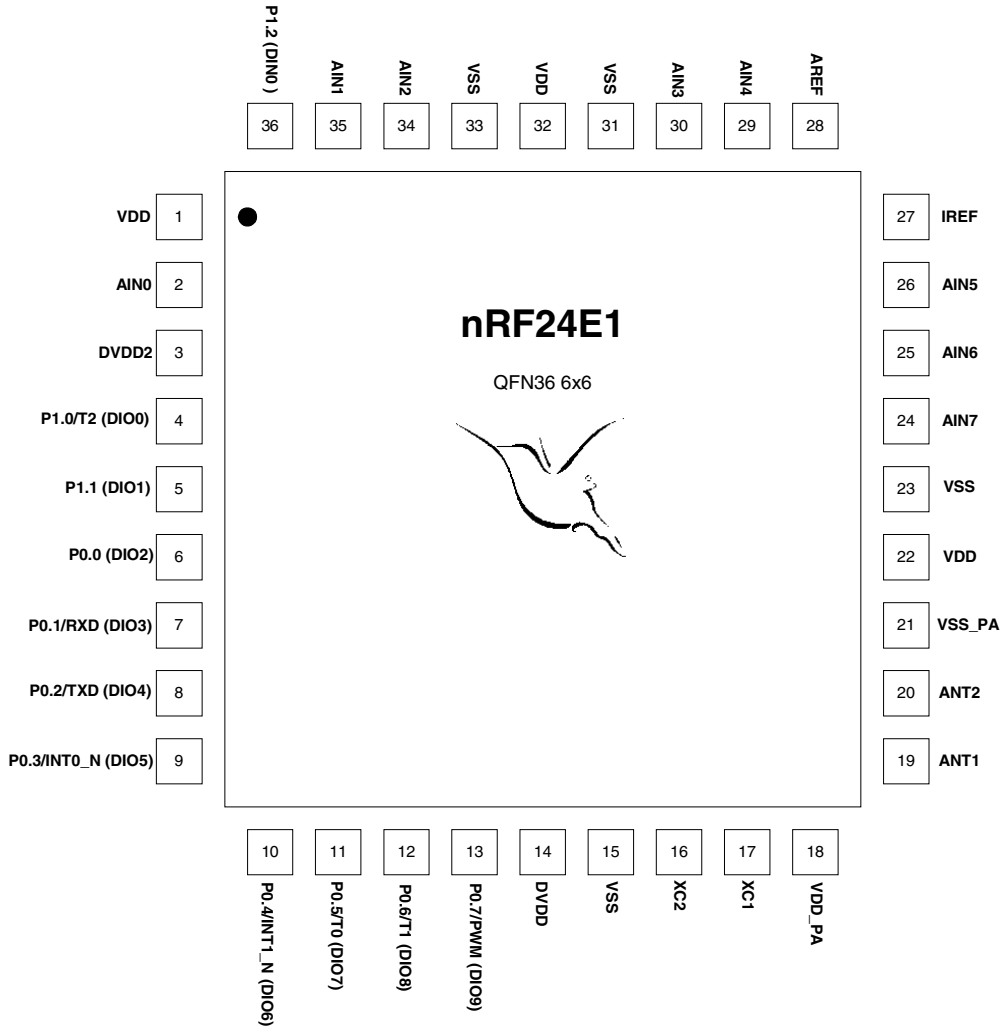



Figure 1-1 nRF24E1 block diagram plus external components

## 1.3 Pin Diagram




| Pin | Name | Pin function | Description |
|---|---|---|---|
| 1 | VDD | Power | Power Supply (1.9-3.6 V DC) |
| 2 | AIN0 | Analog input | ADC input 0 |
| 3 | DVDD2 | Regulated power | Digital Power Supply , must be connected to regulator output DVDD |
| 4 | P1.0/T2 | Digital I/O | Port 1, bit 0 or T2 timer input or SPI clock or DIO0 |
| 5 | P1.1 | Digital I/O | Port 1, bit 1 or SPI dataout or DIO1 |
| 6 | P0.0 | Digital I/O | Port 0, bit 0 or EEPROM.CSN or DIO2 |
| 7 | P0.1/RXD | Digital I/O | Port 0, bit 1 or UART.RXD or DIO3 |
| 8 | P0.2/TXD | Digital I/O | Port 0, bit 2 or UART.TXD or DIO4 |
| 9 | P0.3/INT0_N | Digital I/O | Port 0, bit 3 or INT0_N interrupt or DIO5 |

| | | | |
|---|---|---|---|
| 10 | P0.4/INT1_N | Digital I/O | Port 0, bit 4 or INT1_N interrupt or DIO6 |
| 11 | P0.5/T0 | Digital I/O | Port 0, bit 5 or T0 timer input or DIO7 |
| 12 | P0.6/T1 | Digital I/O | Port 0, bit 6 or T1 timer input or DIO8 |
| 13 | P0.7/PWM | Digital I/O | Port 0, bit 7 or PWM output or DIO9 |
| 14 | DVDD | Regulator output | Digital voltage regulator output for de-coupling and feed to DVVD2 |
| 15 | VSS | Power | Ground (0V) |
| 16 | XC2 | Analog output | Crystal Pin 2 |
| 17 | XC1 | Analog input | Crystal Pin 1 |
| 18 | VDD_PA | Regulator output | DC supply (+1.8V) to RF Power Amplifier (ANT1,ANT2) only |
| 19 | ANT1 | RF | Antenna interface 1 |
| 20 | ANT2 | RF | Antenna interface 2 |
| 21 | VSS_PA | Power | Ground (0V) |
| 22 | VDD | Power | Power Supply (1.9-3.6 V DC) |
| 23 | VSS | Power | Ground (0V) |
| 24 | AIN7 | Analog input | ADC input 7 |
| 25 | AIN6 | Analog input | ADC input 6 |
| 26 | AIN5 | Analog input | ADC input 5 |
| 27 | IREF | Analog input | Connection to external Bias reference resistor |
| 28 | AREF | Analog input | ADC reference voltage |
| 29 | AIN4 | Analog input | ADC input 4 |
| 30 | AIN3 | Analog input | ADC input 3 |
| 31 | VSS | Power | Ground (0V) |
| 32 | VDD | Power | Power Supply (1.9-3.6 V DC) |
| 33 | VSS | Power | Ground (0V) |
| 34 | AIN2 | Analog input | ADC input 2 |
| 35 | AIN1 | Analog input | ADC input 1 |
| 36 | P1.2 | Digital input | Port 1, bit 2 or SPI datain or DIN0 |

Table 1-3 : nRF24E1 pin function

## 1.4 Glossary of Terms

| Term | Description |
|---|---|
| ADC | Analog to Digital Converter |
| CLK | Clock |
| CRC | Cyclic Redundancy Check |
| CS | Chip Select |
| CE | Chip Enable |
| DR | Data Ready |
| FS | Full Scale |
| GFSK | Gaussian Frequency Shift Keying |
| GPIO | General Purpose In Out |
| ISM | Industrial-Scientific-Medical |
| kSPS | kilo Samples per Second |
| MCU | Microcontroller Unit |
| OD | Overdrive |
| P0 (or P1) | (8051) In / Out Port 0 (or Port 1) |
| PWM | Pulse Width Modulation |
| PWR_DWN | Power Down |
| PWR_UP | Power Up |
| RTC | Real Time Clock |
| RX | Receive |
| SFR | (8051) Special Function Register |
| SPI | Serial Peripheral Interface |
| SPS | Samples per Second |
| ST_BY | Standby |
| TX | Transmit |
| XTAL | Crystal (oscillator) |

# 2 ARCHITECTURAL OVERVIEW

This section will give a brief overview of each of the blocks in the block diagram in Figure 1-1.

## 2.1 Microcontroller

The nRF24E1 microcontroller is instruction set compatible with the industry standard 8051. Instruction timing is slightly different from the industry standard, typically each instruction will use from 4 to 20 clock cycles, compared with 12 to 48 for the "standard". The interrupt controller is extended to support 5 additional interrupt sources; ADC, SPI, RF receiver 1, RF receiver 2 and wakeup timer. There are also 3 timers which are 8052 compatible, plus some extensions, in the microcontroller core. An 8051 compatible UART that can use timer1 or timer2 for baud rate generation in the traditional asynchronous modes is included. The CPU is equipped with 2 data pointers to facilitate easier moving of data in the XRAM area, which is a common 8051 extension. The microcontroller clock is derived directly from the crystal oscillator.

### *2.1.1 Memory configuration*

The microcontroller has a 256 byte data ram (8052 compatible, with the upper half only addressable by register indirect addressing). A small ROM of 512 bytes, contains a bootstrap loader that is executed automatically after power on reset or if initiated by software later. The user program is normally loaded into a 4k byte RAM[1] from an external serial EEPROM by the bootstrap loader. The 4k byte RAM may also (partially) be used for data storage in some applications.

### *2.1.2 Boot EEPROM/FLASH*

If the mask ROM option is not used, the program code for the device must be loaded from an external non-volatile memory. The default boot loader expects this to be a "generic 25320" EEPROM with SPI interface. These memories are available from several vendors with supply ranges down to 1.8V. The SPI interface uses the pins P1.2/DIN0 (EEPROM SDO), P1.0/DIO0 (EEPROM SCK), P1.1/DIO1 (EEPROM SDI) and P0.0/DIO2 (EEPROM CSN). When the boot is completed, the P1.2/DIN0, P1.0/DIO0 and P1.1/DIO1 pins may be used for other purposes such as other SPI devices or GPIO.

### *2.1.3 Register map*

The SFR (Special Function Registers) control several of the features of the nRF24E1. Most of the nRF24E1 SFRs are identical to the standard 8051 SFRs. However, there are additional SFRs that control features that are not available in the standard 8051. The SFR map is shown in the table below. The registers with grey background are registers with industry standard 8051 behavior. Note that the function of P0 and P1 are somewhat different from the "standard" even if the conventional addresses (0x80 and 0x90) are used

[1] Optionally this 4k block of memory can be configured as 2k mask ROM and 2k RAM or 4 k mask ROM

| | X000 | X001 | X010 | X011 | X100 | X101 | X110 | X111 |
|---|---|---|---|---|---|---|---|---|
| **F8** | EIP | | | | | | | |
| **F0** | B | | | | | | | |
| **E8** | EIE | | | | | | | |
| **E0** | ACC | | | | | | | |
| **D8** | EICON | | | | | | | |
| **D0** | PSW | | | | | | | |
| **C8** | T2CON | | RCAP2L | RCAP2H | TL2 | TH2 | | |
| **C0** | | | | | | | | |
| **B8** | IP | | | | T1_1V2 | T2_1V2 | DEV_ OFFSET | |
| **B0** | | RSTREAS | SPI _DATA | SPI _CTRL | SPI CLK | TICK_ DV | CK_ CTRL | TEST_ MODE |
| **A8** | IE | PWM CON | PWM DUTY | REGX _MSB | REGX _LSB | REGX _CTRL | | |
| **A0** | RADIO (P2) | ADCCON | ADC DATAH | ADC DATAL | ADC STATIC | | | |
| **98** | SCON | SBUF | | | | | | |
| **90** | P1 | EXIF | MPAGE | | P0_DIR | P0_ALT | P1_DIR | P1_ALT |
| **88** | TCON | TMOD | TL0 | TL1 | TH0 | TH1 | CKCON | SPC_FNC |
| **80** | P0 | SP | DPL | DPH | DPL1 | DPH1 | DPS | PCON |

Table 2-1 : SFR Register map

## 2.2 PWM

The nRF24E1 has one programmable PWM output, which is the alternate function of PO.7 at pin DIO9.
The resolution of the PWM is software programmable to 6, 7 or 8 bits.
The frequency of the PWM signal is programmable via a 6 bit prescaler from the XTAL oscillator.
The duty cycle is programmable between 0% and 100% via one 8-bit register.

## 2.3 SPI

nRF24E1 features a simple single buffered SPI master. The 3 lines of the SPI bus (SDI, SCK and SDO) are multiplexed (by writing to register SPI_CTRL) between the GPIO pins (P1.2/DIN0, P1.0/DIO0 and P1.1/DIO1) and the RF transceiver. The SPI hardware does not generate any chip select signal. The programmer will typically use GPIO bits (from port P0) to act as chip selects for one or more external SPI devices. When the SPI interfaces the RF transceiver, the chip selects are available in an internal GPIO port, P2.

## 2.4 Port Logic

The device has 1 general purpose input and 10 general purpose bi-directional pins. These are by default configured as GPIO pins controlled by the ports P0 (DIO2 to DIO9) and P1 (DIO0, DIO1, DIN0) of the microcontroller.

Most of the GPIO pins can be used for multiple purposes under program control. The alternate functions include two external interrupts, UART RXD and TXD, a SPI master port, three enable/count signals for the timers and the PWM output.

## 2.5 Power Management

The nRF24E1 can be set into a low power down mode under program control, and also the ADC and RF subsystems can be turned on or off under program control. The CPU will stop, but all RAM's and registers maintain their values. The low power RC oscillator is running, and so are the watchdog and the RTC wakeup timer (if enabled by software). The current consumption in this mode is typically 2µA.
The device can exit the power down mode by an external pin (INT0_N or INT1_N) if enabled, by the wakeup timer if enabled or by a watchdog reset.

## 2.6 RTC Wakeup Timer, Watchdog and RC Oscillator

The nRF24E1 contains a low power RC oscillator which can not be disabled, so it will run continuously as long as VDD ≥ 1.8V.

RTC Wakeup Timer and Watchdog are two 16 bit programmable timers that run on the RC oscillator LP_OSC clock. The resolution of the watchdog and wakeup timer is programmable from approximately 300µs to approximately 80ms. By default the resolution is 10ms. The wakeup timer can be started and stopped by user software. The watchdog is disabled after a reset, but if activated it can not be disabled again, except by another reset

## 2.7 XTAL Oscillator

Both the microcontroller, ADC and RF front end run on a crystal oscillator generated clock. A range of crystals frequencies from 4 to 20 MHz may be utilised, but 16 MHz is recommended since it gives best over all performance. For details, please see Crystal Specification on page 96. The oscillator may be started and stopped as requested by software.

## 2.8 AD Converter

The nRF24E1 AD converter has 10 bit dynamic range and linearity with a conversion time of 48 CPU instruction cycles per 10-bit result.
The reference for the AD converter is software selectable between the AREF input and an internal 1.22V bandgap reference.

The converter has 9 inputs selectable by software. Selecting one of the inputs 0 to 7 will convert the voltage on the respective AIN0 to AIN7 pin.
Input 8 enables software to monitor the nRF24E1 supply voltage by converting an internal input that is VDD/3 with the 1.22V internal reference selected.
The AD converter is typically used in a start/stop mode. The sampling time is then under software control.
The converter is by default configured as 10 bits. For special requirements, the AD converter can be configured by software to perform 6, 8 or 12 bit conversions. The converter may also be used in differential mode with AIN0 used as inverting input and one of the other 7 external inputs used as noninverting input. In that case the conversion time can be reduced to approximately 2 µs.

## 2.9 Radio Transceiver

The transceiver part of the circuit has identical functionality to the nRF2401 single chip RF transceiver. It is accessed through an internal parallel port and / or an internal SPI. The data ready signals for each DuoCeiver™ receiver output can be programmed as interrupts to the microcontroller or polled via a GPIO port.
nRF2401 is a radio transceiver for the world wide 2.4 - 2.5 GHz ISM band. The transceiver consists of a fully integrated frequency synthesizer, a power amplifier, a modulator and two receiver units. Output power and frequency channels and other RF parameters are easily programmable by use of the RADIO register, SFR 0xA0. RF current consumption is only 10.5 mA in TX mode (output power -5dBm) and 18 mA in RX mode. For power saving the transceiver can be turned on / off under software control. Further information about the nRF2401 chip can be found at our website http://www.nordicsemi.no.

# 3 I/O PORTS

The nRF24E1 have two IO ports located at the default locations for P0 and P1 in standard 8051, but the ports are fully bi-directional CMOS and the direction of each pin is controlled by a _DIR and an _ALT bit for each bit as shown in the table below.

| Pin | Default function | Alternate=1 | SPI_CTRL=01 |
|---|---|---|---|
| DIN0 | P1.2 | | SPI_DI |
| DIO0 | P1.0 | T2 (timer2 input) | SPI_SCK |
| DIO1 | P1.1 | | SPI_DO |
| DIO2 | P0.0[2] | EEPROM_CSN | |
| DIO3 | P0.1 | RXD (UART) | |
| DIO4 | P0.2 | TXD (UART) | |
| DIO5 | P0.3 | INT0_N (interrupt) | |
| DIO6 | P0.4 | INT1_N (interrupt) | |
| DIO7 | P0.5 | T0 (timer0 input) | |
| DIO8 | P0.6 | T1 (timer1 input) | |
| DIO9 | P0.7 | PWM | |

Table 3-1 : Port functions

## 3.1 I/O port behavior during RESET

During the period the internal reset is active (regardless of whether or not the clock is running), all the port pins are configured as inputs. When program execution starts, the DIO ports are still configured as inputs and the program will need to set the _ALT and/or the _DIR register for the pins that should be used as outputs.

## 3.2 Port 0 (P0)

P0_ALT and P0_DIR control the P0 port function in that order of priority. If the alternate function for port p0.n is set (by P0_ALT.n = 1) the pin will be input or output as required by the alternate function (UART, external interrupt, timer inputs or PWM output), except that the UART RXD direction will still depend on P0_DIR.1.

To use INT0_N or INT1_N, the corresponding alternate function must be activated, P0_ALT.3 / P0_ALT.4

When the P0_ALT.n is not set, bit ‘n’ of the port is a GPIO function with the direction controlled by P0_DIR.n.

P0.0 is always a GPIO. It will be activated by the default boot loader after reset and should be connected to the CSN of the boot flash.

[2] Reserved for use as EEPROM_CSN, works as GPIO P0.0 independent of the “Alternate setting”

| Pin | Data in P0_ALT.n,P0_DIR.n | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 10 | | 11 | | 00 | | 01 | |
| P0.0 (DIO2) | P0.0 | Out | P0.0 | In | P0.0 | Out | P0.0 | In |
| P0.1 (DIO3) | RXD | Out | RXD | In | P0.1 | Out | P0.1 | In |
| P0.2 (DIO4) | TXD | Out | TXD | Out | P0.2 | Out | P0.2 | In |
| P0.3 (DIO5) | INT0_N | In | INT0_N | In | P0.3 | Out | P0.3 | In |
| P0.4 (DIO6) | INT1_N | In | INT1_N | In | P0.4 | Out | P0.4 | In |
| P0.5 (DIO7) | T0 | In | T0 | In | P0.5 | Out | P0.5 | In |
| P0.6 (DIO8) | T1 | In | T1 | In | P0.6 | Out | P0.6 | In |
| P0.7 (DIO9) | PWM | Out | PWM | Out | P0.7 | Out | P0.7 | In |

Table 3-2 : Port 0 (P0) functions

Port 0 is controlled by SFR-registers 0x80, 0x94 and 0x95 listed in the table below.

| Addr SFR (hex) | R/W | #bit | Init value (hex) | Name | Function |
|---|---|---|---|---|---|
| 80 | R/W | 8 | FF | P0 | Port 0, pins DIO9 to DIO2 |
| 94 | R/W | 8 | FF | P0_DIR | Direction for each bit of Port 0<br>0: Output, 1: Input<br>Direction is overridden if alternate function is selected for a pin. |
| 95 | R/W | 8 | 00 | P0_ALT | Select alternate functions for each pin of P0, if corresponding bit in P0_ALT is set, as listed in Table 3-2 : Port 0 (P0) functions, P0.0 has no alternate function,as it is intended as CS for external boot flash memory. It will function as a GPIO bit regardless of P0_ALT.0 |

Table 3-3 : Port 0 control and data SFR-registers

## 3.3 Port 1 (P1 or SPI port)

The P1 port consists of only 3 pins, one of which is an hardwired input. The function is controlled by SPI_CTRL.

When SPI_CTRL is 01, the port is used as a SPI master port. The GPIO bits in port P0 may be used as chip select(s). For timing diagram, please see Figure 3-1 : SPI interface timing.

When not used as SPI port, P0_ALT.0 will force P1.0 to be the timer T2 input, P1.1 is now a GPIO. When P0_ALT.0 is 0, also P1.0 is a GPIO.

P1.2 (DIN0) is always an input.

| **Pin** | **SPI_CTRL = 01** | | **SPI_CTRL != 01** | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | **P1_ALT.n = 1** | | **P1_ALT.n = 0** | | | |
| | | | | | **P1_DIR.n = 0** | | **P1_DIR.n = 1** | |
| P1.0 (DIO0) | SCK | Out | T2 | In | P1.0 | In | P1.0 | Out |
| P1.1 (DIO1) | SDO | Out | P1.1 | In[3] | P1.1 | In | P1.1 | Out |
| P1.2 (DIN0) | SDI | In | P1.2 | In | P1.2 | In | P1.2 | In |

Table 3-4 : Port 1 (P1) functions

Port 1 is controlled by SFR-registers 0x90, 0x96 and 0x97, and only the 3 lower bits of the registers are used.

| **Addr SFR (hex)** | **R/W** | **#bit** | **Init value (hex)** | **Name** | **Function** |
|---|---|---|---|---|---|
| 90 | R/W | 3 | FF | P1 | Port 1, pins DIN0, DIO1 and DIO0 |
| 96 | R/W | 3 | FF | P1_DIR | Direction for each bit of Port 1<br>0: Output, 1: Input<br>Direction is overridden if alternate function is selected for a pin, or if SPI_CTRL=01.<br>bit0, DIN0 is always input. |
| 97 | R/W | 3 | 00 | P1_ALT | Select alternate functions for each pin of P1 if corresponding bit in P1_ALT is set, as listed in Table 3-4 : Port 1 (P1) functions<br>If SPI_CTRL is '01', the P1 port is used as SPI master data and clock :<br>2 -> SDI – input to nRF24E1 from slave<br>1 -> SDO – output from nRF24E1 to slave<br>0 -> SCK – output from nRF24E1 to slave |

Table 3-5 : Port 1 control and data SFR-registers

[3] P1.1 is actually under control of P1_DIR.1 even when P1_ALT.1 is 1, since there is no alternate function for this pin.

P1 may also be configured as a SPI master port , and is then controlled by the 3 SFR registers 0xB2, 0xB3, 0xB4 as shown in the table below.

| Addr SFR (hex) | R/ W | #bit | Init (hex) | Name | Function |
|---|---|---|---|---|---|
| B2 | R/W | 8 | 0 | SPI_DATA | SPI data input/output |
| B3 | R/W | 2 | 0 | SPI_CTRL | 00 -> SPI not used no clock generated<br>01 -> SPI connected to port P1 (as for booting) another GPIO must be used as chip select (see also Table 3-4 : Port 1 (P1) functions)<br>10 -> SPI connected to RADIO transmitter/receiver 1 for TX or RX or for transceiver configuration<br>11 -> SPI connected to RADIO receiver 2 for RX Chip select is a bit of RADIO register (see Table 4-2 : RADIO register ) |
| B4 | R/W | 2 | 0 | SPICLK | Divider factor from CPU clock to SPI clock<br>00: 1/8 of CPU clock frequency<br>01: 1/16 of CPU clock frequency<br>10: 1/32 of CPU clock frequency<br>11: 1/64 of CPU clock frequency<br>The CPU clock is the oscillator generated clock described in Crystal Specification page 96 |

Table 3-6 : SPI control and data SFR-registers

### *3.3.1 SPI interface operation*

Whenever SPI_DATA register is written to, a sequence of 8 pulses is started on SCK, and the 8 bits of SPI_DATA register are clocked out on SDO with msb first. Simultaneously 8 bits from SDI are clocked into SPI_DATA register. Ouput data is shifted on negedge SCK, and input data is read on posedge SCK. This is illustrated in Figure 3-1 : SPI interface timing. When the 8 bits are done, SPI_READY interrupt (EXIF.5) goes active, and the 8 bits from SDI may be read from SPI_DATA register. The EXIF.5 bit must be cleared before starting another SPI transaction by writing to SPI_DATA register again.
SCK, SDO and SDI may be external pins or internal signals, as defined in SPI_CTRL register.

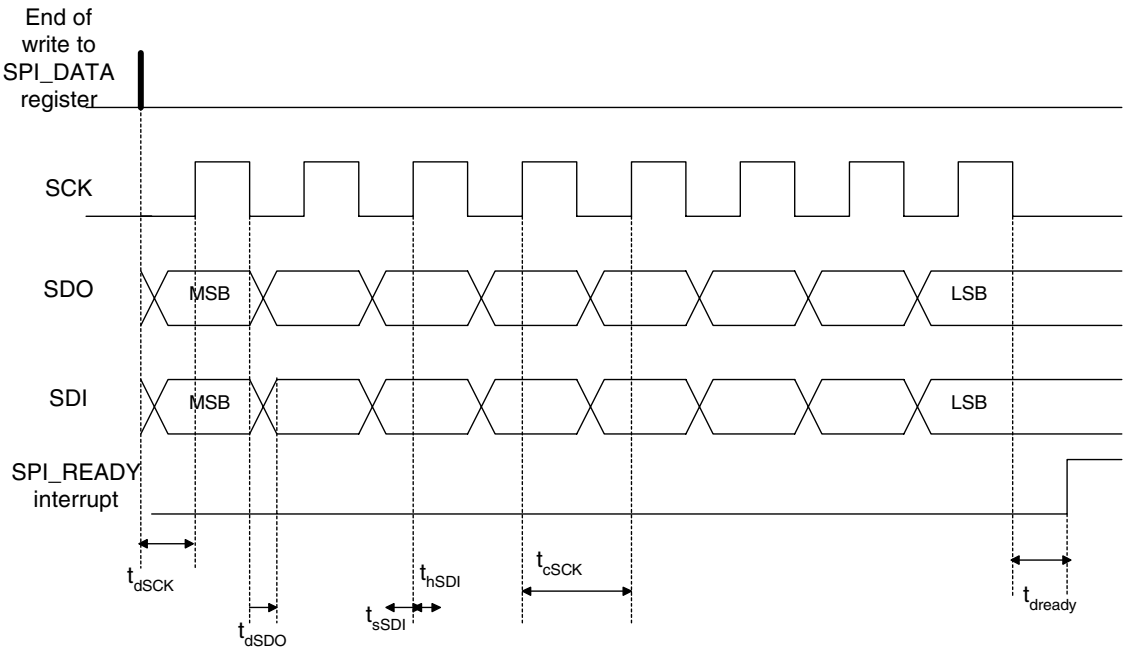



Figure 3-1 : SPI interface timing

$t_{cSCK}$ : SCK cycle time, as defined by SPICLK register.

$t_{dSCK}$ : time from writing to SPI_DATA register to first SCK pulse,

$t_{dSCK} = t_{cSCK} / 2$

$t_{dSDO}$ : delay from negedge SCK to new SDO output data, may vary from -40ns to 40ns

$t_{sSDI}$ : SDI setup time to posedge SCK, $t_{sSDI} > 45$ns.

$t_{hSDI}$ : SDI hold time to posedge SCK, $t_{hSDI} > 0$ns.

$t_{dready}$ : time from last SCK pulse to SPI_READY interrupt goes active

$t_{dready} = 7$ CPU clock cycles

Note that the above delay, setup and hold time numbers only apply for SPI connected to Port 1; as when SPI is connected to the Radio, SCK,SDO,SDI are all internal signals, not visible to the user.

Minimum time between two consecutive SPI transactions will be :

$8.5\ t_{cSCK} + t_{dready} + t_{SW}$

where $t_{SW}$ is the time taken by the software to process SPI_READY interrupt, and write to SPI_DATA register.

# 4 nRF2401 2.4GHz TRANSCEIVER SUBSYSTEM

## 4.1 RADIO port (Port 2)

The transceiver is controlled by the RADIO port. The RADIO port uses the address normally used by port P2 in standard 8051. However since the radio transceiver is on chip, the port is not bi-directional. The power on default values in the port "latch" also differs from traditional 8051 to match the requirements of the radio transceiver subsystem.
Operation of the transceiver is controlled by SFR registers RADIO and SPI_CTRL:

| Addr SFR (hex) | R/W | #bit | Init value (hex) | Name | Function |
|---|---|---|---|---|---|
| A0 | R/W | 8 | 80 | RADIO | General purpose IO for interface to nRF2401 radio transceiver subsystem |
| B3 | R/W | 2 | 0 | SPI_CTRL | 00 -> SPI not used<br>01 -> SPI connected to port P1 (boot)<br>10 -> SPI connected to nRF2401 CH1<br>11 -> SPI connected to nRF2401 RX CH2 |

Table 4-1 : nRF2401 2.4GHz transceiver subsystem control registers - SFR 0xA0 and 0xB3

The bits of the RADIO register correspond to similar pins of the nRF2401 single chip, as shown in Table 4-2 : RADIO register . In the documentation the pin names are used, so please note that setting or reading any of these nRF2401 pins, means to write or read the RADIO SFR register accordingly. Please also note that in the transceiver documentation the notation MCU means the onchip 8051 compatible microcontroller.

| RADIO register bit | corresponding pin name on single chip nRF2401 2.4GHz Transceiver |
|---|---|
| **Read :** | |
| 7: 0 (not used) | |
| 6: DR2, data ready from receiver 2 (available also as interrupt) | DR2 |
| 5: CLK2, clock for receiver 2 data out | CLK2 |
| 4: DOUT2, data out from receiver 2 | DOUT2 |
| 3: 0 (not used) | |
| 2: DR1, data ready from receiver 1 (available also as interrupt) | DR1 |
| 1: CLK1, clock for receiver 1 data out | CLK1 |
| 0: DATA, data out from receiver 1 | DATA |

| **Write :** | |
|---|---|
| 7: PWR_UP, power on radio | PWR_UP |
| 6: CE, Activate RX or TX mode | CE |
| 5: CLK2, clock for receiver 2 data out | CLK2 |
| 4: Not used | |
| 3: CS, Chip select configuration mode | CS |
| 2: Not used | |
| 1: CLK1, clock for data input or receiver 1 data out | CLK1 |
| 0: DATA, configuration or TX data input | DATA |

Table 4-2 : RADIO register - SFR 0xA0, default initial data value is 0x80.
Note : Some of the pins are overridden when SPI_CTRL=1x, see Table 4-3 : Transceiver SPI interface.

### *4.1.1 Controlling the transceiver via SPI interface.*

It is more convenient to use the built-in SPI interface to do the most common transceiver operations as RF configuration and ShockBurst™ RX or TX. Please see Table 3-6 : SPI control and data SFR-registers for use of SPI interface. The radio port will be connected in different ways to the SPI hardware when SPI_CTRL is '1x'. When SPI_CTRL is '0x', all radio pins are connected directly to their respective port pins.

| SPI signal | SPI_CTRL=10 (binary) | SPI_CTRL=11 |
|---|---|---|
| CS<br>(active high) | RADIO_wr.6 (CE) for ShockBurst™<br>RADIO_wr.3 (CS) for Configuration | RADIO_wr.6 (CE) |
| SCK | nRF2401/CLK1 | nRF2401/CLK2 |
| SDI | nRF2401/DATA | nRF2401/DOUT2 |
| SDO | nRF2401/DATA | not used |
| ShockBurst™ data ready | RADIO_rd.2 (DR1) | RADIO_rd.6 (DR2) |

Table 4-3 : Transceiver SPI interface.

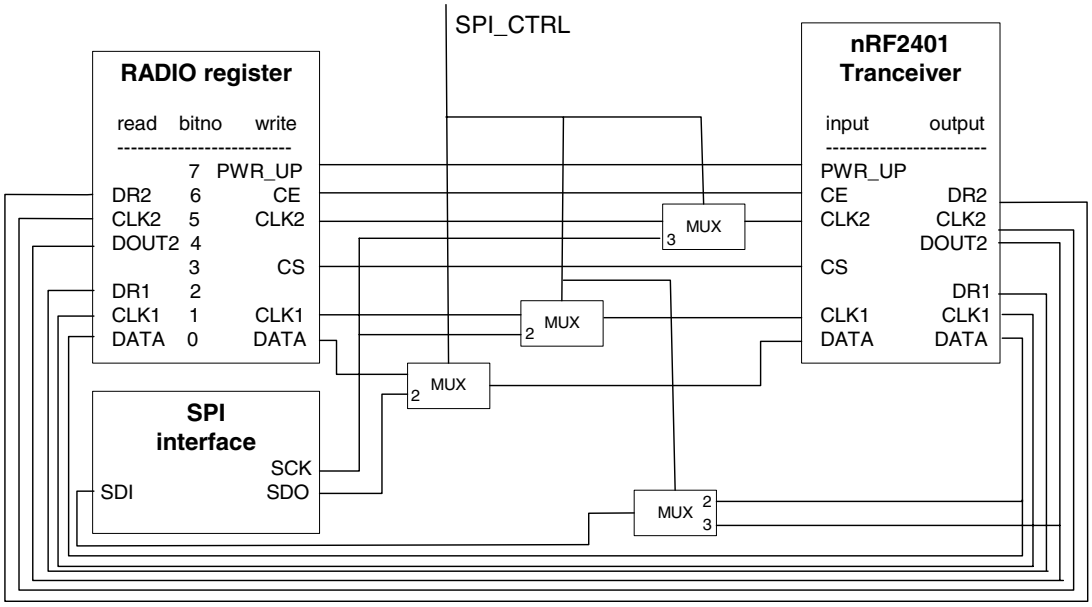



Figure 4-1 : Transceiver interface

### *4.1.2 RADIO port behavior during RESET*

During the period the internal reset is active (regardless of whether or not the clock is running), the RADIO outputs that control the nRF2401 transceiver subsystem are forced to their respective default values (RADIO.3=0 (CS), RADIO.6=0 (CE) RADIO.7=1 (PWR_UP)). When program execution starts, these ports will remain at those default levels until the programmer actively changes them by writing to the RADIO register.

## 4.2 Modes of operation

### *4.2.1 Overview*

The nRF2401 subsystem can be set in the following main modes depending on three control pins:

| **Mode** | PWR_UP | CE | CS |
|---|---|---|---|
| Active (RX/TX) | 1 | 1 | 0 |
| Configuration | 1 | 0 | 1 |
| Stand by | 1 | 0 | 0 |
| Power down | 0 | X | X |

Table 4-4 nRF2401 subsystem main modes

### *4.2.2 Active modes*

The nRF2401 subsystem has two active (RX/TX) modes:

- ShockBurst™
- Direct Mode (not supported by nRF24E1)

The device functionality in these modes is decided by the content of a configuration word. This configuration word is presented in the configuration section. Please note that Direct mode is not supported, as this will require a more powerful CPU than 8051.

### *4.2.3 ShockBurst™*

The ShockBurst™ technology uses on-chip FIFO to clock in data at a low data rate and transmit at a very high rate thus enabling extremely power reduction.
When operating the nRF2401 subsystem in ShockBurst™, you gain access to the high data rates (1 Mbps) offered by the 2.4 GHz band without the need of a costly, high-speed microcontroller (MCU) for data processing.
By putting all high speed signal processing related to RF protocol on-chip, the nRF24E1 offers the following benefits:

- Highly reduced current consumption
- Lower system cost (facilitates use of less expensive microcontroller)
- Greatly reduced risk of 'on-air' collisions due to short transmission time

The nRF2401 subsystem can be programmed using a simple 3-wire interface where the data rate is decided by the speed of the CPU.
By allowing the digital part of the application to run at low speed while maximizing the data rate on the RF link, the ShockBurst™ mode reduces the average current consumption in applications considerably.

#### 4.2.3.1 ShockBurst™ principle

When the nRF2401 subsystem is configured in ShockBurst™, TX operation is conducted in the following way (10 kbps for the example only).

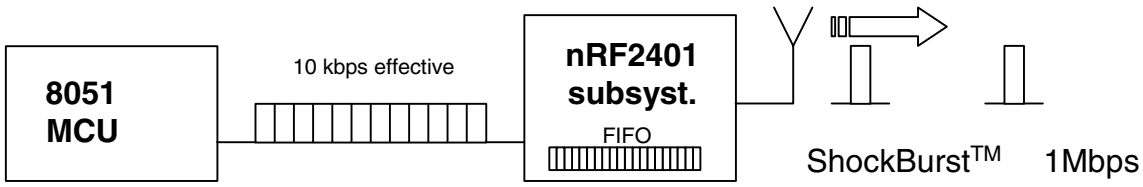



Figure 4-2Clocking in data with CPU and sending with ShockBurst™ technology

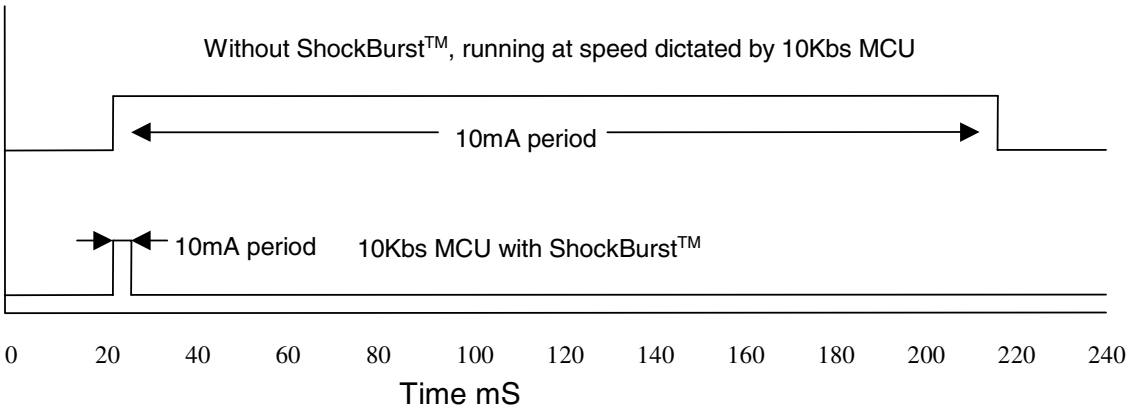



Figure 4-3 RF Current consumption with & without ShockBurst™ technology

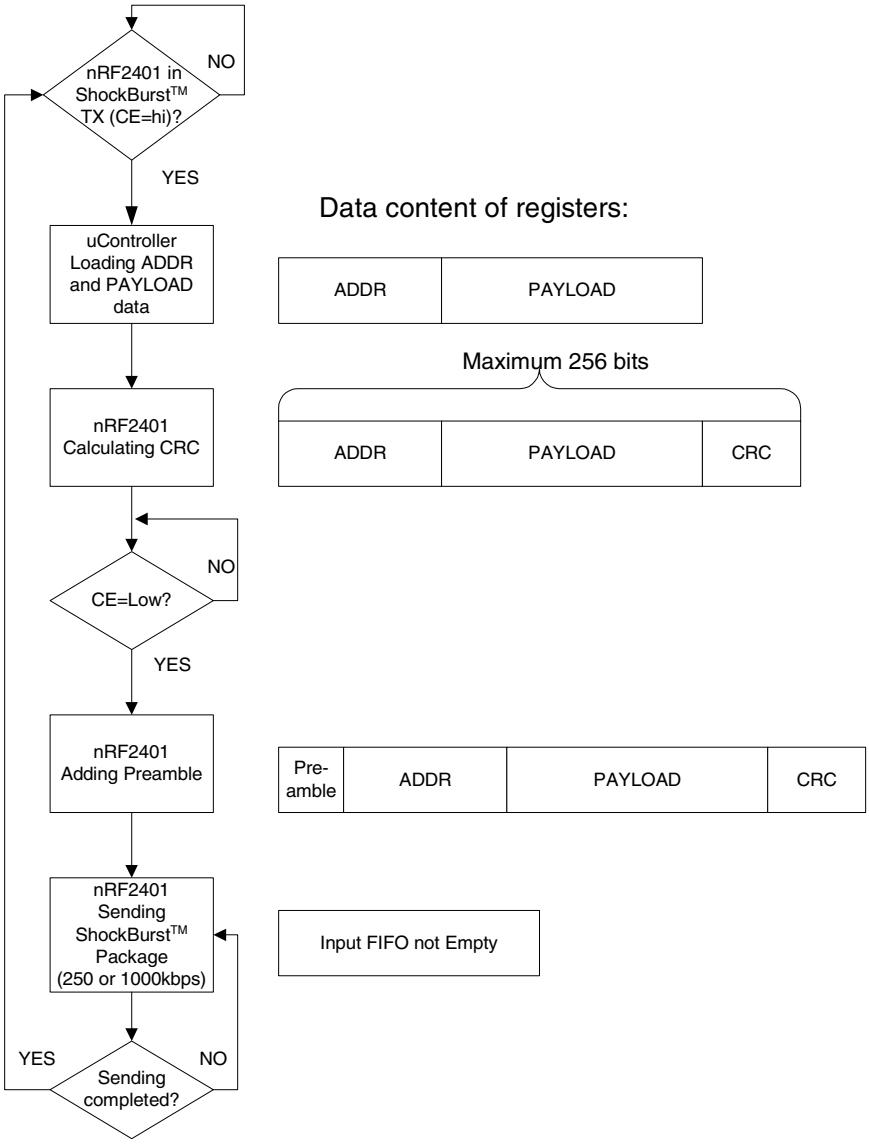



Figure 4-4 Flow Chart ShockBurst™ Transmit of nRF2401 subsystem

4.2.3.2 ShockBurst™ Transmit:

4.2.3.2.1 CPU interface pins: CE, CLK1, DATA

1. When the application CPU has data to send, set CE high. This activates nRF2401 on-board data processing.
2. The address of the receiving node (RX address) and payload data is clocked into the nRF2401 subsystem. The application protocol or CPU sets the speed <1Mbps (ex: 10kbps).
3. CPU sets CE low, this activates a ShockBurst™ transmission.
4. ShockBurst™:
   - RF front end is powered up
   - RF package is completed (preamble added, CRC calculated)
   - Data is transmitted at high speed (250 kbps or 1 Mbps configured by user).

- nRF2401 subsystem returns to stand by when finished

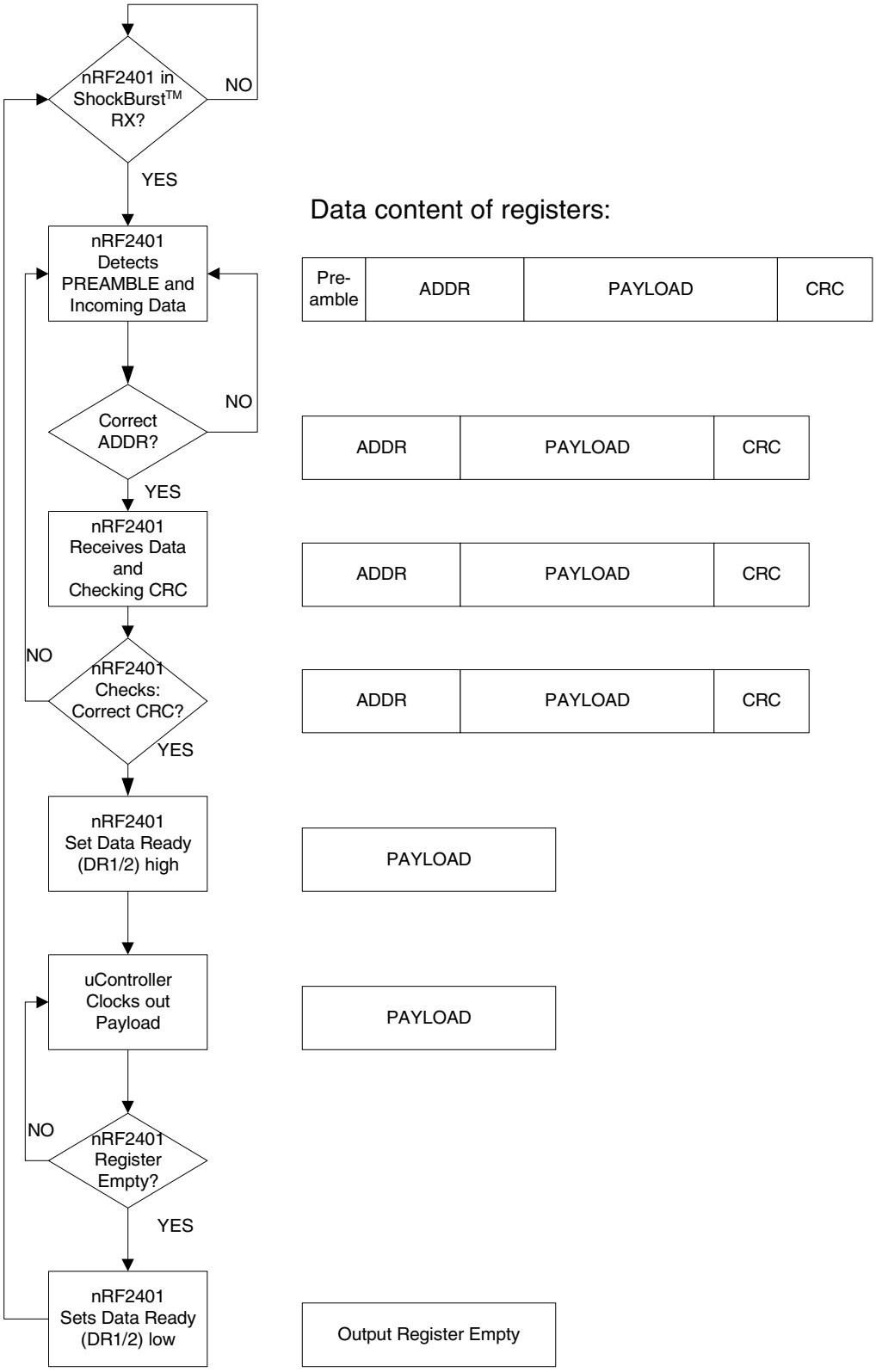



Figure 4-5 Flow Chart ShockBurst™ Receive of nRF2401 subsystem

4.2.3.3 ShockBurst™ Receive:

4.2.3.3.1 CPU interface pins: CE, DR1, CLK1 and DATA (one RX channel receive)

1. Correct address and size of payload of incoming RF packages are set when nRF2401 subsystem is configured to ShockBurst™ RX.
2. To activate RX, set CE high.
3. After 200 µs settling, nRF2401 subsystem is monitoring the air for incoming communication.
4. When a valid package has been received (correct address and CRC found), nRF2401 subsystem removes the preamble, address and CRC bits.
5. nRF2401 subsystem then notifies (interrupts) the CPU by setting the DR1 pin high.
6. CPU may set the CE low to disable the RF front end (low current mode).
7. The CPU will clock out just the payload data at a suitable rate (ex. 10 kbps).
8. When all payload data is retrieved nRF2401 subsystem sets DR1 low again, and is ready for new incoming data package if CE is kept high during data download. If the CE was set low, a new start up sequence can begin, see Figure 4-14.

### *4.2.4 DuoCeiver™ Simultaneous Two Channel Receive Mode*

In ShockBurst™ mode the nRF24E1 can facilitate simultaneous reception of two parallel independent frequency channels at the maximum data rate.
This means:

- nRF24E1 can receive data from two 1 Mbps transmitters (ex: nRF24E1, nRF2401 or nRF2402) 8 MHz (8 frequency channels) apart through one antenna interface.
- The output from the two data channels is fed to two separate sets of interface pins.
  - Data channel 1: CLK1, DATA, and DR1
  - Data channel 2: CLK2, DOUT2, and DR2

The DuoCeiver™ technology provides 2 separate dedicated data channels for RX and replaces the need for two, stand alone receiver systems.

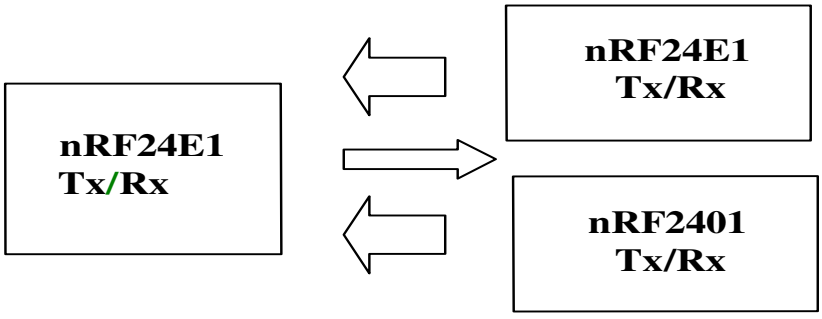



Figure 4-6 Simultaneous 2 channel receive on nRF24E1

There is one absolute requirement for using the second data channel. For the nRF24E1 to be able to receive at the second data channel the frequency channel must be 8MHz

higher than the frequency of data channel 1. The nRF2401 subsystem must be programmed to receive at the frequency of data channel 1. No time multiplexing is used in nRF2401 subsystem to fulfil this function. In direct mode the CPU must be able to handle two simultaneously incoming data packets if it is not multiplexing between the two data channels. In ShockBurst™ it is possible for the CPU to clock out one data channel at a time while data on the other data channel waits for CPU availability, without any lost data packets, and by doing so reduce the needed performance of the CPU.

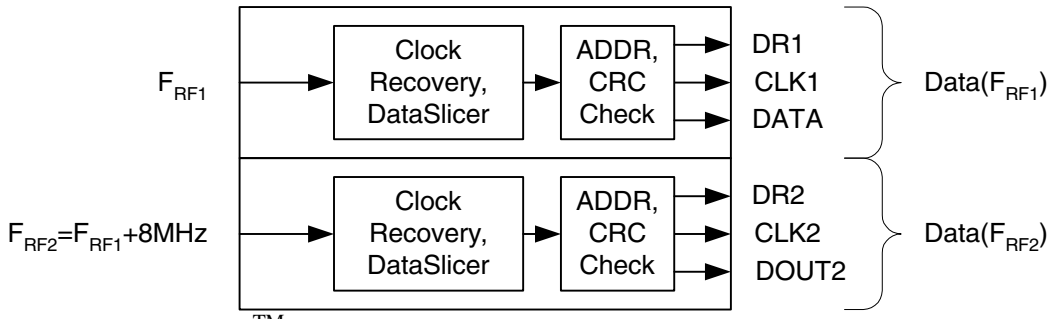



Figure 4-7 DuoCeiver™ with two simultaneously independent receive channels.

## 4.3 Device configuration

All configuration of the nRF2401 subsystem is done via a 3-wire interface interface (CS, CLK1 and DATA) to a single configuration register. The configuration word can be up to 18 bytes long. The configuration bits (DATA) must be clocked (by CLK1) into nRF2401 subsystem, with msb first, while CS=1. No more than 18 bytes may be downloaded.

### *4.3.1 Configuration for ShockBurst™ operation*

The configuration word in ShockBurst™ enables the nRF2401 subsystem to handle the RF protocol. Once the protocol is completed and loaded into nRF2401 subsystem only one byte, bit[7:0], needs to be updated during actual operation.

The configuration blocks dedicated to ShockBurst™ is as follows:

- Payload section width: Specifies the number of payload bits in a RF package. This enables the nRF2401 subsystem to distinguish between payload data and the CRC bytes in a received package.
- Address width: Sets the number of bits used for address in the RF package. This enables the nRF2401 subsystem to distinguish between address and payload data.
- Address (RX Channel 1 and 2): Destination address for received data.
- CRC: Enables on-chip CRC generation and de-coding.

NOTE:
These configuration blocks, with the exception of the CRC, are dedicated for the packages that a nRF2401 subsystem is to receive.
In TX mode, the CPU must generate an address and a payload section that fits the configuration of the nRF2401 subsystem that is to receive the data.
When using the nRF2401 subsystem on-chip CRC feature ensure that CRC is enabled and uses the same length for both the TX and RX devices.

| PRE-AMBLE | ADDRESS | PAYLOAD | CRC |
|---|---|---|---|

Figure 4-8Data packet set-up

### *4.3.2 Configuration for Direct Mode operation*

For direct mode operation only the two first bytes (bit[15:0]) of the configuring word is relevant.

### *4.3.3 Configuration Word overview*

| | Bit position | Number of bits | Name | Function |
|---|---|---|---|---|
| **ShockBurst™ configuration** | 143:120 | 24 | TEST | Reserved for testing |
| | 119:112 | 8 | DATA2_W | Length of data payload section RX channel 2 |
| | 111:104 | 8 | DATA1_W | Length of data payload section RX channel 1 |
| | 103:64 | 40 | ADDR2 | Up to 5 byte address for RX channel 2 |
| | 63:24 | 40 | ADDR1 | Up to 5 byte address for RX channel 1 |
| | 23:18 | 6 | ADDR_W | Number of address bits (both RX channels). |
| | 17 | 1 | CRC_L | 8 or 16 bit CRC |
| | 16 | 1 | CRC_EN | Enable on-chip CRC generation/checking. |
| | | | | |
| **General device configuration** | 15 | 1 | RX2_EN | Enable two channel receive mode |
| | 14 | 1 | CM | Communication mode (Direct or ShockBurst™) |
| | 13 | 1 | RFDR_SB | RF data rate (1Mbps requires 16MHz crystal) |
| | 12:10 | 3 | XO_F | Crystal frequency |
| | 9:8 | 2 | RF_PWR | RF output power |
| | 7:1 | 7 | RF_CH# | Frequency channel |
| | 0 | 1 | RXEN | RX or TX operation |

Table 4-5 Table of configuration words.

The configuration word is shifted in MSB first on positive CLK1 edges. New configuration is enabled on the falling edge of CS.

NOTE.
On the falling edge of CS, the nRF2401 subsystem updates the number of bits actually shifted in during the last configuration.
Ex:
If the nRF2401 subsystem is to be configured for 2 channel RX in ShockBurst™, a total of 120 bits must be shifted in during the first configuration after VDD is applied.

Once the wanted protocol, modus and RF channel are set, only one bit (RXEN) is shifted in to switch between RX and TX.

### *4.3.4 Configuration Word Detailed Description*

The following describes the function of the 144 bits (bit 143 = MSB) that is used to configure the nRF2401 subsystem.
General Device Configuration: bit[15:0]
ShockBurst™ Configuration: bit[119:0]
Test Configuration: bit[143:120]

| MSB | TEST | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| D143 | D142 | D141 | D140 | D139 | D138 | D137 | D136 | |
| Reserved for testing | | | | | | | | |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | Default |

| MSB | TEST | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D135 | D134 | D133 | D132 | D131 | D130 | D129 | D128 | D127 | D126 | D125 | D124 | D123 | D122 | D121 | D120 |
| Reserved for testing | | | | | | | | | | | | | | Close PLL in TX | |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | Default |

| DATA2_W | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| D119 | D118 | D117 | D116 | D115 | D114 | D113 | D112 | |
| Data width channel 2 in # of bits excluding addr/crc | | | | | | | | |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Default |

| DATA1_W | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| D111 | D110 | D109 | D108 | D107 | D106 | D105 | D104 | |
| Data width channel 1 in # of bits excluding addr/crc | | | | | | | | |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Default |

| ADDR2 | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D103 | D102 | D101 | …. | D71 | D70 | D69 | D68 | D67 | D66 | D65 | D64 | |
| Channel 2 Address RX (up to 40bit) | | | | | | | | | | | | |
| 0 | 0 | 0 | … | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | Default |

| ADDR1 | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D63 | D62 | D61 | …. | D31 | D30 | D29 | D28 | D27 | D26 | D25 | D24 | |
| Channel 1 Address RX (up to 40bit) | | | | | | | | | | | | |
| 0 | 0 | 0 | … | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | Default |

| ADDR_W | | | | | | |
|---|---|---|---|---|---|---|
| D23 | D22 | D21 | D20 | D19 | D18 | |
| Address width in # of bits (both channels) | | | | | | |
| 0 | 0 | 1 | 0 | 0 | 0 | Default |

| CRC | | |
|---|---|---|
| D17 | D16 | |
| CRC Mode 1 = 16bit, 0 = 8bit | CRC 1 = enable; 0 = disable | |
| 0 | 1 | Default |

| RF-Programming | | | | | | | | | | | | | | | LSB | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | |
| Two Ch. | BUF | OD | XO Frequency | | | RF Power | | Channel selection | | | | | | | RXEN | |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Default |

Table 4-6 Configuration data word

The MSB bit should be loaded first into the configuration register.
Default configuration word: h8E08.1C20.2000.0000.00E7.0000.0000.E721.0F04.

#### 4.3.4.1 ShockBurst™ configuration:

The section bit[119:16] contains the segments of the configuration register dedicated to ShockBurst™ operational protocol. After VDD is turned on ShockBurst™ configuration is done once and remains set whilst VDD is present. During operation only the first byte for frequency channel and RX/TX switching need to be changed.

##### 4.3.4.1.1 PLL_CTRL

| PLL_CTRL | | |
|---|---|---|
| **D121** | **D120** | **PLL** |
| 0 | 0 | Open TX/Closed RX |
| 0 | 1 | Open TX/Open RX |
| 1 | 0 | Closed TX/Closed RX |
| 1 | 1 | Closed TX/Open RX |

Table 4-7 PLL setting.

Bit 121-120:

PLL_CTRL: Controls the setting of the PLL for test purposes. With closed PLL in TX no deviation will be present. For normal operational mode these two bits must both be low.

##### 4.3.4.1.2 DATAx_W

| DATA2_W | | | | | | | |
|---|---|---|---|---|---|---|---|
| 119 | 118 | 117 | 116 | 115 | 114 | 113 | 112 |

| DATA1_W | | | | | | | |
|---|---|---|---|---|---|---|---|
| 111 | 110 | 109 | 108 | 107 | 106 | 105 | 104 |

Table 4-8 Number of bits in payload.

Bit 119 – 112:

DATA2_W: Length of RF package payload section for receive-channel 2.

Bit 111 – 104:

DATA1_W: Length of RF package payload section for receive-channel 1.

NOTE:

The total number of bits in a ShockBurst™ RF package may not exceed 256! Maximum length of payload section is hence given by:

$$DATAx\_W(bits) = 256 - ADDR\_W - CRC$$

Where:

ADDR_W: length of RX address set in configuration word bit [23:18]
CRC: check sum, 8 or 16 bits set in configuration word bit [17]
PRE: preamble, 8 bits are automatically included

Shorter address and CRC leaves more room for payload data in each package.

4.3.4.1.3 ADDRx

| **ADDR2** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 103 | 102 | 101 | …. | 71 | 70 | 69 | 68 | 67 | 66 | 65 | 64 |

| ADDR1 | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 63 | 62 | 61 | …. | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |

Table 4-9 Address of receiver 2 and receiver 1.

Bit 103 – 64:

ADDR2: Receiver address channel 2, up to 40 bit.

Bit 63 – 24: ADDR1

ADDR1: Receiver address channel 1, up to 40 bit.

NOTE!

Bits in ADDRx exceeding the address width set in ADDR_W are redundant and can be set to logic 0.

4.3.4.1.4 ADDR_W & CRC

| **ADDR_W** | | | | | | **CRC_L** | **CRC_EN** |
|---|---|---|---|---|---|---|---|
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |

Table 4-10 Number of bits reserved for RX address + CRC setting.

Bit 23 – 18:

ADDR_W: Number of bits reserved for RX address in ShockBurst™ packages.

NOTE:

Maximum number of address bits is 40 (5 bytes). Values over 40 in ADDR_W are not valid.

Bit 17:

CRC_L: CRC length to be calculated in ShockBurst™.
Logic 0: 8 bit CRC
Logic 1: 16 bit CRC

Bit: 16:

CRC_EN: Enables on-chip CRC generation (TX) and verification (RX).
Logic 0: On-chip CRC generation/checking disabled
Logic 1: On-chip CRC generation/checking enabled

NOTE:

An 8 bit CRC will increase the number of payload bits possible in each ShockBurst™ data packet, but will also reduce the system integrity.

4.3.4.2 General RF configuration:

This section of the configuration word handles RF and device related parameters.

4.3.4.2.1 Modes

| RX2_EN | CM | RFDR_SB | XO_F | | | RF_PWR | |
|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |

Table 4-11 RF operational settings.

Bit 15:

RX2_EN:

Logic 0: One channel receive
Logic 1: Two channels receive

NOTE:

In two channels receive, the nRF24E1 receives on two, separate frequency channels simultaneously. The frequency of receive channel 1 is set in the configuration word bit[7-1], receive channel 2 is always 8 channels (8 MHz) above receive channel 1.

Bit 14:

Communication Mode:

Logic 0: nRF2401 subsystem operates in direct mode.
Logic 1: nRF2401 subsystem operates in ShockBurst™ mode

Bit 13:

RF Data Rate:

Logic 0: 250 kbps
Logic 1: 1 Mbps

NOTE:

Utilizing 250 kbps instead of 1Mbps will improve the receiver sensitivity by 10 dB. 1Mbps requires 16MHz crystal.

Bit 12-10:

XO_F: Selects the nRF24E1 crystal frequency to be used:

| XO Frequency Selection | | | |
|---|---|---|---|
| **D12** | **D11** | **D10** | **Crystal Frequency [MHz]** |
| 0 | 0 | 0 | 4 |
| 0 | 0 | 1 | 8 |
| 0 | 1 | 0 | 12 |
| 0 | 1 | 1 | 16 |
| 1 | 0 | 0 | 20 |

Table 4-12 Crystal frequency setting.

Please also see Table 14-2 Crystal specification of the nRF24E1

Bit 9-8:

RF_PWR: Sets nRF24E1 RF output power in transmit mode:

| RF Output Power | | |
|---|---|---|
| **D9** | **D8** | **P [dBm]** |
| 0 | 0 | -20 |
| 0 | 1 | -10 |
| 1 | 0 | -5 |
| 1 | 1 | 0 |

Table 4-13 RF output power setting.

4.3.4.2.2 RF channel & direction

| RF_CH# | | | | | | | RXEN |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Table 4-14 Frequency channel and RX / TX setting.

Bit 7 – 1:

RF_CH#: Sets the frequency channel the nRF24E1 operates on.

The channel frequency in ***transmit*** is given by:

$$Channel_{RF} = 2400\ MHz + RF\_CH\# \cdot 1.0\ MHz$$

RF_CH #: between 2400MHz and 2527MHz may be set.

The channel frequency in ***data channel 1*** is given by:

$$Channel_{RF} = 2400\ MHz + RF\_CH\# \cdot 1.0\ MHz$$

RF_CH #: between 2400MHz and 2524MHz may be set.

NOTE:
The channels above 83 can only be utilized in certain territories (ex: Japan)

The channel frequency in ***data channel 2*** is given by:

$$Channel_{RF} = 2400\ MHz + RF\_CH\# \cdot 1.0\ MHz + 8MHz \text{ (Receive at PIN\#4)}$$

RF_CH #: between 2408MHz and 2524MHz may be set.

Bit 0:

Set active mode:

Logic 0: transmit mode
Logic 1: receive mode

## 4.4 Data package Description

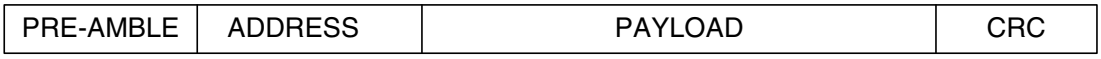



Figure 4-9 Data Package Diagram

The data packet for both ShockBurst™ mode and direct mode communication is divided into 4 sections. These are:

| | |
|---|---|
| **1. PREAMBLE** | • The preamble field is required in ShockBurst™ and Direct modes.<br>• Preamble is 8 bits in length and is dependent of the first address bit.<br>PREAMBLE 1st ADDR-BIT<br>01010101 0<br>10101010 1<br>• Preamble is automatically added to the data packet in ShockBurst™ and thereby gives extra space for payload. In Direct mode MCU must handle preamble.<br>In ShockBurst™ mode RX, the preamble is removed from the received output data, in direct mode the preamble is transparent to the output data. |
| **2 ADDRESS** | • The address field is required in ShockBurst™ mode.[4]<br>• 8 to 40 bits length.<br>• Address automatically removed from received packet in ShockBurst™ mode. In Direct mode MCU must handle address. |
| **3 PAYLOAD** | • The data to be transmitted<br>• In ShockBurst™ mode payload size is 256 bits minus the following: (Address: 8 to 40 bits. + CRC 8 or 16 bits).<br>• In Direct mode the maximum packet size (length) is for 1Mbps 4000 bits (4ms). |
| **4 CRC** | • The CRC is optional in ShockBurst™ mode, and is not used in Direct mode.<br>• 8 or 16 bits length<br>• The CRC is removed from the received output data in ShockBurst™ RX. |

Table 4-15 Data package description

[4] Suggestions for the use of addresses in ShockBurst™: In general more bits in the address gives less false detection, which in the end may give lower data packet loss.

A. The address made by (5, 4, 3, or 2) equal bytes are not recommended because it in general will make the packet-error-rate increase.
B. Addresses where the level shift only one time (i.e. 0x000FFFFFFF) could often be detected in noise that may give a false detection, which again may give raised packet-error-rate.
C. First byte of address should not start with 0x55.. or 0xAA.. as this may be interpreted as part of preamble, causing address mismatch for the rest of the address

## 4.5 Important RF Timing Data

The following timing applies for operation of nRF2401 subsystem.

### *4.5.1 nRF2401 subsystem Timing Information*

| nRF2401 subsystem timing | Max. | Min. | Name |
|---|---|---|---|
| PWR_DWN ➔ Configuration ST_BY mode | 3ms | | Tpd2sby |
| PWR_DWN➔ Active mode (RX/TX) | 3ms | | Tpd2a |
| ST_BY ➔ TX ShockBurst™ | 195µs | | Tsby2txSB |
| ST_BY ➔ RX mode | 202µs | | Tsby2rx |
| Minimum delay from CS to data. | | 5µs | Tcs2data |
| Minimum delay from CE to data. | | 5µs | Tce2data |
| Minimum delay from DR1/2 to clk. | | 50ns | Tdr2clk |
| Maximum delay from clk to data. | 50ns | | Tclk2data |
| Delay between edges | | 50ns | Td |
| Setup time | | 500ns | Ts |
| Hold time | | 500ns | Th |
| Delay to finish internal GFSK data | | 1/data rate | Tfd |
| Minimum input clock high | | 500ns | Thmin |

Table 4-16 Operational timing for nRF2401 subsystem

When the nRF2401 subsystem is in power down it must always settle in stand by for Tpd2sby (3ms) before it can enter configuration or one of the active modes.

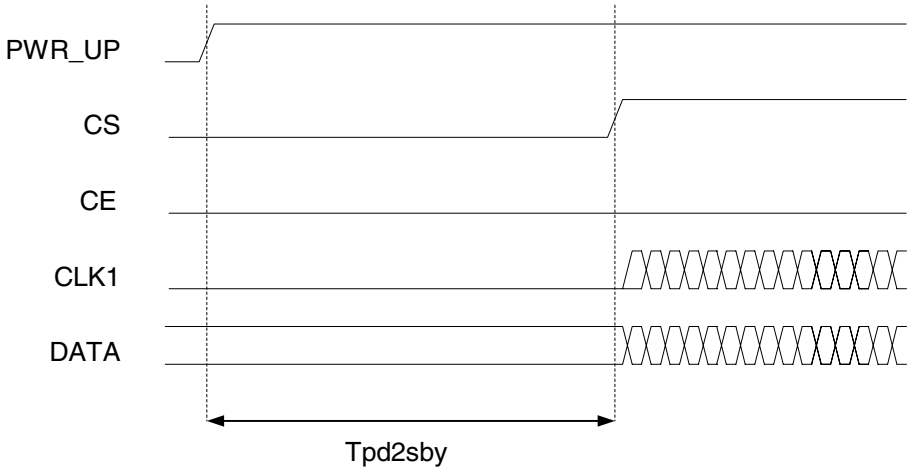



Figure 4-10 Timing diagram for power down (or VDD off) to configuration mode for nRF2401 subsystem.

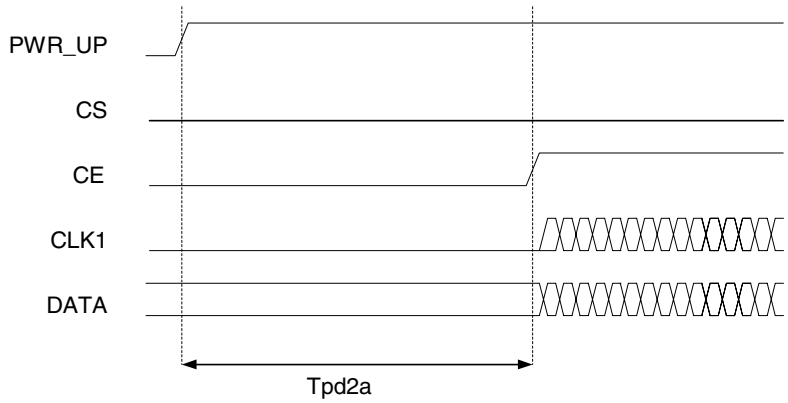



Figure 4-11 Power down (or VDD off) to active mode

Note that the configuration word will be lost when VDD is turned off and that the device then must be configured before going to one of the active modes. If the device is configured one can go directly from power down to the wanted active mode.

**Note:**

CE and CS may not be high at the same time. Setting one or the other decides whether configuration or active mode is entered.

### *4.5.2 Configuration mode timing*

When one or more of the bits in the configuration word needs to be changed the following timing apply.

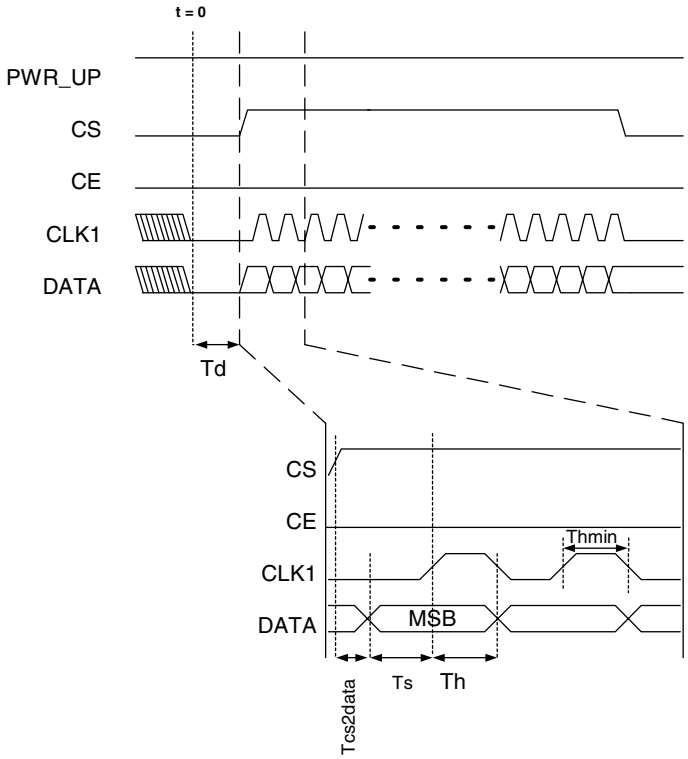



Figure 4-12 Timing diagram for configuration of nRF2401 subsystem

If configuration mode is entered from power down, CS can be set high after Tpd2sby as shown in Figure 4-10.

### *4.5.3 ShockBurst™ Mode timing*

**ShockBurst™ TX:**

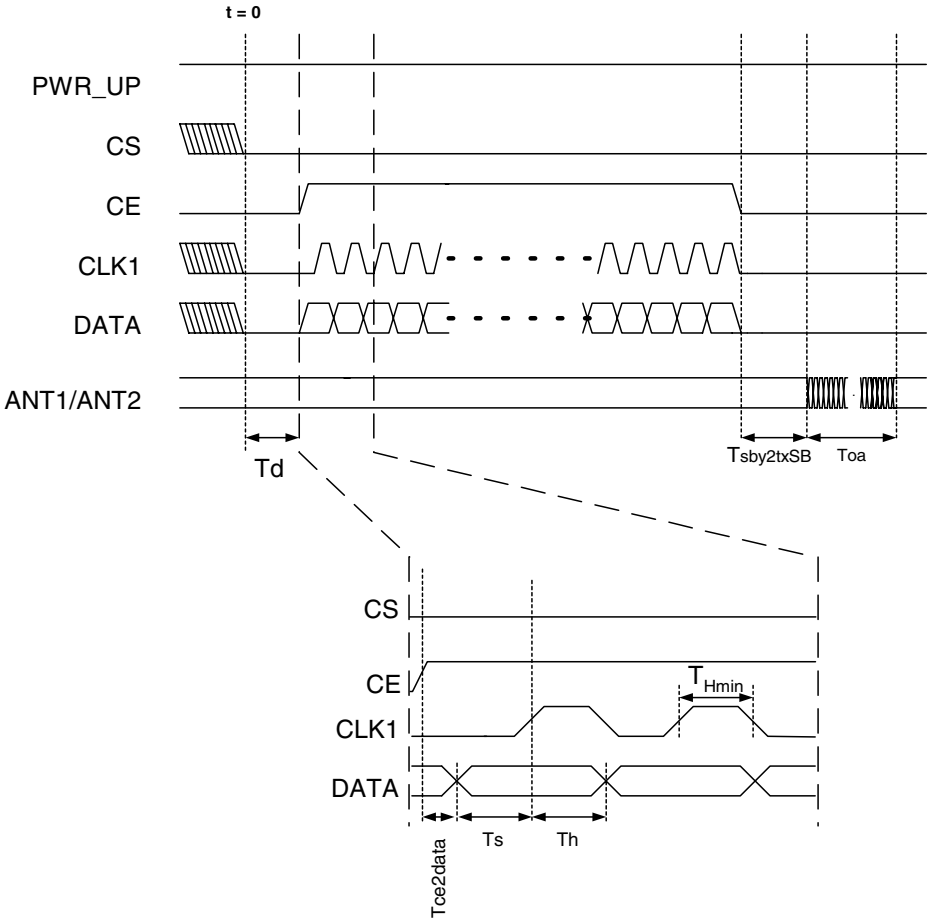



Figure 4-13 Timing of ShockBurst™ in TX

The package length and the data rate give the delay Toa (time on air), as shown in the equation. Databits are the total number of bits, including any CRC and preamble bits which may be added.

$$T_{OA} = 1/datarate \cdot (\#databits + 1)$$

## ShockBurst™ RX:

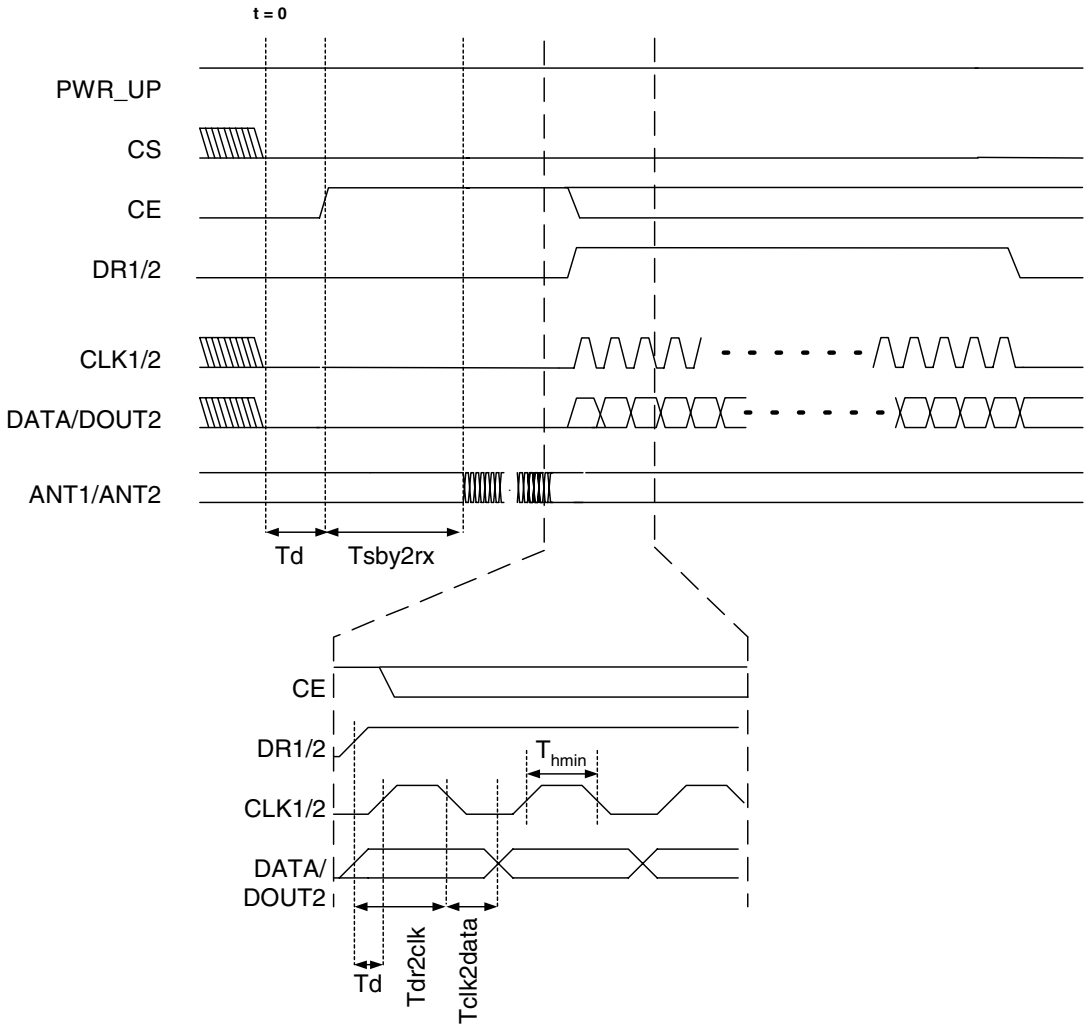



Figure 4-14 Timing of ShockBurst™ in RX

The CE may be kept high during downloading of data, but the cost is higher current consumption (19mA) and the benefit is short start-up time (200µs) when DR1 goes low.

# 5 A/D CONVERTER

The AD converter subsystem has 10 bit dynamic range and linearity when used at the Nyquist rate. With lower signal frequencies and post filtering, up to 12 bits resolution is possible. The reference for the AD converter is selectable between the AREF input and an internal 1.22V bandgap reference.

The converter default setting is 10 bits. For special requirements, the AD converter can be configured to perform 6, 8, 10 or 12 bit conversions. The converter may also be used in differential mode with AIN0 used as inverting input and one of the other 7 external inputs used as noninverting input. In differential mode a slightly improvement (e.g. 2dB for a 10 bit conversion) in SNR may be expected.

The AD converter is interfaced to the microcontroller via 4 registers. ADCCON (0xA1) contains the most commonly used control functions like channel and reference selection, power on and start stop control. ADCSTATIC (0xA4) contains infrequently used control functions that will normally not be changed by nRF24E1 applications. The high part of the result is available in the ADCDATAH (0xA2) register, whereas the ADCDATAL (0xA3) will hold the low part of the result (if any) and the end of conversion together with overflow status bits.

The complete AD subsystem is switched off by clearing bit NPD (ADCCON.5). The AD converter is normally clocked by the CPU clock divided by 32 (125 to 625 kHz), and the ADC will produce 2 bits of result per clock cycle.

## 5.1 A/D converter subsystem block diagram

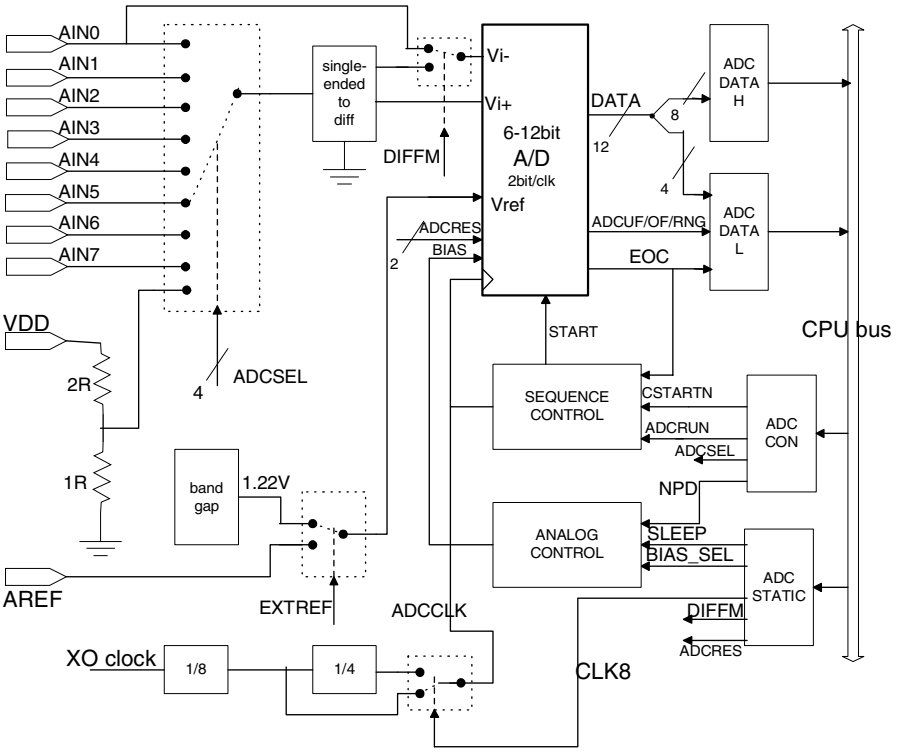



Figure 5-1 : Block diagram of A/D converter

## 5.2 A/D converter registers

### *5.2.1 ADCCON register, SFR 0xA1*

| Bit(s) | Name | Function |
|---|---|---|
| 7 | CSTARTN | Toggle H -> L -> H to start A/D conversion.<br>This bit is internally synchronized to the ADC clock<br>Ignored if ADCRUN is set.. |
| 6 | ADCRUN | Set to have the A/D converter run continuously<br>CSTARTN is ignored in this case |
| 5 | NPD | Set to 0 to put A/D converter in power down state |
| 4 | EXTREF | Select reference for A/D converter<br>0: Use internal band gap reference (nominally 1.22V)<br>1: Use external pin AREF for reference<br>Ignored if ADCSEL=8. |
| 3 - 0 | ADCSEL | Select input AIN0 to AIN7<br>ADCSEL=8 will select internal VDD/3, and also automatically select internal bandgap reference<br>For n=0..7, ADCSEL=n will select input pin AINn |

Table 5-1 : ADCCON register, SFR 0xA1, default initial data value is 0x80.

### *5.2.2 ADCSTATIC register, SFR 0xA4*

| Bit(s) | Name | Function |
|---|---|---|
| 7 | DIFFM | Enable differential measurements, AIN0 must be used as inverting input and one of the other inputs AIN1 to AIN7, as selected by ADCSEL, must be used as noninverting input. |
| 6 | SLEEP | Set A/D converter in a reduced power mode |
| 5 | CLK8 | 0 : ADCCLK frequency = CPU clock divided by 32<br>1 : ADCCLK frequency = CPU clock divided by 8 |
| 4 – 2 | ADCBIAS | Control A/D converter bias current<br>No need to change for nRF24E1 operation |
| 1 - 0 | ADCRES | Select A/D converter resolution<br>00: 6-bit, result in ADCDATAH 5-0<br>01: 8-bit, result in ADCDATAH<br>10: 10-bit, result in ADCDATAH,ADCDATAL.7-6<br>11: 12-bit, result in ADCDATAH,ADCDATAL.7-4 |

Table 5-2 : ADCSTATIC register, SFR 0xA4, default initial data value is 0x0A.

### *5.2.3 ADCDATAH register, SFR 0xA2*

| Bit(s) | Name | Function |
|---|---|---|
| 7 - 0 | ADCDATAH | Most significant 8 bits of A/D converter result. For 6-bit conversions ADCDATAH.7-6 is ‘00’ |

### *5.2.4 ADCDATAL register, SFR 0xA3*

| Bit(s) | Name | Function |
|---|---|---|
| 7 - 4 | ADCDATAL | Least significant part of A/D converter result when resolution is 12 or 10 bits, leftjustified. For 10-bit conversions ADCDATAH.5-4 is ‘00’ |
| 3 | | not used |
| 2 | ADCUF | Underflow in conversion. Data is all 0’s |
| 1 | ADCOF | Overflow in conversion. Data is all 1’s |
| 0 | ADCRNG | Overflow or underflow in conversion<br>(ADCUF \| ADCOF) |

Table 5-3 : ADC data SFR-registers, SFR 0xA2 and 0xA3.

## 5.3 A/D converter usage

### *5.3.1 End of conversion.*

A signal ADC_EOC is available in the EXIF.4 bit (Interrupt 2 flag) and it is set to 1 by A/D converter when a conversion (single step or continuous mode) is completed, see Table 7-4 : EXIF Register – SFR 0x91. For timing of ADC_EOC, see Figure 5-3 and Figure 5-4

### *5.3.2 Measurements with external reference*

When EXTREF (ADCCON.4) is set to 1 and ADCSEL (ADCCON.3-0) selects an input $AIN_i$ ( i.e. $AIN_0$ to $AIN_7$), the result in ADCDATA is directly proportional to the ratio between the voltage on the selected input, and the voltage on pin AREF.

$AIN_i$ voltage = AREF voltage * ADCDATA / $2^{**N}$

Where N is the number of bits set in ADCRES (ADCSTATIC.1-0) and ADCDATA is the resulting bits in ADCDATAH (and ADCDATAL if N > 8).

For differential measurements a simular equation apply :

($AIN_i$ - $AIN_0$)voltage = AREF voltage * (ADCDATA -$2^{**(N-1)}$) / $2^{**N}$

This mode of operation is normally selected for sources where the voltage is depending on the supply voltage (or another variable voltage), like shown in Figure 5-2 below. The resistor R1 is selected to keep AREF ≤ 1.5V for the maximum VDD voltage.

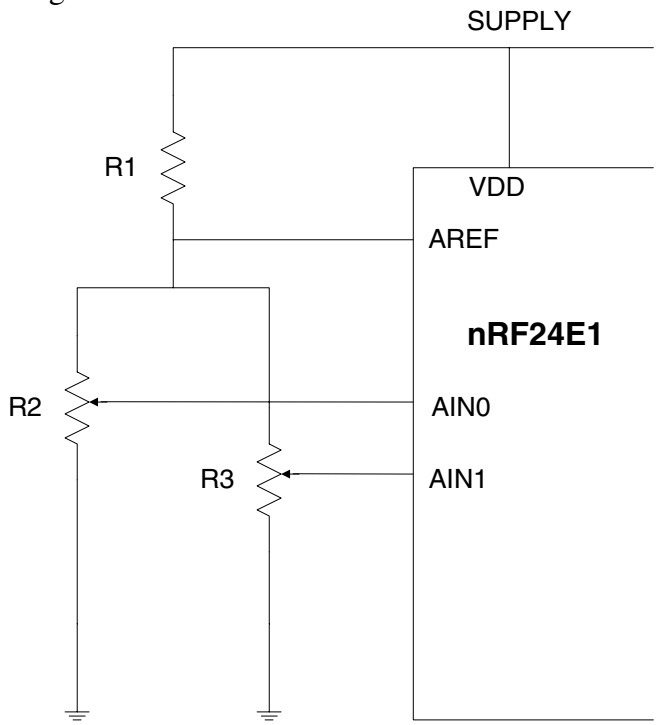



Figure 5-2 Typical use of A/D with 2 ratiometric inputs

### *5.3.3 Measurements with internal reference*

When EXTREF (ADCCON.4) is set to 0 and ADCSEL (ADCCON.3-0) selects an input $AIN_i$ (i.e. $AIN_0$ to $AIN_7$), the result in ADCDATA is directly proportional to the ratio between the voltage on the selected input, and the internal bandgap reference (nominally 1.22V).

if single ended input : $AIN_i$ voltage = 1.22 V * ADCDATA / $2^{**N}$

if differential input : ($AIN_i$ - $AIN_0$) voltage = 1.22 V * (ADCDATA -$2^{**(N-1)}$) / $2^{**N}$

Where N is the number of bits set in ADCRES (ADCSTATIC.1-0) and ADCDATA is the result bits in ADCDATAH (and ADCDATAL if N > 8).

This mode of operation is normally selected for sources where the voltage is not depending on the supply voltage.

### *5.3.4 Supply voltage measurement*

When ADCSEL (ADCCON.3-0) is set to 8, the ADC will use the internal bandgap reference (nominally 1.22V), and the input is 1/3 of the voltage on the VDD pins. The result in ADCDATA is thus directly proportional to the VDD voltage.

VDD voltage = 3.66 V * ADCDATA / 2**N

Where N is the number of bits set in ADCRES (ADCSTATIC.1-0) and ADCDATA is the result bits in ADCDATAH (and ADCDATAL if N > 8).

## 5.4 A/D Converter timing

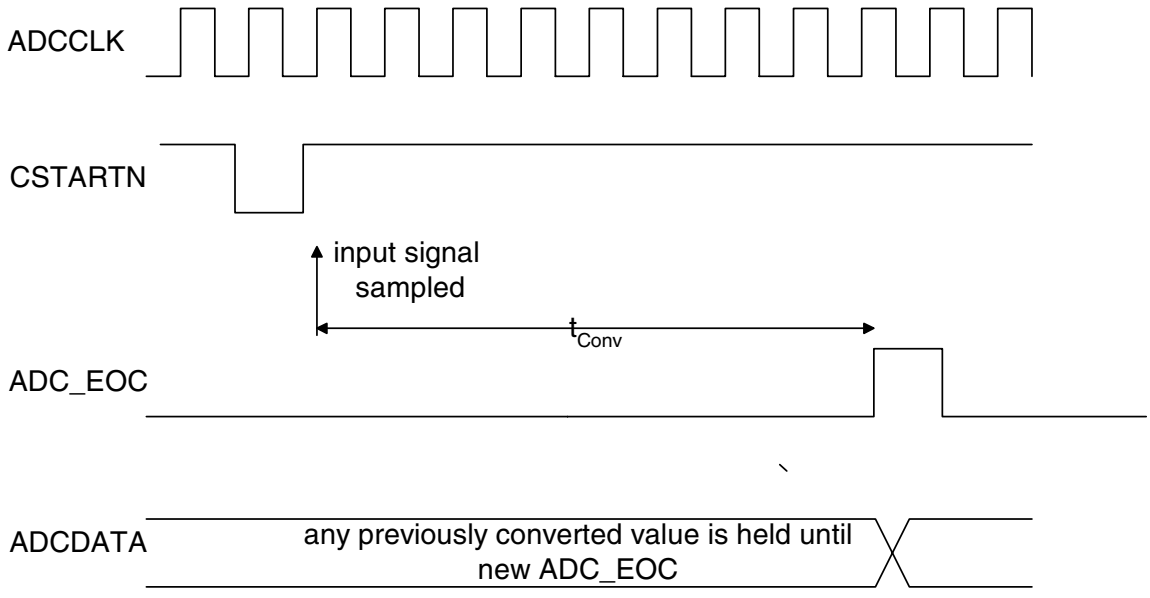



Figure 5-3 : Timing diagram single step conversion.

ADCRUN=0, and conversion is started at first posedge ADCCLK after CSTARTN has gone high. A pulse is generated on ADC_EOF when the converted value is available on the ADCDATA bus. Conversion time $t_{Conv}$ depends on resolution, $t_{Conv}$ = N/2 + 3 clock cycles, where N is number of resolution bits. In the figure a 10 bit conversion is shown. Minimum width of a CSTARTN pulse is 1 clock cycle. If a new CSTARTN pulse comes before previous conversion has finished, the previous conversion will be aborted.

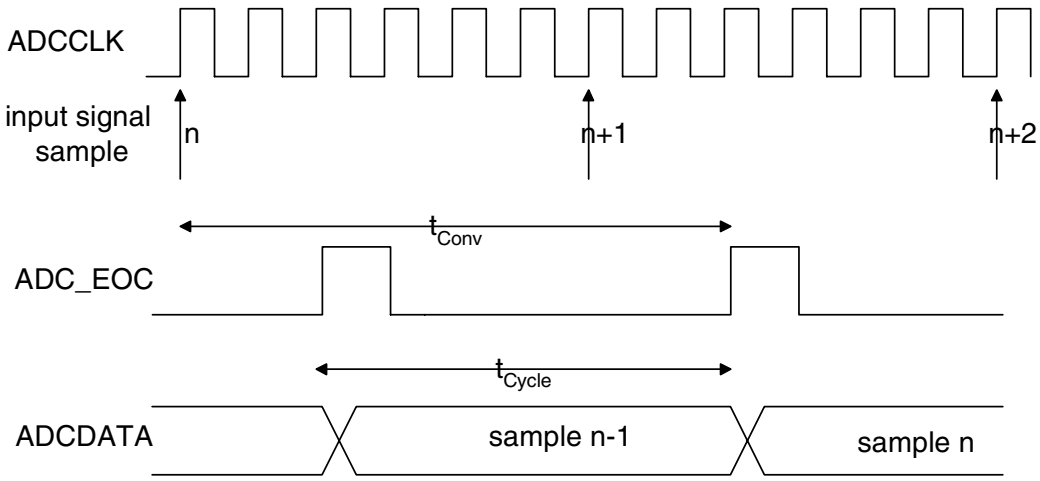



Figure 5-4 : Timing diagram continuous mode conversion.

ADCRUN=1, and CSTARTN is ignored. Cycle time $t_{Cycle}$ is the time between each conversion. $t_{Cycle}$ = N/2 +1 clock cycles, where N is number of resolution bits. The figure is showing 10 bit conversions.

## 5.5 Analog interface guidelines

The input impedance of analog inputs should preferably be in range 100-1000 Ω, and in any case be less than 10 kΩ. Small capacitors on inputs (e.g. 200pF) are recommended for decoupling, see also Figure 15-1 for application example.
If AIN inputs goes beyond the selected reference voltage, the ADC will clip and the result will be the maximum code. Absolute maximum for any AIN voltage is 2.0V.

# 6 PWM

The nRF24E1 PWM output is a one-channel PWM with a 2 register interface. The first register, PWMCON, enables PWM function and PWM period length, which is the number of clock cycles for one PWM period, as shown in the table below. The other register, PWMDUTY, controls the duty cycle of the PWM output signal. When this register is written, the PWM signal will change immediately to the new value. This can result in 4 transitions within one PWM period, but the transition period will always have a "DC value" between the "old" sample and the "new" sample.

The table shows how PWM frequency (or period length) and PWM duty cycle are controlled by the settings in the two PWM SFR-registers. For a crystal frequency of 16 MHz, PWM frequency range will be about 1-253 kHz.

| **PWMCON[7:6]** | **PWM frequency** | **PWMDUTY (duty cycle)** |
|---|---|---|
| 00 | 0 (PWM module inactive) | 0 |
| 01 | $f_{XO} \cdot \frac{1}{63 \cdot (PWMCON[5:0]+1)}$ | $\frac{PWMDUTY[5:0]}{63}$ |
| 10 | $f_{XO} \cdot \frac{1}{127 \cdot (PWMCON[5:0]+1)}$ | $\frac{PWMDUTY[6:0]}{127}$ |
| 11 | $f_{XO} \cdot \frac{1}{255 \cdot (PWMCON[5:0]+1)}$ | $\frac{PWMDUTY}{255}$ |

PWM is controlled by SFR 0xA9 and 0xAA.

| **Addr SFR (hex)** | **R/W** | **#bit** | **Init (hex)** | **Name** | **Function** |
|---|---|---|---|---|---|
| A9 | R/W | 8 | 0 | PWMCON | PWM control register<br>7-6: Enable / period length select<br>00: Disable PWM<br>01: Period length is 6 bit<br>10: Period length is 7 bit<br>11: Period length is 8 bit<br>5-0: PWM frequency prescale factor (see table above) |
| AA | R/W | 8 | 0 | PWMDUTY | PWM duty cycle (6 to 8 bits according to period length) |

Table 6-1 : PWM control registers - SFR 0xA9 and 0xAA

# 7 INTERRUPTS

nRF24E1 supports the following interrupt sources:

| Interrupt signal | Description |
|---|---|
| INT0_N | External interrupt, active low, configurable as edge-sensitive or level-sensitive, at Port P0.3 |
| TF0 | Timer 0 interrupt |
| INT1_N | External interrupt, active low, configurable as edge-sensitive or level-sensitive, at Port P0.4 |
| TF1 | Timer 1 interrupt |
| TF2 or EXF2 | Timer 2 interrupt |
| TI or RI | Receive/transmit interrupt from Serial Port |
| int2 | Internal ADC_EOC (end of AD conversion) interrupt |
| int3 | Internal SPI_READY interrupt |
| int4 | Internal RADIO.DR1 interrupt (a packet is ready from receiver 1) |
| int5 | Internal RADIO.DR2 interrupt (a packet is ready from receiver 2) |
| wdti | Internal RTC wakeup timer interrupt |

Table 7-1 : nRF24E1 interrupt sources

## 7.1 Interrupt SFRs

The following SFRs are associated with interrupt control:
- IE – SFR 0xA8 (Table 7-2)
- IP – SFR 0xB8 (Table 7-3)
- EXIF – SFR 0x91 (Table 7-4)
- EICON – SFR 0xD8 (Table 7-5)
- EIE – SFR 0xE8 (Table 7-6)
- EIP – SFR 0xF8 (Table 7-7)

The IE and IP SFRs provide interrupt enable and priority control for the standard interrupt unit, as with industry standard 8051. The EXIF, EICON, EIE, and EIP registers provide flags, enable control, and priority control for the extended interrupt unit.

Table 7-2 explains the bit functions of the IE register.

| Bit | Function |
|---|---|
| IE.7 | EA - Global interrupt enable. Controls masking of all interrupts. EA = 0 disables all interrupts (EA overrides individual interrupt enable bits). When EA = 1, each interrupt is enabled or masked by its individual enable bit. |
| IE.6 | Reserved. Read as 0. |
| IE.5 | ET2 - Enable Timer 2 interrupt. ET2 = 0 disables Timer 2 interrupt (TF2). ET2 = 1 enables interrupts generated by the TF2 or EXF2 flag. |

| | |
|---|---|
| IE.4 | ES - Enable Serial Port interrupt. ES = 0 disables Serial Port interrupts (TI and RI). ES = 1 enables interrupts generated by the TI or RI flag. |
| IE.3 | ET1 - Enable Timer 1 interrupt. ET1 = 0 disables Timer 1 interrupt (TF1). ET1 = 1 enables interrupts generated by the TF1 flag. |
| IE.2 | EX1 - Enable external interrupt 1. EX1 = 0 disables external interrupt 1 (INT1_N). EX1 = 1 enables interrupts generated by the INT1_N pin. |
| IE.1 | ET0 - Enable Timer 0 interrupt. ET0 = 0 disables Timer 0 interrupt (TF0). ET0 = 1 enables interrupts generated by the TF0 flag. |
| IE.0 | EX0 - Enable external interrupt 0. EX0 = 0 disables external interrupt 0 (INT0_N). EX0 = 1 enables interrupts generated by the INT0_N pin. |

Table 7-2 : IE Register – SFR 0xA8

Table 7-3 explains the bit functions of the IP register.

| Bit | Function |
|---|---|
| IP.7 | Reserved. Read as 1. |
| IP.6 | Reserved. Read as 0. |
| IP.5 | PT2 - Timer 2 interrupt priority control. PT2 = 0 sets Timer 2 interrupt (TF2) to low priority. PT2 = 1 sets Timer 2 interrupt to high priority. |
| IP.4 | PS - Serial Port interrupt priority control. PS = 0 sets Serial Port interrupt (TI or RI) to low priority. PS = 1 sets Serial Port interrupt to high priority. |
| IP.3 | PT1 - Timer 1 interrupt priority control. PT1 = 0 sets Timer 1 interrupt (TF1) to low priority. PT1 = 1 sets Timer 1 interrupt to high priority. |
| IP.2 | PX1 - External interrupt 1 priority control. PX1 = 0 sets external interrupt 1 (INT1_N) to low priority. PT1 = 1 sets external interrupt 1 to high priority. |
| IP.1 | PT0 - Timer 0 interrupt priority control. PT0 = 0 sets Timer 0 interrupt (TF0) to low priority. PT0 = 1 sets Timer 0 interrupt to high priority. |
| IP.0 | PX0 - External interrupt 0 priority control. PX0 = 0 sets external interrupt 0 (INT0_N) to low priority. PT0 = 1 sets external interrupt 0 to high priority. |

Table 7-3 : IP Register – SFR 0xB8

Table 7-4 explains the bit functions of the EXIF register.

| Bit | Function |
|---|---|
| EXIF.7 | IE5 - Interrupt 5 flag. IE5 = 1 indicates that a rising edge was detected on the RADIO.DR2 signal.(see ch. 5.1.RADIO) IE5 must be cleared by software. Setting IE5 in software generates an interrupt, if enabled. |
| EXIF.6 | IE4 - Interrupt 4 flag. IE4 = 1 indicates that a rising edge was detected on the RADIO.DR1 signal.(see ch. 5.1.RADIO) IE4 must be cleared by software. Setting IE4 in software generates an interrupt, if enabled. |

| | |
|---|---|
| EXIF.5 | IE3 - Interrupt 3 flag. IE3 = 1 indicates that the internal SPI module has sent or received 8 bits, and is ready for a new command. IE3 must be cleared by software. Setting IE3 in software generates an interrupt, if enabled. |
| EXIF.4 | IE2 - Interrupt 2 flag. IE2 = 1 indicates that a rising edge was detected on the ADC_EOC signal. (see ch.5.3.1 End of conversion.) IE2 must be cleared by software. Setting IE2 in software generates an interrupt, if enabled. |
| EXIF.3 | Reserved. Read as 1. |
| EXIF.2-0 | Reserved. Read as 0. |

Table 7-4 : EXIF Register – SFR 0x91

Table 7-5 explains the bit functions of the EICON register.

| Bit | Function |
|---|---|
| EICON.7 | Not used. |
| EICON.6 | Reserved. Read as 1. |
| EICON.5 | Reserved. Read as 0. |
| EICON.4 | Reserved. Read as 0. |
| EICON.3 | WDTI - RTC wakeup timer interrupt flag. WDTI = 1 indicates a wakeup timer interrupt was detected. WDTI must be cleared by software before exiting the interrupt service routine. Otherwise, the interrupt occurs again. Setting WDTI in software generates a wakeup timer interrupt, if enabled. |
| EICON.2-0 | Reserved. Read as 0. |

Table 7-5 : EICON Register – SFR 0xD8

Table 7-6 explains the bit functions of the EIE register.

| Bit | Function |
|---|---|
| EIE.7-5 | Reserved. Read as 1. |
| EIE.4 | EWDI - Enable RTC wakeup timer interrupt. EWDI = 0 disables wakeup timer interrupt (wdti). EWDI = 1 enables interrupts generated by wakeup. |
| EIE.3 | EX5 - Enable interrupt 5. EX5 = 0 disables interrupt 5 (RADIO.DR2). EX5 = 1 enables interrupts generated by the RADIO.DR2 signal. |
| EIE.2 | EX4 - Enable interrupt 4. EX4 = 0 disables interrupt 4 (RADIO.DR1). EX4 = 1 enables interrupts generated by the RADIO.DR1 signal. |
| EIE.1 | EX3 - Enable interrupt 3. EX3 = 0 disables interrupt 3 (SPI_READY). EX3 = 1 enables interrupts generated by the SPI_READY signal. |
| EIE.0 | EX2 - Enable interrupt 2. EX2 = 0 disables interrupt 2 (ADC_EOC). EX2 = 1 enables interrupts generated by the ADC_EOC signal. |

Table 7-6 : EIE Register – SFR 0xE8

Table 7-7 explains the bit functions of the EIP register.

| Bit | Function |
|---|---|
| EIP.7-5 | Reserved. Read as 1. |
| EIP.4 | PWDI - RTC wakeup timer interrupt priority control. WDPI = 0 sets wakeup timer interrupt (wdti) to low priority. PS = 1 sets wakeup timer interrupt to high priority. |
| EIP.3 | PX5 - interrupt 5 priority control. PX5 = 0 sets interrupt 5 (RADIO.DR2) to low priority. PX5 = 1 sets interrupt 5 to high priority. |
| EIP.2 | PX4 - interrupt 4 priority control. PX4 = 0 sets interrupt 4 (RADIO.DR1) to low priority. PX4 = 1 sets interrupt 4 to high priority. |
| EIP.1 | PX3 - interrupt 3 priority control. PX3 = 0 sets interrupt 3 (SPI_READY) to low priority. PX3 = 1 sets interrupt 3 to high priority. |
| EIP.0 | PX2 - interrupt 2 priority control. PX2 = 0 sets interrupt 2 (ADC_EOC) to low priority. PX2 = 1 sets interrupt 2 to high priority. |

Table 7-7 : EIP Register – SFR 0xF8

## 7.2 Interrupt Processing

When an enabled interrupt occurs, the CPU vectors to the address of the interrupt service routine (ISR) associated with that interrupt, as listed in Table 7-8. The CPU executes the ISR to completion unless another interrupt of higher priority occurs. Each ISR ends with an RETI (return from interrupt) instruction. After executing the RETI, the CPU returns to the next instruction that would have been executed if the interrupt had not occurred.

| Interrupt | Description | Natural Priority (lowest number gives highest priority) | Interrupt Vector |
|---|---|---|---|
| INT0_N | External interrupt 0 | 1 | 0x03 |
| TF0 | Timer 0 interrupt | 2 | 0x0B |
| INT1_N | External interrupt 1 | 3 | 0x13 |
| TF1 | Timer 1 interrupt | 4 | 0x1B |
| TI or RI | Serial Port transmit or receive | 5 | 0x23 |
| TF2 or EXF2 | Timer 2 interrupt | 6 | 0x2B |
| int2 | ADC_EOC interrupt | 8 | 0x43 |
| int3 | SPI_READY interrupt | 9 | 0x4B |
| int4 | RADIO.DR1 interrupt | 10 | 0x53 |
| int5 | RADIO.DR2 interrupt | 11 | 0x5B |
| wdti | RTC wakeup timer interrupt | 12 | 0x63 |

Table 7-8 : Interrupt Natural Vectors and Priorities

An ISR can only be interrupted by a higher priority interrupt. That is, an ISR for a low-level interrupt can be interrupted only by a high-level interrupt. The CPU always completes the instruction in progress before servicing an interrupt. If the instruction in progress is RETI, or a write access to any of the IP, IE, EIP, or EIE SFRs, the CPU completes one additional instruction before servicing the interrupt.

## 7.3 Interrupt Masking

The EA bit in the IE SFR (IE.7) is a global enable for all interrupts. When EA = 1, each interrupt is enabled/masked by its individual enable bit. When EA = 0, all interrupts are masked. Table 7-9 provides a summary of interrupt sources, flags, enables, and priorities.

| Interrupt | Description | Flag | Enable | Control |
|---|---|---|---|---|
| INT0_N | External interrupt 0 | TCON.1 | IE.0 | IP.0 |
| TF0 | Timer 0 interrupt | TCON.5 | IE.1 | IP.1 |
| INT1_N | External interrupt 1 | TCON.3 | IE.2 | IP.2 |
| TF1 | Timer 1 interrupt | TCON.7 | IE.3 | IP.3 |
| TI or RI | Serial Port transmit or receive | SCON.0 (RI), SCON.1 (TI) | IE.4 | IP.4 |
| TF2 or EXF2 | Timer 2 interrupt | T2CON.7 (TF2), T2CON.6 (EXF2) | IE.5 | IP.5 |
| int2 | ADC_EOC interrupt | EXIF.4 | EIE.0 | EIP.0 |
| int3 | SPI_READY interrupt | EXIF.5 | EIE.1 | EIP.1 |
| int4 | RADIO.DR1 interrupt | EXIF.6 | EIE.2 | EIP.2 |
| int5 | RADIO.DR2 interrupt | EXIF.7 | EIE.3 | EIP.3 |
| wdti | RTC wakeup timer interrupt | EICON.3 | EIE.4 | EIP.4 |

Table 7-9 : Interrupt Flags, Enables, and Priority Control

## 7.4 Interrupt Priorities

There are two stages of interrupt priority assignment: interrupt level and natural priority. The interrupt level (high, or low) takes precedence over natural priority. All interrupts can be assigned either high or low priority. In addition to an assigned priority level (high or low), each interrupt has a natural priority, as listed in Table 7-8. Simultaneous interrupts with the same priority level (for example, both high) are resolved according to their natural priority. For example, if INT0_N and int2 are both programmed as high priority, INT0_N

takes precedence. Once an interrupt is being serviced, only an interrupt of higher priority level can interrupt the service routine of the interrupt currently being serviced.

## 7.5 Interrupt Sampling

The internal timers and serial port generate interrupts by setting their respective SFR interrupt flag bits. The CPU samples external interrupts once per instruction cycle, at the rising edge of CPU_clk at the end of cycle C4.
The INT0_N and INT1_N signals are both active low and can be programmed through the IT0 and IT1 bits in the TCON SFR to be either edge-sensitive or level-sensitive. For example, when IT0 = 0, INT0_N is level-sensitive and the CPU sets the IE0 flag when the INT0_N pin is sampled low. When IT0 = 1, INT0_N is edge-sensitive and the CPU sets the IE0 flag when the INT0_N pin is sampled high then low on consecutive samples. To ensure that edge-sensitive interrupts are detected, the corresponding ports should be held high for four clock cycles and then low for four clock cycles. Level-sensitive interrupts are not latched and must remain active until serviced.

## 7.6 Interrupt Latency

Interrupt response time depends on the current state of the CPU. The fastest response time is five instruction cycles: one to detect the interrupt, and four to perform the LCALL to the ISR.The maximum latency (thirteen instruction cycles) occurs when the CPU is currently executing an RETI instruction followed by a MUL or DIV instruction. The thirteen instruction cycles in this case are: one to detect the interrupt, three to complete the RETI, five to execute the DIV or MUL, and four to execute the LCALL to the ISR.
For the maximum latency case, the response time is 13 x 4 =52clock cycles.

## 7.7 Interrupt Latency from Power Down Mode.

nRF24E1 may be set into Power Down Mode by writing 0x2 or 0x3 to SFR 0xB6, register CK_CTRL. The CPU will then perform a controlled shutdown of clock and power regulator. The system can only be restarted from pins INT0_N or INT1_N, or an RTC wakeup or a Watchdog reset. In this case the CPU cannot respond until the clock and power regulator have restarted, which may take 3 to 4 LP_OSC cycles. This delay may vary from 0.6ms to 4 ms depending on processing, temperature and supply voltage. In the same way, the shutdown also takes from 2 to 3 LP_OSC cycles, which will be in the range of 0.4 - 3ms.

## 7.8 Single-Step Operation

The nRF24E1 interrupt structure provides a way to perform single-step program execution. When exiting an ISR with an RETI instruction, the CPU will always execute at least one instruction of the task program. Therefore, once an ISR is entered, it cannot be re-entered until at least one program instruction

is executed. To perform single-step execution, program one of the external interrupts (for example, INT0_N) to be level-sensitive and write an ISR for that interrupt that terminates as follows:

```
JNB TCON.1,$ ;    wait for high on INT0_N
JB TCON.1,$ ;     wait for low on INT0_N
RETI ;           return for ISR
```

The CPU enters the ISR when INT0_N goes low, then waits for a pulse on INT0_N. Each time INT0_N is pulsed, the CPU exits the ISR, executes one program instruction, then re-enters the ISR.

# 8 WAKEUP TIMER AND WATCHDOG

## 8.1 Tick calibration

The "TICK" is an interval that is nominally 10ms long. This interval is the unit of resolution both for the watchdog and the RTC wakeup timer. The LP_OSC clock source of the "TICK" is very inaccurate, and may vary from 6ms to 30ms depending on processing, temperature and supply voltage. That means that Watchdog and RTC may not be used for any accurate timing functions.

The accuracy can be improved by calibrating the TICK value at regular intervals. The register TICK_DV controls how many LP_OSC periods elapse between each TICK. The frequency of the LP_OSC (between 1 kHz and 5 kHz) can be measured by timer2 in capture mode with t2ex enabled (EXEN2=1). The signal connected to t2ex has exactly half the frequency of LP_OSC. The 16-bit difference between two consecutive captures in SFR-registers{RCAP2H,RCAP2L} is proportional to the LP_OSC period. For details about timer2 see ch. 10.8.3 and Figure 10-5 : Timer 2 – Timer/Counter with Capture

TICK is controlled by SFR 0xB5.

| Addr SFR | R/W | #bit | Init hex | Name | Function |
|---|---|---|---|---|---|
| B5 | R/W | 8 | 1D | TICK_DV | Divider that's used in generating TICK from LP_OSC frequency.<br>$f_{TICK} = f_{LP\_OSC} / (1 + TICK\_DV)$<br>The default value gives a TICK of 10ms nominal as default. |

Table 8-1 : TICK control register - SFR 0xB5

## 8.2 RTC Wakeup timer

The RTC is a simple 16 bit down counter that produces an interrupt and reloads automatically when the count reaches zero. This process is initially disabled, and will be enabled with the first write to the timer latch. Writing the timer latch will always be followed by a reload of the counter. The counter may be disabled again by writing a disable opcode to the control register. Both the latch and the counter value may be read by giving the respective codes in the control register, see description in Table 8-2

This counter is used for a wakeup sometime in the future (a relative time wakeup call). If 'N' is written to the counter, the first wakeup will happen from somewhere between 'N+1' and 'N+2' "TICK" from the completion of the write, thereafter a new wakeup is issued every "N+1" "TICK" until the unit is disabled or another value is written to the latch.
The wakeup timer is connected to the WDTI interrupt of the CPU. The programmer may poll the EICON.3 flag or enable the interrupt. If the oscillator is stopped, the wakeup interrupt will restart the oscillator regardless of the state of EIE.4 interrupt enable.
The nRF24E1 do not provide any "absolute time functions". Absolute time functions in nRF24E1 can well be handled in software since our RAM is continuously powered even when in sleep mode. There will be an application note with the required code to implement the complete absolute time function using some 100 bytes of code and 12 IRAM locations (with 2 alarms).

## 8.3 Watchdog

The watchdog is activated upon writing 0x08 to its control register SFR 0xAD. It can not be disabled by any other means than a reset. The watchdog register is loaded by writing a 16-bit value to the two 8-bit data registers (SFR 0xAB and 0xAC) and then the writing the correct opcode to the control register. The watchdog will then count down towards 0 and when 0 is reached the complete microcontroller will be reset . To avoid the reset, the software must load new values into the watchdog register sufficiently often.

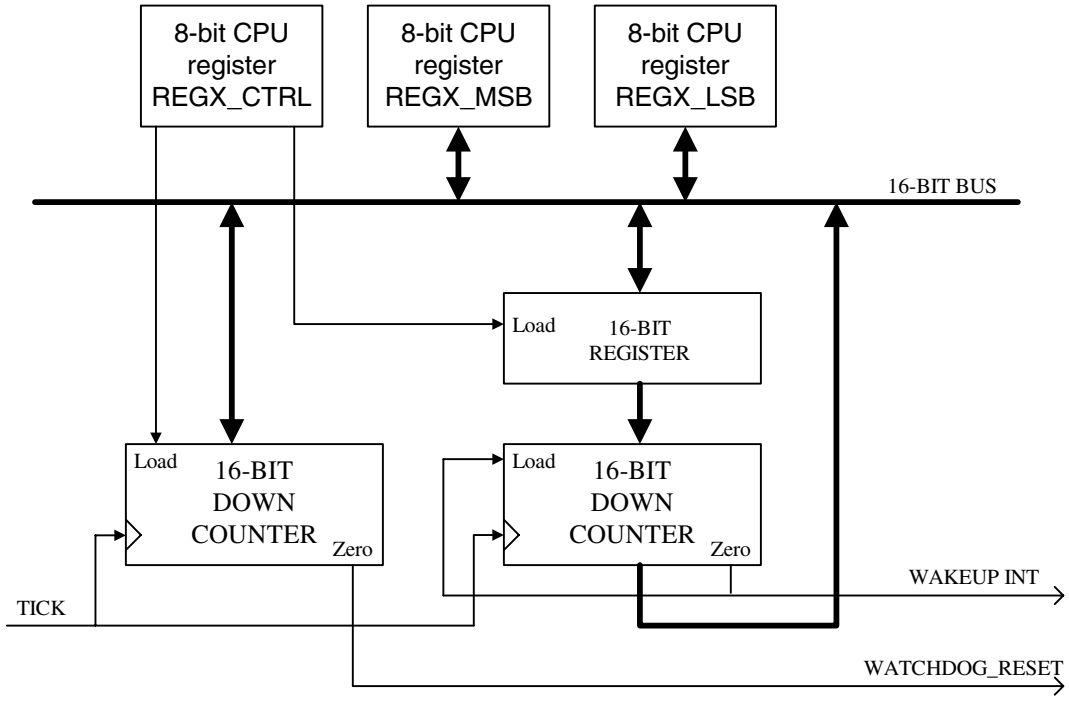



Figure 8-18-2 : RTC and watchdog block diagram

RTC and Watchdog are controlled by SFRs 0xAB, 0xAC and 0xAD. These 3 registers REGX_MSB, REGX_LSB and REGX_CTRL are used to interface the blocks running on the slow LP_OSC clock. The 16-bit register {REGX_MSB, REGX_LSB} can be written or read as two bytes from the CPU. Typical sequences are:

**Write:** Wait until not busy.
Write REGX_MSB, Write REGX_LSB, Write REGX_CTRL

**Read:** Wait until not busy.
Write REGX_CTRL, Wait until not busy.
Read REGX_MSB, Read REGX_LSB

Note : please also wait until not busy before accessing SFR 0xB6 CK_CTRL (page 58)

| Addr SFR (hex) | R/W | # bit | Init (hex) | Name | Function |
|---|---|---|---|---|---|
| AB | R/W | 8 | 0 | REGX_MSB | Most significant part of 16 bit register for interface to Watchdog and RTC |
| AC | R/W | 8 | 0 | REGX_LSB | Least significant part of 16 bit register for interface to Watchdog and RTC |

| | | | | | |
|---|---|---|---|---|---|
| AD | R/W | 5 | 0 | REGX_ CTRL | Control for 16 bit register for interface to Watchdog and RTC. Bit 4 is only available on read and is used to flag the interface unit as busy.<br>Bits 3:0 is read/write with the encoding:<br>0 000: Read from WD register (16 bit)<br>1 000: Write to WD register (16 bit)<br>0 010: Read from RTC latch register (16 bit)<br>1 010: Write to RTC latch register (16 bit)<br>0 011: Read from RTC counter reg. (16 bit)<br>1 011: Disable RTC counter (no data) |

Table 8-2 : RTC and Watchdog SFR-registers

## 8.4 Reset

nRF24E1 can be reset either by the on-chip power-on reset circuitry or by the on-chip watchdog counter.

### *8.4.1 Power-on Reset*

The power-on reset circuitry keeps the chip in power-on-reset state until the supply voltage reaches VDDmin. At this point the internal voltage generators and oscillators start up, the SFRs are initialized to their reset values, as listed in Table 10-10, and thereafter the CPU begins program execution at the standard reset vector address 0x0000. The startup time from power-on reset is about 14 LP_OSC cycles, which in total may vary from 3 to 15ms depending on processing, temperature and supply voltage.

### *8.4.2 Watchdog Reset*

If the Watchdog reset signal goes active, nRF24E1 enters the same reset sequence as for power-on reset, that is the internal voltage generators and oscillators start up, the SFRs are initialized to their reset values, as listed in Table 10-10, and thereafter the CPU begins program execution at the standard reset vector address 0x0000. (of the existing program, there is no reboot) The startup time from watchdog reset is somewhat shorter, 12 LP_OSC cycles, which in total may vary from 2.5 to 13ms depending on processing, temperature and supply voltage.

### *8.4.3 Program reset address*

The program reset address is controlled by the RSTREAS register, SFR 0xB1, see Table 8-3 This register shows which reset source that caused the last reset, and provides a choice of two different program start addresses. The default value is power-on reset, which starts the boot loader, while a watchdog reset does not reboot, but restarts at address 0 of the already loaded program.

| Addr SFR (hex) | R/W | #bit | Init (hex) | Name | Function |
|---|---|---|---|---|---|
| B1 | R/W | 2 | 02 | RSTREAS | bit 0: Reason for last reset<br>0: POR<br>1: Any other reset source<br>Clear this bit in software to force a reboot after jump to zero (boot loader will load code RAM if this bit is 0)<br>bit 1: Use IROM for reset vector<br>0: Reset vectors to 0x0000.<br>1: Reset vectors to 0x8000. |

Table 8-3 Reset control registe - SFR 0xB1.

# 9 POWER SAVING MODES

nRF24E1 provides the two industry standard 8051 power saving modes: idle mode and stop mode, but with only minor power saving; therefore also a non standard power-down mode is provided, where both oscillator and internal power regulators are turned off to achieve more power saving.

The bits that control entry into idle and stop modes are in the PCON register at SFR address 0x87, listed in Table 9-1. The bits that control entry into power down mode are in the CK_CTRL register at SFR address 0xB6, listed in Table 9-2

| Bit | Function |
|---|---|
| PCON.7 | SMOD – Serial Port baud-rate doubler enable. When SMOD = 1, the baud rate for Serial Port is doubled. |
| PCON.6–4 | Reserved. |
| PCON.3 | GF1 – General purpose flag 1. Bit-addressable, general purpose flag for software control. |
| PCON.2 | GF0 – General purpose flag 0. Bit-addressable, general purpose flag for software control. |
| PCON.1 | STOP – Stop mode select. Setting the STOP bit places the nRF24E1 in stop mode. |
| PCON.0 | IDLE – Idle mode select. Setting the IDLE bit places the nRF24E1 in idle mode. |

Table 9-1 : PCON Register – SFR 0x87

## 9.1 Idle Mode

An instruction that sets the IDLE bit (PCON.0) causes the nRF24E1 to enter idle mode when that instruction completes. In idle mode, CPU processing is suspended and internal registers and memory maintain their current data. However, unlike the standard 8051, the CPU clock is not disabled internally, thus not much power is saved.

There are two ways to exit idle mode: activate any enabled interrupt or watchdog reset. Activation of any enabled interrupt causes the hardware to clear the IDLE bit and terminate idle mode. The CPU executes the ISR associated with the received interrupt. The RETI instruction at the end of the of ISR returns the CPU to the instruction following the one that put the nRF24E1 into idle mode. A watchdog reset causes the nRF24E1 to exit idle mode, reset internal registers, execute its reset sequence and begin program execution at the standard reset vector address 0x0000.

## 9.2 Stop Mode

An instruction that sets the STOP bit (PCON.1) causes the nRF24E1 to enter stop mode when that instruction completes. Stop mode is identical to idle mode, except that the only way to exit stop mode is by watchdog reset Since there is little power saving, stop mode is not recommended, as it is more efficient to use power down mode.

## 9.3 Power down mode

An instruction that sets the STOP_CLOCK bit (SFR 0xB6 CK_CTRL.1) causes the nRF24E1 to enter power down mode when that instruction completes. In power down mode, CPU processing is suspended, while internal registers and memories maintain their current data. The CPU will perform a controlled shutdown of clock and power regulators. But the transceiver subsystem has to be disabled separately by setting RADIO.7=0 before stopping the clock.
The system can only be restarted from a low level on pin INT0_N (P0.3) or INT1_N (P0.4) if enabled (by P0_ALT), or an RTC wakeup interrupt or a Watchdog reset. This will cause the hardware to clear the CK_CTRL.1 bit and terminate power down mode. If there is an enabled interrupt associated with the wakeup event, the CPU executes the ISR associated with that interrupt immediately after power and clocks are restored. The RETI instruction at the end of the of ISR returns the CPU to the instruction following the one that put the nRF24E1 into power down mode. A watchdog reset causes the nRF24E1 to exit power down mode, reset internal registers, execute its reset sequence and begin program execution at the standard reset vector address 0x0000.

Note : Before accessing the CK_CTRL register, make sure that the busy bit of RTC/Watchdog SFR 0xAD, bit 4 (page 56) is not set

| Bit | Function |
|---|---|
| CK_CTRL .0 | Not used |
| CK_CTRL .1 | STOP_CLOCK. Setting the STOP_CLOCK bit places the nRF24E1 in power down mode. |

Table 9-2 : CK_CTRL register - SFR 0xB6

### *9.3.1 Clarification about wakeup and interrupt from external events*

1: Wakeup and interrupt on pins P0.4 and P0.3 are intended to be parallel exclusive functions.
2: Interrupt circuitry is not active during power down and wakeup not active during power up.

3: however when the nRF24E1 is started by one of these pins, the event will be captured in the interrupt circuitry also and an interrupt MAY be delivered if enabled. A level interrupt will always be delivered (even if pin has returned high). A falling edge interrupt may be delivered.

### *9.3.2 Startup time from Power down mode.*

Startup time consists of a number of LP_OSC cycles + a number of XTAL clock cycles. $f_{LP\_OSC}$ may vary from 1 to 5.5kHz over voltage and temperature, but can be measured as described on page 53. $f_{XTAL}$ depends on the selected crystal, as described on page 96. Because frequency $f_{XTAL}$ is much higher, startup time is dominated by $f_{LP\_OSC}$.
Startup times are summarized in the table below :

| Reason of startup | Startup time in $f_{LP\_OSC}$ cycles | Startup time in $f_{XTAL}$ cycles | Example of total startup time<br>if $f_{LP\_OSC}$ =3kHz<br>if $f_{XTAL}$ =16MHz |
|---|---|---|---|
| Power-on reset | 14-15 | 24 | 4.8 ms |
| Watchdog reset | 12 | 24 | 4.0 ms |
| External interrupt | 3-4 | max 52,<br>see ch. 7.6 | 1.2 ms |
| RTC interrupt | 3 | max 52<br>see ch. 7.6 | 1.0 ms |

Table 9-3 : Startup times from Power down mode

# 10 MICROCONTROLLER

The embedded microcontroller is the DW8051 MacroCell from Synopsys which is similar to the Dallas DS80C320 in terms of hardware features and instruction-cycle timing.

## 10.1 Memory Organization

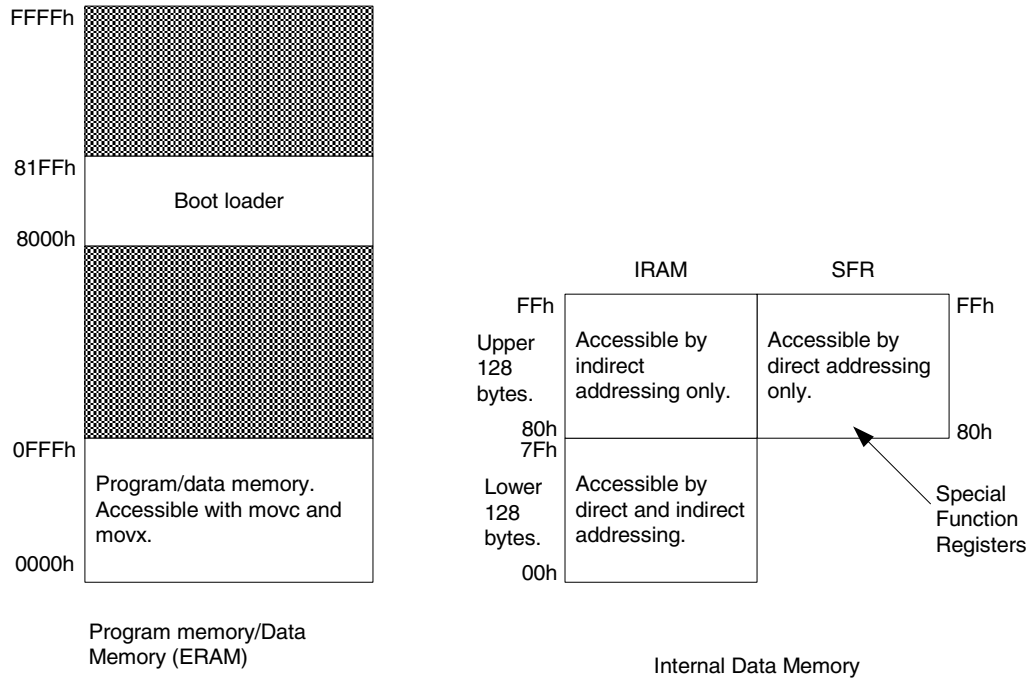



Figure 10-1 : Memory Map and Organization

### *10.1.1 Program Memory/Data Memory*

The nRF24E1 has 4KB of program memory available for user programs located at the bottom of the address space as shown in Figure 10-1. This memory also function as a random access memory and can be accessed with the movx and movc instructions.

After power on reset the boot loader loads the user program from the external serial EEPROM and stores it from address 0 in this memory.

10.1.1.1 Memory paging

A Special function register, MPAGE, at SFR address 0x92 provides memory paging function. During MOVX A, @Ri and MOVX @Ri, A instructions, the contents of the MPAGE register are placed on the upper eight address bits of memory address.

### *10.1.2 Internal Data Memory*

The Internal Data Memory, illustrated in Figure 10-1, consists of:

- 128 bytes of registers and scratchpad memory accessible through direct or indirect addressing (addresses 0x00–0x7F).
- 128 bytes of scratchpad memory accessible through indirect addressing (0x80–0xFF).
- 128 special function registers (SFRs) accessible through direct addressing.

The lower 32 bytes form four banks of eight registers (R0–R7). Two bits on the program status word (PSW) select which bank is in use. The next sixteen bytes form a block of bit-addressable memory space at bit addresses 0x00–0x7F. All of the bytes in the lower 128 bytes are accessible through direct or indirect addressing. The SFRs and the upper 128 bytes of RAM share the same address range (0x80-0xFF). However, the actual address space is separate and is differentiated by the type of addressing. Direct addressing accesses the SFRs, while indirect addressing accesses the upper 128 bytes of RAM. Most SFRs are reserved for specific functions, as described in 10.6Special Function Registers on page 69. SFR addresses ending in 0h or 8h are bit-addressable.

## 10.2 Program format in external EEPROM

The table below shows the layout of the first few bytes of the EEPROM image.

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| 0: | Version (now 00) | | Reserved (now 00) | | SPEED | XO_FREQ | | |
| 1: | Offset to start of user program (N) | | | | | | | |
| 2: | Number of 256 byte blocks in user program (includes block 0 that is not full) | | | | | | | |
| … | Optional User data, not interpreted by boot loader | | | | | | | |
| … | … | | | | | | | |
| N: | First byte of user program, goes into ERAM at 0x0000 | | | | | | | |
| N+1: | Second byte of user program, goes into ERAM at 0x0001 | | | | | | | |
| | … | | | | | | | |

Table 10-1 : EEPROM layout

The contents of the 4 lowest bits in the first byte is used by the boot loader to set the correct SPI frequency. These fields are encoded as shown below:

SPEED (bit 3): EEPROM max speed
0 = 1MHz
1 = 0.5MHz

XO_FREQ (bits 2,1 and 0): Crystal oscillator frequency
000 = 4MHz,
001 = 8MHz,
010 = 12MHz,
011 = 16MHz,
100 = 20MHz

The program eeprep can be used to add this header to a program file.

Command format: eeprep [options] <infile> <outfile>
<infile> is the output file of an assembler or compiler
<outfile> is a file suitable for programming the EEPROM (above format with no user data).
Both files are "Intelhex" format.
The options available for eeprep are:

-c n Set crystal frequency in MHz. Valid numbers are 4, 8, 12, 16 (default) and 20
-i Ignore checksums
-p n Set program memory size (default 4096 bytes)
-s Select slow EEPROM clock (500KHz)

## 10.3 Instruction Set

All nRF24E1 instructions are binary-code–compatible and perform the same functions that they do in the industry standard 8051. The effects of these instructions on bits, flags, and other status functions is identical to the industry-standard 8051. However, the timing of the instructions is different, both in terms of number of clock cycles per instruction cycle and timing within the instruction cycle.
The instruction set is fully compatible to the instruction set of nRF24E2.
Table 10-3 to Table 10-8 lists the nRF24E1 instruction set and the number of instruction cycles required to complete each instruction.

| Symbol | Function |
|---|---|
| A | Accumulator |
| Rn | Register R0–R7 |
| direct | Internal register address |
| @Ri | Internal register pointed to by R0 or R1 (except MOVX) |
| rel | Two's complement offset byte |
| bit | Direct bit address |
| #data | 8-bit constant |
| #data 16 | 16-bit constant |
| addr 16 | 16-bit destination address |
| addr 11 | 11-bit destination address |

Table 10-2 : Legend for Instruction Set Table

Table 10-3 to Table 10-8 define the symbols and mnemonics used in Table 10-2.

| Arithmetic Instructions | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **Byte** | **Instr. Cycles** | **Hex Code** |
| ADD A, Rn | Add register to A | 1 | 1 | 28–2F |
| ADD A, direct | Add direct byte to A | 2 | 2 | 25 |
| ADD A, @Ri | Add data memory to A | 1 | 1 | 26–27 |
| ADD A, #data | Add immediate to A | 2 | 2 | 24 |
| ADDC A, Rn | Add register to A with carry | 1 | 1 | 38–3F |
| ADDC A, direct | Add direct byte to A with carry | 2 | 2 | 35 |
| ADDC A, @Ri | Add data memory to A with carry | 1 | 1 | 36–37 |
| ADDC A, #data | Add immediate to A with carry | 2 | 2 | 34 |
| SUBB A, Rn | Subtract register from A with borrow | 1 | 1 | 98–9F |
| SUBB A, direct | Subtract direct byte from A with borrow | 2 | 2 | 95 |
| SUBB A, @Ri | Subtract data memory from A with borrow | 1 | 1 | 96–97 |
| SUBB A, #data | Subtract immediate from A with borrow | 2 | 2 | 94 |
| INC A | Increment A | 1 | 1 | 04 |
| INC Rn | Increment register | 1 | 1 | 08–0F |
| INC direct | Increment direct byte | 2 | 2 | 05 |
| INC @Ri | Increment data memory | 1 | 1 | 06–07 |
| DEC A | Decrement A | 1 | 1 | 14 |
| DEC Rn | Decrement register | 1 | 1 | 18–1F |
| DEC direct | Decrement direct byte | 2 | 2 | 15 |
| DEC @Ri | Decrement data memory | 1 | 1 | 16–17 |
| INC DPTR | Increment data pointer | 1 | 3 | A3 |
| MUL AB | Multiply A by B | 1 | 5 | A4 |
| DIV AB | Divide A by B | 1 | 5 | 84 |
| DA A | Decimal adjust A | 1 | 1 | D4 |
|  | | | | |

Table 10-3 : nRF24E1 Instruction Set, Arithmetic Instructions.

| Logical Instructions | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **Byte** | **Instr. Cycles** | **Hex Code** |
| ANL A, Rn | AND register to A | 1 | 1 | 58–5F |
| ANL A, direct | AND direct byte to A | 2 | 2 | 55 |
| ANL A, @Ri | AND data memory to A | 1 | 1 | 56–57 |
| ANL A, #data | AND immediate to A | 2 | 2 | 54 |
| ANL direct, A | AND A to direct byte | 2 | 2 | 52 |
| ANL direct, #data | AND immediate data to direct byte | 3 | 3 | 53 |
| ORL A, Rn | OR register to A | 1 | 1 | 48–4F |
| ORL A, direct | OR direct byte to A | 2 | 2 | 45 |
| ORL A, @Ri | OR data memory to A | 1 | 1 | 46–47 |
| ORL A, #data | OR immediate to A | 2 | 2 | 44 |
| ORL direct, A | OR A to direct byte | 2 | 2 | 42 |
| ORL direct, #data | OR immediate data to direct byte | 3 | 3 | 43 |
| XRL A, Rn | Exclusive-OR register to A | 1 | 1 | 68–6F |
| XRL A, direct | Exclusive-OR direct byte to A | 2 | 2 | 65 |
| XRL A, @Ri | Exclusive-OR data memory to A | 1 | 1 | 66–67 |
| XRL A, #data | Exclusive-OR immediate to A | 2 | 2 | 64 |
| XRL direct, A | Exclusive-OR A to direct byte | 2 | 2 | 62 |
| XRL direct, #data | Exclusive-OR immediate to direct byte | 3 | 3 | 63 |
| CLR A | Clear A | 1 | 1 | E4 |
| CPL A | Complement A | 1 | 1 | F4 |
| SWAP A | Swap nibbles of A | 1 | 1 | C4 |
| RL A | Rotate A left | 1 | 1 | 23 |
| RLC A | Rotate A left through carry | 1 | 1 | 33 |
| RR A | Rotate A right | 1 | 1 | 03 |
| RRC A | Rotate A right through carry | 1 | 1 | 13 |
| All mnemonics are copyright © Intel Corporation 1980. | | | | |

Table 10-4 : nRF24E1 Instruction Set, Logical Instructions.

| Boolean Instructions | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **Byte** | **Instr. Cycles** | **Hex Code** |
| CLR C | Clear carry | 1 | 1 | C3 |
| CLR bit | Clear direct bit | 2 | 2 | C2 |
| SETB C | Set carry | 1 | 1 | D3 |
| SETB bit | Set direct bit | 2 | 2 | D2 |
| CPL C | Complement carry | 1 | 1 | B3 |
| CPL bit | Complement direct bit | 2 | 2 | B2 |
| ANL C, bit | AND direct bit to carry | 2 | 2 | 82 |
| ANL C, /bit | AND direct bit inverse to carry | 2 | 2 | B0 |
| ORL C, bit | OR direct bit to carry | 2 | 2 | 72 |
| ORL C, /bit | OR direct bit inverse to carry | 2 | 2 | A0 |
| MOV C, bit | Move direct bit to carry | 2 | 2 | A2 |
| MOV bit, C | Move carry to direct bit | 2 | 2 | 92 |
| All mnemonics are copyright © Intel Corporation 1980. | | | | |

Table 10-5 : nRF24E1 Instruction Set, Boolean Instructions.

| Data Transfer Instructions | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **Byte** | **Instr. Cycles** | **Hex Code** |
| MOV A, Rn | Move register to A | 1 | 1 | E8–EF |
| MOV A, direct | Move direct byte to A | 2 | 2 | E5 |
| MOV A, @Ri | Move data memory to A | 1 | 1 | E6–E7 |
| MOV A, #data | Move immediate to A | 2 | 2 | 74 |
| MOV Rn, A | Move A to register | 1 | 1 | F8–FF |
| MOV Rn, direct | Move direct byte to register | 2 | 2 | A8–AF |
| MOV Rn, #data | Move immediate to register | 2 | 2 | 78–7F |
| MOV direct, A | Move A to direct byte | 2 | 2 | F5 |
| MOV direct, Rn | Move register to direct byte | 2 | 2 | 88–8F |
| MOV direct, direct | Move direct byte to direct byte | 3 | 3 | 85 |
| MOV direct, @Ri | Move data memory to direct byte | 2 | 2 | 86–87 |
| MOV direct, #data | Move immediate to direct byte | 3 | 3 | 75 |
| MOV @Ri, A | Move A to data memory | 1 | 1 | F6–F7 |
| MOV @Ri, direct | Move direct byte to data memory | 2 | 2 | A6–A7 |

| | | | | |
|---|---|---|---|---|
| MOV @Ri, #data | Move immediate to data memory | 2 | 2 | 76–77 |
| MOV DPTR, #data | Move immediate to data pointer | 3 | 3 | 90 |
| MOVC A, @A+DPTR | Move code byte relative DPTR to A | 1 | 3 | 93 |
| MOVC A, @A+PC | Move code byte relative PC to A | 1 | 3 | 83 |
| MOVX A, @Ri | Move external data (A8) to A | 1 | 2–9* | E2–E3 |
| MOVX A, @DPTR | Move external data (A16) to A | 1 | 2–9* | E0 |
| MOVX @Ri, A | Move A to external data (A8) | 1 | 2–9* | F2–F3 |
| MOVX @DPTR, A | Move A to external data (A16) | 1 | 2–9* | F0 |
| PUSH direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH A, Rn | Exchange A and register | 1 | 1 | C8–CF |
| XCH A, direct | Exchange A and direct byte | 2 | 2 | C5 |
| XCH A, @Ri | Exchange A and data memory | 1 | 1 | C6–C7 |
| XCHD A, @Ri | Exchange A and data memory nibble | 1 | 1 | D6–D7 |
| All mnemonics are copyright © Intel Corporation 1980. | | | | |

Table 10-6 : nRF24E1 Instruction Set, Data Transfer Instructions.

* Number of cycles is 2 + CKCON.2-0. (CKCON.2-0 is the integer value of the 3LSB of SFR 0x8E CKCON). Default is 3 cycles.

| Branching Instructions | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **Byte** | **Instr. Cycles** | **Hex Code** |
| ACALL addr 11 | Absolute call to subroutine | 2 | 3 | 11–F1 |
| LCALL addr 16 | Long call to subroutine | 3 | 4 | 12 |
| RET | Return from subroutine | 1 | 4 | 22 |
| RETI | Return from interrupt | 1 | 4 | 32 |
| AJMP addr 11 | Absolute jump unconditional | 2 | 3 | 01–E1 |
| LJMP addr 16 | Long jump unconditional | 3 | 4 | 02 |
| SJMP rel | Short jump (relative address) | 2 | 3 | 80 |
| JC rel | Jump on carry = 1 | 2 | 3 | 40 |
| JNC rel | Jump on carry = 0 | 2 | 3 | 50 |
| JB bit, rel | Jump on direct bit = 1 | 3 | 4 | 20 |
| JNB bit, rel | Jump on direct bit = 0 | 3 | 4 | 30 |
| JBC bit, rel | Jump on direct bit = 1 and clear | 3 | 4 | 10 |
| JMP @A+DPTR | Jump indirect relative DPTR | 1 | 3 | 73 |
| JZ rel | Jump on accumulator = 0 | 2 | 3 | 60 |
| JNZ rel | Jump on accumulator /= 0 | 2 | 3 | 70 |
| CJNE A, direct, rel | Compare A, direct JNE relative | 3 | 4 | B5 |
| CJNE A, #d, rel | Compare A, immediate JNE relative | 3 | 4 | B4 |
| CJNE Rn, #d, rel | Compare reg, immediate JNE relative | 3 | 4 | B8–BF |
| CJNE @Ri, #d, rel | Compare ind, immediate JNE relative | 3 | 4 | B6–B7 |
| DJNZ Rn, rel | Decrement register, JNZ relative | 2 | 3 | D8–DF |
| DJNZ direct, rel | Decrement direct byte, JNZ relative | 3 | 4 | D5 |
| All mnemonics are copyright © Intel Corporation 1980. | | | | |

Table 10-7 : nRF24E1 Instruction Set, Branching Instructions.

| Miscellaneous Instructions | | | | |
|---|---|---|---|---|
| **Mnemonic** | **Description** | **Byte** | **Instr. Cycles** | **Hex Code** |
| NOP | No operation | 1 | 1 | 00 |
| There is an additional reserved opcode (A5) that performs the same function as NOP.<br>All mnemonics are copyright © Intel Corporation 1980. | | | | |

Table 10-8 : nRF24E1 Instruction Set, Miscellaneous Instructions.

## 10.4 Instruction Timing

Instruction cycles in the nRF24E1 are four clock cycles in length, as opposed to twelve clock cycles per instruction cycle in the standard 8051. This translates to a 3X improvement in execution time for most instructions. However, some instructions require a different number of instruction cycles on the nRF24E1 than they do on the standard 8051. In the standard 8051, all instructions except for MUL and DIV take one or two instruction cycles to complete. In the nRF24E1 architecture, instructions can take between one and five instruction cycles to complete. For example, in the standard 8051, the instructions MOVX A, @DPTR and MOV direct, direct each take two instruction cycles (twenty-four clock cycles) to execute. In the nRF24E1 architecture, MOVX A, @DPTR takes two instruction cycles (eight clock cycles) and MOV direct, direct takes three instruction cycles (twelve clock cycles). Both instructions execute faster on the nRF24E1 than they do on the standard 8051, but require different numbers of clock cycles.

For timing of real-time events, use the numbers of instruction cycles from Table 10-3 to Table 10-8 to calculate the timing of software loops. The bytes column of these table indicates the number of memory accesses (bytes) needed to execute the instruction. In most cases, the number of bytes is equal to the number of instruction cycles required to complete the instruction. However, as indicated in Table 10-3, there are some instructions (for example, DIV and MUL) that require a greater number of instruction cycles than memory accesses.By default, the nRF24E1 timer/counters run at twelve clock cycles per increment so that timer-based events have the same timing as with the standard 8051. The timers can be configured to run at four clock cycles per increment to take advantage of the higher speed of the nRF24E1.

## 10.5 Dual Data Pointers

The nRF24E1 employs dual data pointers to accelerate data memory block moves. The standard 8051 data pointer (DPTR) is a 16-bit value used to address external data RAM or peripherals. The nRF24E1 maintains the standard data pointer as DPTR0 at SFR locations 0x82 and 0x83. It is not necessary to modify code to use DPTR0. The nRF24E1 adds a second data pointer (DPTR1) at SFR locations 0x84 and 0x85. The SEL bit in the DPTR Select register, DPS (SFR 0x86), selects the active pointer. When SEL = 0, instructions that use the DPTR will use DPL and DPH. When SEL=1, instructions that use the DPTR will use DPL1 and DPH1. SEL is the bit 0 of SFR location 0x86. No other bits of SFR location 0x86 are used. All DPTR-related instructions use the currently selected data pointer. To switch the active pointer, toggle the SEL bit. The fastest way to do so is to use the increment instruction (INC DPS). This requires only one instruction to switch from a source address to a destination address, saving application code from having to save source and destination addresses when doing a block move.

Using dual data pointers provides significantly increased efficiency when moving large blocks of data.

The SFR locations related to the dual data pointers are:

- 0x82 DPL DPTR0 low byte
- 0x83 DPH DPTR0 high byte
- 0x84 DPL1 DPTR1 low byte

- 0x85 DPH1 DPTR1 high byte
- 0x86 DPS DPTR Select (LSB)

## 10.6 Special Function Registers

The Special Function Registers (SFRs) control several of the features of the nRF24E1. Most of the nRF24E1 SFRs are identical to the standard 8051 SFRs. However, there are additional SFRs that control features that are not available in the standard 8051. Table 10-9 lists the nRF24E1 SFRs and indicates which SFRs are not included in the standard 8051 SFR space. When writing software for the nRF24E1, use equate statements to define the SFRs that are specific to the nRF24E1 and custom peripherals. In Table 10-9, SFR bit positions that contain a 0 or a 1 cannot be written to and, when read, always return the value shown (0 or 1). SFR bit positions that contain "–" are available but not used. Table 10-10 shows the value of each SFR, after power-on reset or a watchdog reset, together with a pointer to a detailled description of each register. Please note that any unused address in the SFR address space is reserved and should not be written to.

Notes to Table 10-9 on next page :
(1) Not part of standard 8051 architecture.
(2) Registers unique to nRF24E1
(3) P0 and P1 differ from standard 8051

| Addr | Register | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|
| 0x80 | P0(3) | Port 0 | | | | | | | |
| 0x81 | SP | Stack pointer | | | | | | | |
| 0x82 | DPL | Data pointer 0, low byte | | | | | | | |
| 0x83 | DPH | Data pointer 0, high byte | | | | | | | |
| 0x84 | DPL1(1) | Data pointer 1, low byte | | | | | | | |
| 0x85 | DPH1(1) | Data pointer 1, high byte | | | | | | | |
| 0x86 | DPS(1) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | SEL |
| 0x87 | PCON | SMOD | - | 1 | 1 | GF1 | GF0 | STOP | IDLE |
| 0x88 | TCON | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
| 0x89 | TMOD | GATE | C/T | M1 | M0 | GATE | C/T | M1 | M0 |
| 0x8A | TL0 | Timer/counter 0 value, low byte | | | | | | | |
| 0x8B | TL1 | Timer/counter 1 value, low byte | | | | | | | |
| 0x8C | TH0 | Timer/counter 0 value, high byte | | | | | | | |
| 0x8D | TH1 | Timer/counter 1 value, high byte | | | | | | | |
| 0x8E | CKCON(1) | - | - | T2M | T1M | T0M | MD2 | MD1 | MD0 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0x8F | SPC_FNC(1) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | WRS |
| 0x90 | P1(3) | - | - | - | - | - | Port 1 bit 2:0 | | |
| 0x91 | EXIF(1) | IE5 | IE4 | IE3 | IE2 | 1 | 0 | 0 | 0 |
| 0x92 | MPAGE(1) | program/data memory page address | | | | | | | |
| 0x94 | P0_DIR(2) | Direction of Port 0 | | | | | | | |
| 0x95 | P0_ALT(2) | Alternate functions of Port 0 | | | | | | | |
| 0x96 | P1_DIR(2) | - | - | - | - | - | Direction of Port 1 | | |
| 0x97 | P1_ALT(2) | - | - | - | - | - | alt.funct.of Port 1 | | |
| 0x98 | SCON | SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |
| 0x99 | SBUF | Serial port data buffer | | | | | | | |
| 0xA0 | RADIO(2) | PWR_UP | DR2/CE | CLK2 | DOUT2 | CS | DR1 | CLK1 | DATA |
| 0xA1 | ADCCON(2) | CSTRTN | ADCRUN | NPD | EXTREF | ADCSEL | | | |
| 0xA2 | ADCDATAH(2) | High bits of ADC result | | | | | | | |
| 0xA3 | ADCDATAL(2) | Low bits of ADC result | | | | - | ADCUF | ADCOF | ADCRNG |
| 0xA4 | ADCSTATIC(2) | DIFFM | SLEEP | CLK8 | ADCBIAS | | ADCRES | | |
| 0xA8 | IE | EA | 0 | ET2 | ES | ET1 | EX1 | ET0 | EX0 |
| 0xA9 | PWMCON(2) | PWM_LENGTH | | PWM_PRESCALE | | | | | |
| 0xAA | PWMDUTY(2) | PWM_DUTY_CYCLE | | | | | | | |
| 0xAB | REGX_MSB(2) | High byte of Watchdog/RTC register | | | | | | | |
| 0xAC | REGX_LSB(2) | Low byte of Watchdog/RTC register | | | | | | | |
| 0xAD | REGX_CTRL(2) | - | - | - | Control of REGX_MSB and REGX_LSB | | | | |
| 0xB1 | RSTREAS(2) | - | - | - | - | - | - | RFLR | |
| 0xB2 | SPI_DATA(2) | SPI_DATA input/output bits | | | | | | | |
| 0xB3 | SPI_CTRL(2) | - | - | - | - | - | - | SPI_CTRL | |
| 0xB4 | SPICLK(2) | - | - | - | - | - | - | SPICLK | |
| 0xB5 | TICK_DV(2) | TICK_DV | | | | | | | |
| 0xB6 | CK_CTRL(2) | - | - | - | - | - | - | CK_CTRL | |
| 0xB8 | IP | 1 | 0 | PT2 | PS | PT1 | PX1 | PT0 | PX0 |
| 0xC8 | T2CON | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 |
| 0xCA | RCAP2L | Timer/counter 2 capture or reload, low byte | | | | | | | |
| 0xCB | RCAP2H | Timer/counter 2 capture or reload, high byte | | | | | | | |
| 0xCC | TL2 | Timer/counter 2 value, low byte | | | | | | | |
| 0xCD | TH2 | Timer/counter 2 value, high byte | | | | | | | |
| 0xD0 | PSW | CY | AC | F0 | RS1 | RS0 | OV | F1 | P |
| 0xD8 | EICON(1) | - | 1 | 0 | 0 | WDTI | 0 | 0 | 0 |
| 0xE0 | ACC | Accumulator register | | | | | | | |
| 0xE8 | EIE(1) | 1 | 1 | 1 | EWDI | EX5 | EX4 | EX3 | EX2 |
| 0xF0 | B | B-register | | | | | | | |
| 0xF8 | EIP(1) | 1 | 1 | 1 | PWDI | PX5 | PX4 | PX3 | PX2 |
| 0xFE | HWREV | Device hardware revision number | | | | | | | |
| 0xFF | ----- | Reserved, do not use | | | | | | | |

Table 10-9 **:** Special Function Registers summary

| Register | Addr | Reset value | Description |
|---|---|---|---|
| ACC | 0xE0 | 0x00 | Accumulator register |
| ADCCON | 0xA1 | 0x80 | Table 5-1, page 41 |
| ADCDATAH | 0xA2 | read only | Table 5-3, page 42 |
| ADCDATAL | 0xA3 | read only | Table 5-3, page 42 |
| ADCSTATIC | 0xA4 | 0x0A | Table 5-2, page 42 |
| B | 0xF0 | 0x00 | B-register |
| CK_CTRL | 0xB6 | 0x00 | Table 9-2, page 58 |
| CKCON | 0x8E | 0x01 | Table 10-15, page 78 |
| DPH | 0x83 | 0x00 | ch.10.5, page 68 |
| DPH1 | 0x85 | 0x00 | ch.10.5, page 68 |
| DPL | 0x82 | 0x00 | ch.10.5, page 68 |
| DPL1 | 0x84 | 0x00 | ch.10.5, page 68 |
| DPS | 0x86 | 0x00 | ch.10.5, page 68 |
| EICON | 0xD8 | 0x40 | Table 7-5, page 49 |
| EIE | 0xE8 | 0xE0 | Table 7-6, page 49 |
| EIP | 0xF8 | 0xE0 | Table 7-7, page 50 |
| EXIF | 0x91 | 0x08 | Table 7-4, page 49 |
| HWREV | 0xFE | 0x00,read only | hardware revision no |
| IE | 0xA8 | 0x00 | Table 7-2, page 48 |
| IP | 0xB8 | 0x80 | Table 7-3, page 48 |
| MPAGE | 0x92 | 0x00 | ch.10.1.1.1, page 60 |
| P0 | 0x80 | 0xFF | Table 3-3, page 14 |
| P0_ALT | 0x95 | 0x00 | Table 3-3, page 14 |
| P0_DIR | 0x94 | 0xFF | Table 3-3, page 14 |
| P1 | 0x90 | 0xFF | Table 1-1, page 15 |
| P1_ALT | 0x97 | 0x00 | Table 3-5, page 15 |
| P1_DIR | 0x96 | 0xFF | Table 3-5, page 15 |
| PCON | 0x87 | 0x30 | Table 9-1, page 57 |
| PSW | 0xD0 | 0x00 | Table 10-11, page 72 |
| PWMCON | 0xA9 | 0x00 | Table 6-1, page 46 |
| PWMDUTY | 0xAA | 0x00 | Table 6-1, page 46 |
| RADIO | 0xA0 | 0x80 | Table 4-2, page 19 |
| RCAP2H | 0xCB | 0x00 | ch.10.8.3.3, page 80 |
| RCAP2L | 0xCA | 0x00 | ch.10.8.3.3, page 80 |
| REGX_CTRL | 0xAD | 0x00 | Table 8-2, page 56 |
| REGX_LSB | 0xAC | 0x00 | Table 8-2, page 56 |
| REGX_MSB | 0xAB | 0x00 | Table 8-2, page 56 |
| RSTREAS | 0xB1 | 0x02 | Table 8-3, page 57 |
| SBUF | 0x99 | 0x00 | ch.10.9, page 82 |
| SCON | 0x98 | 0x00 | Table 10-19, page 83 |
| SP | 0x81 | 0x07 | Stack pointer |
| SPC_FNC | 0x8F | 0x00 | do not use |
| SPI_CTRL | 0xB3 | 0x00 | Table 3-6, page 16 |
| SPI_DATA | 0xB2 | 0x00 | Table 3-6, page 16 |
| SPICLK | 0xB4 | 0x00 | Table 3-6, page 16 |
| T2CON | 0xC8 | 0x00 | Table 10-16, page 79 |
| TCON | 0x88 | 0x00 | Table 10-14, page 75 |
| TH0 | 0x8C | 0x00 | ch.10.8, page 74 |
| TH1 | 0x8D | 0x00 | ch.10.8, page 74 |
| TH2 | 0xCD | 0x00 | ch.10.8, page 74 |
| TICK_DV | 0xB5 | 0x1D | Table 8-1, page 53 |
| TL0 | 0x8A | 0x00 | ch.10.8, page 74 |
| TL1 | 0x8B | 0x00 | ch.10.8, page 74 |
| TL2 | 0xCC | 0x00 | ch.10.8, page 74 |
| TMOD | 0x89 | 0x00 | Table 10-13, page 75 |

Table 10-10 **:** Special Function Register reset values and description, alphabetically.

Table 10-11 lists the functions of the bits in the PSW register.

| Bit | Function |
|---|---|
| PSW.7 | CY - Carry flag. Set to 1 when last arithmetic operation resulted in a carry (during addition) or borrow (during subtraction); otherwise cleared to 0 by all arithmetic operations. |
| PSW.6 | AC - Auxiliary carry flag. Set to 1 when last arithmetic operation resulted in a carry into (during addition) or borrow from (during subtraction) the high-order nibble; otherwise cleared to 0 by all arithmetic operations. |
| PSW.5 | F0 - User flag 0. Bit-addressable, general purpose flag for software control. |
| PSW.4 | RS1 - Register bank select bit 1. Used with RS0 to select a register blank in internal RAM. |
| PSW.3 | RS0 - Register bank select bit 0, decoded as:<br>RS1 RS0 Bank selected<br>0 0 Register bank 0, addresses 0x00-0x07<br>0 1 Register bank 1, addresses 0x08-0x0F<br>1 0 Register bank 2, addresses 0x10-0x17<br>1 1 Register bank 3, addresses 0x18-0x1F |
| PSW.2 | OV - Overflow flag. Set to 1 when last arithmetic operation resulted in a carry (addition), borrow (subtraction), or overflow (multiply or divide); otherwise cleared to 0 by all arithmetic operations. |
| PSW.1 | F1 - User flag 1. Bit-addressable, general purpose flag for software control. |
| PSW.0 | P - Parity flag. Set to 1 when modulo-2 sum of 8 bits in accumulator is 1 (odd parity); cleared to 0 on even parity. |

Table 10-11 : PSW Register – SFR 0xD0

## 10.7 SFR registers unique to nRF24E1

The table below lists the SFR registers that are unique to nRF24E1 (not part of standard 8051 register map) The registers P0, P1 and RADIO use the addresses for the ports P0, P1 and P2 in a standard 8051. Whereas the functionality of these ports is similar to that of the corresponding ports in standard 8051, it is not identical.

| **Addr SFR** | **R/W** | **#bit** | **Init hex** | **Name** | **Function** |
|---|---|---|---|---|---|
| 80* | R/W | 8 | FF | P0 | Port 0, pins DIO9 to DIO2 |
| 90* | R/W | 8(3) | FF | P1[5] | Port 1, pins DIN0, DI1, DI0 |
| 94 | R/W | 8 | FF | P0_DIR | Direction of each GPIO bit of port 0 |
| 95 | R/W | 8 | 00 | P0_ALT | Select alternate functions for each pin of port 0 |
| 96 | R/W | 8(3) | FF | P1_DIR | Direction for each GPIO bit of port 1 |
| 97 | R/W | 8(3) | 00 | P1_ALT | Select alternate functions for each pin of port 1 |

* This bit addressable register differs in usage from “standard 8051”

5 Only 3 lower bits are meaningful in P1 and corresponding P1_DIR and P1_ALT

| Addr SFR | R/W | #bit | Init hex | Name | Function |
|---|---|---|---|---|---|
| A0* | R/W | 8 | 80 | RADIO | General purpose IO for interface to 2401 radio, for details see ch. 4 nRF2401 2.4GHz TRANSCEIVER SUBSYSTEM |
| A1 | R/W | 8 | 80 | ADCCON | ADC control register |
| A2 | R | 8 | XX | ADCDATAH | High 8 bits of ADC result |
| A3 | R | 8 | XX | ADCDATAL | Low bits of ADC result (if any) and status |
| A4 | R/W | 6 | 0A | ADCSTATIC | Static configuration data for ADC: |
| A9 | R/W | 8 | 0 | PWMCON | PWM control register |
| AA | R/W | 8 | 0 | PWMDUTY | PWM duty cycle |
| AB | R/W | 8 | 0 | REGX_MSB | High part of 16 bit register for interface to Watchdog and RTC |
| AC | R/W | 8 | 0 | REGX_LSB | Low part of 16 bit register for interface to Watchdog and RTC |
| AD | R/W | 5 | 0 | REGX_CTRL | Control of interface to Watchdog and RTC. |
| B1 | R/W | 2 | 02 | RSTREAS | Reset status and control |
| B2 | R/W | 8 | 0 | SPI_DATA | SPI data input/output |
| B3 | R/W | 2 | 0 | SPI_CTRL | 00 -> SPI not used 01 -> connect to P1<br>10 or 11 -> connect to RADIO |
| B4 | R/W | 2 | 0 | SPICLK | Divider from CPU clock to SPI clock |
| B5 | R/W | 8 | 1D | TICK_DV | TICK Divider. |
| B6 | W | 2 | 0 | CK_CTRL | Clock control |
| B7 | R | 4 | 0 | TEST_MODE | Test mode register.<br>This register must always be 0 in normal mode. |
| BC | RW | 8 | # | T1_1V2 | Another 3 test mode registers.<br>Initial values must not be changed. |
| BD | RW | 8 | # | T2_1V2 | |
| BE | RW | 4 | # | DEV_OFFSET | |

Table 10-12 : SFR registers unique to nRF24E1

## 10.8 Timers/Counters

The nRF24E1 includes three timer/counters (Timer 0, Timer 1 and Timer 2). Each timer/counter can operate as either a timer with a clock rate based on the CPU clock , or as an event counter clocked by the t0 pin (Timer 0), t1 pin (Timer 1), or the t2 pin (Timer 2). These pins are alternate function bits of Port 0 and 1 as this : t0 is P0.5, t1 is P0.6 and t2 is P1.0, for details please see ch. 3 I/O PORTS.
Each timer/counter consists of a 16-bit register that is accessible to software as three SFRs: (Table 10-9 **:** Special Function Registers)

- Timer 0 - TL0 and TH0
- Timer 1 - TL1 and TH1
- Timer 2 - TL2 and TH2

### *10.8.1 Timers 0 and 1*

Timers 0 and 1 each operate in four modes, as controlled through the TMOD SFR (Table 10-13) and the TCON SFR (Table 10-14). The four modes are:

- 13-bit timer/counter (mode 0)
- 16-bit timer/counter (mode 1)
- 8-bit counter with auto-reload (mode 2)
- Two 8-bit counters (mode 3, Timer 0 only)

| Bit | Function |
|---|---|
| TMOD.7 | GATE - Timer 1 gate control. When GATE = 1, Timer 1 will clock only when external interrupt INT1_N = 1 and TR1 (TCON.6) = 1. When GATE = 0, Timer 1 will clock only when TR1 = 1, regardless of the state of INT1_N. |
| TMOD.6 | C/T - Counter/Timer select. When C/T = 0, Timer 1 is clocked by CPU_clk/4 or CPU_clk/12, depending on the state of T1M (CKCON.4). When C/T = 1, Timer 1 is clocked by the t1 pin. |
| TMOD.5 | M1 - Timer 1 mode select bit 1. |
| TMOD.4 | M0 - Timer 1 mode select bit 0, decoded as:<br>M1 M0 Mode<br>00 Mode 0 : 13-bit counter<br>01 Mode 1 : 16-bit counter<br>10 Mode 2 : 8-bit counter with auto-reload<br>11 Mode 3 : Two 8-bit counters |
| TMOD.3 | GATE - Timer 0 gate control. When GATE = 1, Timer 0 will clock only when external interrupt INT0_N = 1 and TR0 (TCON.4) = 1. When GATE = 0, Timer 0 will clock only when TR0 = 1, regardless of the state of INT0_N. |
| TMOD.2 | C/T - Counter/Timer select. When C/T = 0, Timer 0 is clocked by CPU_clk/4 or CPU_clk/12, depending on the state of T0M (CKCON.3). When C/T = 1, Timer 0 is clocked by the t0 pin. |
| TMOD.1 | M1 - Timer 0 mode select bit 1. |
| TMOD.0 | M0 - Timer 0 mode select bit 0, decoded as:<br>M1 M0 Mode<br>00 Mode 0 : 13-bit counter<br>01 Mode 1 : 16-bit counter<br>10 Mode 2 : 8-bit counter with auto-reload<br>11 Mode 3 : Two 8-bit counters |

Table 10-13 : TMOD Register – SFR 0x89

| Bit | Function |
|---|---|
| TCON.7 | TF1 - Timer 1 overflow flag. Set to 1 when the Timer 1 count overflows and cleared when the CPU vectors to the interrupt service routine. |
| TCON.6 | TR1 - Timer 1 run control. Set to 1 to enable counting on Timer 1. |
| TCON.5 | TF0 - Timer 0 overflow flag. Set to 1 when the Timer 0 count overflows and cleared when the CPU vectors to the interrupt service routine. |
| TCON.4 | TR0 - Timer 0 run control. Set to 1 to enable counting on Timer 0. |
| TCON.3 | IE1 - Interrupt 1 edge detect. If external interrupt 1 is configured to be edge-sensitive (IT1 = 1), IE1 is set by hardware when a negative edge is detected on the INT1_N external interrupt pin and is automatically cleared when the CPU vectors to the corresponding interrupt service routine. In edge-sensitive mode, IE1 can also be cleared by software. If external interrupt 1 is configured to be level-sensitive (IT1 = 0), IE1 is set when the INT1_N pin is low and cleared when the INT1_N pin is high. In level-sensitive mode, software cannot write to IE1. |
| TCON.2 | IT1 - Interrupt 1 type select. When IT1 = 1, the nRF24E1 detects external interrupt pin INT1_N on the falling edge (edge-sensitive). When IT1 = 0, the nRF24E1 detects INT1_N as a low level (level-sensitive). |
| TCON.1 | IE0 - Interrupt 0 edge detect. If external interrupt 0 is configured to be edge-sensitive (IT0 = 1), IE0 is set by hardware when a negative edge is detected on the INT0_N external interrupt pin and is automatically cleared when the CPU vectors to the corresponding interrupt service routine. In edge-sensitive mode, IE0 can also be cleared by software. If external interrupt 0 is configured to be level-sensitive (IT0 = 0), IE0 is set when the INT0_N pin is low and cleared when the INT0_N pin is high. In level-sensitive mode, software cannot write to IE0. |
| TCON.0 | IT0 - Interrupt 0 type select. When IT0 = 1, the nRF24E1 detects external interrupt INT0_N on the falling edge (edge-sensitive). When IT0 = 0, the nRF24E1 detects INT0_N as a low level (level-sensitive). |

Table 10-14 : TCON Register – SFR 0x88

#### 10.8.1.1 Mode 0

Mode 0 operation, illustrated in Figure 10-2 : Timer 0/1 – Modes 0 and 1, is the same for Timer 0 and Timer 1. In mode 0, the timer is configured as a 13-bit counter that uses bits 0–4 of TL0 (or TL1) and all eight bits of TH0 (or TH1). The timer enable bit (TR0/TR1) in the TCON SFR starts the timer. The C/T bit selects the timer/counter clock source, CPU_clk or t0/t1. The timer counts transitions from the selected source as long as the GATE bit is 0, or the GATE bit is 1 and the corresponding interrupt pin (INT0_N or INT1_N) is deasserted. INT0_N and INT1_N are alternate function bits of Port0, please seeTable 3-1 : Port functions. When the 13-bit count increments from 0x1FFF (all ones), the counter rolls over to all zeros, the TF0 (or TF1) bit is set in the TCON SFR, and the t0_out (or t1_out) pin goes high for one clock cycle. The upper three bits of TL0 (or TL1) are indeterminate in mode 0 and must be masked when the software evaluates the register.

#### 10.8.1.2 Mode 1

Mode 1 operation is the same for Timer 0 and Timer 1. In mode 1, the timer is configured as a 16-bit counter. As illustrated in Figure 10-2 : Timer 0/1 – Modes 0

and 1, all eight bits of the LSB register (TL0 or TL1) are used. The counter rolls over to all zeros when the count increments from 0xFFFF. Otherwise, mode 1 operation is the same as mode 0.

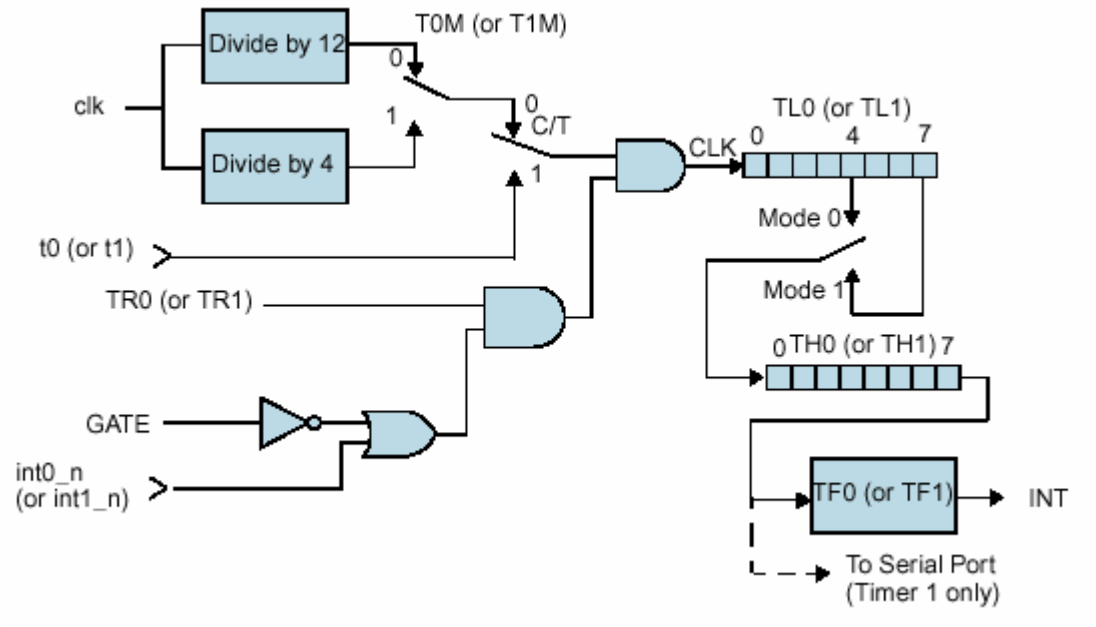



Figure 10-2 : Timer 0/1 – Modes 0 and 1

10.8.1.3 Mode 2

Mode 2 operation is the same for Timer 0 and Timer 1. In mode 2, the timer is configured as an 8-bit counter, with automatic reload of the start value. The LSB register (TL0 or TL1) is the counter, and the MSB register (TH0 or TH1) stores the reload value. As illustrated in Figure 10-3 : Timer 0/1 – Mode 2, mode 2 counter control is the same as for mode 0 and mode 1. However, in mode 2, when TL*n* increments from 0xFF, the value stored in TH*n* is reloaded into TLn.

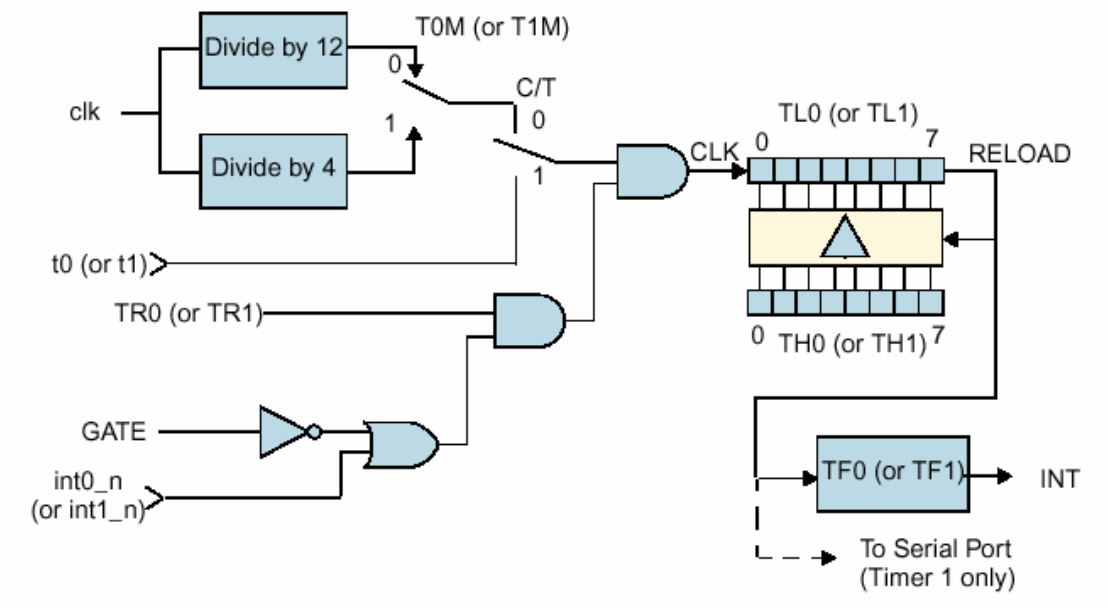



Figure 10-3 : Timer 0/1 – Mode 2

10.8.1.4 Mode 3

In mode 3, Timer 0 operates as two 8-bit counters, and Timer 1 stops counting and holds its value. As shown in Figure 10-4 : Timer 0 – Mode 3, TL0 is configured as an 8-bit counter controlled by the normal Timer 0 control bits. TL0 can count either CPU clock cycles (divided by 4 or by 12) or high-to-low transitions on t0, as determined by the C/T bit. The GATE function can be used to give counter enable control to the INT0_N signal.

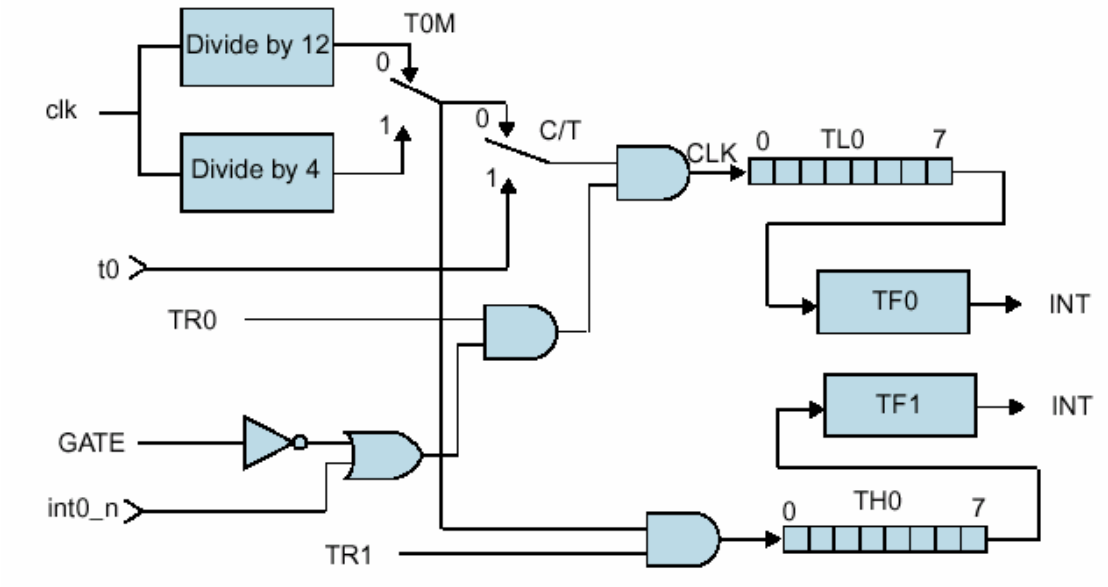



Figure 10-4 : Timer 0 – Mode 3

TH0 functions as an independent 8-bit counter. However, TH0 can count only CPU clock cycles (divided by 4 or by 12). The Timer 1 control and flag bits (TR1 and TF1) are used as the control and flag bits for TH0.
When Timer 0 is in mode 3, Timer 1 has limited usage because Timer 0 uses the Timer 1 control bit (TR1) and interrupt flag (TF1). Timer 1 can still be used for baud rate generation and the Timer 1 count values are still available in the TL1 and TH1 registers.Control of Timer 1 when Timer 0 is in mode 3 is through the Timer 1 mode bits. To turn Timer 1 on, set Timer 1 to mode 0, 1, or 2. To turn Timer 1 off, set it to mode 3. The Timer 1 C/T bit and T1M bit are still available to Timer 1. Therefore, Timer 1 can count CPU_clk/4, CPU_clk/12, or high-to-low transitions on the t1 pin. The Timer 1 GATE function is also available when Timer 0 is in mode 3.

### *10.8.2 Timer Rate Control*

The default timer clock scheme for the nRF24E1 timers is twelve CPU clock cycles per increment, the same as in the standard 8051. However, in the nRF24E1, the instruction cycle is four clock cycles.
Using the default rate (twelve clocks per timer increment) allows existing application code with real-time dependencies, such as baud rate, to operate properly. However, applications that require fast timing can set the timers to increment every four clock cycles by setting bits in the Clock Control register (CKCON) at SFR location 0x8E, described in Table 10-15 : CKCON Register – SFR 0x.
The CKCON bits that control the timer clock rates are:

CKCON bit Counter/Timer
- 5 Timer 2
- 4 Timer 1
- 3 Timer 0

When a CKCON register bit is set to 1, the associated counter increments at four-clock intervals. When a CKCON bit is cleared, the associated counter increments at twelve-clock intervals. The timer controls are independent of each other. The default setting for all three timers is 0; that is, twelve-clock intervals. These bits have no effect in counter mode.

| Bit | Function |
|---|---|
| CKCON.7,6 | Reserved |
| CKCON.5 | T2M – Timer 2 clock select. When T2M = 0, Timer 2 uses CPU_clk/12 (for compatibility with 80C32); when T2M = 1, Timer 2 uses CPU_clk/4. This bit has no effect when Timer 2 is configured for baud rate generation. |
| CKCON.4 | T1M – Timer 1 clock select. When T1M = 0, Timer 1 uses CPU_clk/12 (for compatibility with 80C32); when T1M = 1, Timer 1 uses CPU_clk/4. |
| CKCON.3 | T0M – Timer 0 clock select. When T0M = 0, Timer 0 uses CPU_clk/12 (for compatibility with 80C32); when T0M = 1, Timer 0 uses CPU_clk/4. |
| CKCON.2–0 | MD2, MD1, MD0 – Control the number of cycles to be used for external MOVX instructions; number of cycles is 2 + { MD2, MD1, MD0} |

Table 10-15 : CKCON Register – SFR 0x8E,
default initial data value is 0x01, i.e. MOVX takes 3 cycles.

### *10.8.3 Timer 2*

Timer 2 runs only in 16-bit mode and offers several capabilities not available with Timers 0 and 1. The modes available with Timer 2 are:

- 16-bit timer/counter
- 16-bit timer with capture
- 16-bit auto-reload timer/counter
- Baud-rate generator

The SFRs associated with Timer 2 are:

- T2CON – SFR 0xC8; refer to Table 10-16 : T2CON Register – SFR 0x
- RCAP2L – SFR 0xCA – Used to capture the TL2 value when Timer 2 is configured for capture mode, or as the LSB of the 16-bit reload value when Timer 2 is configured for auto-reload mode.
- RCAP2H – SFR 0xCB – Used to capture the TH2 value when Timer 2 is configured for capture mode, or as the MSB of the 16-bit reload value when Timer 2 is configured for auto-reload mode.
- TL2 – SFR 0xCC – Lower eight bits of the 16-bit count.
- TH2 – SFR 0xCD – Upper eight bits of the 16-bit count.

| Bit | Function |
|---|---|
| T2CON.7 | TF2 - Timer 2 overflow flag. Hardware will set TF2 when Timer 2 overflows from 0xFFFF. TF2 must be cleared to 0 by the software. TF2 will only be set to a 1 if RCLK and TCLK are both cleared to 0. Writing a 1 to TF2 forces a Timer 2 interrupt if enabled. |
| T2CON.6 | EXF2 - Timer 2 external flag. Hardware will set EXF2 when a reload or capture is caused by a high-to-low transition on the t2ex pin, and EXEN2 is set. EXF2 must be cleared to 0 by the software. Writing a 1 to EXF2 forces a Timer 2 interrupt if enabled. |
| T2CON.5 | RCLK - Receive clock flag. Determines whether Timer 1 or Timer 2 is used for Serial port timing of received data in serial mode 1 or 3. RCLK = 1 selects Timer 2 overflow as the receive clock. RCLK = 0 selects Timer 1 overflow as the receive clock. |
| T2CON.4 | TCLK - Transmit clock flag. Determines whether Timer 1 or Timer 2 is used for Serial port timing of transmit data in serial mode 1 or 3. TCLK =1 selects Timer 2 overflow as the transmit clock. TCLK = 0 selects Timer 1 overflow as the transmit clock. |
| T2CON.3 | EXEN2 - Timer 2 external enable. EXEN2 = 1 enables capture or reload to occur as a result of a high-to-low transition on t2ex, if Timer 2 is not generating baud rates for the serial port. EXEN2 = 0 causes Timer 2 to ignore all external events at t2ex. |
| T2CON.2 | TR2 - Timer 2 run control flag. TR2 = 1 starts Timer 2. TR2 = 0 stops Timer 2. |
| T2CON.1 | C/T2 - Counter/timer select. C/T2 = 0 selects a timer function for Timer 2. C/T2 = 1 selects a counter of falling transitions on the t2 pin. When used as a timer, Timer 2 runs at four clocks per increment or twelve clocks per increment as programmed by CKCON.5, in all modes except baud-rate generator mode. When used in baud-rate generator mode, Timer 2 runs at two clocks per increment, independent of the state of CKCON.5. |
| T2CON.0 | CP/RL2 - Capture/reload flag. When CP/RL2 = 1, Timer 2 captures occur on high-to-low transitions of t2ex, if EXEN2 = 1. When CP/RL2 = 0, auto-reloads occur when Timer 2 overflows or when high-to-low transitions occur on t2ex, if EXEN2 = 1. If either RCLK or TCLK is set to 1, CP/RL2 will not function, and Timer 2 will operate in auto-reload mode following each overflow. |

Table 10-16 : T2CON Register – SFR 0xC8

10.8.3.1 Timer 2 Mode Control

Table 10-17 summarizes how the SFR bits determine the Timer 2 mode.

| RCLK | TCLK | CP/RL2 | TR2 | Mode |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 16-bit timer/counter with capture |
| 0 | 0 | 0 | 1 | 16-bit timer/counter with auto-reload |
| 1 | X | X | 1 | Baud-rate generator |
| X | 1 | X | 1 | Baud-rate generator |
| X | X | X | 0 | Off |

Table 10-17 : Timer 2 Mode Control Summary

10.8.3.2 16-Bit Timer/Counter Mode

Figure 10-5 : Timer 2 – Timer/Counter with Capture illustrates how Timer 2 operates in timer/counter mode with the optional capture feature. The C/T2 bit determines whether the 16-bit counter counts clock cycles (divided by 4 or 12), or high-to-low transitions on the t2 pin. The TR2 bit enables the counter. When the count increments from 0xFFFF, the TF2 flag is set, and t2_out goes high for one clock cycle.

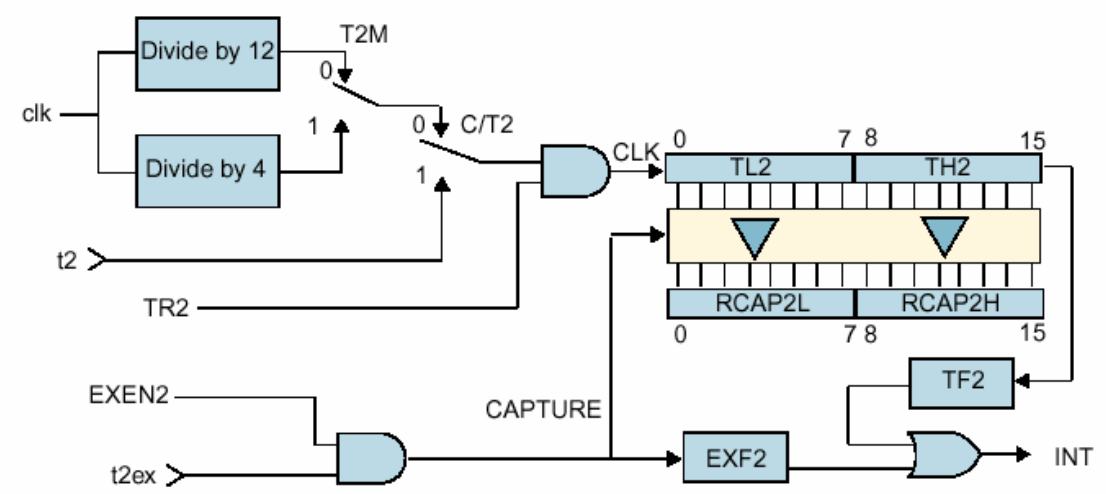



Figure 10-5 : Timer 2 – Timer/Counter with Capture

10.8.3.3 16-Bit Timer/Counter Mode with Capture

The Timer 2 capture mode, illustrated in Figure 10-5 : Timer 2 – Timer/Counter with Capture, is the same as the 16-bit timer/counter mode, with the addition of the capture registers and control signals. The CP/RL2 bit in the T2CON SFR enables the capture feature. When CP/RL2 = 1, a high-to-low transition on t2ex when EXEN2 = 1 causes the Timer 2 value to be loaded into the capture registers (RCAP2L and RCAP2H).

10.8.3.4 16-Bit Timer/Counter Mode with Auto-Reload

When CP/RL2 = 0, Timer 2 is configured for the auto-reload mode illustrated in Figure 10-6 : Timer 2 – Timer/Counter with Auto-Reload. Control of counter input is the same as for the other 16-bit counter modes. When the count increments from 0xFFFF, Timer 2 sets the TF2 flag and the starting value is reloaded into TL2 and

TH2. The software must preload the starting value into the RCAP2L and RCAP2H registers.
When Timer 2 is in auto-reload mode, a reload can be forced by a high-to-low transition on the t2ex pin, if enabled by EXEN2 = 1.

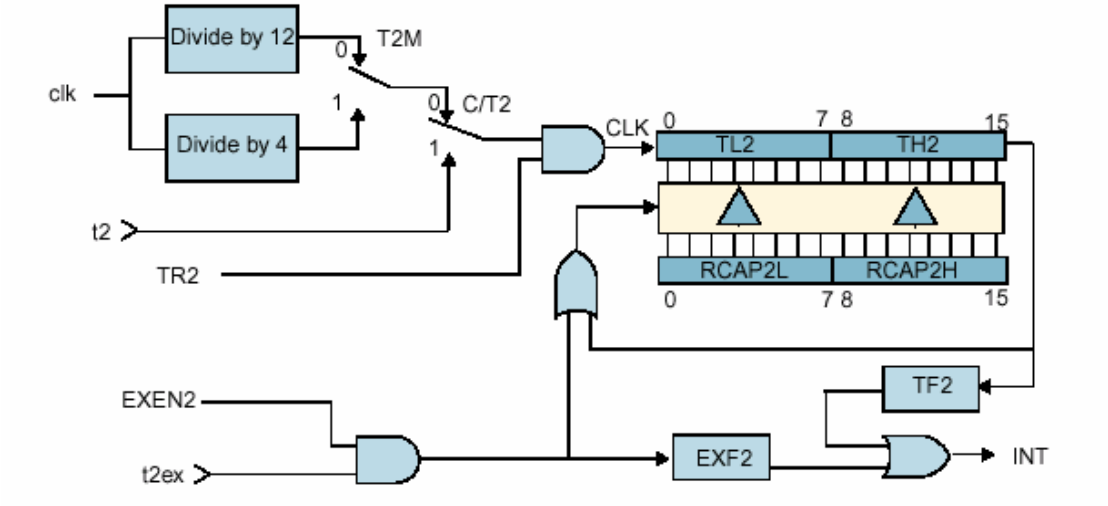



Figure 10-6 : Timer 2 – Timer/Counter with Auto-Reload

10.8.3.5 Baud Rate Generator Mode

Setting either RCLK or TCLK to 1 configures Timer 2 to generate baud rates for Serial port in serial mode 1 or 3. In baud-rate generator mode, Timer 2 functions in auto-reload mode. However, instead of setting the TF2 flag, the counter overflow generates a shift clock for the serial port function. As in normal auto-reload mode, the overflow also causes the preloaded start value in the RCAP2L and RCAP2H registers to be reloaded into the TL2 and TH2 registers.
When either TCLK = 1 or RCLK = 1, Timer 2 is forced into auto-reload operation, regardless of the state of the CP/RL2 bit.
When operating as a baud rate generator, Timer 2 does not set the TF2 bit. In this mode, a Timer 2 interrupt can be generated only by a high-to-low transition on the t2ex pin setting the EXF2 bit, and only if enabled by EXEN2 = 1.The counter time base in baud-rate generator mode is CPU_clk/2. To use an external clock source, set C/T2 to 1 and apply the desired clock source to the t2 pin.

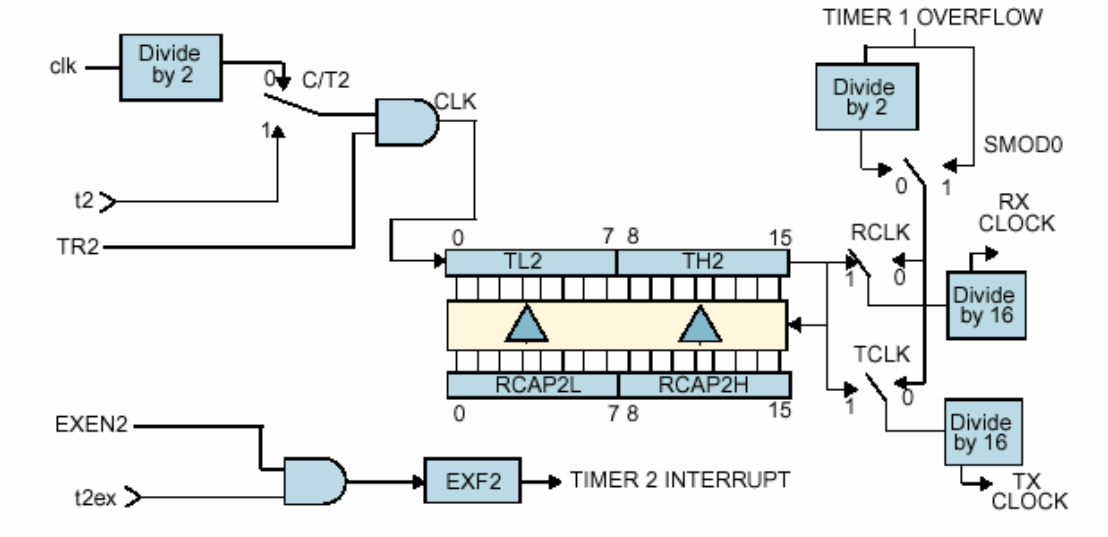



Figure 10-7 : Timer 2 – Baud Rate Generator Mode

## 10.9 Serial Interface

The nRF24E1 is configured with one serial port, which is identical in operation to the standard 8051 serial port. The two serial port pins rxd and txd are available as alternate functions of P0.1 and P0.2, for details please see ch. 3 I/O PORTS.
The serial port can operate in synchronous or asynchronous mode. In synchronous mode, the nRF24E1 generates the serial clock and the serial port operates in half-duplex mode. In asynchronous mode, the serial port operates in full-duplex mode. In all modes, the nRF24E1 buffers receive data in a holding register, enabling the UART to receive an incoming word before the software has read the previous value.
The serial port can operate in one of four modes, as outlined in Table 10-18

| Mode | Sync/A sync | Baud Clock | Data Bits | Start/ Stop | 9th Bit Function |
|---|---|---|---|---|---|
| 0 | Sync | CPU_clk/4 or CPU_clk/12 | 8 | None | None |
| 1 | Async | Timer 1 or Timer 2 | 8 | 1 start, 1 stop | None |
| 2 | Async | CPU_clk/32 or CPU_clk/64 | 9 | 1 start, 1 stop | 0, 1, parity |
| 3 | Async | Timer 1 or Timer 2 | 9 | 1 start, 1 stop | 0, 1, parity |

Table 10-18 : Serial Port Modes

The SFRs associated with the serial port are:
- SCON – SFR 0x98 – Serial port control (Table 10-19)
- SBUF – SFR 0x99 – Serial port buffer

| Bit | Function |
|---|---|
| SCON.7 | SM0 - Serial port mode bit 0. |
| SCON.6 | SM1 - Serial port mode bit 1, decoded as:<br>SM0 SM1 Mode<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 3 |
| SCON.5 | SM2 - Multiprocessor communication enable. In modes 2 and 3, SM2 enables the multiprocessor communication feature. If SM2 = 1 in mode 2 or 3, RI will not be activated if the received 9th bit is 0. If SM2 = 1 in mode 1, RI will be activated only if a valid stop is received. In mode 0, SM2 establishes the baud rate: when SM2 = 0, the baud rate is CPU_clk/12; when<br>SM2 = 1, the baud rate is CPU_clk/4. |
| SCON.4 | REN - Receive enable. When REN = 1, reception is enabled. |
| SCON.3 | TB8 - Defines the state of the 9th data bit transmitted in modes 2 and 3. |
| SCON.2 | RB8 - In modes 2 and 3, RB8 indicates the state of the 9th bit received. In mode 1, RB8 indicates the state of the received stop bit. In mode 0, RB8 is<br>not used. |
| SCON.1 | TI - Transmit interrupt flag. Indicates that the transmit data word has been shifted out. In mode 0, TI is set at the end of the 8th data bit. In all other modes, TI is set when the stop bit is placed on the txd pin. TI must be cleared by the software. |
| SCON.0 | RI – Receive interrupt flag. Indicates that a serial data word has been received. In mode 0, RI is set at the end of the 8th data bit. In mode 1, RI is set after the last sample of the incoming stop bit, subject to the state of SM2.In modes 2 and 3, RI is set at the end of the last sample of RB8. RI must be cleared by the software. |

Table 10-19 : SCON Register – SFR 0x98

### *10.9.1 Mode 0*

Serial mode 0 provides synchronous, half-duplex serial communication. For Serial Port 0, both serial data input and output occur on rxd pin, and txd provides the shift clock for both transmit and receive. The rxd and txd pins are alternate function bits of Port 0, please also see Table 3-2 : Port 0 (P0) functions for port and pin configuration. The lack of open drain ports on nRF24E1 makes it a programmer responsibility to control the direction of the rxd pin.

The serial mode 0 baud rate is either CPU_clk/12 or CPU_clk/4, depending on the state of the SM2. When SM2 = 0, the baud rate is CPU_clk/12; when SM2 = 1, the baud rate is CPU_clk/4.

Mode 0 operation is identical to the standard 8051. Data transmission begins when an instruction writes to the SBUF SFR. The UART shifts the data out, LSB first, at the selected baud rate, until the 8-bit value has been shifted out.

Mode 0 data reception begins when the REN bit is set and the RI bit is cleared in the corresponding SCON SFR. The shift clock is activated and the UART shifts data in on each rising edge of the shift clock until eight bits have been received. One machine cycle after the 8th bit is shifted in, the RI bit is set and reception stops until the software clears the RI bit.

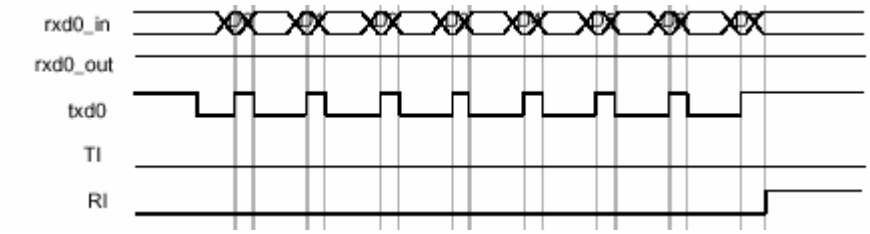



Figure 10-8 : Serial Port Mode 0 receive timing for low-speed (CPU_clk/12) operation.

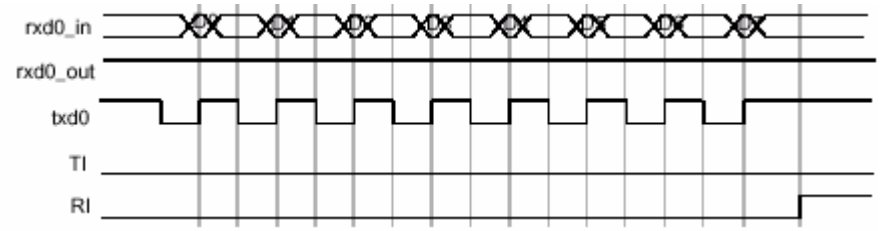



Figure 10-9 : Serial Port Mode 0 receive timing for high-speed (CPU_clk/4) operation

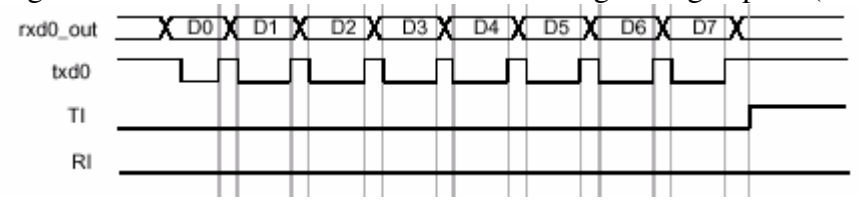



Figure 10-10 : Serial Port Mode 0 transmit timing for high-speed (CPU_clk/4) operation

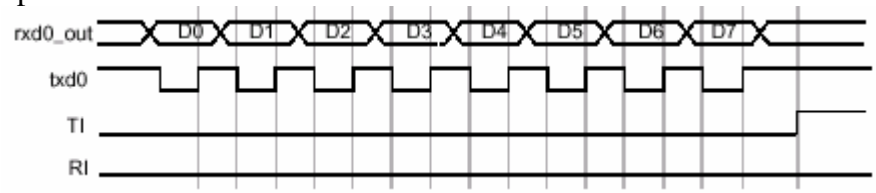



Figure 10-11 : Serial Port Mode 0 transmit timing for high-speed (CPU_clk/4) operation

### *10.9.2 Mode 1*

Mode 1 provides standard asynchronous, full-duplex communication, using a total of ten bits: one start bit, eight data bits, and one stop bit. For receive operations, the stop bit is stored in RB8. Data bits are received and transmitted LSB first.

10.9.2.1 Mode 1 Baud Rate

The mode 1 baud rate is a function of timer overflow. Serial port can use either Timer 1 or Timer 2 to generate baud rates. Each time the timer increments from its maximum count (0xFF for Timer 1 or 0xFFFF for Timer 2), a clock is sent to the baud-rate circuit. The clock is then divided by 16 to generate the baud rate. When using Timer 1, the SMOD bit selects whether or not to divide the Timer 1 rollover rate by 2. Therefore, when using Timer 1, the baud rate is determinedby the equation:

$$\text{Baud Rate} = \frac{2^{SMOD}}{32} \times \text{Timer 1 Overflow}$$

SMOD is SFR bit PCON.7

When using Timer 2, the baud rate is determined by the equation:

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow}}{16}$$

To use Timer 1 as the baud-rate generator, it is best to use Timer 1 mode 2 (8-bit counter with auto-reload), although any counter mode can be used. The Timer 1 reload value is stored in the TH1 register, which makes the complete formula for Timer 1:

$$\text{Baud Rate} = \frac{2^{SMOD}}{32} \times \frac{\text{clk}}{4 \times (256 - \text{TH1})}$$

The 4 in the denominator in the above equation can be obtained by setting the T1M bit in the CKCON SFR. To derive the required TH1 value from a known baud rate (when TM1 = 0), use the equation:

$$\text{TH1} = 256 - \frac{2^{SMOD} * clk}{128 * \text{Baud Rate}}$$

You can also achieve very low serial port baud rates from Timer 1 by enabling the Timer 1 interrupt, configuring Timer 1 to mode 1, and using the Timer 1 interrupt to initiate a 16-bit software reload. Table Table 10-20 lists sample reload values for a variety of common serial port baud rates.

| Desired Baud Rate | SMOD | C/T | Timer 1 Mode | TH1 Value for 16 MHz CPU clk | TH1 Value for 8 MHz CPU clk |
|---|---|---|---|---|---|
| 19.2 Kb/s | 1 | 0 | 2 | 0xF3 | - |
| 9.6 Kb/s | 1 | 0 | 2 | 0xE6 | 0xF3 |
| 4.8 Kb/s | 1 | 0 | 2 | 0XcC | 0xE6 |
| 2.4 Kb/s | 1 | 0 | 2 | 0x98 | 0xCC |
| 1.2 Kb/s | 1 | 0 | 2 | 0x30 | 0x98 |

Table 10-20 : Timer 1 Reload Values for Serial Port Mode 1 Baud Rates

To use Timer 2 as the baud-rate generator, configure Timer 2 in auto-reload mode and set the TCLK and/or RCLK bits in the T2CON SFR. TCLK selects Timer 2 as the baud-rate generator for the transmitter; RCLK selects Timer 2 as the baud-rate generator for the receiver. The 16-bit reload value for Timer 2 is stored in the RCAP2L and RCA2H SFRs, which makes the equation for the Timer 2 baud rate:

$$\text{Baud Rate} = \frac{clk}{32 \times (65536 - \{\text{RCAP2H}, \text{RCAP2L}\})}$$

where RCAP2H,RCAP2L is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned number. The 32 in the denominator is the result of the CPU_clk signal being divided by 2 and the Timer 2 overflow being divided by 16. Setting TCLK or RCLK to 1 automatically causes the CPU_clk signal to be divided by 2, as shown in Figure

10-7 : Timer 2 – Baud Rate Generator Mode, instead of the 4 or 12 determined by the T2M bit in the CKCON SFR.
To derive the required RCAP2H and RCAP2L values from a known baud rate, use the equation:

$$\text{RCAP2H,RCAP2L} = 65536 - \frac{clk}{32 \times \text{Baud Rate}}$$

Table Table 10-21 lists sample values of RCAP2L and RCAP2H for a variety of desired baud rates.

| Baud Rate | C/T2 | 16 MHz CPU clk | |
|---|---|---|---|
| | | RCAP2H | RCAP2L |
| 57.6 Kb/s | 0 | 0xFF | 0xF7 |
| 19.2 Kb/s | 0 | 0xFF | 0xE6 |
| 9.6 Kb/s | 0 | 0xFF | 0xCC |
| 4.8 Kb/s | 0 | 0xFF | 0x98 |
| 2.4 Kb/s | 0 | 0xFF | 0x30 |
| 1.2 Kb/s | 0 | 0xFE | 0x5F |

Table 10-21 : Timer 2 Reload Values for Serial Port Mode 1 Baud Rates

When either RCLK or TCLK is set, the TF2 flag will not be set on a Timer 2 rollover, and the t2ex reload trigger is disabled.

10.9.2.2 Mode 1 Transmit

Figure 10-12 illustrates the mode 1 transmit timing. In mode 1, the UART begins transmitting after the first rollover of the divide-by-16 counter after the software writes to the SBUF register. The UART transmits data on the txd pin in the following order: start bit, eight data bits (LSB first), stop bit. The TI bit is set two clock cycles after the stop bit is transmitted.

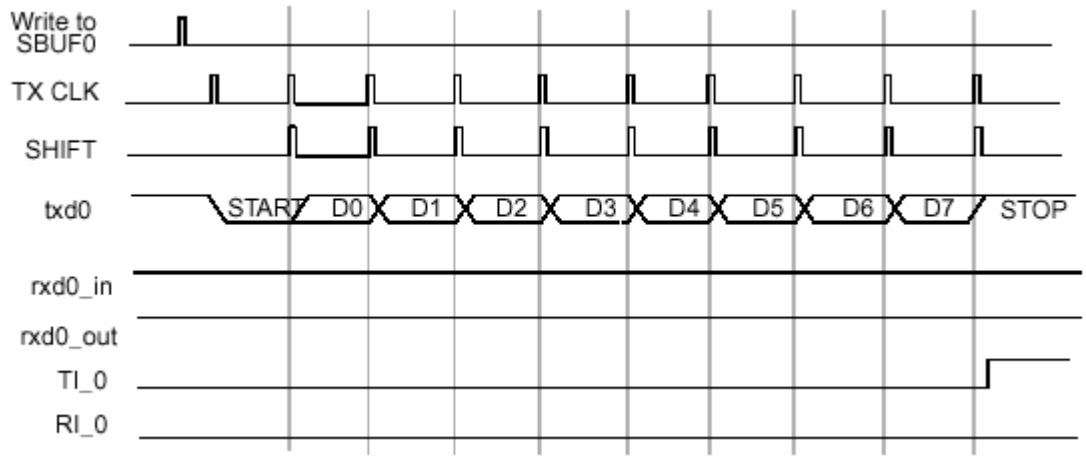



Figure 10-12 : Serial port Mode 1 Transmit Timing

10.9.2.3 Mode 1 Receive

Figure 18 illustrates the mode 1 receive timing. Reception begins at the falling edge of a start bit received on rxd_in, when enabled by the REN bit. For this purpose, rxd_in is sampled sixteen times per bit for any baud rate. When a falling edge of a start bit is detected, the divide-by-16 counter used to generate the receive clock is reset to align the counter rollover to the bit boundaries.

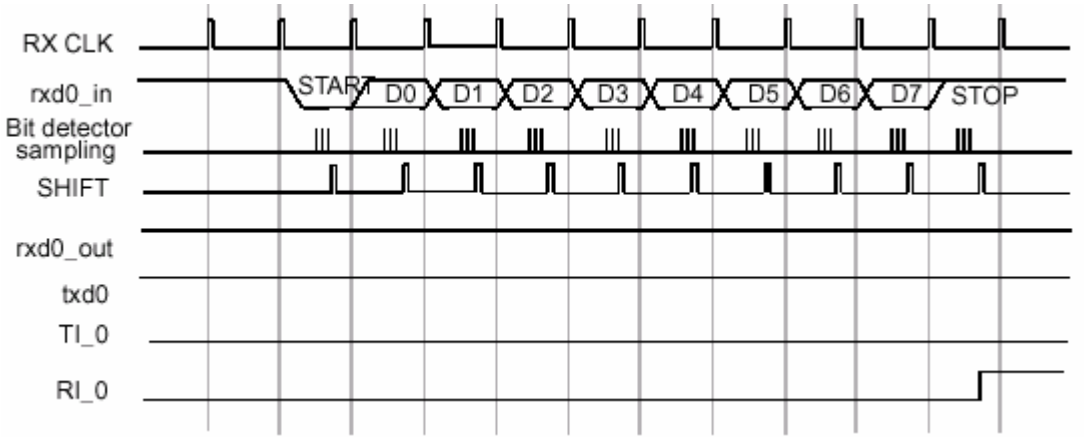



Figure 10-13 : Serial port Mode 1 Receive Timing

For noise rejection, the serial port establishes the content of each received bit by a majority decision of three consecutive samples in the middle of each bit time. This is especially true for the start bit. If the falling edge on rxd_in is not verified by a majority decision of three consecutive samples (low), then the serial port stops reception and waits for another falling edge on rxd_in.
At the middle of the stop bit time, the serial port checks for the following conditions:

- RI = 0
- If SM2 = 1, the state of the stop bit is 1
  (if SM2 = 0, the state of the stop bit does not matter)

If the above conditions are met, the serial port then writes the received byte to the SBUF register, loads the stop bit into RB8, and sets the RI bit. If the above conditions are not met, the received data is lost, the SBUF register and RB8 bit are not loaded, and the RI bit is not set. After the middle of the stop bit time, the serial port waits for another high-to-low transition on the rxd_in pin.
Mode 1 operation is identical to that of the standard 8051 when Timers 1 and 2 use CPU_clk/12 (the default).

### *10.9.3 Mode 2*

Mode 2 provides asynchronous, full-duplex communication, using a total of eleven bits:

- One start bit

- Eight data bits
- One programmable 9th bit
- One stop bit

The data bits are transmitted and received LSB first. For transmission, the 9th bit is determined by the value in TB8. To use the 9th bit as a parity bit, move the value of the P bit (SFR PSW.0) to TB8.

The mode 2 baud rate is either CPU_clk/32 or CPU_clk/64, as determined by the SMOD bit. The formula for the mode 2 baud rate is:

$$\text{Baud Rate} = \frac{2^{SMOD} * clk}{64}$$

Mode 2 operation is identical to the standard 8051.

#### 10.9.3.1 Mode 2 Transmit

Figure 10-14 illustrates the mode 2 transmit timing. Transmission begins after the first rollover of the divide-by-16 counter following a software write to SBUF . The UART shifts data out on the txd pin in the following order: start bit, data bits (LSB first), 9th bit, stop bit. The TI bit is set when the stop bit is placed on the txd pin.

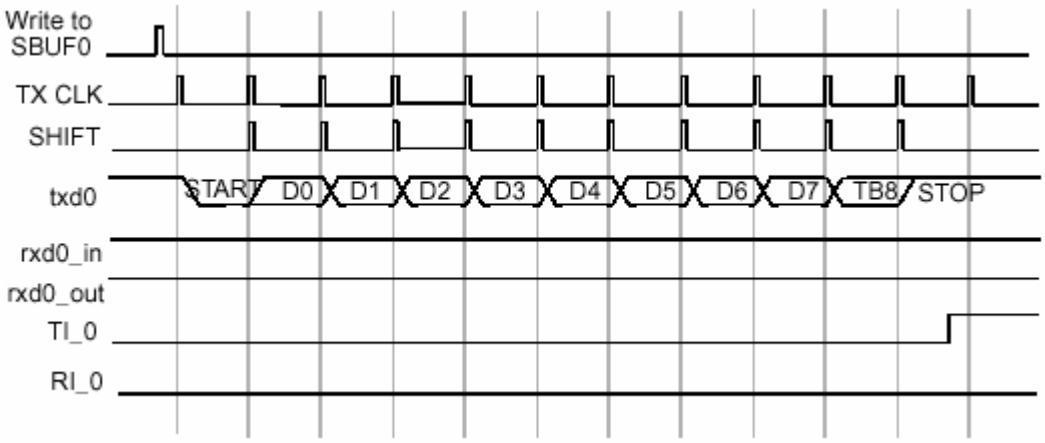



Figure 10-14 : Serial port Mode 2 Transmit Timing

#### 10.9.3.2 Mode 2 Receive

Figure 10-15 illustrates the mode 2 receive timing. Reception begins at the falling edge of a start bit received on rxd_in, when enabled by the REN bit. For this purpose, rxd_in is sampled sixteen times per bit for any baud rate.When a falling edge of a start bit is detected, the divide-by-16 counter used to generate the receive clock is reset to align the counter rollover to the bit boundaries.

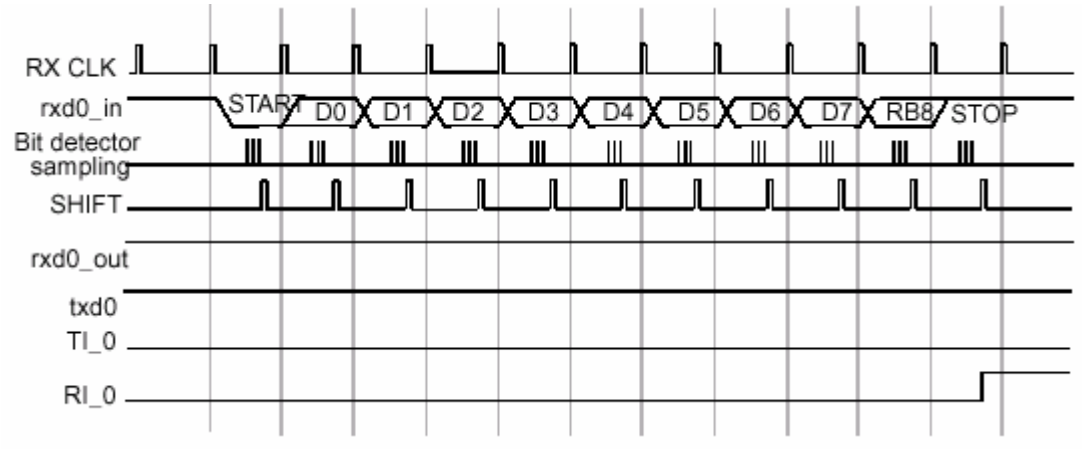



Figure 10-15 : Serial port Mode 2 Receive Timing

For noise rejection, the serial port establishes the content of each received bit by a majority decision of three consecutive samples in the middle of each bit time. This is especially true for the start bit. If the falling edge on rxd_in is not verified by a majority decision of three consecutive samples (low), then the serial port stops reception and waits for another falling edge on rxd_in.

At the middle of the stop bit time, the serial port checks for the following conditions:

- RI = 0
- If SM2 = 1, the state of the stop bit is 1
  (if SM2 = 0, the state of the stop bit does not matter)

If the above conditions are met, the serial port then writes the received byte to the SBUF register, loads the 9th received bit into RB8, and sets the RI bit. If the above conditions are not met, the received data is lost, the SBUF register and RB8 bit are not loaded, and the RI bit is not set. After the middle of the stop bit time, the serial port waits for another high-to-low transition on the rxd_in.

### *10.9.4 Mode 3*

Mode 3 provides asynchronous, full-duplex communication, using a total of eleven bits:

- One start bit
- Eight data bits
- One programmable 9th bit
- One stop bit; the data bits are transmitted and received LSB first

The mode 3 transmit and receive operations are identical to mode 2. The mode 3 baud rate generation is identical to mode 1. That is, mode 3 is a combination of mode 2 protocol and mode 1 baud rate. Figure 10-16 illustrates the mode 3 transmit timing. Mode 3 operation is identical to that of the standard 8051 when Timers 1 and 2 use CPU_clk/12 (the default).

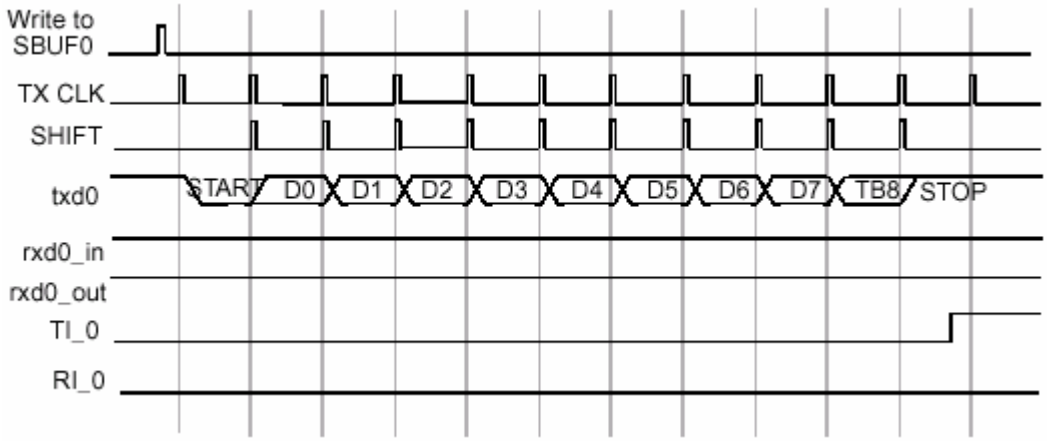



Figure 10-16 : Serial port Mode 3 Transmit Timing

Figure 10-17 illustrates the mode 3 receive timing.

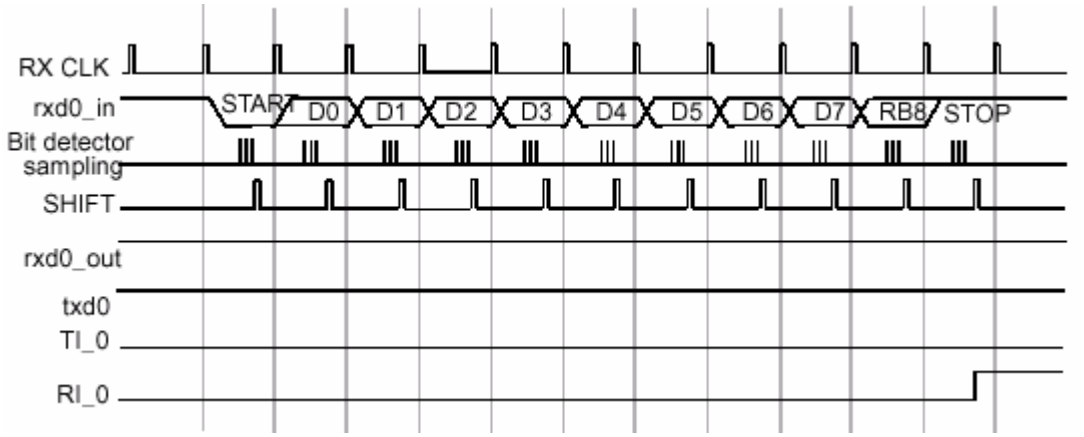



Figure 10-17 : Serial port Mode 3 Receive Timing

### *10.9.5 Multiprocessor Communications*

The multiprocessor communication feature is enabled in modes 2 and 3 when the SM2 bit is set in the SCON SFR for a serial port. In multiprocessor communication mode, the 9th bit received is stored in RB8 and, after the stop bit is received, the serial port interrupt is activated only if RB8 = 1. A typical use for the multiprocessor communication feature is when a master wants to send a block of data to one of several slaves. The master first transmits an address byte that identifies the target slave. When transmitting an address byte, the master sets the 9$^{th}$ bit to 1; for data bytes, the 9th bit is 0.

When SM2 = 1, no slave will be interrupted by a data byte. However, an address byte interrupts all slaves so that each slave can examine the received address byte to determine whether that slave is being addressed. Address decoding must be done by software during the interrupt service routine. The addressed slave clears its SM2 bit and prepares to receive the data bytes. The slaves that are not being addressed leave the SM2 bit set and ignore the incoming data bytes.

# 11 ELECTRICAL SPECIFICATIONS

Conditions: VDD = +3V, VSS = 0V, $T_A$ = - 40ºC to + 85ºC

| Symbol | Parameter (condition) | Notes | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| | **Operating conditions** | | | | | |
| VDD | Supply voltage | | 1.9 | 3.0 | 3.6 | V |
| TEMP | Operating Temperature | | -40 | +27 | +85 | ºC |
| | **Digital input pin** | | | | | |
| $V_{IH}$ | HIGH level input voltage | | VDD- 0.3 | | VDD | V |
| $V_{IL}$ | LOW level input voltage | | VSS | | 0.3 | V |
| Ci | input capacitance | | | 0.55 | | pF |
| $Ii_L$ | input leakage current | | | 0.08 | | nA |
| | **Digital output pin** | | | | | |
| $V_{OH}$ | HIGH level output voltage ($I_{OH}$=0.5mA) | | VDD- 0.3 | | VDD | V |
| $V_{OL}$ | LOW level output voltage ($I_{OL}$=-0.5mA) | | VSS | | 0.3 | V |
| | **Microcontroller** | | | | | |
| $f_{XTAL}$ | Crystal frequency | 2) | 4 | | 20 | MHz |
| $f_{LP\_OSC}$ | Low power RC oscillator frequency | I | 1 | | 5.5 | KHz |
| $I_{VDD\_MCU}$ | Supply current @16MHz @3V | | | 3 | | mA |
| $I_{VDD\_pwd}$ | Average Supply current in power down | | | 2 | | µA |
| | **General RF conditions** | | | | | |
| $f_{OP}$ | Operating frequency | 1) | 2400 | | 2524 | MHz |
| Δf | Frequency deviation | | | ±156 | | kHz |
| $R_{GFSK}$ | Data rate ShockBurst™ | | >0 | | 1000 | kbps |
| $F_{CHANNEL}$ | Channel spacing | | | 1 | | MHz |
| | **Transmitter operation** | | | | | |
| $P_{RF}$ | Maximum Output Power | 4) | | 0 | +4 | dBm |
| $P_{RFC}$ | RF Power Control Range | | 16 | 20 | | dB |
| $P_{RFCR}$ | RF Power Control Range Resolution | | | | ±3 | dB |
| $P_{BW}$ | 20dB Bandwidth for Modulated Carrier | | | | 1000 | kHz |
| $P_{RF2}$ | 2nd Adjacent Channel Transmit Power 2MHz | | | | -20 | dBm |
| $P_{RF3}$ | 3rd Adjacent Channel Transmit Power 3MHz | | | | -40 | dBm |
| $I_{VDD\_TX0}$ | Supply current @ 0dBm output power | 5) | | 13 | | mA |
| $I_{VDD\_TX20}$ | Supply current @ -20dBm output power | 5) | | 9 | | mA |
| | **Receiver operation** | | | | | |
| $I_{VDD\_RX}$ | Supply current one receiver @250kbps | 3) | | 18 | | mA |
| $I_{VDD\_RX}$ | Supply current one receiver @1000kbps | 3) | | 19 | | mA |
| $I_{VDD\_RX2}$ | Supply current two receivers @250kbps | 3) | | 23 | | mA |
| $I_{VDD\_RX2}$ | Supply current two receivers @1000kbps | 3) | | 25 | | mA |
| $RX_{SENS}$ | Sensitivity at 0.1%BER (@250kbps) | | | -90 | | dBm |
| $RX_{SENS}$ | Sensitivity at 0.1%BER (@1000kbps) | | | -80 | | dBm |
| $C/I_{CO}$ | C/I Co-channel | 6) | | 10 | | dB |
| $C/I_{1ST}$ | 1st Adjacent Channel Selectivity C/I 1MHz | 6) | | -20 | | dB |
| $C/I_{2ND}$ | 2nd Adjacent Channel Selectivity C/I 2MHz | 6) | | -37 | | dB |
| $C/I_{3RD}$ | 3rd Adjacent Channel Selectivity C/I 3MHz | 6) | | -43 | | dB |
| $RX_B$ | Blocking Data Channel 2 | | | -41 | | dB |
| | | | | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | **ADC operation** | | | | | |
| DNL | Differential Nonlinearity $f_{IN}$ = 0.9991 kHz | I | | ±0.5 | | LSB |
| INL | Integral Nonlinearity $f_{IN}$ = 0.9991 kHz | I | | ±0.75 | | LSB |
| SNR | Signal to Noise Ratio (DC input) | V | | 59 | | dBFS |
| $V_{OS}$ | Midscale offset | I | | ± 1 | | %FS |
| $\varepsilon_G$ | Gain Error | I | | ±1 | | %FS |
| SNR | Signal to Noise Ratio (without harmonics) $f_{IN}$ = 10 kHz | V | 53 | 58 | | dBFS |
| SFDR | Spurious Free Dynamic Range $f_{IN}$ = 10 kHz | V | | 65 | | dB |
| $V_{BG}$ | Internal reference | I | 1.1 | 1.22 | 1.3 | V |
| | Internal reference voltage drift | V | | 100 | | ppm/°C |
| $V_{FS}$ | Reference voltage input (external ref) | I | 0.8 | | 1.5 | V |
| $F_{S6}$ | 6 bit conversion | IV | $f_{XTAL}$ / 160 | | $f_{XTAL}$ / 128 | SPS |
| $F_{S8}$ | 8 bit conversion | IV | $f_{XTAL}$ / 192 | | $f_{XTAL}$ / 160 | SPS |
| $F_{S10}$ | 10 bit conversion | IV | $f_{XTAL}$ / 224 | | $f_{XTAL}$ / 192 | SPS |
| $F_{S12}$ | 12 bit conversion | IV | $f_{XTAL}$ / 256 | | $f_{XTAL}$ / 224 | SPS |
| $I_{ADC}$ | Supply current ADC operation | I | | 1 | | mA |
| $t_{NPD}$ | Start-up time from ADC Power down | I | | 15 | | µs |

NOTES:

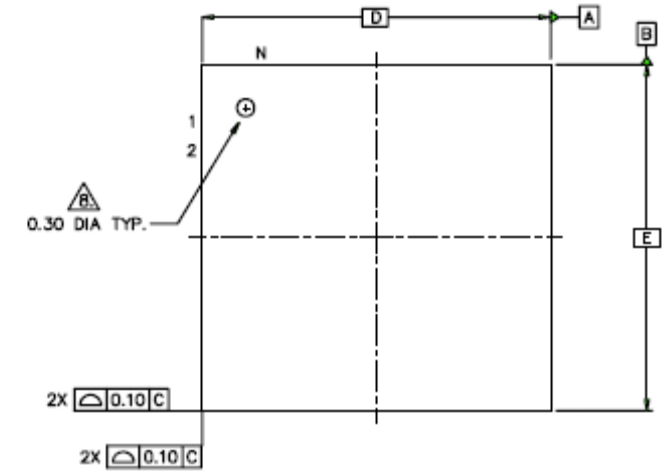
1) Usable band is determined by local regulations
2) The crystal frequency may be chosen from 5 different values (4, 8, 12, 16, and 20MHz) which are specified in the nRF2401 configuration word, please seeTable 14-2 Crystal specification of the nRF24E1. 16MHz is required for 1Mbps operation.
3) Current for nRF2401 RF subsystem only.
4) Antenna load impedance = 100Ω+j175Ω
5) Current for nRF2401 RF subsystem only. Antenna load impedance = 100Ω+j175Ω. Effective data rate 250kbps or 1Mbps.
6) 250kbps.

I ) Test Level I: 100% production tested at +25°C
II ) Test Level II: 100% production tested at +25°C and sample tested at specified temperatures
III ) Test Level III: Sample tested only
IV ) Test Level IV: Parameter is guaranteed by design and characterization testing
V ) Test Level V: Parameter is typical value only
VI) Test Level VI: 100% production tested at +25°C. Guaranteed by design and characterization testing for industrial temperature range

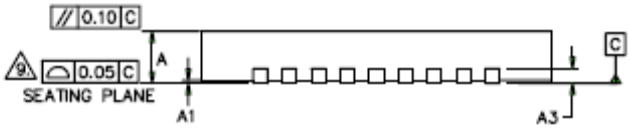
Table 11-1 : nRF24E1 Electrical specifications

# 12 PACKAGE OUTLINE

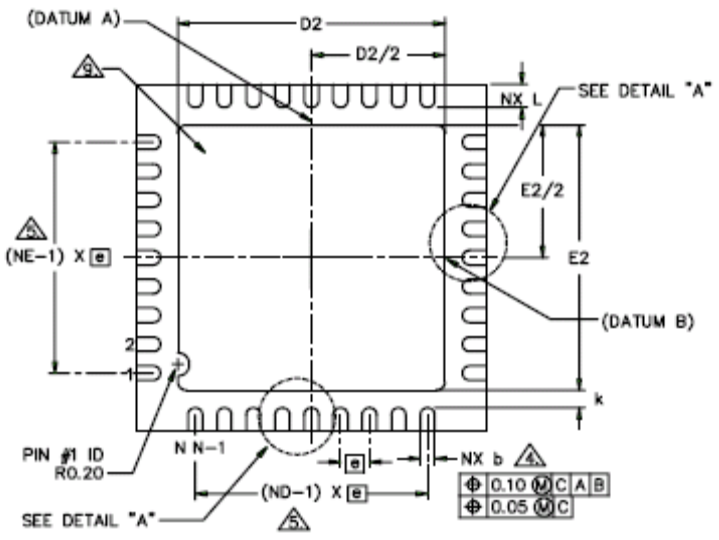
nRF24E1G uses the QFN36 6x6 package, with matt tin plating.



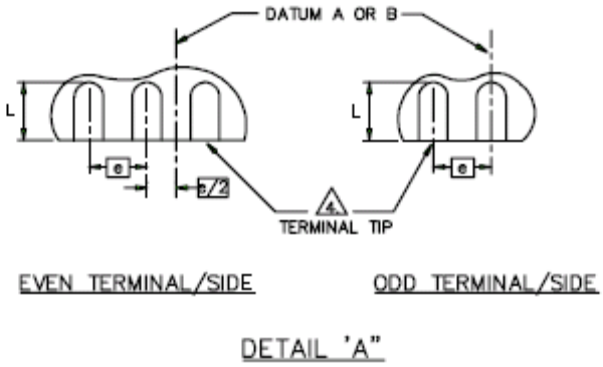

TOP VIEW




SIDE VIEW




BOTTOM VIEW




| Package Type | | A | A1 | A3 | K | D/E | e | D2/E2 | L | b |
|---|---|---|---|---|---|---|---|---|---|---|
| Saw QFN36 | **Min** | 0.80 | 0.00 | | | | | 4.50 | 0.35 | 0.18 |
| (6x6 mm) | **Typ.** | 0.85 | 0.02 | 0.20 | 0.20 | 6.0 BSC[6] | 0.50 | 4.60 | 0.40 | 0.25 |
| | **Max** | 0.90 | 0.05 | REF. | min | | BSC | 4.70 | 0.45 | 0.30 |

Figure 12-1 : nRF24E1 Package outline. Dimensions are in mm.

## 12.1 Package marking:

| | | | | | |
|---|---|---|---|---|---|
| n | R | F | | *B* | *X* |
| 2 | 4 | E | 1 | G | |
| *Y* | *Y* | *W* | *W* | *L* | *L* |

Abbreviations:

B – Build Code, i.e. unique code for production sites, package type and test platform
X – "X" grade, i.e. Engineering Samples (optional)
YY – 2 digit Year number
WW – 2 digit Week number
LL – 2 letter wafer lot number code

[6] BSC: Basic Spacing between Centers, ref. JEDEC standard 95, page 4.17-11/A

# 13 ABSOLUTE MAXIMUM RATINGS

**Supply voltages**
VDD ..... - 0.3V to + 3.6V
VSS ..... 0V

**Input voltage**
For analog pins, AIN0 to AIN7 and AREF:
$V_{IA}$ ..... - 0.3V to 2.0 V
For all other pins:
$V_I$ ..... - 0.3V to VDD + 0.3V

**Output voltage**
$V_O$ ..... - 0.3V to VDD + 0.3V

**Total Power Dissipation**
$P_D$ ($T_A$=85°C) ..... 60mW

**Temperatures**
Operating Temperature…. - 40°C to + 85°C
Storage Temperature……. - 40°C to + 125°C

**Moisture Sensitivity Level**
MSL2@260ºC, three times reflow

*Note: Stress exceeding one or more of the limiting values may cause permanent damage to the device.*

### *13.1.1 ATTENTION!*

Electrostatic Sensitive Device
Observe Precaution for handling.

# 14 Peripheral RF Information

### *14.1.1 Antenna output*

The ANT1 & ANT2 output pins provide a balanced RF output to the antenna. The pins must have a DC path to VDD, either via a RF choke or via the center point in a dipole antenna. The load impedance seen between the ANT1/ANT2 outputs should be in the range 200-700Ω. A load of 100Ω+j175Ω is recommended for maximum output power (0dBm). Lower load impedance (for instance 50 Ω) can be obtained by fitting a simple matching network.

### *14.1.2 Output Power adjustment*

| Power setting bits of configuring word | RF output power | DC current consumption |
|---|---|---|
| 11 | 0 dBm ±3dB | 16.0 mA |
| 10 | -5 dBm ±3dB | 13.5 mA |
| 01 | -10 dBm ±3dB | 12.4 mA |
| 00 | -20 dBm ±3dB | 11.8 mA |

Conditions: VDD = 3.0V, VSS = 0V, $T_A$ = 27ºC, Load impedance = 100Ω+j175Ω.

Table 14-1 RF output power setting for the nRF24E1.

### *14.1.3 Crystal Specification*

Tolerance includes initially accuracy and tolerance over temperature and aging.

| Frequency | $C_L$ | ESR | $C_{0max}$ | Tolerance |
|---|---|---|---|---|
| 4 MHz | 12pF | 150 Ω | 7.0pF | ±30ppm |
| 8 MHz | 12pF | 100 Ω | 7.0pF | ±30ppm |
| 12 MHz | 12pF | 100 Ω | 7.0pF | ±30ppm |
| 16 MHz | 12pF | 100 Ω | 7.0pF | ±30ppm |
| 20 MHz | 12pF | 100 Ω | 7.0pF | ±30ppm |

Table 14-2 Crystal specification of the nRF24E1.

To achieve a crystal oscillator solution with low power consumption and fast start-up time, it is recommended to specify the crystal with a low value of crystal load capacitance. Specifying $C_L$=12pF is OK, but it is possible to use up to 16pF. Specifying a lower value of crystal parallel equivalent capacitance, $C_0$ will also work, but this can increase the price of the crystal itself. Typically $C_0$=1.5pF at a crystal specified for $C_0$_max=7.0pF.
The selected frequency value must also be set into the nRF2401 configuration word, please see Table 4-12 Crystal frequency setting.

## 14.2 PCB layout and de-coupling guidelines

A well-designed PCB is necessary to achieve good RF performance. Keep in mind that a poor layout may lead to loss of performance, or even functionality, if due care is not taken. A fully qualified RF-layout for the nRF24E1 and its surrounding components, including matching networks, can be downloaded from **www.nordicsemi.no**.

A PCB with a minimum of two layers including a ground plane is recommended for optimum performance. The nRF24E1 DC supply voltage should be de-coupled as close as possible to the VDD pins with high performance RF capacitors, see Table 15-1. It is preferable to mount a large surface mount capacitor (e.g. 4.7µF tantalum) in parallel with the smaller value capacitors. The nRF24E1 supply voltage should be filtered and routed separately from the supply voltages of any digital circuitry.

Long power supply lines on the PCB should be avoided. All device grounds, VDD connections and VDD bypass capacitors must be connected as close as possible to the nRF24E1 IC. For a PCB with a topside RF ground plane, the VSS pins should be connected directly to the ground plane. For a PCB with a bottom ground plane, the best technique is to have via holes as close as possible to the VSS pads. One via hole should be used for each VSS pin.

Full swing digital data or control signals should not be routed close to the crystal or the power supply lines.

## 14.3 Reflow information

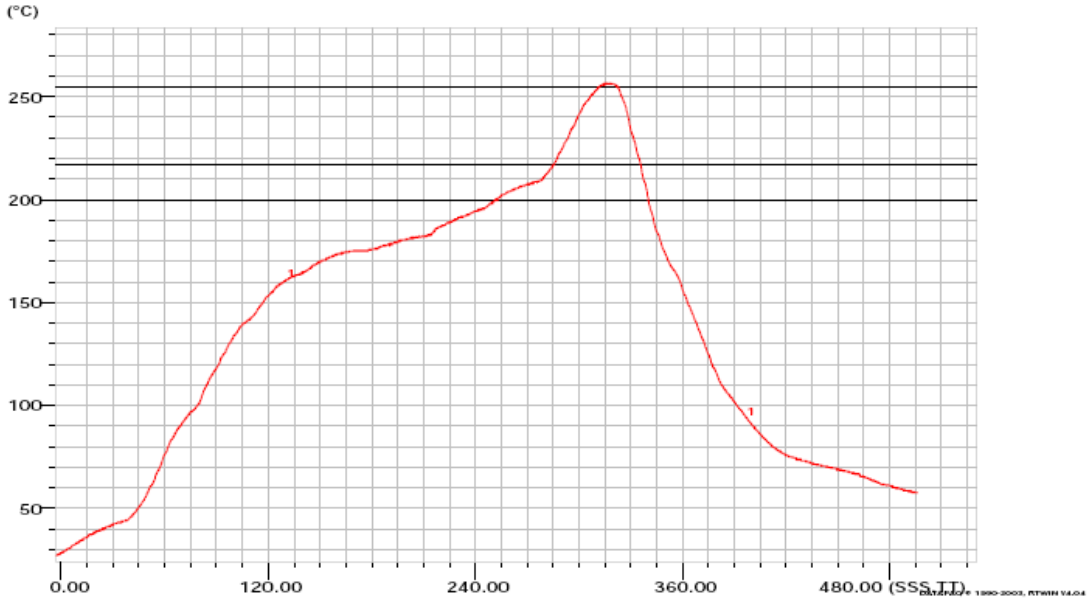



Figure 14-1 Reflow Soldering profile, GREEN

| | |
|---|---|
| Ramp rate (RT-150ºC) | 1.38 ºC/s |
| Pre-heat (150-200ºC) | 134 s |
| Dwell @217ºC | 50 s |
| Dwell @245ºC | 10 s |
| Ramp up | 1.42 ºC/s |
| Ramp down | 2.59 ºC/s |
| Peak temperature | 257 ºC |
| Time from RT to PT | 320 s |

# 15 Application example

## 15.1 nRF24E1 with single ended matching network

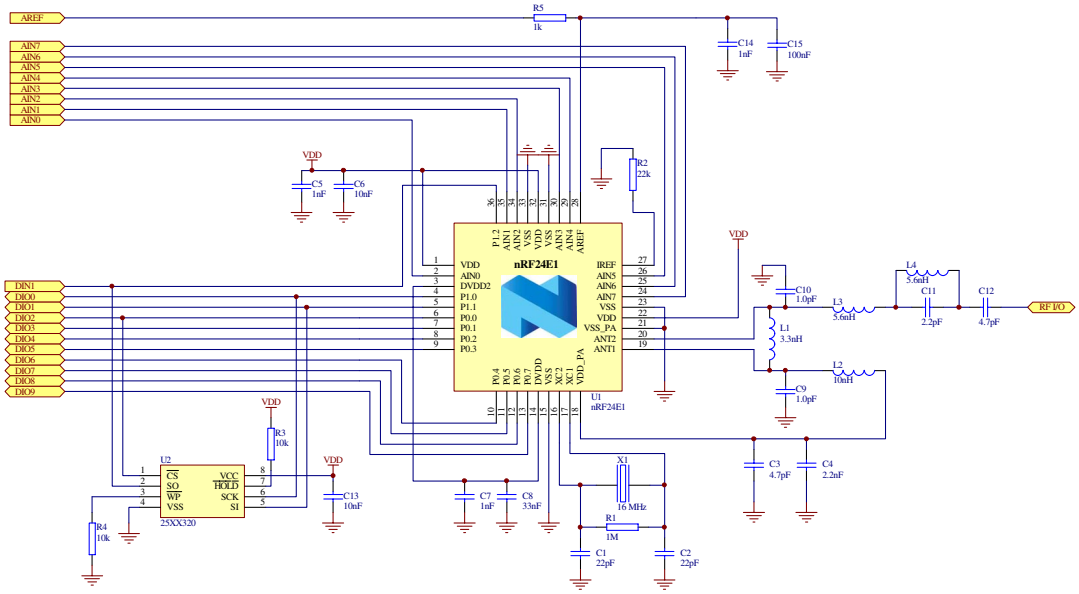



Figure 15-1 nRF24E1 schematic for RF layout with single end 50Ω antenna

| Component | Description | Size | Value | Tolerance | Units |
|---|---|---|---|---|---|
| C1 | Capacitor ceramic, 50V, NPO | 0603 | 22 | ±5% | pF |
| C2 | Capacitor ceramic, 50V, NPO | 0603 | 22 | ±5% | pF |
| C3 | Capacitor ceramic, 50V, NPO | 0603 | 4.7 | ±5% | pF |
| C4 | Capacitor ceramic, 50V, X7R | 0603 | 2.2 | ±10% | nF |
| C5 | Capacitor ceramic, 50V, X7R | 0603 | 1.0 | ±10% | nF |
| C6 | Capacitor ceramic, 50V, X7R | 0603 | 10 | ±10% | nF |
| C7 | Capacitor ceramic, 50V, X7R | 0603 | 1 | ±10% | nF |
| C8 | Capacitor ceramic, 50V, X7R | 0603 | 33 | ±10% | nF |
| C9 | Capacitor ceramic, 50V, NPO | 0603 | 1.0 | ± 0.1 pF | pF |
| C10 | Capacitor ceramic, 50V, NPO | 0603 | 1.0 | ± 0.1 pF | pF |
| C11 | Capacitor ceramic, 50V, NPO | 0603 | 2.2 | ± 0.25 pF | pF |
| C12 | Capacitor ceramic, 50V, NPO | 0603 | 4.7 | ± 0.25 pF | pF |
| C13 | Capacitor ceramic, 50V, X7R | 0603 | 10 | ±10% | nF |
| C14 | Capacitor ceramic, 50V, X7R | 0603 | 1.0 | ±10% | nF |
| C15 | Capacitor ceramic, 50V, X7R | 0603 | 100 | ±10% | nF |
| L1 | Inductor, wire wound [2)] | 0603 | 3.3 | ± 5% | nH |
| L2 | Inductor, wire wound [2)] | 0603 | 10 | ± 5% | nH |
| L3 | Inductor, wire wound [2)] | 0603 | 5.6 | ± 5% | nH |
| L4 | Inductor, wire wound [2)] | 0603 | 5.6 | ± 5% | nH |
| R1 | Resistor | 0603 | 1.0 | ±5% | MΩ |
| R2 | Resistor | 0603 | 22 | ±1% | kΩ |
| R3 | Resistor | 0603 | 10 | ±5% | kΩ |
| R4 | Resistor | 0603 | 10 | ±5% | kΩ |
| R5 | Resistor | 0603 | 1 | ±5% | kΩ |
| U1 | **nRF24E1** transceiver | QFN36 / 6x6 | nRF24E1 | | |
| X1 | Crystal, CL = 12pF,<br>ESR < 100 ohm | LxWxH =<br>4.0x2.5x0.8 | 16[1)] | +/- 30 ppm | MHz |
| U2 | 4 kbyte serial EEPROM with SPI interface | SO8 | 2XX320 | | |

Table 15-1 Recommended components (BOM) in nRF24E1 with antenna matching network

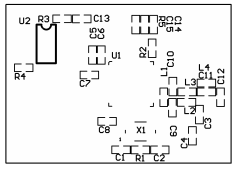
[2)] Wire wound inductors are recommended, other can be used if their self-resonant frequency (SRF) is > 2.7 GHz

[1)] **nRF24E1** can operate at several crystal frequencies, please refer to 96.

## 15.2 PCB layout example

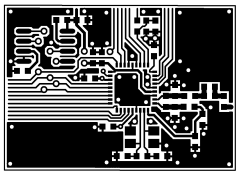
Figure 15-2 shows a PCB layout example for the application schematic in Figure 15-1.

A double-sided FR-4 board of 1.6mm thickness is used. This PCB has a ground plane on the bottom layer. Additionally, there are ground areas on the component side of the board to ensure sufficient grounding of critical components. A large number of via holes connect the top layer ground areas to the bottom layer ground plane.

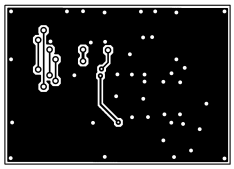


Top silk screen

No components in bottom layer



Top view



Bottom view

Figure 15-2 nRF24E1 RF layout with single ended connection to 50Ω antenna and 0603 size passive components

# 16 Table of Figures

# 17 Table of Tables

# 18 DEFINITIONS

| **Data sheet status** | |
|---|---|
| Objective product specification | This datasheet contains target specifications for product development. |
| Preliminary product specification | This datasheet contains preliminary data; supplementary data may be published from Nordic Semiconductor ASA later. |
| Product specification | This datasheet contains final product specifications. Nordic Semiconductor ASA reserves the right to make changes at any time without notice in order to improve design and supply the best possible product. |
| **Limiting values** | |
| Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Specifications sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability. | |
| **Application information** | |
| Where application information is given, it is advisory and does not form part of the specification. | |

Table 18-1 :Definitions

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function or design. Nordic Semiconductor does not assume any liability arising out of the application or use of any product or circuits described herein.

## LIFE SUPPORT APPLICATIONS

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

Product Specification: Revision Date: 17.03.2006.

Datasheet order code: 170306-nRF24E1.

# YOUR NOTES

# Nordic Semiconductor ASA – World Wide Distributors

**For Your nearest dealer, please see http://www.nordicsemi.no**

**For technical support: http://www.nordicsemi.no/wireless_support**

**Main Office:**
Vestre Rosten 81, N-7075 Tiller, Norway
Phone: +47 72 89 89 00, Fax: +47 72 89 89 89

**Visit the Nordic Semiconductor ASA website at http://www.nordicsemi.no**

# X-ON Electronics

Largest Supplier of Electrical and Electronic Components

***Click to view similar products for*** RF Transceiver ***category:***

***Click to view products by*** Nordic ***manufacturer:***

Other Similar products are found below :

ADRV9026BBCZ ADRV9026-MB/PCBZ ADL6316ACCZ ADL6316ACCZ-R7 ADL6317ACCZ SYN480R VI4455 NRF51822-CEAA-R (E0) ADF7242BCPZ-RL AT86RF232-ZX ADF7021-NBCPZ-RL Si4055-C2A-GM ADRV9008BBCZ-2 ADF7030-1BSTZN-RL AD9874ABSTRL ADF7020-1BCPZ-RL7 ADF7020BCPZ ADF7020BCPZ-RL ADF7021BCPZ ADF7021BCPZ-RL ADF7021BCPZ-RL7 ADF7021-NBCPZ ADF7021-VBCPZ ADF7023-JBCPZ ADF7025BCPZ ADF7241BCPZ ADRV9029BBCZ AT86RF231-ZU AT86RF232-ZXR AT86RF233-ZU ATA8520-GHQW FM11NC08S MC3361BPL-D16-T SX1236IMLTRT HT9170D BGT24MTR11E6327XUMA1 BGT24MTR12E6327XUMA1 MAX2510EEI+ MAX2511EEI+ MAX2510EEI+T MAX2832ETM+ MAX7030LATJ+ SX1212IWLTRT MAX9947ETE+ AT86RF212B-ZU ATA5429-PLSW NRF24LE1-O17Q32-R AT86RF233-ZUR nRF24L01P-R SI4463-C2A-GM