nRF2401+
Design document

July 29
2015

This document describes the embedded software design of the Wireless nRF2401+ module.

Version 1.0

Revision History

Date	Version	Description	Author	Role
29/7/2015	1.0	Document creation	Ibrahim Mostafa	Junior ES Engineer

Table of Contents

Table of Figures

1 Introduction

1.1 Purpose

The purpose of this document is to describe the detailed design of the nRF2401+ module and how it works.

1.2 Definitions, Acronyms, and Abbreviations

ACK	*Acknowledgement*
CRC	Cyclic Redundancy Check
MSB	Most significant Bit/Byte
LSB	Least significant Bit/Byte

1.3 References

Item	Name	link
[1]	nRF24L01P_Product_Specification_1_0	

1.4 Overview

The nRF24L01+ is a single chip 2.4GHz transceiver with an embedded baseband protocol engine (Enhanced ShockBurst™). You can operate and configure the nRF24L01+ through a Serial Peripheral Interface (SPI).
The embedded baseband protocol engine (Enhanced ShockBurst™) is based on packet communication and supports various modes from manual operation to advanced autonomous protocol operation. Internal FIFOs ensure a smooth data flow between the radio front end and the system's MCU. Enhanced ShockBurst™ reduces system cost by handling all the high speed link layer operations.

1.5 Folders and files structure

nRF2401+ module was implemented by three files: nRF2401.c, nRF2401_cfg.h and nRF2401.h .

1.6 Features

- 126 RF channels
- 250kbps, 1 and 2Mbps air data rate
- 1 to 32 bytes dynamic payload length
- Automatic packet handling
- 1.9 to 3.6V supply range
- 5V tolerant inputs
- separate 32 bytes TX and RX FIFOs

1.7 Pin Function

1	VCC	Power Supply (+1.9V - +3.6V DC)
2	GND	Ground
3	CSN	Digital Input SPI Chip Select
4	CE	Digital Input Chip Enable Activates RX or TX mode
5	MOSI	Digital Input SPI Slave Data Input
6	SCK	Digital Input SPI Clock
7	IRQ	Digital Output Mask-able interrupt pin. Active low, when it is active high, there is an interrupt (received data or ACK for the sent data or maximum Retransmitted, to know what of them, read the Status register)
8	MISO	Digital Output SPI Slave Data Output, with tristate option

2 Detailed Design

2.1 Operational Mode

The General idea of any Operation is that enabling the power on mode of the module and establish the Setting to transmit or receive by sending Commands to the module by SPI then write or read the registers in the module to make the required operation.

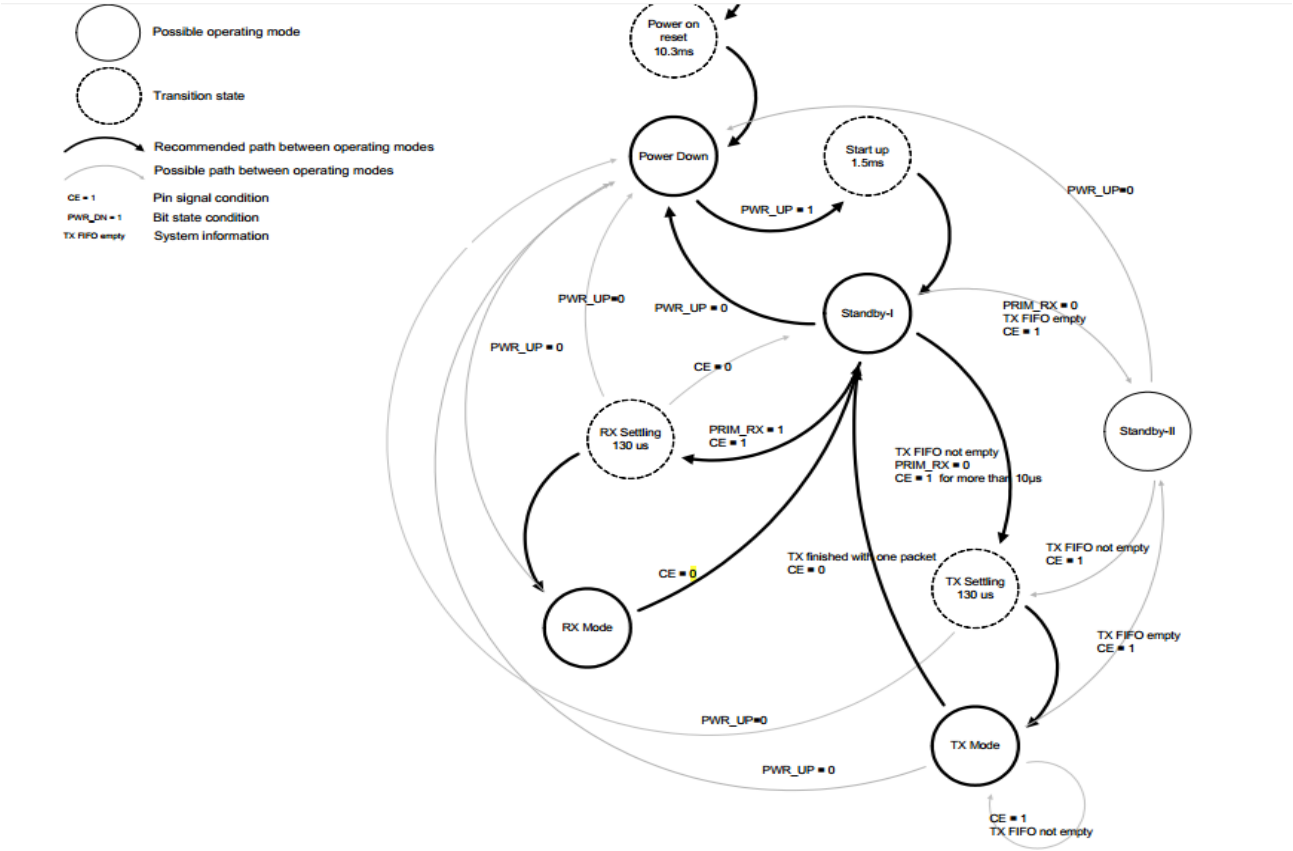
By setting the PWR_UP bit in the CONFIG register to 1, the device enters standby-I mode.

Bit rate, RF channel, Power Amplifier, define Static or dynamic Payload length, Payload length if static, Auto ACK or not, Auto retransmitted or not, Address of RX and TX and anther Features should be defined in the initialization.

The RX mode is an active mode where the nRF24L01+ radio is used as a receiver. To enter this mode, the nRF24L01+ must have the PWR_UP bit, PRIM_RX bit and the CE pin set high then if data is received, RX Flag will be one and IRQ will interrupt the controller.

The TX mode is an active mode for transmitting packets. To enter this mode, the nRF24L01+ must have the PWR_UP bit set high, PRIM_RX bit set low, a payload in the TX FIFO and a high pulse on the CE for more than 10µs. if Auto ACK is enabled, IRQ pin will interrupt the controller and Auto retransmitted is enabled and reached the maximum time and counts, IRQ will interrupt the controller.

The nRF24L01+ stays in TX mode until it finishes transmitting a packet. If CE = 0, nRF24L01+ returns to standby-I mode.



2.2 Enhanced ShockBurst™

2.2.1 Features

- 1 to 32 bytes dynamic payload length
- Automatic packet handling
- Automatic packet transaction handling (make RX transmit ACK to TX and can send Payload with this ACK and auto retransmit feature for defined time and counts if Packet is lost)
- Auto Acknowledgement with payload
- Auto retransmit
- 6 data pipe MultiCeiver™ (logical channels for one physical channel) for 1:6 star networks

2.2.2 Overview

Enhanced ShockBurst™ uses ShockBurst™ for automatic packet handling and timing. During transmit, ShockBurst™ assembles the packet and clocks the bits in the data packet for transmission.
During receive, ShockBurst™ constantly searches for a valid address in the demodulated signal. When ShockBurst™ finds a valid address, it processes the rest of the packet and validates it by CRC.

2.2.3 Enhanced Shockburst™ packet format

Preamble 1 byte	Address 3-5 byte	Packet Control Field 9 bit	Payload 0 - 32 byte	CRC 1-2 byte

The preamble is a bit sequence used to synchronize the receivers demodulator to the incoming bit stream.

Address preventing accidental cross talk between multiple nRF24L01+ systems, configure from AW register to be 3, 4 or 5 bytes.

Packet Control Field is used to determine the packet is retransmitted or first time transmit, define it used ACK or nit and the width of data (Payload) in the packet if it is static, make RX side as TX side as number of send bytes except it is dynamic, you don't need to define it.

The CRC is the mandatory error detection mechanism in the packet. If RX CRC doesn't equal TX CRC, Packet is rejected.

2.2.4 Commands Table

First, Command (1Byte) is send then (if needed) send/get data or write/read register.

The serial shifting SPI commands is in the following format:

<Command word: MSBit to LSBit (one byte)>

<Data bytes: LSByte to MSByte, MSBit in each byte first>

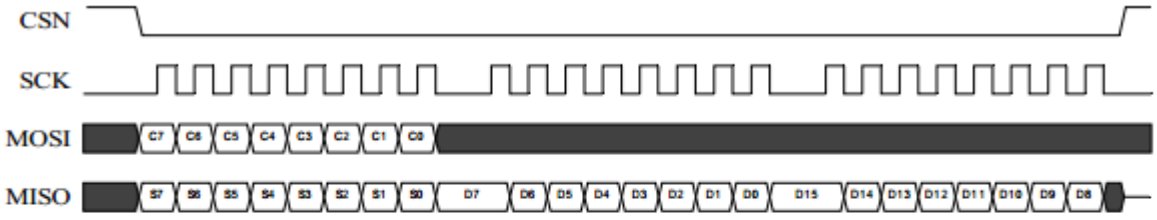


Figure 26. SPI read operation

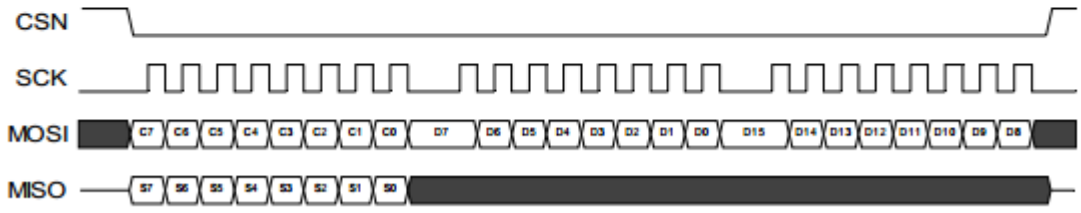


Figure 27. SPI write operation

Command name	Command word (binary)	# Data bytes	Operation
R_REGISTER	000A AAAA	1 to 5 LSByte first	Read command and status registers. AAAAA = 5 bit Register Map Address
W_REGISTER	001A AAAA	1 to 5 LSByte first	Write command and status registers. AAAAA = 5 bit Register Map Address Executable in power down or standby modes only.
R_RX_PAYLOAD	0110 0001	1 to 32 LSByte first	Read RX-payload: 1 – 32 bytes. A read operation always starts at byte 0. Payload is deleted from FIFO after it is read. Used in RX mode.
W_TX_PAYLOAD	1010 0000	1 to 32 LSByte first	Write TX-payload: 1 – 32 bytes. A write operation always starts at byte 0 used in TX payload.
FLUSH_TX	1110 0001	0	Flush TX FIFO, used in TX mode
FLUSH_RX	1110 0010	0	Flush RX FIFO, used in RX mode Should not be executed during transmission of acknowledge, that is, acknowledge package will not be completed.
REUSE_TX_PL	1110 0011	0	Used for a PTX device Reuse last transmitted payload. TX payload reuse is active until W_TX_PAYLOAD or FLUSH TX is executed. TX payload reuse must not be activated or deacti-vated during package transmission.
R_RX_PL_WID[a]	0110 0000	1	Read RX payload width for the top R_RX_PAYLOAD in the RX FIFO. **Note:** Flush RX FIFO if the read value is larger than 32 bytes.
W_ACK_PAYLOAD[a]	1010 1PPP	1 to 32 LSByte first	Used in RX mode. Write Payload to be transmitted together with ACK packet on PIPE PPP. (PPP valid in the range from 000 to 101). Maximum three ACK packet payloads can be pending. Payloads with same PPP are handled using first in - first out principle. Write payload: 1– 32 bytes. A write operation always starts at byte 0.
W_TX_PAYLOAD_NO ACK[a]	1011 0000	1 to 32 LSByte first	Used in TX mode. Disables AUTOACK on this specific packet.
NOP	1111 1111	0	No Operation. Might be used to read the STATUS register

2.2.5 Register Map

There are almost 29 register, each one is 1byte and has bits to control the operations, to learn more look for it in the data sheet.

3 Driver Functions

3.1 Internal Functions

3.1.1 *nRF_writeRegister*

Format	RF_writeRegister (U8_t Reg_Add, U8_t value)
Description	to write defined value to a specific register
Argument	U8_t Reg_Add: register address U8_t value : value wanted to write to this register
Return value	NONE

3.2 Global Functions

3.2.1 EF_void_nRF_init

Format	EF_void_nRF_init();
Description	initialize the SPI ,External Interrupt and nRF module
Argument	NONE
Return value	NONE

3.2.2 EF_void_nRF_TXSetup

Format	EF_void_nRF_TXSetup(void);
Description	establish the module to transmit
Argument	NONE
Return value	NONE

3.2.3 EF_BOOLEAN_nRF_SendData

Format	EF_BOOLEAN_nRF_SendData(U8_t Data, U8_t DataLength);
Description	transmit data by the nRF module
Argument	U8_t Data: pointer to transmitted data U8_t DataLength: no of bytes of transmitted data
Return value	BOOLEAN to check for Errors

3.2.4 EF_BOOLEAN_nRF_RXSetup

Format	EF_BOOLEAN_nRF_RXSetup()
Description	establish the module to receive
Argument	NONE
Return value	NONE

3.2.5 EF_BOOLEAN_nRF_GetData

Format	EF_BOOLEAN_nRF_GetData(U8_t Data , U8_t DataLength);
Description	receive data by the nRF module
Argument	U8_t Data: pointer to received data U8_t DataLength: no of bytes of received data
Return value	BOOLEAN to check for Errors

3.2.6 EF_u8_nRF_ReadRegister

Format	EF_u8_nRF_ReadRegister(U8_t Reg_Add);
Description	read specific register located the nRF module
Argument	U8_t Reg_Add: address of wanted register
Return value	the data located in this register