nRF24LE1 with support for infrared protocols

nAN-19

Application Note v1.0

Key words

- Transmit and receive infrared signals with nRF24LE1
- Three protocols: RC-5, NEC and SIRC
- Timer-controlled transmission and reception

2010-12-15

Contents

1 Introduction

RF technology offers new possibilities for more advanced remote controls in consumer electronics like TVs, DVD players and PVRs. However, at least for a transitional period, customers will also have older equipment that only supports IR (infrared) remote controls.

To control this older equipment, you need legacy support for IR.

This application note describes how to transmit and receive IR signals using Nordic Semiconductor's nRF24LE1. The nRF24LE1 uses an infrared LED connected to the integrated PWM (pulse width modulation) module in nRF24LE1 to transmit IR signals. An IR receiver is connected to interrupt pins to detect incoming IR signals.

Note: This application note focuses on the nRF24LE1 device, but the principles covered here are general and could also apply to other Nordic Semiconductor products. If you apply these principles to other devices, expect to rewrite some of the code. Note incidentally that the nRF24LU1 has no PWM output.

2 Infrared protocols

IR communication is coded light signals. The light has wavelengths that are invisible to the human eye. To transmit and receive IR signals, you need an IR emitter and an IR receiver. The IR emitter is a simple Light Emitting Diode (LED) which emits light in the IR band, and the IR receiver is a phototransistor that is sensitive to IR light.

The IR protocols are usually pulse code modulated, meaning the light is switched ON/OFF with a specific carrier frequency that in turn is modulated with user data. Modulation is carried out to ensure noise immunity from other IR sources, for example sunlight.

Figure 1. shows a typical light pattern for an IR protocol. The dashed line represents the envelope representing the user data, and the solid lines shows the ON/OFF switching of the IR LED with the higher carrier frequency.

Figure 1. Example of IR modulation pattern

The IR receivers are often tuned to one carrier frequency, for example, 38 kHz. The IR receiver filters out the carrier frequency, and outputs only the data represented by the envelope (dashed line in Figure 1.).

2.1 Protocols

There are several communication protocols used in IR remote controls. The protocols use different carrier frequencies, different bit patterns and are usually incompatible with each other.

Carrier frequency is in the region of 30 kHz to 60 kHz, where 36 kHz, 38 kHz and 40 kHz are most frequently used. The carrier frequencies also have different duty cycles which are specified by the protocols. A lower duty cycle saves power since the IR LED is on for a shorter period of time.

Three commonly used protocols are RC-5, NEC and SIRC. These protocols are examples of different formats for transferring data.

In addition to different bit coding of the user data the protocols define a header or a preamble that is used to synchronize the receiver. These headers can be a couple of bits or some sequence that differs from the data bit pattern. The protocols also define a time interval between messages when the IR diode should be off.

2.1.1 RC-5 protocol

The RC-5 protocol was developed by Philips and is one of the main IR protocols in use. The protocol has been adopted by several brands worldwide. The IR carrier frequency is 36 kHz and uses a duty cycle of 1/3 or 1/4. The protocol uses Manchester encoding which means that the data symbols have constant length. A logical '0' is transmitted as a burst of light lasting for 889 µs followed by a space of 889 µs. A logical '1' is the opposite; a space followed by a burst (See Figure 2. and Figure 3.).

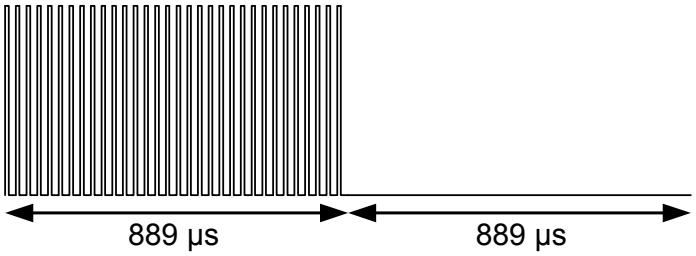


Figure 2. RC-5 logical "0"

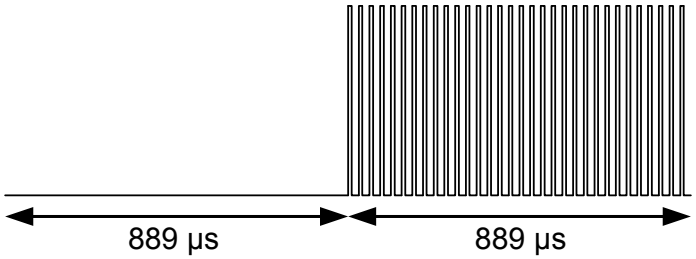


Figure 3. RC-5 logical "1"

A RC-5 message or packet consists of 14 bits. There are two start bits that always are '1'. The third bit is a toggle bit that flips between '1' and '0' for each button press.

Following the three bits are five address bits. These bits are used to identify the equipment that should respond to the command. After the address, there are six bits for command. Each bit takes 1.778 ms to transfer, so that the entire sequence lasts for 24.892 ms.

As long as a key is pressed, the message will be repeated every 114 ms and the toggle bit will remain in the same state during the repeated messages. Figure 4. shows an example of an RC-5 message.

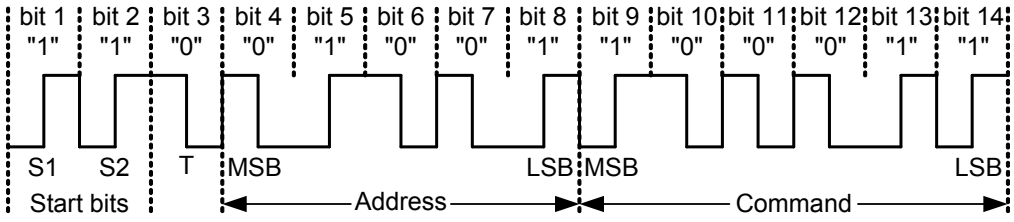


Figure 4. Example of an RC-5 sequence

2.1.2 NEC protocol

The NEC protocol was developed by NEC, and is now adopted by many companies. The protocol uses pulse distance encoding of the bits, and has a header pulse, a dedicated pulse for repeated transmissions and a pulse to mark the end of a message. The carrier frequency is 38 kHz and the duty cycle is 1/3 or 1/4. The header is a burst of light lasting for 9 ms followed by a 4.5 ms space. All bits start with a burst of light lasting for 562.5 µs followed by a space of 562.5 µs if the bit is '0', or a space of 1687.5 µs if the bit is '1'. Figure 5., Figure 6. and Figure 7. show some of the pulse sequences in the NEC protocol.

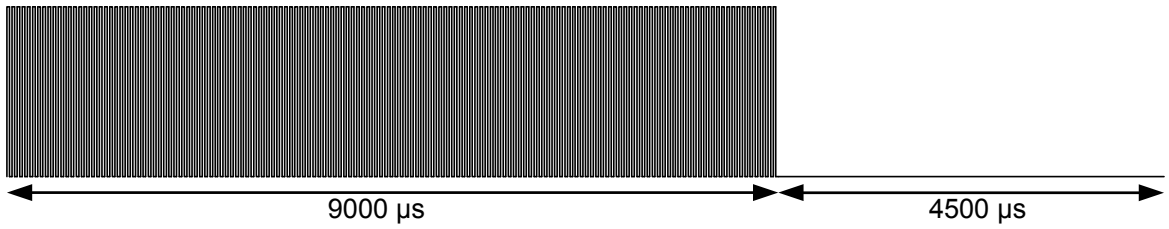


Figure 5. NEC header pulse

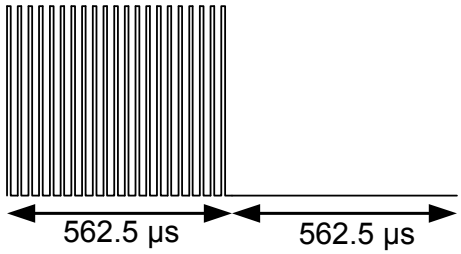


Figure 6. NEC logical "0"

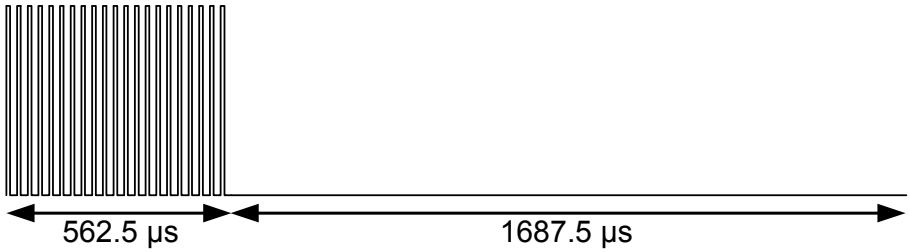


Figure 7. NEC logical "1"

The standard NEC protocol consists of an 8-bit address and an 8-bit command. In addition, an inverted version of the address and the command is sent, resulting in a total of 32 bits. Because each bit has an

inverse, the transmit time for the sequence is constant and lasts for 67.5 ms. Figure 8. shows an example of an NEC sequence.

Figure 8. Example of a NEC sequence

As long as a key is pressed, a repeat code will be sent every 108 ms (shown in Figure 9.). The repeat code consists of a 9 ms burst followed by a 2.25 ms space, and finally the 562.5 µs end of message pulse.

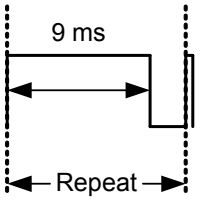


Figure 9. NEC repeat sequence

The 8-bit address space sets a limit of 256 possible values. However, this number proved to be too small and there was a need for more addresses. Because of this, the extended NEC protocol was defined.

The extended protocol is similar to the standard protocol except that the inverted address is replaced by eight, regular address bits, giving a total of 16 bits for the address.

2.1.3 SIRC protocol

SIRC stands for Sony Infrared Remote Control and was developed by Sony. The SIRC protocol uses pulse length encoding of the bits, and the sequence starts with a header pulse. The protocol uses a carrier frequency of 40 kHz and the recommended duty cycle is 1/3 or 1/4. The bits are represented by the width of the pulses meaning that all spaces are of equal length: 600 µs. The header pulse is 2.4 ms, the logical '0' pulse is 600 µs and the logical '1' pulse lasts for 1200 µs. See Figure 10., Figure 11. and Figure 12. on page 7.

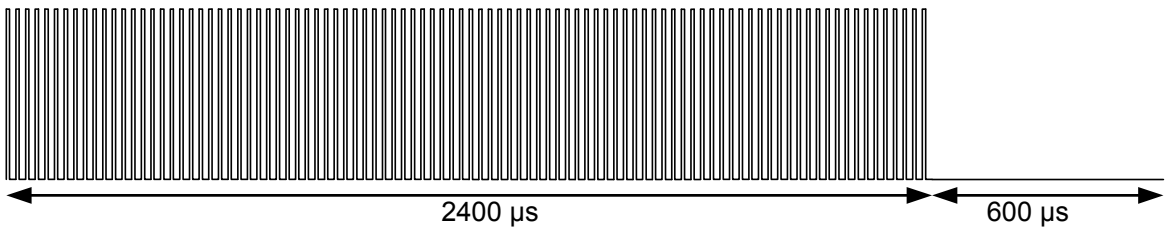


Figure 10. SIRC header pulse

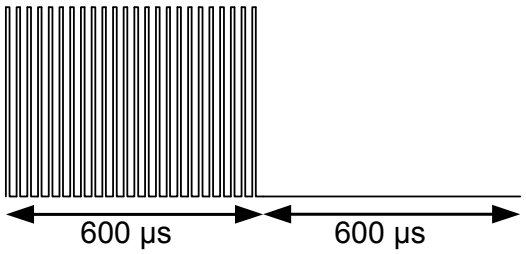


Figure 11. SIRC logical "0"

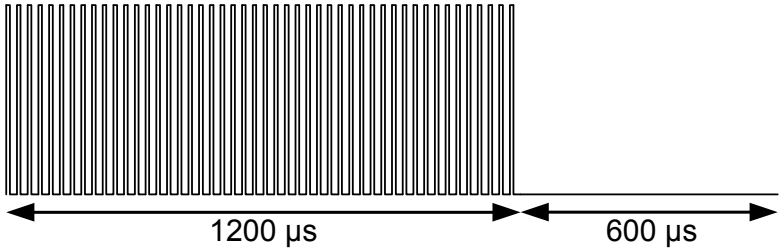


Figure 12. SIRC logical "1"

The protocol comes in 12-bit, 15-bit and 20-bit versions. The sequences of the three versions are:

- 12-bit version: 7-bit command and a 5-bit address
- 15-bit version: 7-bit command and an 8-bit address
- 20-bit version: 7-bit command, a 5-bit address and an 8-bit extended word.

There are different versions and different bit times for logical '1' and logical '0'. Therefore, the sequence time varies from 17.4 ms for the 12-bit version with only '0' transmitted, to 39 ms for the 20-bit version with only '1' transmitted. The message is repeated every 45 ms while a key is pressed. Figure 13. shows an example of a sequence of the 12-bit version of the SIRC protocol.

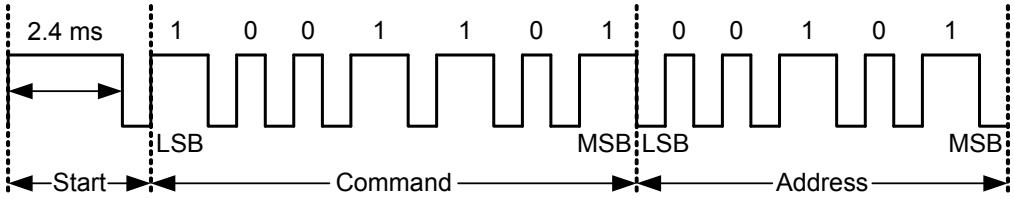


Figure 13. Example of a 12-bit SIRC sequence

3 Implementation

This chapter presents the software and hardware needed for you to successfully transmit and receive infrared signals.

3.1 Transmitter hardware

The necessary signal to pulse the IR LED is easily generated with the nRF24LE1's built-in PWM module. Figure 14. on page 8 shows the external circuitry needed for an nRF24LE1-based IR transmitter.

An IR LED from Vishay Semiconductors (TSAL6200) is controlled through a transistor (Fairchild Semiconductor FDV303N) because the current needed to emit a strong enough IR signal exceeds the drive capabilities of the nRF24LE1 I/O ports.

The gate of the transistor is connected to one of the PWM outputs of nRF24LE1. A resistor in series with the diode is used to limit the current flowing through the diode. A pull up resistor is connected in parallel with the diode to ensure that the drain of the transistor is pulled to VDD when the transistor is shut off . This prevents parasitic current drain from the battery when the LED is off.

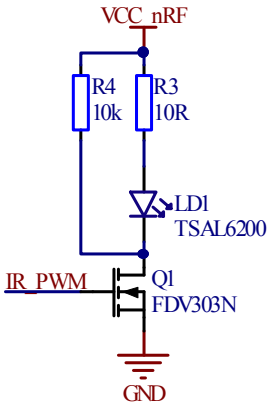


Figure 14. IR emitter diode with external circuitry

The LED in an IR transmitter drains up to 100 mA of current at full power. The power supply used must be capable of delivering these peak currents. Since the IR is sent in bursts, a large reservoir capacitor (10µF) on the power supply will reduce the voltage drops of the supply when the IR LED is on. If the supply voltage drops below the minimum supply voltage of nRF24LE1(1.9V) during the IR pulse, nRF24LE1 will reset, and operation will no longer be possible.

3.2 Transmitter software

Protocol-specific information like timing, frequency, duty cycle and so on, are stored in a data structure. A function is called to start a transmission (example: `lib_ir_transmit_sirc_command()`.) The function sets up the data structure and loads the protocol information. A flag is set to indicate that data is ready to be transmitted. After this, a timer interrupt function (`lib_ir_transmit_timer_irq_function()`) handles all the transmission. After a transmission is finished, a call-back function is called to notify the main program.

To transmit an IR command, the nRF24LE1 uses the built-in PWM module that outputs the carrier signal with correct frequency and duty cycle. One of the generic timers in nRF24LE1 is used to control the encoding of user data on the IR carrier. The timer is handled outside of the library, and the timer period must be given in the initialization function as a number in µs. The timer period should be short enough to give a good resolution for all the supported protocols meaning below 60 µs. For each timer interrupt the `lib_ir_transmit_timer_irq_function()` function must be called. This function will keep track of the bit sequence in the IR protocols when sending data. Figure 15. on page 9 shows a flow chart for the timer interrupt function.

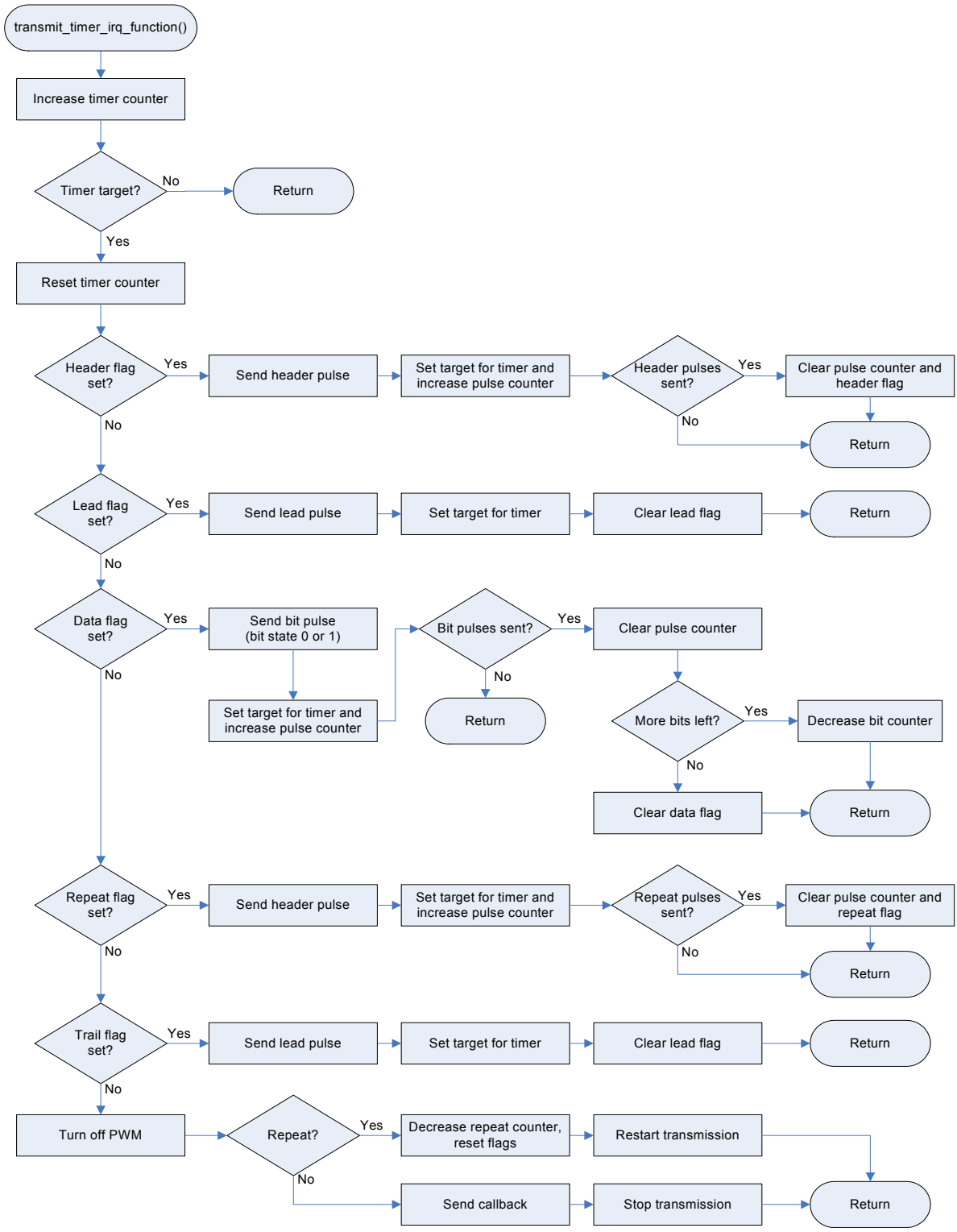


Figure 15. Flow chart for transmit timer interrupt function: lib_ir_transmit_timer_irq_function()

3.3 Receiving hardware

The IR receiver is a combination of a light-sensitive transducer and a demodulator. The receiver has a band-pass filter which is tuned to a specific, carrier frequency. The receiver must therefore be chosen according to the IR protocol used. In an example shown in Figure 16. a receiver tuned to 38 kHz is used, but, because of the filter characteristics, both 36 kHz and 40 kHz may also be detected. The other required external circuitry for an IR receiver is a resistor and a capacitor. Figure 16. shows the IR receiver from Vishay Semiconductors (TSOP34338), where the **OUT** pin is connected directly to the nRF24LE1.

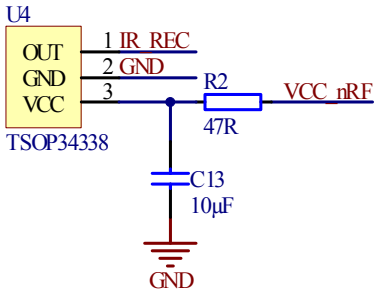


Figure 16. IR receiver and external circuitry

3.4 Receiving software

Protocol-specific information like timing, frequency, duty cycle and so on, are the same as for the transmit library. `lib_ir_receive_command()` is called, with the desired protocol as input parameter, to initiate a reception. The function loads the IR protocol information and sets a flag to indicate that it is ready to receive data. A timer interrupt function (`lib_ir_receive_timer_irq_function ()`) will reset the counters until the first edge of an IR signal is received. After the first edge is detected, the timer interrupt function will sample and decode the incoming signal. The library checks the level of the input signal (high or low) at given intervals according to the specified protocol. If the incoming signal is not the correct protocol, the library will fail to decode the message and give an error. When the data is received, a call back function is called with the received data to notify the main program.

As for the transmit library, the receive library uses a timer to control the timing of the IR protocol. This timer must also be handled outside of the library and could be the same timer as for the transmitter. The timer period must be given in the initialization function in the same way as for the transmit library. The timer interrupt function (`lib_ir_receive_timer_irq_function ()`) must be called on every timer interrupt. This function will keep track of the timing when receiving data. In addition, function (`lib_ir_receive_edge_irq_function()`) should be called when there is a falling or rising edge of the IR signal. Figure 17. on page 11 shows a flow chart for the timer interrupt function, and Figure 18. on page 12 shows the flow chart for edge detect interrupt function.

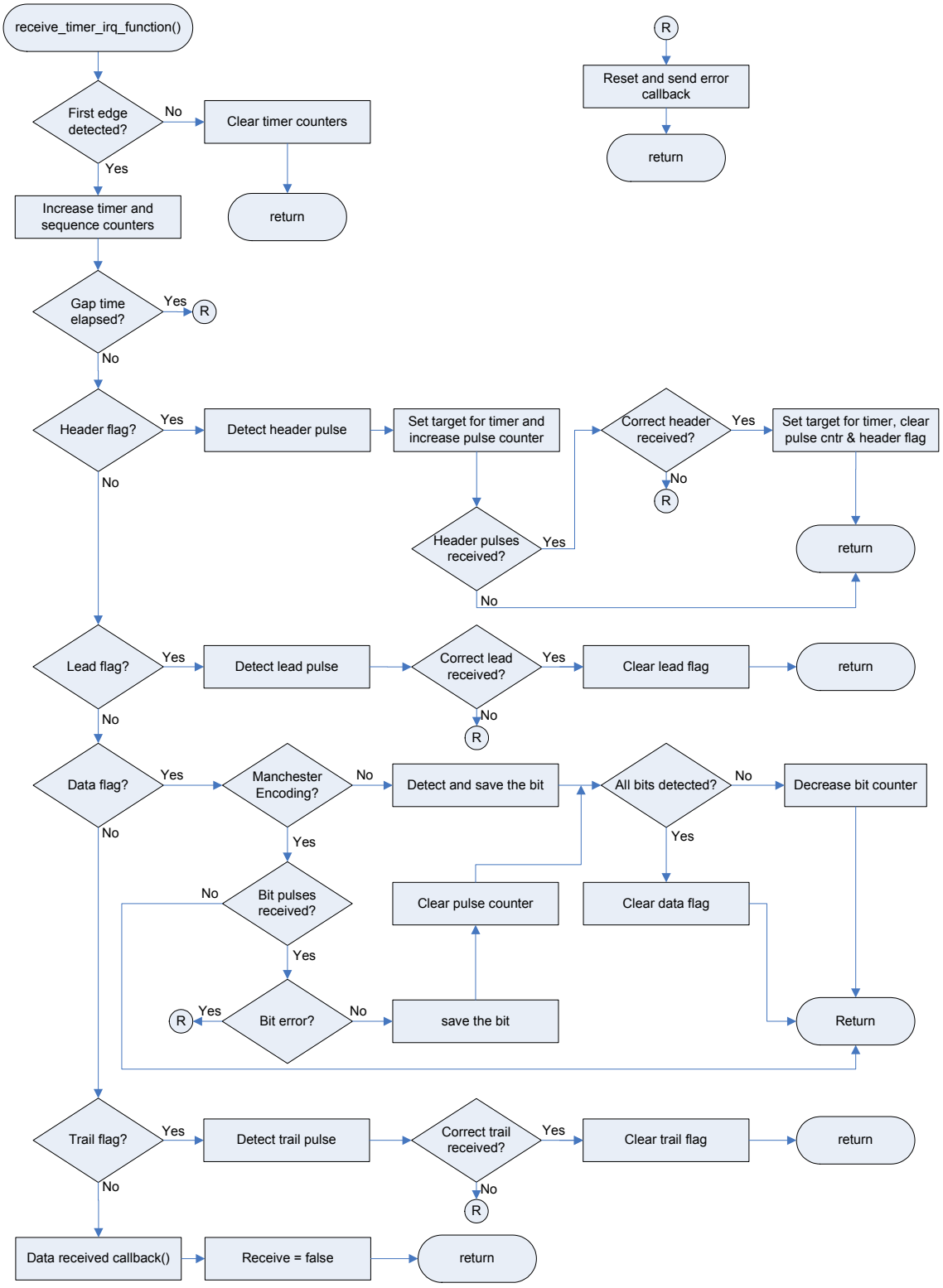


Figure 17. Flow chart for receive timer interrupt function: lib_ir_receive_timer_irq_function ()

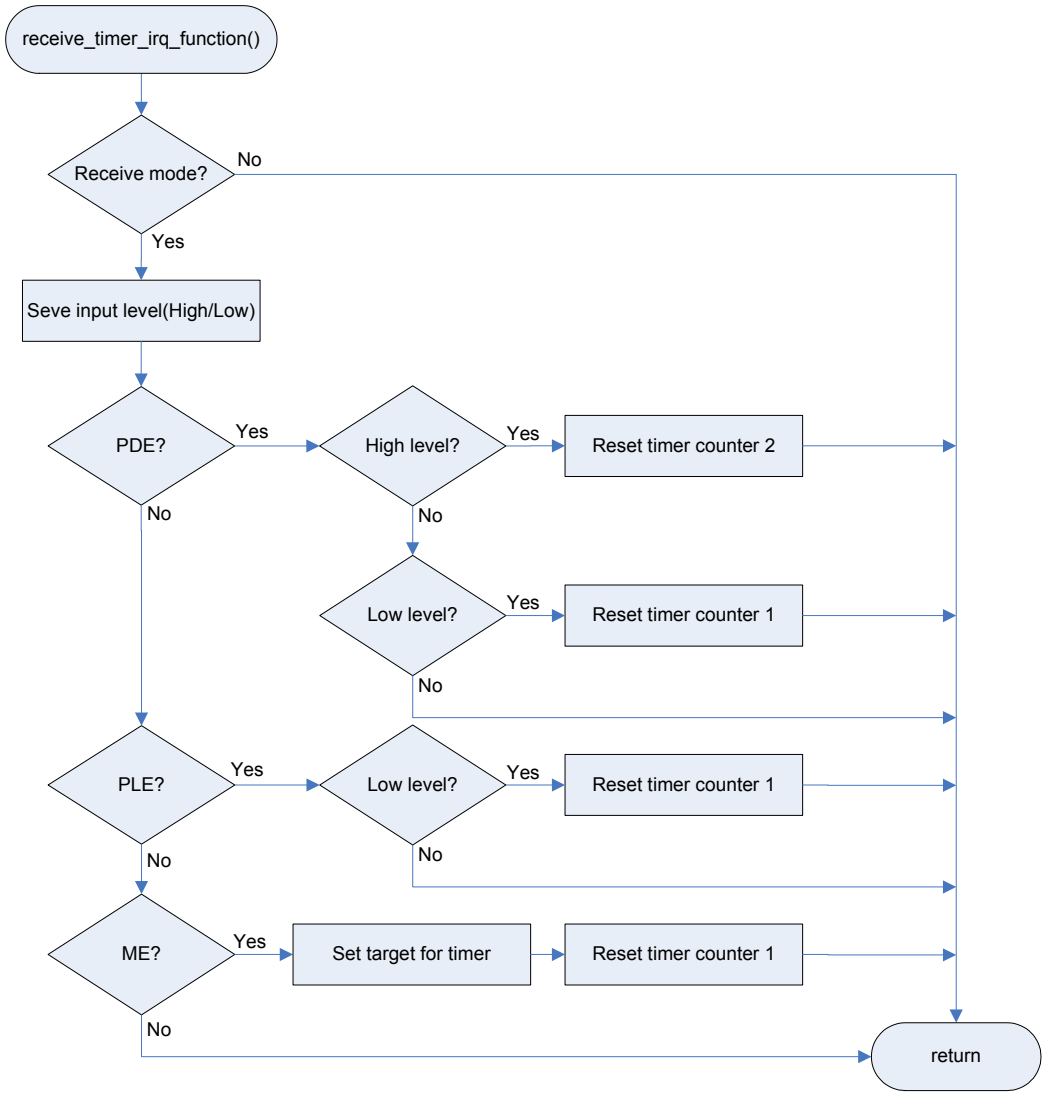


Figure 18. Flow chart for edge detect interrupt function: lib_ir_receive_edge_irq_function()

4 Glossary

Term	Description
LED	Light Emitting Diode
ME	Manchester Encoding
PDE	Pulse Distance Encoding
PLE	Pulse Length Encoding
PVR	Personal Video Recorder
PWM	Pulse-width Modulation
RF	Radio Frequency

Table 1. Glossary

5 References

http://www.sbprojects.com/knowledge/ir/ir.htm
http://wiki.altium.com/display/ADOH/Philips+RC5+Infrared+Transmission+Protocol
http://wiki.altium.com/display/ADOH/NEC+Infrared+Transmission+Protocol
http://www.sbprojects.com/knowledge/ir/nec.htm
http://www.sbprojects.com/knowledge/ir/sirc.htm
http://picprojects.org/projects/sirc/sonysirc.pdf
http://www.vishay.com/docs/81010/tsal6200.pdf
http://www.fairchildsemi.com/ds/FD/FDV303N.pdf
http://www.vishay.com/docs/81737/tsop341.pdf

Liability disclaimer

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

Life support applications

Nordic Semiconductor's 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.

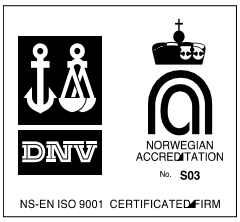
Contact details

For your nearest dealer, please see http://www.nordicsemi.com

Receive available updates automatically by subscribing to eNews from our homepage or check our website regularly for any available updates.

Main office:

Otto Nielsens veg 12
7004 Trondheim
Phone: +47 72 89 89 00
Fax: +47 72 89 89 89
www.nordicsemi.com



Revision History

Date	Version	Description
December 2010	1.0	