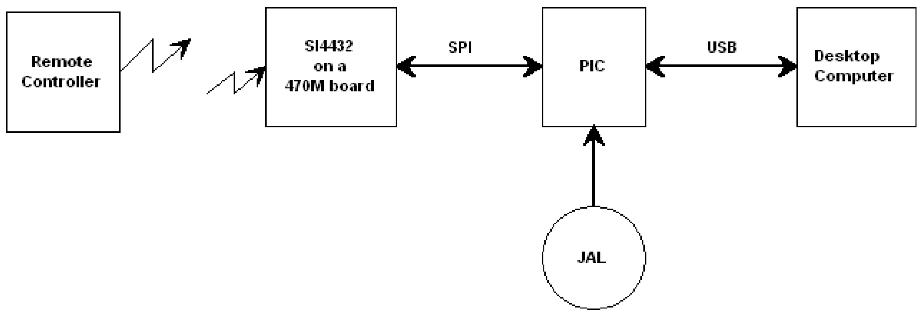
ook modulation

last updated: nov. 2014, SM

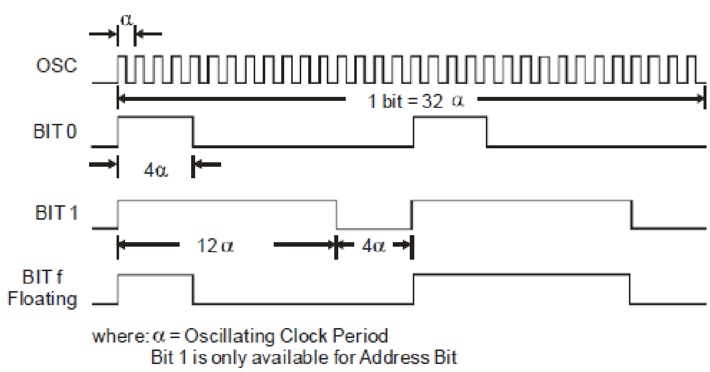
Introduction

On-Off-Keying (OOK) is a very simple modulation techniek used in infrared control and many (cheap) remote home control systems like lamp controls, door openers etc. Most of the remote control units works with a carrier frequency of 434 MHz or 868 MHz.
The SI443 is a cheap RF transceiver (both sender and receiver) specially meant for packet transmission in the UHF-band (200 MHz ... 900 MHz). This relative low frequency, compared to the popular 2.4 GHz band, has the advantage of a longer range and better passing through obstacles like walls. Although the SI4432 is not well suited for raw OOK transmission, experiments shows that it's quiet well possible to build a reasonable reliable OOK transmitter with a PIC and some JAL code. The JAL library "SI4432_support.jal" contains all necessary functions. This package also contains a number of desktop programs to view and analyse the signals. In the picture below "Remote Controler" is in fact a Remote Receiver.



The modulation signal

Often a PT2262 (SC2262 is equivalent) is used as the encoder to generate the modulation signal. The PT2262 sends a 12 digit, trinary coded, word followed by a sync bit. Normally a sync is in front of the data but here the sync is after the data !! Trinary code means that each digit can have one out of three values: zero, one and float. The MSB of the 12 digit word contains the address and the LSB contains the data. At least 6 of the digits needs to be the address, and the address can extend to 11 digits. Of course, the more address bits, the less data bits. The difference between address digits and data digits becomes clear when using the standard decoder (PT2272 or SC2272), the address digits are only used as inputs for comparison of the incoming signal, while data digits can be used both as an input or as an output with a logic zero or one.



SI4432 implementation

As shown above, we have a data word followed by a sync bit. The SI4432 is especially designed for packet transmission, and in standard mode generates pre-amble, sync words, header words and a CRC word. Preferable we don't want to generate the modulation by JAL, but we want to use the SI4432's FIFO and let the SI4432 do the Baudrate timing. We can easily disable sync words, header and CRC words, but we can't disable the preamble. Experiments shows that the preamble (long series of zeroes and ones), if held as short as possible, won't heart a good OOK generation. The registers settings are best taken from the hacked Excel file. The preamble is set to zero (or one), which will produce 4 bits of preamble.

Data bytes

Each period of 4 alfa is represented as 1 bit. So one bit from the OOK modulation is represented in1 byte to the SI4432. This results in the following table:

OOK	SI4432 modulation
0	0x88
1	0xEE

float	0x8E
sync	0x80, 0x00, 0x00, 0x00

For the GOA remote control set the following codes are valid

Switch	On			Off		
A1	0FFF	0FFF	FFF1	0FFF	0FFF	FFF0
A2	0FFF	F0FF	FFF1	0FFF	F0FF	FFF0
A3	0FFF	FF0F	FFF1	0FFF	FF0F	FFF0
B1	F0FF	0FFF	FFF1	F0FF	0FFF	FFF0
B2	F0FF	F0FF	FFF1	F0FF	F0FF	FFF0
B3	F0FF	FF0F	FFF1	F0FF	FF0F	FFF0
C1	FF0F	0FFF	FFF1	FF0F	0FFF	FFF0
C2	FF0F	F0FF	FFF1	FF0F	F0FF	FFF0
C3	FF0F	FF0F	FFF1	FF0F	FF0F	FFF0
D1	FFF0	0FFF	FFF1	FFF0	0FFF	FFF0
D2	FFF0	F0FF	FFF1	FFF0	F0FF	FFF0
D3	FFF0	FF0F	FFF1	FFF0	FF0F	FFF0

The first 7 bits change for different addresses and switches. The last bit (bit 12) is used as an On-Off Signal. So we implemented the following code in JAL :

```
procedure SI4432_OOK_Write_1 ( byte in Switch,
                                bit in On_Off,
                                byte in Repeat_Count  ) is
  var byte OOK_A1 [] = { 0x88, 0x8E, 0x8E, 0x8E, 0x88, 0x8E, 0x8E }
  var byte OOK_A2 [] = { 0x88, 0x8E, 0x8E, 0x8E, 0x8E, 0x88, 0x8E }
  var byte OOK_A3 [] = { 0x88, 0x8E, 0x8E, 0x8E, 0x8E, 0x8E, 0x88 }

;  var byte OOK_B1 [] = { 0x8E, 0x88, 0x8E, 0x8E, 0x88, 0x8E, 0x8E }
;  var byte OOK_B2 [] = { 0x8E, 0x88, 0x8E, 0x8E, 0x8E, 0x88, 0x8E }
;  var byte OOK_B3 [] = { 0x8E, 0x88, 0x8E, 0x8E, 0x8E, 0x8E, 0x88 }
;
;  var byte OOK_C1 [] = { 0x8E, 0x8E, 0x88, 0x8E, 0x88, 0x8E, 0x8E }
;  var byte OOK_C2 [] = { 0x8E, 0x8E, 0x88, 0x8E, 0x8E, 0x88, 0x8E }
;  var byte OOK_C3 [] = { 0x8E, 0x8E, 0x88, 0x8E, 0x8E, 0x8E, 0x88 }
;
;  var byte OOK_D1 [] = { 0x8E, 0x8E, 0x8E, 0x88, 0x88, 0x8E, 0x8E }
;  var byte OOK_D2 [] = { 0x8E, 0x8E, 0x8E, 0x88, 0x8E, 0x88, 0x8E }
;  var byte OOK_D3 [] = { 0x8E, 0x8E, 0x8E, 0x88, 0x8E, 0x8E, 0x88 }

  var byte Data [] = { 0x8E,
                       0x8E, 0x8E, 0x8E, 0xEE,
                       0x80 }

  if On_Off then Data[4] = 0xEE
  else           Data[4] = 0x88
  end if
```

The FIFO will be filled with one of the OOK_xx arrays plus the Data-array.
The last byte of the Data-array contains the first part of the sync word.

Sync word

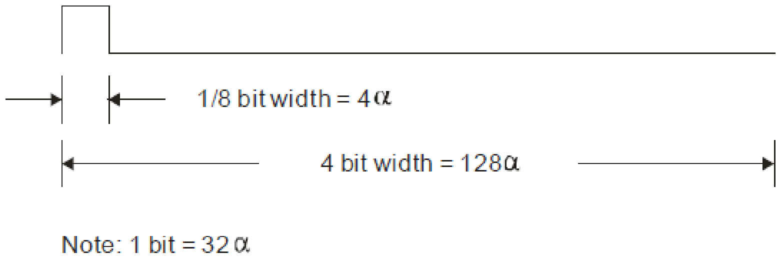
In OOK signals the syn words is placed behind the data word.

A group of Code Bits is called a Code Word. A Code Word consists of 12 AD bits followed by one Sync Bit. The 12 AD bits are interpreted as either address or data bits depending on the PT2272 version used. Please refer to the diagrams below:

PT2272:

A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	SYNC

Synchronous (Sync.) Bit Waveform
The Synchronous Bit Waveform is 4 bits long with 1/8 bit width pulse. Please refer to the diagram below:



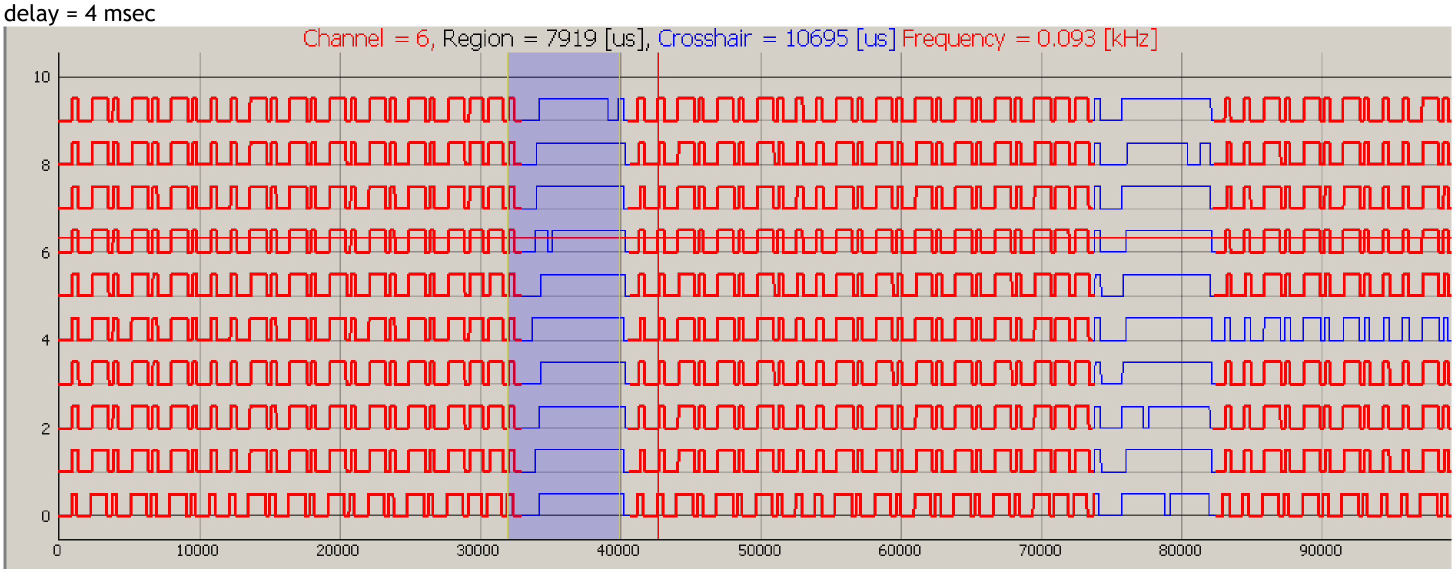
So to be able to use the FIFO, we need to handle the sync bit as a data word. The big trick is to realize that the last part of the sync word, which is the major part, is a zero, so no modulation. This means that we can replace almost the complete sync word with a simple delay in JAL.

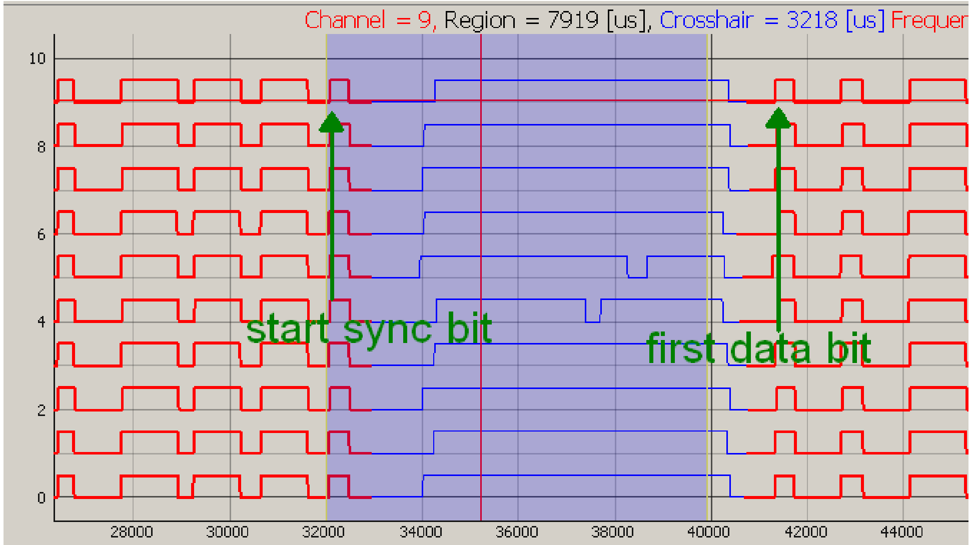
Knowing the Baudrate of the required signal (2.85 kHz) we can calculate that the sync word must have a length of 11.4 msec. Adding an extra delay of 7 msec results in total sync word length of 10.9 msec. In the next paragraph the results of some experiments with the width of the sync pulse are shown.

Sync bit length

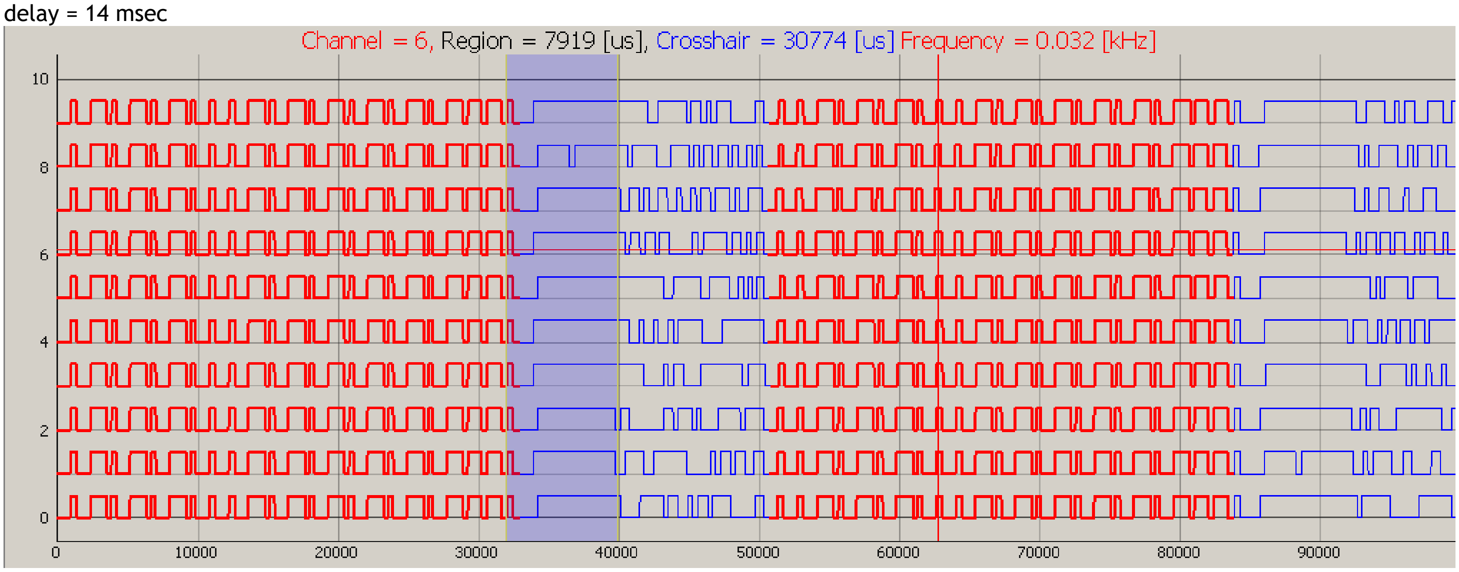
All measurememst were done with the Logic Packet Viewer.
The total length of the sync bit, at a Baudrate of 2.85 kHz, should be 11.4 msec.
This requires with the current JAL routine an extra delay of 7 msec (resulting in a total sync bit length of 10.9 msec) or 8msec (resulting in a total sync bit length of 11.9 msec.
To see how sensitive the PT2272 decoder is, we did some experiments, by toggling a remote switch on-off and measure if all commands were recognized. Theoretically the PT2272 decoder needs 2 consecutive command signals. With 2 consecutive command signals none of the remote switches reacted in any way. With3 consecutive command signals, and an extra delay of 7 or 8 msec, about 50% of the commands were succesfull. With 4 consecutive commands, and a 7 or 8 msec extra delay, we gained a score of 100%.
Probably the RF-receiver has to settle on the incoming signal, so it takes while before valid commands reach the decoder.
With 4 consecutive commands we tried how small and how large we could make the sync bit until the commands becomes unreliable.
The shortest sycn bit that worked reliable was with an extra delay of 4 msec (sync bit 7.9 msec) and the largest reliable sync bit was reached with an extra delay of 14 msec (sync bit 17.9 msec).
Pictures of both of these extreme values are shown below. The streamviewer (also based on a SI4432) can not realiable detect the long low period of the sync bit, so graphs may not show the actual value in that long low period.

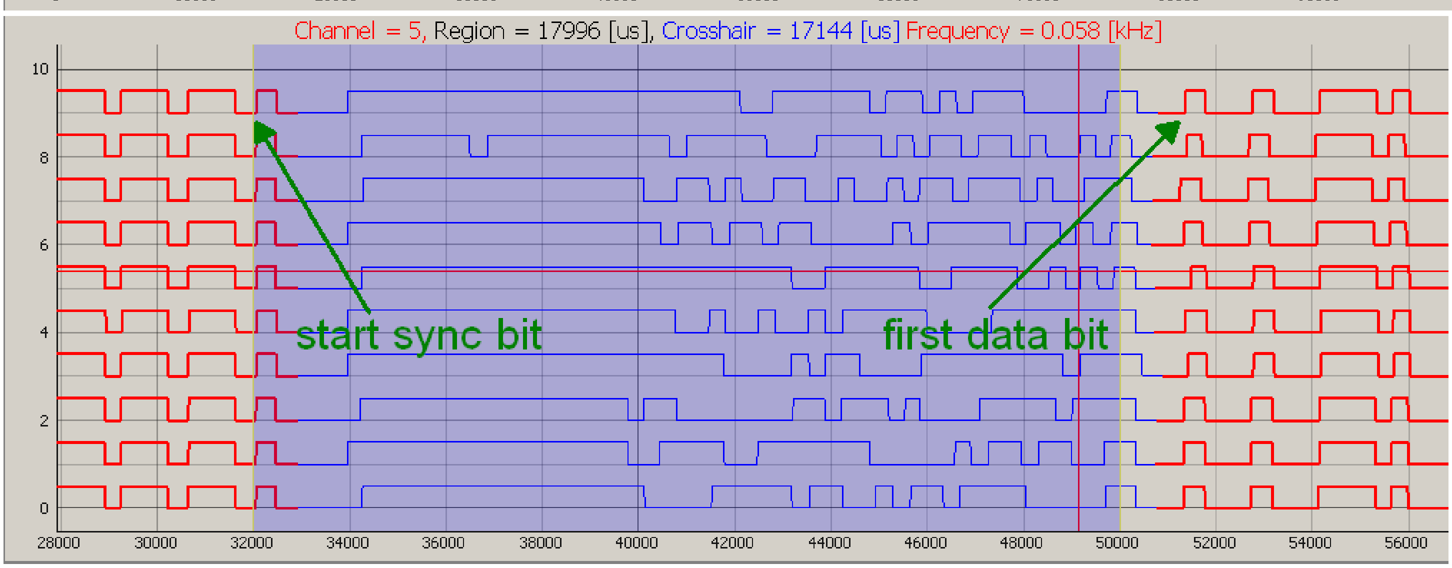
delay = 4 msec





delay = 14 msec





Background Info

exact Baudrate seems to be 2.85 kBaud

clock = 100 usec = 10 kHz
bit = 4 clocks ==> 2.5 kbps

http://community.silabs.com/t5/Silicon-Labs-Knowledge-Base/IF-Filter-BW-in-OOK-mode-for-EZRadioPRO/ta-p/112497

Question

How do I configure EZRadioPRO chips to obtain narrow IF bandwidth in OOK mode?

Answer

Normally, it is not necessary to configure EZRadioPRO chips for narrow IF channel filter selection pre-detection bandwidths while in OOK mode. It is possible to obtain excellent RX sensitivity in OOK mode while still configuring the chip for a relatively wide IF bandwidth.

Many legacy OOK systems use frequency resonators with relatively poor frequency tolerance (e.g., SAW-based resonators) and thus it is often necessary to configure the receiver for a wide reception bandwidth to allow for uncertainty in the exact location of the signal in the frequency spectrum. In fact, the EZRadioPRO Excel Register Calculator Settings worksheet does not allow the user to specify an IF bandwidth in OOK mode less than 75 kHz. Many users may feel that this width of IF bandwidth may negatively impact the RX sensitivity of the system, and desire to configure the chip for a narrower bandwidth.

However, EZRadioPRO devices use a combination of pre-detection and post-detection channel selection filtering in OOK mode. The bandwidth of the pre-detection filter is configurable by the user, and corresponds (approximately) to the value chosen in the Excel Register Calculator Settings worksheet in the 'OOK RX BW (kHz)' entry cell. The bandwidth of the post-detection filter is not explicitly selectale by the user, but instead is calculated based upon the selected data rate and the specified pre-detection filter bandwidth. The resulting post-detection filter bandwidth is typically much narrower than the pre-detection filter bandwidth.

In EZRadioPRO devices the channel select filtering is performed in the digital domain. The selected channel filter response is determined by configuring three parameters in the RX Modem: FILSET, NDEC, and DWN3BYPASS. (On Si443x devices, these parameters may all be found in SPI Register 0x1C.) The FILSET parameter selects from a set of pre-programmed filter tap coefficients, while the decimation parameters NDEC and DWN3BYPASS parameters select a clock rate at which the digital filter will be clocked. A partial table of the filter bandwidths obtainable from combinations of these parameters may be found in 'AN440 Si4430/31/32 Register Description'. For example, if the user desires an IF filter bandwidth of 350 kHz, the nearest available (equal or larger) bandwidth value in this table is 361.8 kHz for FILSET=9, NDEC=0, and DWN3BYPASS=1.

The pre-detection and post-detection filtering in EZRadioPRO devices in OOK mode shares common FILSET and DWN3BYPASS parameters, but use different NDEC values. The FILSET and DWN3BYPASS parameters are chosen to obtain the selected pre-detection bandwidth, assuming a value of NDEC=0. The same FILSET and DWN3BYPASS parameters are then used for the post-detection filter, but a different (non-zero) value of NDEC is used to obtain a bandwidth appropriate for the selected data rate. The parameter values programmed into the RX Modem are the common FILSET and DWN3BYPASS parameters, along with the larger non-zero value of NDEC corresponding to the post-detection filter bandwidth; the chip automatically uses NDEC=0 for the pre-detection filter bandwidth. As the filter decimation ratio is equal to 2^NDEC, there will always be a power-of-2 relationship between the pre-detection filter bandwidth (selected by the user in the 'OOK RX BW (kHz)' entry cell in the Excel Register Calculator worksheet) and the resulting (typically narrower) post-detection filter bandwidth.