

DSpectrumGUI – Rapid Reverse Engineering Guide

Table of Contents

DSpectrumGUI – Rapid Reverse Engineering Guide	L
Capturing transmissions	2
Preparation / Setup	5
Rapid Reverse Engineering14	ŀ
Transmitting the signal)

by nullwolf

Capturing transmissions

- Capture-1 : Connect the RTL-SDR dongle to your computer.
- Capture-2 : Open a new terminal window and type:

osmocom_fft

Capture-3 : Press enter.

A window similar to the following should have spawned.



Capture-4 : Enter the desired frequency in the Center Frequency field. **Note:** the frequency needs to be off-center otherwise the "DC Spike" (shown in red in the screenshot below) will overlap with the signal we are trying to capture.





With a little trial-and-error, we can determine an appropriate offset to use. In this particular example, we ended up with 434.2Mhz being a good frequency to tune into to capture a clean signal on 434.24Mhz. This placed the signal to the right of the DC Spike without overlapping it.

<u>Tip</u>: Make sure you press the Enter button after editing any field in osmocom, otherwise your changes won't be picked up. If any fields have a pale pink background colour, you haven't pressed the enter button and it will ignore your change.



- Capture-5 : We also need to modify the file name so that it describes which device we are capturing and what button was pressed to avoid any later confusion. Note that the f%F, s%S, and t%T fields in the name record the frequency, sample rate, and timestamp of the transmission respectively. It's a good idea to leave those parts of the filename intact and just replace the string "name" with a more descriptive string. In this case, our full file name was: "/tmp/remote-blue-arm-f%F-s%St%T.cfile"
- Capture-6 : With osmocom appropriately configured, press the REC button, wait for your transmission (if you have access to the remote, push the button you trying to capture now). After the transmission, press the REC button again to stop recording and save the transmission. In our case, the file will be saved in the /tmp/ directory as shown in the screenshot above.

by nullwolf

Capture-7 : Repeat steps Capture-5 and Capture-6 as appropriate. For example, in this scenario we need to capture at least the 'arm' and 'disarm' button presses for both remotes.

by nullwolf

Preparation / Setup

Prep-1 :	Navigate to the DSpectrumGUI web application (e.g.
	http://localhost:3001).

- Prep-2 : Authenticate with the default credentials (username: user@example.com, password: password).
- Prep-3 : Click "New Device" and fill out the form with the information you have on hand about the device you are reversing. You should have at least the Frequency and a Name at this stage. You may also have an FCC ID.

A device should be thought of as a type of device (e.g. Wireless Alarm System).

	wolf	
Welcome, some guy	New Device	
Devices	Name	Window Aleree Sustans Demote
About		wieless Adilli System Remote
Logout	FCC ID	
	Modulation	
	Encoding	
	Frequency	433.92
	Baudrate	
	Notes	
		Submit

Prep-4 : Units are individual instances of the device (e.g. 2 remotes came with this alarm system) that you would like to compare the signals of.

Press the "Add Unit" button.

Prep-5 : Type in a short but descriptive name for your unit. In this example, one of the remotes was blue and the other was silver, so I named the first unit "Remote #1 Blue", and the other unit "Remote #2 Silver".

by nullwolf

/dspectrum/							
Welcome, some guy	Device was successful	y created.					×
Devices							
About	Device: W	ireless /	Alarm System Rem	ote			
Logout	A device is effectively	a category. It sho	ould have at least one unit. If you have mult	tiple units of the same type (e.g. 3 identica	al garage doc	or remotes), they are best	represented as multiple
	units of the same devi	ce.					
	Name	Wireless Alarm Sy	rstem Remote		Activ		
	FCC ID				View F	Reversing Worksheet	
	Modulation				Edit De Go Ba	evice ck to Devices	
	Encoding	422.02 Mbz					
	Baudrate	-33.52 Pill2					
	Notes						
	Unite						
	Units						
	Add new	Rem	note #1 (Blue)				Create
	Name		Notes	Captures		Actions	

- Prep-6 : Click the link to one of the new units you added it should have appeared just below the "Create" button you pressed.
- Prep-7 : On the Unit show page, click "Add New Capture" in the "Actions" block on the right.

/dspectrum/											
Welcome, some guy	Unit: Remo	te #1 ((Blue)								
Devices	This unit is a Wireless Ala	rm System Re	emote								
About	Name	Remo	te #1 (Blue)								
Logout	Device ID	12						Acti	ons		
	Device Name	Wirele	ess Alarm System Rem	ote				Add N Edit U	lew Capture nit		
	Notes								Device		
	Captures										
	ID	Name		Raw Binary				1	Actions		
	Automated De	modulati	ion Attempts								
	Raw (No special e	encoding)									
	ID Name		Raw Binary To Hex		Raw Binary To	Ascii				Actions	
	Manchester Enco Defaulting to Manchester	ding er format: "G.	E. Thomas"							Switch to	IEEE 802 Format
	ID Name		Valid?	Manchester Binary			Hex		Ascli	Actions	
	PWM Encoding Attempts to decode as	PWM 66/33 Fc	ormat (i.e. a 1 is 110, a	and a 0 is 100)							Flip bits
	ID Name		Valid?	PWM Binary		Hex		Asc		Actions	

Prep-8 : An instance of Inspectrum will spawn for you. Open your first capture file for this device.





000		inspectrum	
80	Controls	4 MHz	
	Open file		
		3 MHz	
Sample ra	te: 8000000		
Constrants	_		
Spectrogram FFT si	70'	∠NHZ C < > # ■ 00	
700	m:	Execution Name	
Dower m	······································	1 M-2: Dropbox biver-arm-14.330000e+08-s8.000000e+06-t20170515204537.cfile	
Dowor		All My Files	
Fond	an: 🕅	Cloud Drive	
308	65. sar	yAç Applications	
Time selecti	on	-1MH2 Documents	
Enable curso	irs:	Compared and the second	
Symbo	ls: 1 C	-2 MHz O Downloads	
Ra	te:	(f) acton	
Peri	od:	Deleted Users	
Symbol ra	te:	Devices	
Symbol perio	od:	random	
		-4 MHz @ Remote Disc	
		Macintosh HD 2	
		New Folder Options Cancel Open	

Prep-9: Scroll to the right until you find your signal. You will likely find that the same 'packet' is repeated numerous times. These should be identical. Center one of packets in the screen.



Prep-10 : Drag the "Zoom" slider slowly to increase the size of the packet if necessary.

			inspectrum: blue-arm-f4.330000e+08-s8.000000e+06-t201	70515204526.cfile		
Controls	00 MHz	0.780000	0.790000	0.800000	0.810000	0.820000
Open file						
Complexeter 0000000	3 MHz					
Sample rate: 8000000	100000					
Spectrogram	-					
Spectrogram	2 MH2					
70000	San an air	The second is shown where the	and all the same same		factor and a	
20011	1 MHz		रस्वाल स्वयंत्र में स्वयंत्र स्वयंत्र है।			
Power max:						
Power min:						
Scales: 12	A CONTRACTOR					
Time colocition	-1 MHz					
Enable cursors:						
Combala: 1	-2 MHz					
ayinbois: 1	-					
Rate:						
Sumbol rate:	3 MHz					
Symbol period:	Start Start					
-,	-4 MHz					

Prep-11 : Type a number into the "Symbols" field on the left. If in doubt, choose a low number like 5 – lower numbers are usually easier to start off with. In the picture below, I started with 25.

					inspectrum: bl	ie-arm-f4.3	30000e+0	8-\$8.0000	00e+06-t	2017051520	4526.cfile						
00	Controls	4 MHz		0.780000		0.7900	00				0.800000			0.810000		0.82000	00
0	pen file																
		3 MHz															
Sample rate:	8000000																
Encetrogram		-															
FFT size:		2 MH2															
Zoom		NAME OF STREET			a ciera pina pina												
Power max		1 MHz	B COL BRANCE	and a state					Sonald.				States R.				
Rower min																	
Scalae		Contract and the															
ocures.																	
Time selection		-1 MHz															
Enable cursors:																	
Symbols:	26 0	-2 MHz															
Rate	56.3063Hz																
Period	17.76ms	-9 MH+															
Symbol rate:	1.40766kHz	-5 1112															
Symbol period:	710.4µs																
		-4 MHz															

Prep-12 : Drag the "Power min" slider slowly until the background noise fades away and you are left with a crisp representation of the signal you are interested in, as shown in the below picture.



Prep-13 : Line up the start of the grid (drag the unbroken line on the left of the grid overlay) to line up perfectly with the start of the first pulse. Then increase the number of bits (on the left panel) and drag the unbroken white line on the right of the grid overlay, until you have covered the symbols. Line them up so that the shortest possible symbols/spaces are perfectly encapsulated in a single grid. See below.



Here is another copy of the above picture, zoomed in so that you can see that the grid overly is aligned as perfectly as possible with the symbols.

MHz											
MHz	1			-							_
MHz											

- Prep-14 : Once the grid overlay is as perfect as you can make it, make a note of the Symbol rate on the left side panel (e.g. the Symbol Rate here was 2.214).
- Prep-15 : Right click near the pulses and select "Add derived plot" > "Add amplitude plot".



Prep-16 : A red line should have appeared towards above the pulses. Drag the red line down so that it's sitting on top of your pulses.



Prep-17 : A set of green, vertical rectangles that line up with the pulses should have appeared at the bottom. Right click inside this section and then click "Extract symbols..."



- Prep-18 : Inspectrum will close by itself and you will be presented with a form in DSpectrumGUI. Enter a name that describes the context (in this case, the button that was pressed 'arm').
- Prep-19 : Scroll down and look at the encoding tables. In our example, DSpectrumGUI flagged that the modulation type is likely to be PWM 77/25, however, it seems we needed to capture some of



the "empty space" at the end in Inspectrum for this to be valid PWM. DSpectrumGUI identified this error for us, and corrected for it by adding the required number of 0s at the end.

A correctly taken capture would have included 3 cells containing whitespace (or 0s) at the end, as demonstrated in the pictures below.



Here is a zoomed in picture showing the last few pulses more clearly.



by nullwolf

Prep-20 : DSpectrumGUI told us that the encoding is PWM 75/25, and Inspectrum told us the Symbol Rate is 2.214 once we had the grid perfectly set up. The baudrate is Inspectrum's symbol rate multiplied by 1000. So, the baudrate in this example is 2214.

Update the device field in DSpectrumGUI.

Welcome, some guy Devices	Editing d	evice
About	Name	Wireless Alarm System Remote
Logout	FCC ID	Unknown
	Modulation	оок
	Encoding	PWM 75/25
	Frequency	433.92
	Baudrate	2000
	Notes	

by nullwolf

Rapid Reverse Engineering

Reversing-1: Now that we've got all our captures in place, we're ready to begin the fun part – making sense of the data and learning how to create our own valid transmissions!

Click on the "View Reversing Worksheet" link on the right.

Name		Wire	eless /	Alarm	Syste	em Rei	note																
FCC ID		Unk	nown													A	ction	S					
Modulation		00														E	dit Devic	ersing W :e	orksheet				
Encoding		PWN	1 75/2	25													o Back t	o Device					
Frequency		433	.92 M	hz																			
Baudrate		200	0																				
Notes																							
VM Bit	Pos	sitic	n A	nal	ysis	To	ol (7	/5/2	!5)														
longost bi	nary s	tring	is use	d as t	he co	mpari	son ba	seline	(B) b	elow													
iongest bi												12	13	14			18	19	20	21	22	23	24
longest bi							6		8	9	10												
m							6 1		8	9	10											0	
m arm							6 1 1		8 0 0	9 1 1	10 1 1											0	

The Reverse Engineering Worksheet is similar to the device page, but it only shows us the information we need for manual analysis. It also has some features that allow us to visualize the packet structure and make some notes about our observations.

If the device is a reasonably basic one, we should have enough information at the end of this process to learn how to generate our own valid signals to transmit via a tool called RFCat and our Yard-Stick One device.



Welcome, some guy	Reverse	Engir	ne	erir	na	W	or	rks	sh	ee	t: \	Wi	re	less	A	arn	n Sv	/ste	em	Rer	no	te					
Devices	A Reverse Enginee	ring Workshe	eet		rou	nd the	e con	ncept	of s	ection		sectio	on if	portion o		inary t	hat you	define	to have	e a part	icular	neaning	. E.g. i	if bits 0-	-7 repn	esent t	
About	"Device ID" then y	ou can define		section		flect t	that.		worl	kshee	et the		ry ge	nerates	below i	s base	d on the	modu	lation a		oding t		ed for t	the devi			
Logout	Name	Wireless A	Alar	m Syste	m R	emote	e																				
	FCC ID	Unknown																		Acti	ons						
	Modulation	оок																		Go Ba Edit D	ck to D evice	evice Vi					
	Encoding	PWM 75/2	25																								
	Frequency	433.92 MI	hz																								
	Baudrate	2000																									
	Notes																										
	Sections De	efined																									
	Name	Star	rt P	osition								End	d Pos	ition					Colo	ur			Note	35			
	PWM Bit Po	sition A	na	alysis	Т	ool ((75	5/2	5) -	· wi	th S	Sec	tior	າຣ													
	The longest binary	string is use		s the co	mpa		base	eline ((B) t	below																	
		0		1 2	3	4		5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
	Remote A (Blue) arm id: 67																										

Reversing-2 : Scroll down and start defining sections. As this is an alarm system and the remote has numerous buttons, we can assume we're expecting to find that each remote transmits its own ID, and also changes the signal to represent which button was pressed.

We don't need any of these answers yet though, let's make the table look a little easier to analyse at a glance by defining our sections. When in doubt, it is a good idea to break the packet up into 8bit (1byte) chunks and see if that works out.

In programming languages, counting starts from Zero. In developing this program I have made a decision to do the same. So the first 8 bits starts at bit position '0' and ends at bit position '7'. Fill out the form with this information and pick a colour for each section.

Define new section Pick and choose which capture to definition.	o use as a template for each section and generate the binary. Note for this to work as intended, each bit you want to send should be included in a section
Name	
Start Position (bit)	
End Position (bit)	
Notes	
Colour	gray \$

by nullwolf

In this example, there was a single bit left over at the end. We define a section for that also, making the start position and the end position have the same value (24).

You should now be left with something that looks a little like this.

by nulwolf																											
	Sections Defined																										
	Name	Start Po	sition				End	d Posit	tion			C	olour			Note	IS										
	Section 1	0										In	digo			Brea	iking it	up into	8 bits					Edit Dest			
	Section 2	8					15					da	arkred			Brea	iking it	up into	8 bits					Edit			
	Section 3	16					23					da	arkgree	en		Brea	iking it	up into	8 bits					Edit			
	Section 4	24					24					gr	ray			Just	one bit	t left ov	ver.					Edit Dest			
	PWM Bit Posit	ion An og Is used 0 0	naly as th 1	VSIS ne con 2 0	Too nparis 3	OI (7 son be 4	75/2 aseline 5	25) - ■ (B) □ 6	- wit	h S 8	ect	10 1	11	12	13	14	15	16 0	17	18	19	20	21 0	22	23 0	24 0	
	PWM Bit Posit The longest binary stric Remote A (Blue) arm Id: 67	ion An Ig is used 0 0	naly as th 1	2 0	Too nparis 3	ol (7 son be 4 1	75/2 aseline 5 1	25) - ■ (B) E 6 1	- wit below 7 1	8 0	9 1	10 1	11	12 0	13 0	14 0	15 1	16 0	17 0	18	19 1	20 0	21 0	22	23 0	24 0	
	PWM Bit Posit The longest binary strin Remote A (Blue) arm id: 67 Remote A (Blue) disarm bi: 68	ion An ig is used 0 0 0	naly as th 1 1	vsis ne con 2 0	Too nparis 3 1	Ol (7 son be 4 1	75/2 seline 5 1	25) - (B) b 6 1	- wit below 7 1	8 0	9 1	ions 10 1	11 1	12 0	13 0	14 0	15 1	16 0	17 0	18 1 0	19 1	20 0 1	21 0 1	22 0	23 0	24 0	
	PWM Bit Posit The longest binary strif Remote A (Blue) arm Hi: 67 Remote A (Blue) disarm Hi: 68 Remote B (Silver) arm Bi: 69	ion An ig Is used 0 0 0 0	as th 1 1 1	2 0 0	Too nparis 3 1 1 0	ol (7 son ba 4 1 1 0	75/2 sseline 1 1	25) - e (B) b 6 1 1	 • with the second second	8 0	9 1 1	10 1 1	11 1 1	12 0 0	13 0 0	14 0 0	15 1 1	16 0	17 0	18 1 0 1	19 1 0	20 0 1 0	21 0 1 0	22 0 0	23 0 0	24 0 0	
	PWM Bit Posit The longest binary strif Remote A (Blue) arm H2: 67 Remote A (Blue) disarm H2: 68 Remote B (Silver) arm H2: 69 Remote B (Silver) disarm	ion An ig is used 0 0 0 0 0 0	as the second se	YSIS 2 2 0 0 0 0 0 0 0 0 0 0	Too paris 3 1 1 0 0	ol (7 son ba 4 1 1 0 0	5/2 5 1 1 0 0	25) - (B) E 6 1 1 0 0	view view view view view view view view	8 0	9 1 1 1	10 1 1 0	11 1 1 1	12 0 0 0	13 0 0 0	14 0 0	15 1 1 1	16 0 0 0	17 0 0 0	18 1 0 1 1	19 1 1 1 1 1 1 1 1 1 1	20 0 1 0	21 0 1 0	22 0 0 0	23 0 0 0 0 0 0 0 0 0	24 0 0 0	

Reversing-3 : We start making observations about the data. As we go, we should click 'edit' for each section and update it with a new name and notes as appropriate.

The first section seems to change when a different remote is used, but it doesn't seem to change if a different button is pressed on the same remote. We can deduce that this is the device ID.

The second section seems to also contain part of the device ID.

The third section seems to change when a different button is pressed (e.g. 'arm') but it doesn't change when a different remote is used with the same kind of button. This must represent the Function ID.

Update each section as appropriate.

by nullwolf

	ns Defined																									
Name	Start Po	sitio	n E	nd Po	osition	n Co	olour		Note	s																
Device ID	0		7		indigo			Char	iges b	etwe	en dev	ices; st	tatic acr	oss fur	ictions.	Must b	e devid	e id							<u>dit</u> Jestroy	
Device ID	8		1	.5		da	rkred	ł	Also	chang	ges be	etween	device	es and s	tatic be	etween	functio	ns. Mu	st still t	e the o	levice i	d.			E	<u>dit</u> Jestroy
Function Code	16		2	3		da	irkgre	en	Stati	c betv	ween	device	s as lor	ng as sa	me but	tton is p	pressed	. Must	be the	button	. Lets c	all it th	e funct	ion cod	e. 📴	<u>dit</u> Jestroy
Packet Terminator	24		2	4		gr	ay		This	bit is	alway	/s 0 so	far. W	e're jus	going	to assu	ume it's	a paci	ket terr	ninator						<u>dit</u> estroy
PWM Bit Pos	ition .	Ana sed a	aly:	S iS	Toc	ol (7 on ba:	5/2	2 5) · ≘ (₿)	- Wİ below	th S	Sect	tions	\$													
PWM Bit Pos	ition /	Ana sed a	alys 15 the	SIS com	Toc paris) (7 on bas 4	5/2 seline 5	2 5) -	- Wİ below 7	th S	Sect	10 10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
PWM Bit Pos The longest binary str Remote A (Blue) arm Id: 67	ition /	Ana sed a 0	aly: Is the 1	sis com 2 0	Toc paris 3	0 (7 on bas 4	5/2 seline 5	2 5) - (B) 6 1	- wi below 7 1	8 0	9 1	10 1	5 11 1	12 0	13 0	14 0	15	16 0	17 0	18	19	20	21 0	22 0	23 0	24 0
PWM Bit Pos The longest binary str Remote A (Blue) arm Id: 67 Remote A (Blue) disarm Id: 68	ition ,	Ana sed a 0 0	alys s the 1 1	2 0	Toc paris 3 1	0 (7 on bas 4 1	5/2 seline 5 1	25) · • (B) 6 1	- wi below 7 1	th S 8 0	9 1	10 1	11 1 1	12 0	13 0	14 0	15 1 1	16 0	17 0	18 1	19 1	20 0	21 0 1	22 0	23 0	24 0
PWM Bit Pos The longest binary state arm Id: 67 Remote A (Blue) disarm Id: 68 Remote B (Silver) arm Id: 69	ition <i>i</i>	Ana sed a 0 0	alys s the 1 1 1	2 0 0	Toc paris 3 1 1	4 1 0	5/2 seline 1 1	25) · (B) 6 1 1	- wi below 7 1 1	B 0	9 1 1	10 1 1	11 1 1 1	12 0 0	13 0	14 0 0	15 1 1 1	16 0 0	17 0 0	18 1 0	19 1 1 1 1 1 1 1 1 1 1	20 0 1 0	21 0 1 0	22 0 0	23 0 0	24 0
PWM Bit Pos The longest binary sto arm Id: 67 Remote A (Blue) disarm Id: 69 Remote B (Silver) arm Id: 69 Remote B (Silver) disarm Id: 70	ition /	Ana sed a 0 0 0	alys s the 1 1 1	2 0 0 0	Toc paris 1 1 0 0	I (7 on based 1 1 0 0	5/2 seline 1 1 0	25) - (B) 6 1 1 0 0	- wi below 7 1 1 1 1	b 8 0 0	9 1 1 1	tions 10 1 0 0	 11 1 1 1 1 	12 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	13 0 0 0	14 0 0 0	15 1 1 1 1	16 0 0	17 0 0	18 1 0 1 0 0	19 1 1 1 1 1 1 1 1 1 1	20 0 1 0	21 0 1 0	22 0 0	23 0 0 0	24 0 0 0

Reversing-4: We have another look at the above, and try to look out for any other useful information. We realize there's little point to separating the first two sections, so we delete one of them. We now edit the remaining Device ID section so that it covers the whole 16bit (2byte) range.



Name	Start Pos	tion	End P	ositio	n C	olour		Notes	S																	
Device ID	0		15	indigo			Chan	ges b	etwe	en devi	ces; st	atic acr	oss fun	ctions.	Must b	e devic	e Id						E	<u>dit</u> estroy		
Function Code	16		23		da	arkgre	en	Statio	c betv	ween	devices	s as lon	g as sa	me but	ton is p	oressed	l. Must	be the	button.	Lets c	all it th	e funct	ion cod	e. 🛓	dit estroy	
Packet Terminator	24		24		gr	ау		This	bit is	alway	/s 0 so	far. We	're just	going	to assu	ıme it's	a pac	et tern	ninator.					Ē	Edit Destroy	
PWM Bit Pos he longest binary si Remote A (Blue) arm	tring is use	d as f	ysis the con 2 0	Toc nparis 3	ol (7 on ba 4	25/2 seline 5	2 5) -	- wit below 7 1	th S 8 0	9 1	10 10	11	12 0	13 0	14 0	15	16 0	17 0	18	19	20 0	21 0	22 0	23 0	24 0	
Remote A (Blue) arm Id: 67 Remote A (Blue) disarm Id: 68	cring is use	d as f	ysis the con 2 0 0	Toc nparis 3 1	A 4 1	5/2 seline 5 1	2 5) - (B) 1 6 1	- with below 7 1	8 0	9 1	10 1	11 1	12 0	13 0	14 0	15 1	16 0	17 0	18 1 0	19 1	20 0	21 0 1	22 0	23 0	24 0	
Remote A (Blue) arm Id: 67 Remote A (Blue) disarm Id: 68 Remote B (Silver) am Id: 69	cring is use a a a a a a a a a a a a a	nal dast 1 1	ysis the con 2 0 0	Toc nparis 3 1 1	l (7 on ba 4 1 1	2 5/2 seline 1 1	25) - (B) 8 6 1 1 0	 will 7 1 1 	8 0 0	9 1 1	tions 10 1 1 0	11 1 1	12 0	13 0 0	14 0 0	15 1 1	16 0	17 0	18 1 0 1	19 1 0 1	20 0 1 0	21 0 1 0	22 0 0	23 0 0	24 0 0	
Remote A (Blue) arm Id: 67 Remote A (Blue) disarm Id: 68 Remote B (Silver) arm Id: 69 Remote B (Silver) disarm Id: 70	tring is use	Image: 1 Image: 1 Image: 1 Image: 1 Image: 1 Image: 1 Image: 1 Image: 1	ysis the control 2 0 0 0 0 0 0 0 0	Toc paris 3 1 1 0 0	4 1 0 0	25/2 seline 1 1 0 0	25) - (B) (6 1 1 0 0	 with 7 1 1 1 1 	8 0 0	9 1 1 1	10 1 1 0 0 0	11 1 1 1 1 1 1 1	12 0 0 0 0 0 0	13 0 0 0	14 0 0 0 0 0	15 1 1 1 1	16 0	17 0 0 0 0 0	18 1 0 1 1	19 1 1 1 1 1 1 1 1 1 1	20 0 1 0 1	21 0 1 0	22 0 0 0	23 0 0 0 0	24 0 0 0	

by nullwolf

Transmitting the signal

Xmit-1 :Scroll down until you find the "Generate Binary" form. This
will allow us to generate or own binary by taking "sections"
out of specific captures.

This way, if we had a "disarm" message for Remote B, but only an "arm" message for Remote A, and needed to transmit a "disarm" signal impersonating Remote A, we could just cherry pick the components that we need to form our transmission.

Generate Binary

Pick and choose which captures to use as a template for each defined section and generate the above should be included in a section definition or it won't be represented in the generated stri

Device ID from:	
Remote A (Blue) - arm	*
Function Code from:	
Remote B (Silver) - disarm	÷
Packet Terminator from:	
Remote A (Blue) - arm	+
Generate Binary	

Xmit-2 :Select Remote A's Device ID, and the Function Code from
Remote B, and press "Generate Binary".

We now have the binary, and even the code, that we need to type into RFCat in order to disarm Remote B's alarm system.

by nullwolf

RFCat

d.setFreq(433920000)
d.setMdmDRate(2214)
d.setMdmModulation(MOD_ASK_OOK)
d.setMdmSyncMode(0)
d.RFxmit("\x8e\x8e\xee\x8e\xee\x88\x8e\x88\x88\xee\x88\x80")
or if you need to repeat the packet:

Xmit-3 : Connect the Yard Stick One to your computer.

Xmit-4 : Open a new terminal window and type:

rfcat -r

This should open up an interactive command-line application. DSpectrum has already provided you with the code to use here. You can simply copy and paste it and see if it works.

Go back to DSpectrumGUI's "Generate Binary" result. That page should display the RFCat code required towards the bottom of the page.

- Xmit-5 : Copy and paste the code (the lines should start with 'd'), and paste them into RFCat.
- Xmit-6 :Hopefully, when you hit enter on the final line (d.RFxmit...),
the alarm system was disarmed.





The code we copied and pasted looks similar to the example below.

d.setMdmModulation(MOD_ASK_OOK) d.setFreq(433920000) d.setMdmSyncMode(0) d.setMdmDRate(2214) d.RFxmit('PACKET_AS_HEXCODE_HERE\x00\x00\x00\x00*5)

Here is a very brief explanation of what the above code means.

The string "PACKET_AS_HEXCODE_HERE" is replaced by the hex code you'd like to transmit. The "\x00" is a null byte which effectively represents a gap in our transmission. 4 null bytes is usually a sufficient gap to serve as a packet separator. We are repeating our packet 5 times, because we noted that the packets were repeated by the remote. This happens to increase the likelihood that the transmission will be received by the target device, even if there is momentary interference at that frequency.

The MdmDRate is the baud rate we noted earlier. Inspectrum's Symbol Rate is represented differently than the baud rate as it is expressed in Inspectrum. Shifting the decimal place 3 places to the right of the Symbol Rate value will give us the correct baud rate to use in RFCat.

Setting MdmSyncMode ensures that RFCat doesn't prefix a sync-word to the data we'd like to send out. Sync-words are also commonly referred to as a "preamble". This is used by some devices to let the receiver know to expect a packet. Our example device doesn't have a preamble / sync-word, so we set this value to '0' to prevent RFCat from sending one, which it does by default.