Medical Device Security & Diabetes

Heather Mortensen

Computer Security, SEIS 720, 01

12/05/2018

Table of Contents

Introduction

This paper explores Personal Medical Devices (PMDs) that help regulate blood glucose dynamics. What I lack in conciseness, I hope to make up for in completeness by exploring: 1.) Radio Frequency (RF) communications and attack vectors; 2.) RF interference and its detection; 3.) Major exploitations of hardware and software; and 4.) Challenges for medical device manufacturers. To prepare for this paper, I attended the Wild West Hackin' Fest Conference in Deadwood, South Dakota where I participated in a variety of Radio Frequency labs (Figure 1). I also presented at the MedFuse conference at the University of Minnesota on the topic of diabetes technology.

Figure 1. Heather participating in the RF labs at Wild West Hackin' Fest 2018

Every chronic illness group has their own areas of expertise. No one 'lawyers up' like the Epilepsy Foundation. No one does more, with less, to provide remarkable patient advocacy than the National Alliance for Mental Illness. And, no one does tech like diabetics. The diabetic patient community has the widest variety of hardware, software, and data analysis treatment options available to a single chronic illness community. Thus, diabetics have a unique cultural experience with technology and the many ways that it can save your life when it's at its best and kill you when it's not. This makes diabetes an excellent case study for the development of technological applications for other chronic illness communities who currently lack quantitative data regarding their condition.

A lack of quantitative data makes treatment of an illness difficult, if not impossible. There was a time before quality datasets for diabetes. Today, software is changing the paradigm surrounding health and illness. The diabetic patient community is exerting unique influence that drives technological development. However, as a result, there is increasing burden on individual patients to craft and evaluate software, hardware, data system, and security features of new technology.

The first device of interest in this paper is a Dexcom G4 Continuous Blood Glucose Monitor (CGM) shown in figure 2. It is a wireless embedded system with no internet connection. It consists of an interstitial fluid sensor, transmitter, and receiver. Every 5 minutes, the sensor measures the amount of sugar in the interstitial fluid and sends it, via the transmitter, to the receiver where it can be viewed by the user. It is not a *perfect* real time device as there is a 15-minute lag time between actual glucose in and the sensors ability to read and transmit it. Blood glucose data can age rapidly, making a 15-minute lag significant in some circumstances. Accuracy of the data also decreases at blood glucose values below 70, where accuracy is most important. There isn't much symptomatic difference between a value of 100 and 125, but there may be a large difference in symptoms between a value of 25 and 50. Despite these downsides, Dexcom is the industry leader in sensor accuracy. The newest model (G6) has begun to outperform the traditional industry standard – a manual blood glucose meter. The G6 does not require calibration via manual blood glucose meter checks (which have had at least a 20% variability from actual blood glucose since the 1960's).

Figure 2. Tracy Morgan (type II diabetic) using the Dexcom CGM (Reverberi & Oswald,)

The second device that I explore is an Omnipod insulin pump. It is shown in figure 3. It delivers insulin continuously, similar to an intravenous drip. A traditional insulin syringe can deliver a 0.01 mL dosage. The pump, in comparison, can administer a dose as small as 0.0005 mL. Pump hardware systems that work reliably, deliver many smaller insulin boluses (small, short acting dosages) throughout the day, in contrast to a single basal (large, long term baseline dose) injection via syringe. This affords us the ability to change the rate of delivery throughout the day, in concert with changing hormonal influence, aerobic and anerobic exercise levels, rates of circulation, rates of digestion, nutritional intake, and other variables. Smaller dosages are more effective because the body absorbs them faster. Insulin regularly takes full physiological effect within 2 to 4 hours of delivery. Controls implemented by a patient,

therefore, do not take effect instantaneously and predictive algorithms can be exceedingly helpful when available.

Figure 3. Kris Freeman (type I diabetic) using the Omnipod Insulin Pump

Radio Frequency (RF) Communication Summary

Both devices rely on RF communications. RF transmissions are initiated as a carrier wave, a sinusoidal wave with constant frequency. Device communication bandwidths are highlighted in figure 4. In order to transmit data, the carrier wave must be modulated up to the desired transmission frequency. (Seeber, Dec 17, 2013, min 7:30) The simplest form of RF carrier wave modulation is ON/OFF Keying. An example of this is Morse Code, where the carrier wave is simply turned on and off. (Seeber, Dec 17, 2013, min 7:30)

Frequency range		Type	Center frequency	Availability	Licensed users
6.765 MHz	6.795 MHz	A	6.78 MHz	Subject to local acceptance	FIXED SERVICE & Mobile service
13.553 MHz	13.567 MHz	B	13.56 MHz	Worldwide	FIXED & Mobile services except Aeronautical mobile (R) service
26.957 MHz	27.283 MHz	B	27.12 MHz	Worldwide	FIXED & MOBILE SERVICE except Aeronautical mobile service, CB Radio
40.66 MHz	40.7 MHz	B	40.68 MHz	Worldwide	Fixed, Mobile services & Earth exploration-satellite service
433.05 MHz	434.79 MHz	A	433.92 MHz	only in Region 1, subject to local acceptance	AMATEUR SERVICE & RADIOLOCATION SERVICE, additional apply the provisions of footnote 5.280. For Australia see footnote AU.
902 MHz	928 MHz	B	915 MHz	Region 2 only (with some exceptions)	FIXED, Mobile except aeronautical mobile & Radiolocation service; in Region 2 additional Amateur service
2.4 GHz	2.5 GHz	B	2.45 GHz	Worldwide	FIXED, MOBILE, RADIOLOCATION, Amateur & Amateur-satellite service
5.725 GHz	5.875 GHz	B	5.8 GHz	Worldwide	FIXED-SATELLITE, RADIOLOCATION, MOBILE, Amateur & Amateur-satellite service
24 GHz	24.25 GHz	B	24.125 GHz	Worldwide	AMATEUR, AMATEUR-SATELLITE, RADIOLOCATION & Earth exploration-satellite service (active)
61 GHz	61.5 GHz	A	61.25 GHz	Subject to local acceptance	FIXED, INTER-SATELLITE, MOBILE & RADIOLOCATION SERVICE
122 GHz	123 GHz	A	122.5 GHz	Subject to local acceptance	EARTH EXPLORATION-SATELLITE (passive), FIXED, INTER-SATELLITE, MOBILE, SPACE RESEARCH (passive) & Amateur service
244 GHz	246 GHz	A	245 GHz	Subject to local acceptance	RADIOLOCATION, RADIO ASTRONOMY, Amateur & Amateur-satellite service

Figure 4. Bandwidths for the Omnipod (Yellow) and Dexcom (Blue) (ISM band.2018)

The Omnipod insulin pump transmits from a 433.923 MHz carrier wave that is modulated through Binary Frequency Shift Keying (2-FSK) with a 26.37 KHz deviation. (Figure 5) It has a data rate of 40,625 bits per second. It uses Manchester Coding, of which there are two forms and many variations within the literature. Omnipod uses the G.E. Thomas version, as opposed to the IEEE 802.3 version. (Reverberi & Oswald; OpenOmni; Skorobogatov) I illustrated the RF transmission in figure 6, where I assumed 0 to be a high-low transition and 1 to be a low-high transition. (Schultz, 2018)

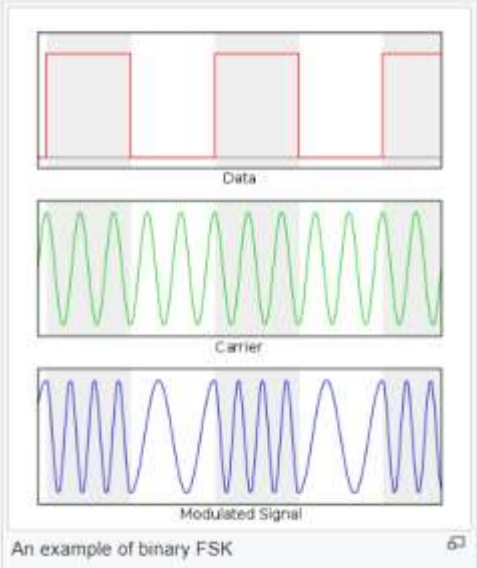
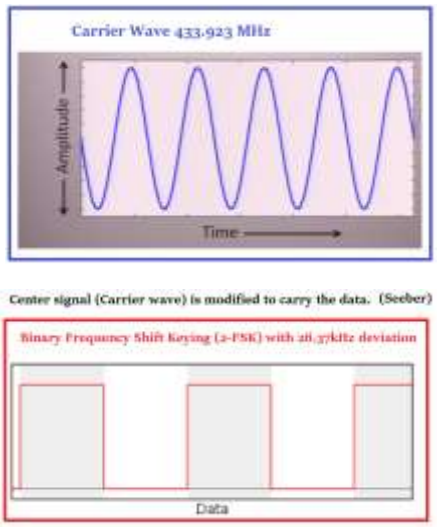


Figure 5. Binary FSK Signal Modulation (Frequency-shift keying.2018)

The Dexcom G4 transmits in the 2.4 – 2.5 GHz Industrial, Scientific, and Medical bandwidth (ISM band). My Dexcom G4 is an older model and is widely known to transmit in plain text, however, "…In 2015, the next generation Dexcom G5 Mobile CGM (G5) system was introduced to the market with further cybersecurity enhancements." (Reverberi & Oswald, 2016) I was unable to identify the nature of these enhancements as there has been little published on newer models. Data transmission consists of four identical transmissions of the data on four different frequencies. Every 5 minutes, Dexcom transmits a blood glucose data reading at 2.425, 2.45, 2.475, and 2.477 GHz using Minimum shift Keying (MSK) modulation. It uses, "…a packet length of 224 bit (including preamble), employs 286.4 kHz channel spacing, and uses a data rate of 49.987 kBit/s." (Reverberi & Oswald, p2) Data travels in one direction, from the sensor and transmitter, to the receiver and uses a rolling code system. This makes it similar to Remote Keyless Entry systems, as noted by Reverberi & Oswald in their paper, "Breaking (and Fixing) a Widely Used Continuous Glucose Monitoring System." (page 2, paragraph 1) For that reason, the next section of this paper explores attack on a common keyfab.



The carrier signal and information signal combine into the modulated signal. (Fletcher, Jan 10, 2018)

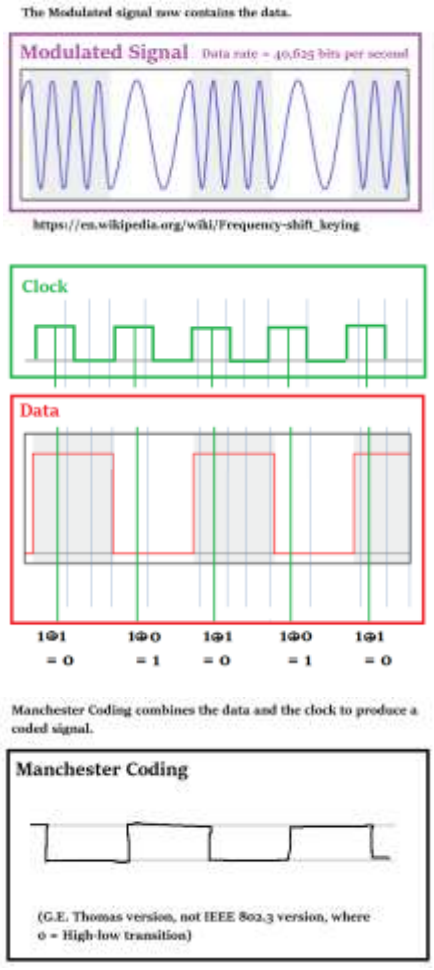
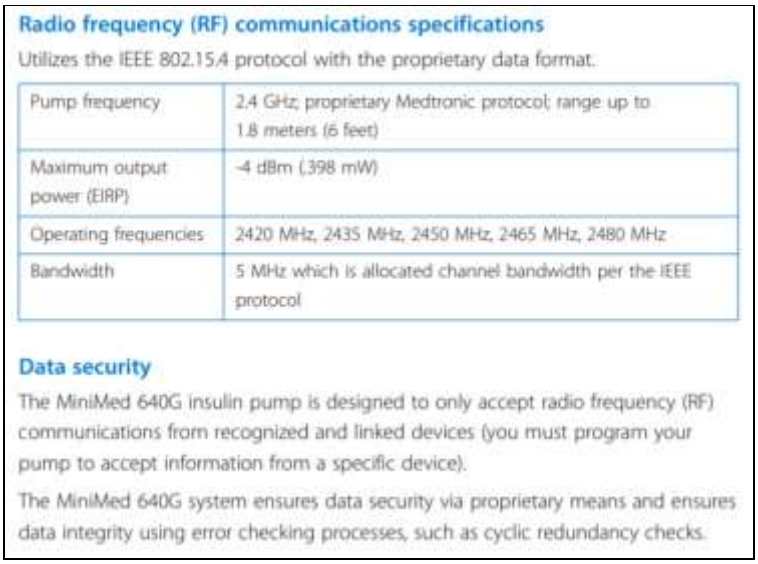


Figure 6. Data Transmission using Radio Frequencies

Why did Dexcom select the ISM band? Why was an alternate bandwidth chosen by Omnipod? Figure 7 displays RF communication details for a 3rd device, the newest model of the Medtronic insulin pump (the 670g). (*MiniMed™ 640G system user guide. Page 271*) It, like the Dexcom CGM and most devices in this space, uses the standard ISM band. Perhaps Dexcom selected this common bandwidth to ease interoperability between devices since Dexcom has, since company inception, striven to partner with other devices. I was approached by a diabetic at the MedFuse Conference who explained that he commonly used Dexcom Developer, software that allows developers access to the API. (https://developer.dexcom.com) (DexCom opens data platform and launches developer program to fuel diabetes app innovation.2017) However, heavy traffic on ISM bandwidth may increase the probability of interference and contribute to the need to send data multiple times in order to guarantee delivery. Channel hoping assists in overcoming lost data due to interference. (433 MHz vs. 2.4 GHz –comparison.)



Radio frequency (RF) communications specifications

Utilizes the IEEE 802.15.4 protocol with the proprietary data format.

Pump frequency	2.4 GHz proprietary Medtronic protocol; range up to 1.8 meters (6 feet)
Maximum output power (EIRP)	-4 dBm (.398 mW)
Operating frequencies	2420 MHz, 2435 MHz, 2450 MHz, 2465 MHz, 2480 MHz
Bandwidth	5 MHz which is allocated channel bandwidth per the IEEE protocol

Data security

The MiniMed 640G insulin pump is designed to only accept radio frequency (RF) communications from recognized and linked devices (you must program your pump to accept information from a specific device).

The MiniMed 640G system ensures data security via proprietary means and ensures data integrity using error checking processes, such as cyclic redundancy checks.

Figure 7. MiniMed 670g RF specifications (MiniMed™ 640G system user guide. Page 271)

Although Omnipod can operate in conjunction with Dexcom, through a middleman software system called Glooko, it has always had very limited cooperative operation with other devices. (Frequently asked questions.) This may, in fact, have something to do with the bandwidth, which is much narrower than the ISM bandwidth. (433 MHz vs. 2.4 GHz –comparison.) Some sources state that operation at the 433 MHz frequency will limit transmission range, compared with ISM. (433 MHz vs. 2.4 GHz –comparison.) The Dexcom CGM delivers data, whereas the Omnipod delivers medication. Therefore, Omnipod is the more critical system for which this shorter transmission range may be a desirable property that provides increased security. However, stronger and longer transmission might also be desirable for a more critical system because it provides more assurance that insulin boluses will be delivered. A contradictory source states that, in general, lower frequencies transmit longer distances, "Yet the tradeoff is antenna size, as the lower frequencies need much longer antennas." (Frenzel, 2008) One benefit of the ISM bandwidth is that rules and regulations regarding operation are consistent worldwide. (Frenzel, 2008) FCC Rules and Regulation regarding the FCC ISM bandwidths can be found at http://afar.net/tutorials/fcc-rules/.

Radio Frequency Interference (RFI) & Detection

I suspected Radio Frequency Interference (RFI) was impacting the ability of my device to function while volunteering at the St Thomas State Fair booth last summer. While there, one of my devices failed. This happens, but is uncommon. One week later, I returned for a second shift. During that shift, both devices failed. That had never happened before. It was so statistically unlikely that I considered the possibility of RFI and wondered how I might detect it, if it existed. I wondered if packet loss, due to RFI, had been the cause that resulted in multiple device failures.

After a decade of working state fair medical aid, I know that Minnesota is the 2nd largest state fair in the country, second only to Texas, which has higher attendance over a shorter duration. That makes it an interesting environment for exploring RFI. Attendance exceeds 200,000 people on weekends. Everything that humans do, you can see them do it at the fair, hence its nickname by medical staff – "The Human Fish Tank." If you're in the right place at the right time, you can see humans give birth, die, or lift police mace off an officer in the crowd and finally settle that lingering 20-year-old grudge. The State Fair is a Twilight Zone of unusual medical happenings. My gut tells me that if RFI were to occur anywhere, it would occur there.

I attempted to imagine what sources of RFI might exist. RFI might stem from the following factors:
1.) Every TV and radio station in the state of Minnesota is present and broadcasting from that location;
2.) Perhaps 200,000 cell phones are confined within a relatively small geographic area (320 acres); and
3.) Heavy police presence (over 70 different agencies) and many firefighters are each carrying a radio. (FAIR HISTORY & ARCHIVES; Nelson, 2018) I was hopeful that RFI might be detectable using Airshark, a software, "…system that detects multiple non-WiFi RF devices in real-time and (by) using only commodity WiFi hardware." (Rayanchu, Banerjee, & Patro,) Unfortunately, Airshark can only identify select devices – such as those listed in figure 8. As sources of interference accumulate, its ability to detect them decreases, "Transmissions from multiple devices that always overlap in time and frequency…can decrease the detection accuracy if the above techniques are used as is." (Rayanchu, Banerjee, & Patro, p9) However, RF interference is detectable in *some* cases, as highlighted below,

"Examples of interference detection are the presence of narrow-band interference (that) appears as a conspicuous spike in the Fourier transform domain; pulsed RFI appears as statistically significant spikes in the time domain; also, many radar returns from the surface have Gaussian statistics due to speckle, and thus the presence of RFI may be detected by signal statistics exhibiting non-Gaussian behavior." (Board on Physics and Astronomy, 2015)

RF device	Airshark-SVM (%) Accuracy/FPR	Airshark-DTree (%)Accuracy/FPR
Analog cordless phone	98.31% / 0.037%	97.73% / 0.012%
Bluetooth (ACL/SCO)	92.03% / 0.094%	91.63% / 0.076%
FHSS cordless phone	98.44% / 0.052%	96.47% / 0.037%
Microwave oven	94.02% / 0.012%	93.16% / 0.06%
ZigBee device	97.49% / 0.048%	96.23% / 0.036%
Video camera	94.24% / 0.08%	92.70% / 0.072%
Audio tx/headphones	92.27% / 0.016%	91.23% / 0.014%
Game controller (Xbox/Wii)	90.32% / 0.064%	91.75% / 0.046%

Table 5: Comparison of SVM and decision tree based approaches. Table shows per-device accuracy in the presence of multiple RF devices. The RSSI of the devices were ≥ -80 dBm.

Figure 8. RF Devices detectable by Airshark (Rayanchu, Banerjee, & Patro, 2011)

Research at the University of Wisconsin Madison led to the development of Airshark. Shravan Rayanchu presents a thorough explanation of their work to Microsoft in the 2012 presentation, "Understanding Wireless Interference in the Unlicensed Band." (A summary of his work is presented at minute 4:34.) Goals during development were to answer these questions: 1.) Is my wireless signal connecting well enough?; 2.) Can we assign a letter grade that describes network quality?; 3.) What is the cause of packet loss when the incidence of packet loss is high?; And, 4.) Can we perform analysis of the network without using custom hardware? (Rayanchu Shravan, 2012)

Airshark is used to detect non-WiFi devices using only a WiFi card available in a typical PC. It helps us to understand the magnitude of RFI that is present. It can identify the type of device that is contributing to RFI and that device's location. However, the typical WiFi card can only process a narrow frequency spectrum of 22 – 40 MHz. To overcome this, they hop around the entire spectrum. This prevents them from seeing everything at once, but provides a small sample of everything. (Rayanchu Shravan, 2012, min 25:50) They have a low sample rate compared with more sophisticated spectrum analyzers. Airshark can detect around ten devices at once. Too many devices will crowd the spectrum. This appeared to be the largest impediment to identifying sources of RFI at the state fair. If a device experienced RFI, it was most likely to occur in environments where the spectrum was crowded and noisy, which simultaneously reduced the likelihood that a specific source could be identified.

Newest models of the Dexcom CGM and Omnipod will use Bluetooth communications. Shravan described detection of Bluetooth signals as "problematic." Bluetooth is a frequency hopper, which makes detection more difficult. Bluetooth detects other signals and hops over them. A crowded spectrum could impact its ability to function. I had hoped that a Dexcom CGM and insulin pumps could be added to the list of identifiable devices by Airshark. Incorporation of medical devices might have value for hospital systems, like the Mayo Clinic, who are working diligently to secure a hospital networks that consists of thousands of devices. Unique characteristics of different devices are used to identify them. Airshark used 'off the shelf' software to, "…capture the temporal and spectral properties of different devices." (Rayanchu Shravan, 2012, min 40:20) However, limitations on Airshark's accuracy in crowded spectrum environments cannot currently be overcome and may require hardware that is specialized, expensive, and very technical to operate.

In general, the presence and influence of RFI may be undetectable for me, personally, in any practical scenario. The Omnipod insulin pump is more likely to fail as a result of RFI, due to characteristics of the bandwidth and its inability to frequency hop to evade interference. However, its bandwidth is far less cluttered than ISM. Considering the Dexcom CGM, The National Academies of Science, Engineering, and Medicine write about sensor technology and RFI,

> **"The sensitivity of a given sensor to RFI is also a function of the specific nature of the RFI experienced. The existence of a wide range of active sensor techniques, as well as a wide variety of RFI environments encountered globally, conspire to make it difficult to characterize the RFI problem and the associated solutions."** (Board on Physics and Astronomy, 2015)

As an increasing number of consumer devices crowd the bandwidth that medical devices operate in, they might be increasingly likely to fail from RFI. The bandwidths used by these devices do not take priority according to FCC guidelines. The Dexcom has the same priority as a cell phone, which may be one reason that frequency hopping is employed. The business trend seems to be pushing devices into Bluetooth in an effort to increase reliability. The FDA is currently studying patient experiences

associated with Bluetooth technology. (https://forum.tudiabetes.org/t/study-volunteers-needed-from-fda/74456) The current Radio frequency Allocations dictated by The US Department of Commerce are shown in figure 9.

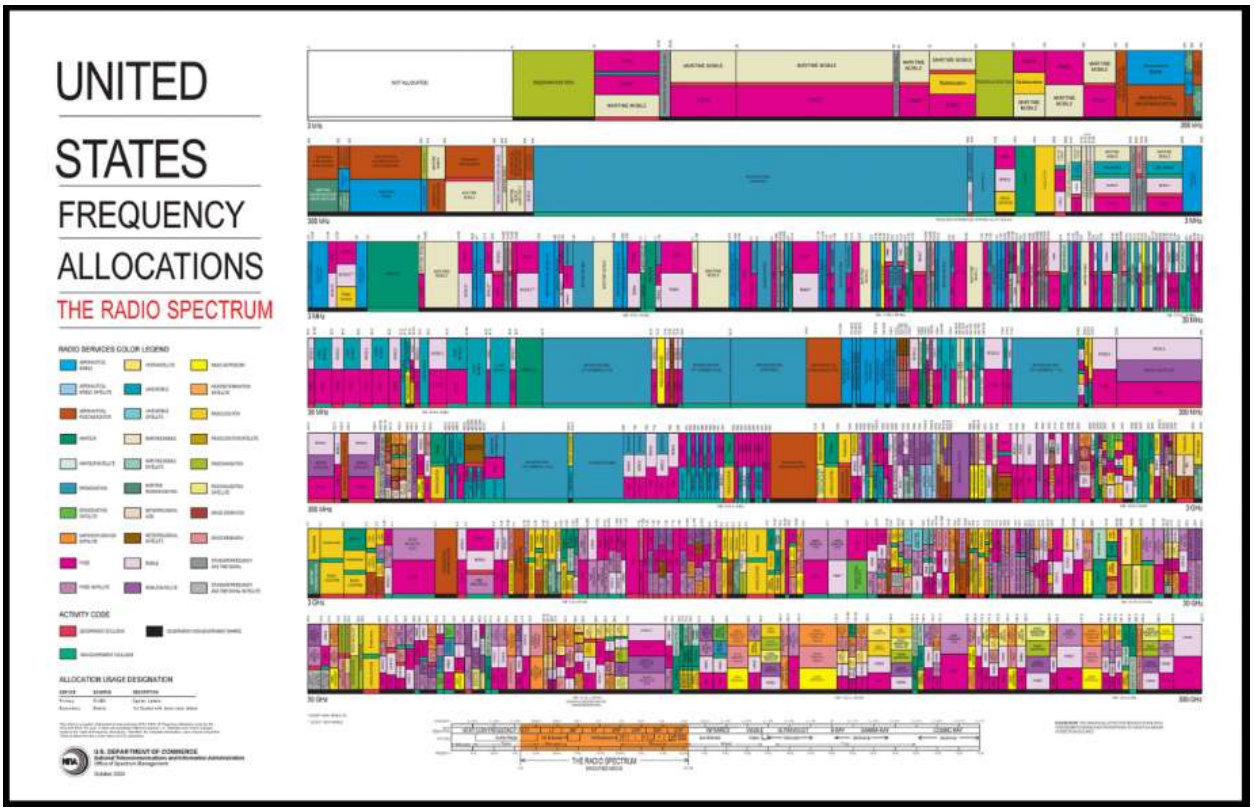


Figure 9. US Radio Spectrum Allocations (United states frequency allocations radio frequency spectrum.2003)

Black Hills Security KeeFab Attack

I took a more detailed look at RF communications in the Keeloq FOB Attack Lab presented at the Wild West Hackin' Fest Conference by Black Hills Security. The purpose of the lab, summarized here, was to verify whether the device was subject to replay attacks, or whether a rolling code algorithm provided sufficient protection. The lab helped participants explore three types of signal modulation through hardware exploitation. (Fletcher, Jan 10, 2018) "The three basic transmission modulation techniques relevant to this lab are amplitude modulation, frequency modulation, and phase modulation." (Fletcher, Jan 10, 2018) However, some authors first classify modulation types as analog or digital.

Amplitude modulation modifies the amplitude of the carrier signal with the data carrying signal, termed the information signal. (figure 10, part a) Frequency and Phase modulation, alternately, use the carrier signal to modify the frequency of the information signal. (figure 10, parts b and c) Several varieties of phase modulation include: 1.) PM (Phase Modulation); 2.) PSK (Phase Shift Keying); and, 3.) BPSK (Binary Phase Shift Keying). The relationship between phase and frequency modulation is identified, "Phase and

frequency are inseparably linked as phase is the integral of frequency. Frequency modulation can be changed to phase modulation by simply adding a CR network to the modulating signal that integrates the modulating signal." (Poole,) I believe that CR in this context refers to a Cognitive Radio network. The relationship between all three variables is shown in figure 10, part d. (Introduction to modulation and different types of modulations.)

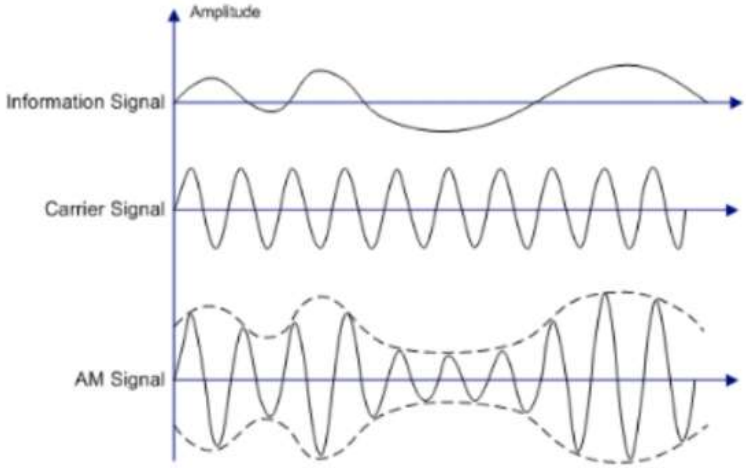


Figure 10, Part a. Amplitude Modulation (Fletcher, Jan 10, 2018)

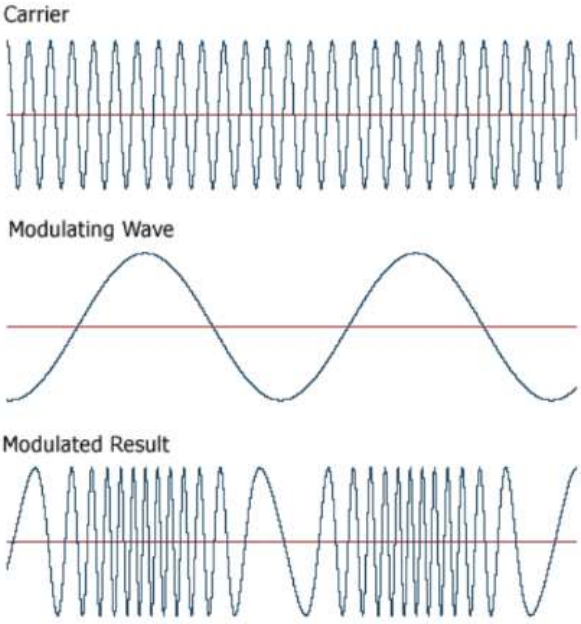


Figure 10, Part b. Frequency Modulation (Fletcher, Jan 10, 2018)

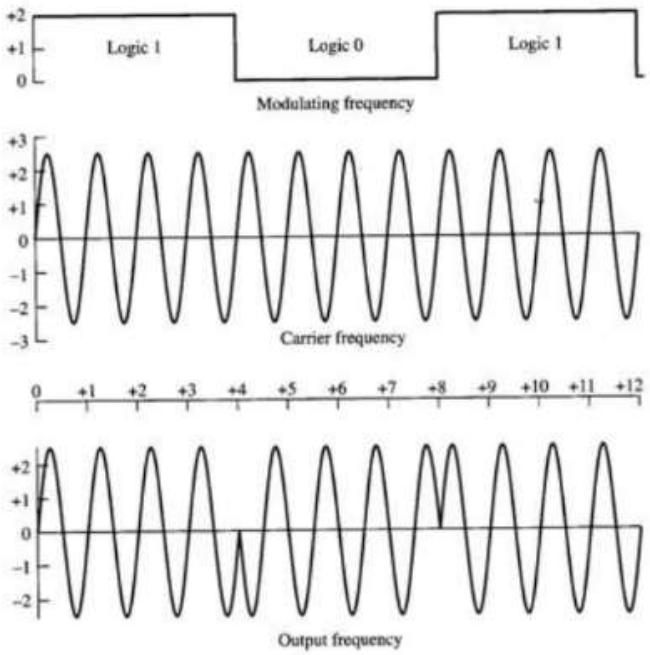


Figure 10, Part c. Phase Modulation – Binary Phase Shift Keying for digital signals (Fletcher, Jan 10, 2018)

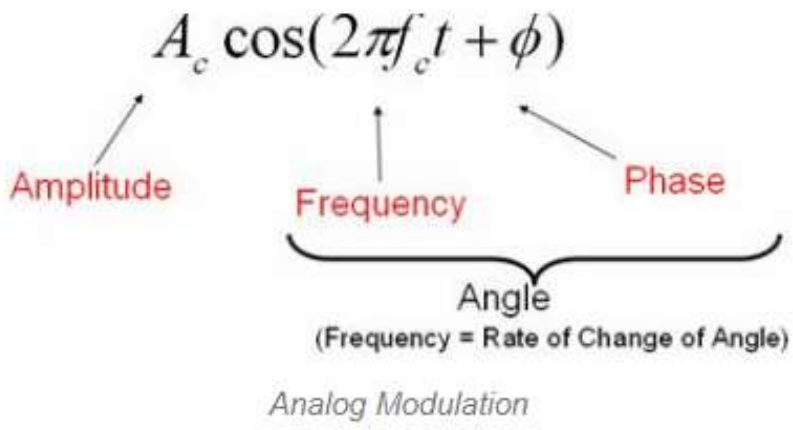


Figure 10, Part d. Relationship between amplitude, frequency, and phase (Introduction to modulation and different types of modulations.)

Figure 10. Three types of signal modulation

On/Off Keying (OOK) was selected for the keyfab device because it only transmits a limited amount of data. It is categorized as a form of amplitude modulation. OOK was used in combination with Pulse Width Modulation encoding and a rolling code security algorithm. Fletcher describes why,

> **"Pulse width modulation is a technique to encode the On-Off Keyed signal to form symbols. The width of the pulses and gaps are interpreted by the receiver to recover the transmitted data…Modern fob transmitters that are used for security purposes (garage door openers and vehicle remote systems) employ a rolling code algorithm to protect the target devices from simple replay or static key discovery attacks."** (Fletcher, Jan 10, 2018)

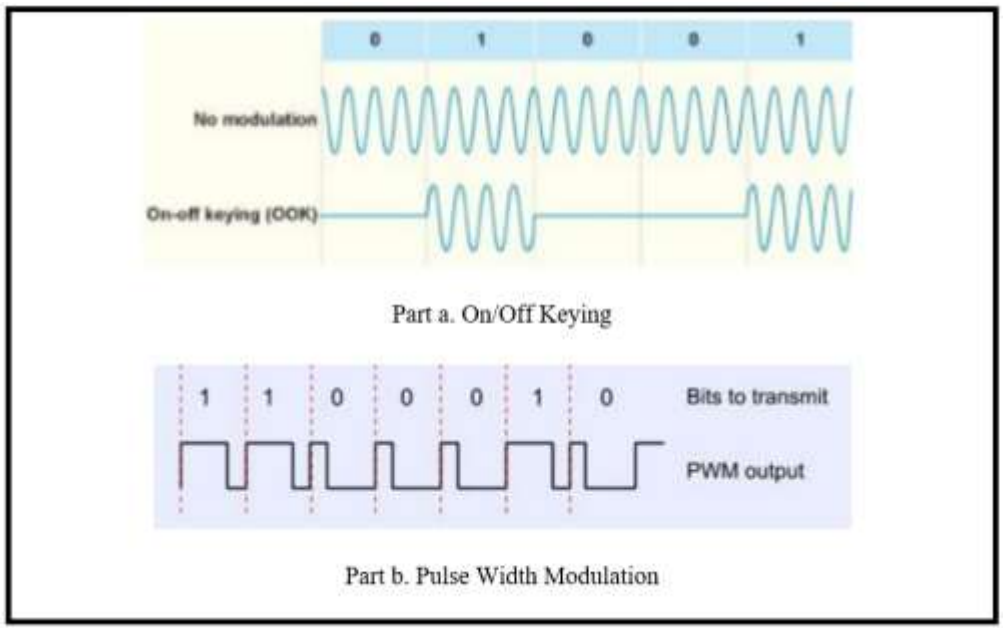


Figure 11. KeyFab Signal (Fletcher, Jan 10, 2018)

Clock synchronization between the transmitter and receiver is critical to the rolling code algorithm. (Fletcher, p4) Details regarding Rolling Code Keeloq can be found in the slides authored by Eisenbarth & Kasper, in the references. It is also shown in figure 12. Fletcher describes synchronization as, "…similar to setting the seed value on a two-factor authentication token." Future key values are generated from a seed value known by both the transmitter and receiver. If signal is lost, then the number of transmissions exceeds the number of transmissions received, and values are deemed to be invalid, "…the two must be re-synchronized to set a matching seed value."

This lab used circuitry from MicroChip Technology Inc, "…to implement rolling code transmitter/receiver pairs for various purposes." A complete list of hardware and software components is included in Appendix A. Fletcher's hardware setup can be viewed there, in Figures A2 and A3.

The first portion of the lab, "…illustrates the code hopping feature through reception and decode of multiple transmitted codes. After confirming that the system does in-fact employ code hopping, we will also confirm that code replay does not result in execution of the targeted feature." (p10) The lab keyfab uses a, "MicroChip HCS200 integrated circuit." (Fletcher, Jan 10, 2018, p8) The associated FCC ID was

located (FCC ID KR55WK48801) and its corresponding data sheet retrieved from http://fccid.io. This provided valuable details regarding the, "transmission frequency, internal photos, and circuit schematics." An ID was not identifiable from the transmitter, so alternately, they opted for, "…stepping through candidate frequencies while operating the device to determine the transmission frequency." (Fletcher, Jan 10, 2018, p8) We will want to find the ideal receiver operating characteristic. If we set the radar too high, our signal will be cluttered. If we set it too low, then we will not see the signal. (Anderson, 2008) Inspection of the chip (MicroChip HCS200 integrated circuit) inside the device and its corresponding data sheet also provided helpful information (see Appendix A, Figure A4).

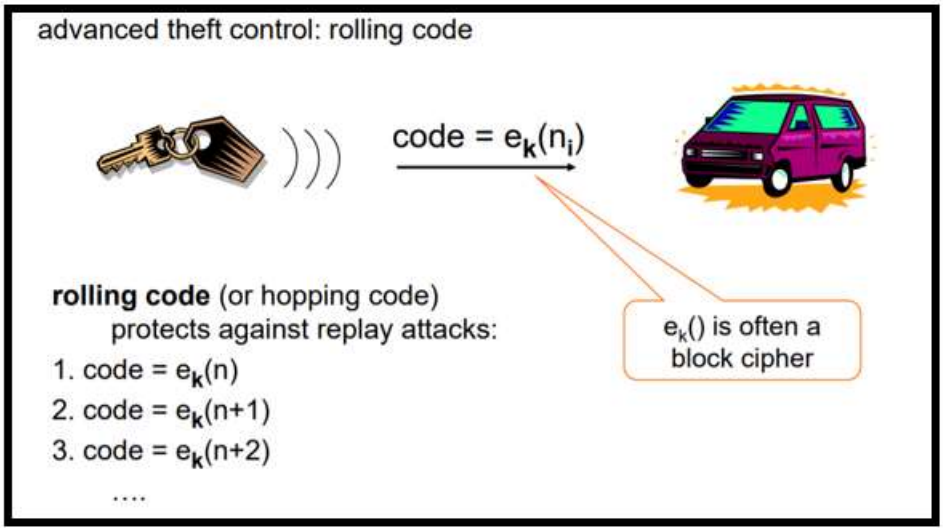


Figure 12. Rolling Code Algorithm (Eisenbarth & Kasper, 2008, Slide 5/50)

In order to verify whether a replay attack is possible, "The RTL-SDR will be used to receive our transmitted signal and the YardStick One will be used to replay the signal…" They are both plugged into the USB ports of different computers. (Fletcher, Jan 10, 2018, p11) We must run the GQRX GUI application. GQRX is an open source software defined radio for producing FFT (Fast Fourier Transforms) plots and waterfalls. (Csete, 2011; Information about FFT in spectrum lab.) We tune our receiver to test the most common frequencies used by keefabs, known to be 315 MHz and 433 MHz. The waterfall should appear at 315 MHz. Set the dB range parameter, "…to be about 75 dB. The display should turn blue and this will make it easier to distinguish the transmitted signal from noise." Next, "…select the Receiver Options tab and set the mode to AM. This will demodulate the AM transmission, so that we can observe the OOK PWM signal. As a side effect, we will also hear the pulses generated by the transmitter as an audible signal." (Fletcher, Jan 10, 2018, p13) The receiver is now configured.

Pressing a keyfab button should activate the light that corresponds to that button on the LED receiver. However, we will only see a spike on the GQRX display if our radio is tuned close enough to the frequency that the device is transmitting on, in this case 433 MHz. Notice that this bandwidth is the same used by the Omnipod insulin pump described earlier in this paper. Since we are near enough the transmission frequency, we ought to see: 1.) A spike on the frequency plot; 2.) A bright yellow line, and 3.) A red line in the FFT waterfall (figure 13). Although we can see that our radio is coarsely tuned, some fine tuning will improve the output. To do this, we line up the peak with the center frequency of the FFT

waterfall. By doing this we see that a more accurate frequency is 433.850 MHz in figure 14. (Fletcher, Jan 10, 2018, pp15-16)

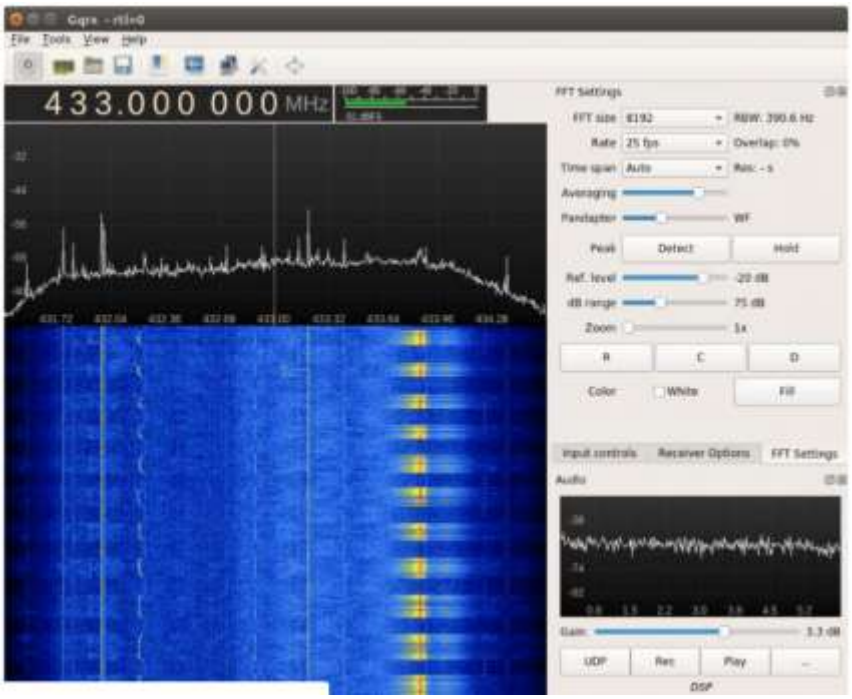


Figure 13. GQRZ FFT waterfall coarsely tuned (Fletcher, Jan 10, 2018, p15)

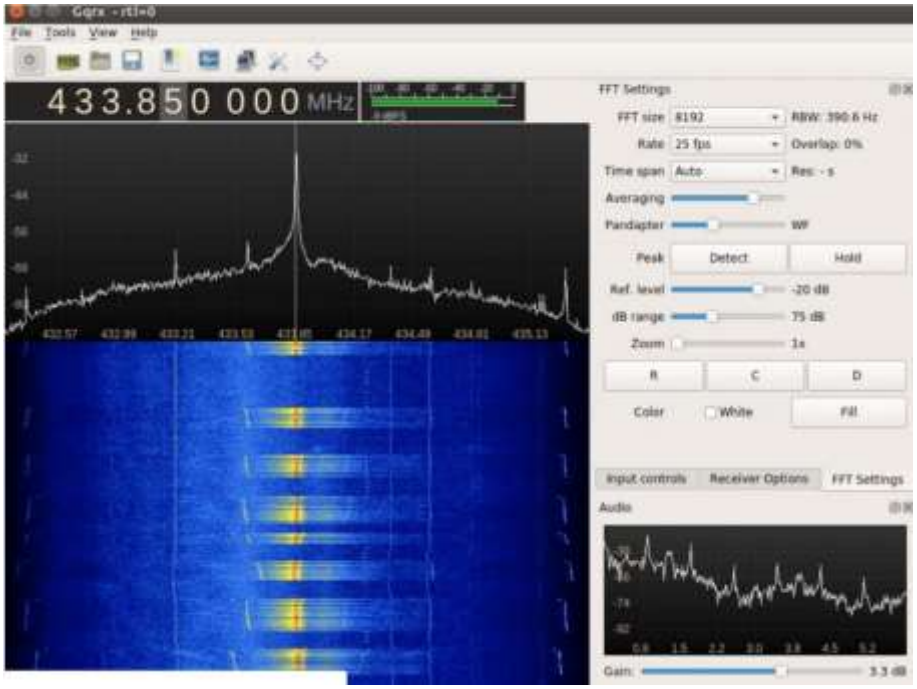


Figure 14. GQRZ FFT waterfall finely tuned (Fletcher, Jan 10, 2018, p16)

Clicking the Record button in Gqrx and pressing the keyfab button multiple times will record, "…the demodulated waveform to a "WAV" file which can be launched using Audacity, an audio software. For details see Fletcher's lab publication, page 16. The code transmitted during each of the button presses must be analyzed to confirm that code hopping is taking place. Button presses appear in figure 15.

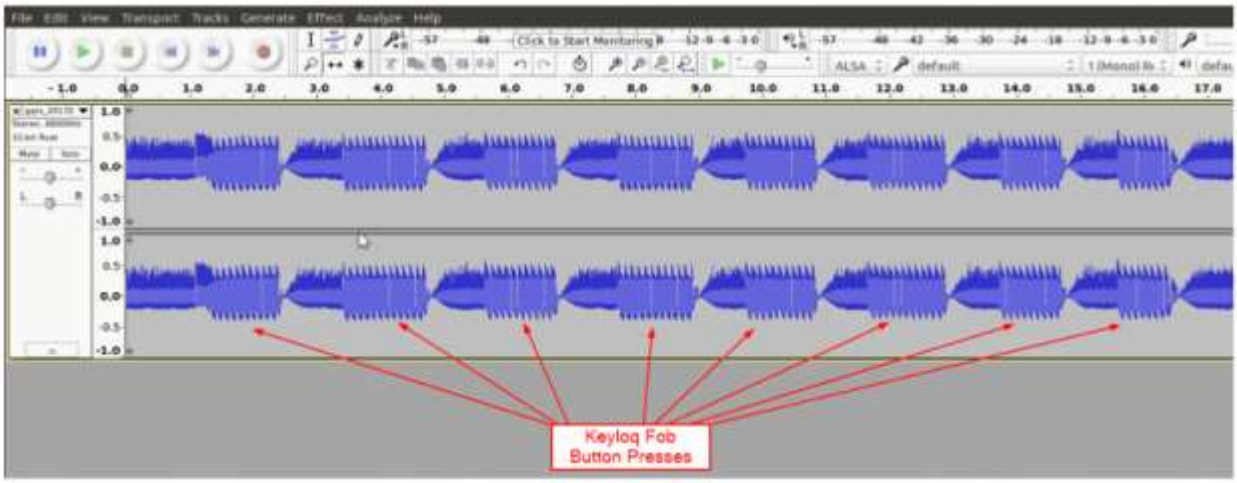


Figure 15. Button presses appear in Audacity (Fletcher, Jan 10, 2018, p17)

An individual button press appears in figure 16. Each button press transmission includes: 1.) A sync word; 2.) The data stream sent to the receiver; 3.) A Pulse Width Modulated (PWM) digital signal that must be decoded into binary and sent to YardStick One. We must attempt to identify underlying patterns in the transmission by using Audacity and then associate them with 0's and 1's. (Figure 17) Fletcher guides us to identify, "…two distinct symbols. The first is a long pulse followed by a short gap and the second is a short pulse followed by a long gap." (Fletcher, Jan 10, 2018, p18)

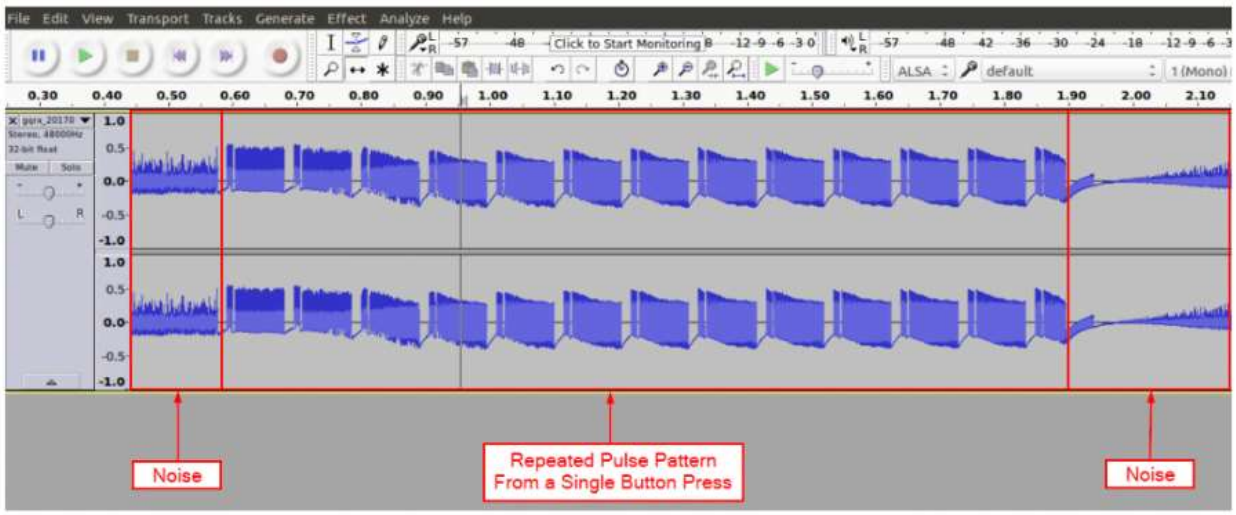


Figure 16. Individual button press (Fletcher, Jan 10, 2018, p17)

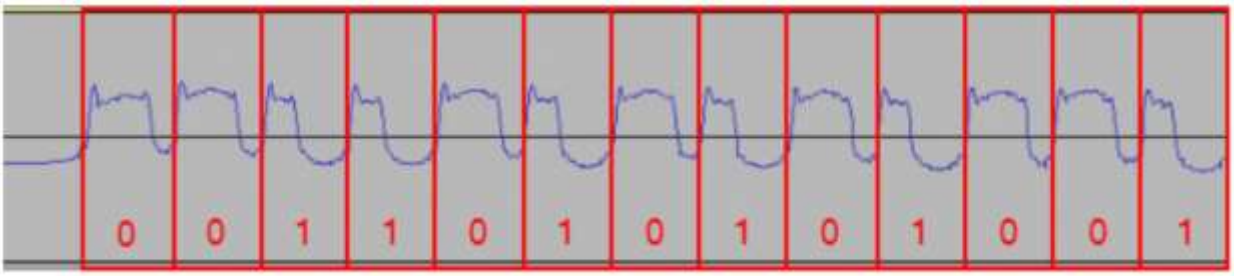


Figure 17. Underlying transmission pattern identified using Audacity (Fletcher, Jan 10, 2018, p19)

Ooktools is a software suite that can relieve us of the need to interpret the entire pattern by hand in order to verify that the rolling code pattern, implemented for security, is working correctly. Ooktools is a primary tool for decoding On-Off Keying bit patterns from PWM signals. We do this by using the Ooktools 'Split Stero to Mono' function and selecting an entire, single section of the PWM code. First, we select the path to our WAV file. Then, open a terminal and run the "ooktools wave binary --source button1_1.way" command. We should see the key data output in the terminal, as shown in figure 18. (Fletcher, Jan 10, 2018, pp19-20) This is repeated for each time the button was pressed in order to compare their outputs in figure 19.

```
root@fieldxpsdr:/home/fletch# ooktools wave binary --source button1.wav
 _ _ _
 ___ ___| |_| |_ ___ ___| |___
| . | . | '_|  _| . | . | |_ -|
|___|___|_,_|_| |___|___|_|___| v1.3
On-off keying tools for your SD-arrrR
https://github.com/leonjza/ooktools

Total Samples: 4649, Min: -11525, Max: 16843, Mean: 2659.0
Cleaning up 4649 data points...
Samples in (Shortest Peak: 17) (Longest Peak: 39)
Math for baud rate will be 1.0/(17/float(48000))
Source wave file has baud rate of: 2823
[ ] indicates number of breaks.
Key Data: 1111111111110001100111011011010111011010110111001111110001100100000000000010011
root@fieldxpsdr:/home/fletch#
```

Figure 18. Key data output of the command "ooktools wave binary --source button1_1.way" (Fletcher, Jan 10, 2018, p20)

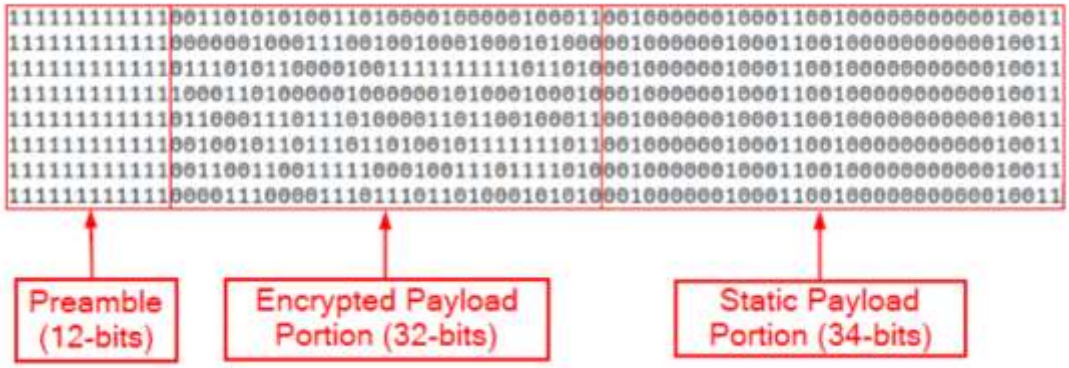


Figure 19. key data output for eight button pushes (Fletcher, Jan 10, 2018, p21)

Fletcher notes that, "…the encrypted payload portion of the transmission does in-fact change with each successive button press." He then uses a python script written to replay the codes captured in figure 19 to see if the button functions are also executed. The script is shown in figure 20. The script performs 3 functions, including: 1.) Configuration of the radio; 2.) Expansion of the binary key into a PWM values, which is converted into hexadecimal; and, 3.) Transmission of the key on the 433.85 MHz frequency using RFCat. In order to run the script, we must know the values of three variables – transmission frequency (433.85 MHz), baud rate (25 fps), and the value of the key (k1 was shown in figure 18).

Figure 20. Fletcher's script for replay attack (Fletcher, Jan 10, 2018, p22)

I used the KeyFab lab as an introduction to understand the Radcliffe Medtronic Insulin pump hack summarized in the following section. Both utilize similar attack methodology on an RF signal, although they have different degrees of complexity. The Keeloq brand keefabs, like the Omnipod insulin pump, operates in the 433 MHz frequency bandwidth. (Fletcher, Jan 10, 2018, p8) We can also draw analogy to efforts by Open Omni to attack the Omnipod Insulin pump. The OpenOmni Github Wiki documentation also identifies use of Binary Frequency Shift Keying (2-FSK).

Details of the original Radcliffe hack

In the 2011 Black Hat Conference titled, "Hacking medical devices for fun and insulin: Breaking the human SCADA system," Jay Radcliffe demonstrated attacks against the Dexcom CGM and a Medtronic insulin pump. His work is summarized in the book, "Security and Privacy for Implantable Medical Device," as "…binary analysis, the ability to disassemble a system's software…(to) completely understand its operation. By inspecting the Java-based configuration program supplied with his own insulin pump…(he) reversed engineered the pump's packet structure, revealing that the pump failed to encrypt the medical data it transmitted or to adequately authenticate the components to one another." (p161)

Medtronic Insulin Pump

He explored the hypothesis that wireless communications are bidirectional and subject to attack by investigating configuration settings in the Java log files. He found that the jar files were not obfuscated. Packet structure, encoding, and decoding were all clearly documented in the comments. RF communications could not be turned off. The most important piece of information required to communicate with the pump was the serial number. He found that the device did not perform any verification procedures to implement new commands. The pump required configuration to communicate with other wireless devices by adding their unique ID. (Radcliffe, min 44) Unique IDs and serial numbers could be obtained by scanning wireless communications. Radcliffe purchased a remote control for his insulin pump from ebay for $20 and confirmed that the RF dongle could be used to communicate with the pumps serial port through RF. (Radcliffe, min 46:45) This was identified as a security risk.

Radcliffe believes that as device functionality becomes more automated and closed loop, the user is less involved in managing device functionality, and as a result, might overlook failures when they occur. As users become increasingly disjointed from device operation, they are less likely to notice device actions that may cause them harm. This is a particular concern for pacemakers, which are almost fully automated and require very minimal user input or attention. The Medtronic pacemaker hack of this summer is discussed later in the paper.

Efforts to place security controls on the device include efforts to limit the transmission distance of the device. Omnipod and Dexcom CGMs will transition into Bluetooth communications. Radcliffe discusses some of the risks, benefits, and security features of Bluetooth at minute 52 of his presentation.

> "**…radio waves in a band of 79 different frequencies (channels) centered on 2.45 GHz, set apart from radio, television and cellphones, and reserved for use by industrial, scientific and medical gadgets. Bluetooth's short-range transmitters have very low power consumption and are more secure than**

wireless networks that operate over longer ranges, such as Wi-Fi." (Comparing wireless communication protocols.2018)

Additionally, as upgraded device models transition communications to Bluetooth, they will use RF chips with built in cryptographic functions. Radcliffe advises the use of Infrared as opposed to RF because it is easy to disable. There is a shorter range of communications. He supports the use of passcodes and verification methods, which as noted elsewhere in this paper, may not be practical or safe to implement in the context of medical devices. Radcliffe mentions a necklace that stops RF communication. He says that many SCADA systems could be attacked by using similar methods.

Dexcom CGM

Radcliff used the FCC disclosure to verify, "unidirectional communication, like a UDP packet." (Radcliffe, 2012, min 17:00) Frequency, Bandwidth, power, modulation type, packet size, and that transmission from the sensor occurs once every 5 minutes. The transmission was captured and translated into binary code. He used the FCC transmitter ID to obtain reports that showed analysis of the signal from Spectrum analyzers and oscilloscopes. (min 18:30) Signal dissection was illustrated in his slides and is included in Appendix B. He read the CGM manufacturer patent to obtain additional information. He physically disassembled the receiver and obtained archived data sheets regarding the wireless chip, which was also used in a variety of SCADA systems. He used two Arduino boards that he describes as capturing a wide spectrum of RF (300 Mhz-900MHz) – an RFM22B by HopeRF and a CC1101 by Texas Instruments. (Radcliffe, minute 20:30) Radcliff reported that his CGM used simple On/Off Keying for signal modulation. That implies that baud rate was equal to the gross bit rate, 8192 bits per second. He reported that every 9 milliseconds, the device transmitted around 76 bits. (Baud.2018; Radcliffe, minute 24:00)

Radcliff describes obstacles in his transition from programming on the software side to programming on the hardware side. In order to program the board's RF deck to receive information, he had to set over 100 registers according to the specifications listed in the manual. He lost time from a common problem encountered by software side programmers when beginning in hardware - a lack of reported error messages. (min 22:00) He describes this as a factor contributing to the rise of hardware hacking.

Radcliffe faced an unknown preamble, syn word, and CRC type and location to verify the integrity of the received packet. (min 25:00) Fortunately, he was able to output the signal module to an oscilloscope that depicted the clock and the data outputted by the device. See figures A and B. Radcliffe was able to obtain 9.3 millisecond transmissions that were five minutes apart. Radcliffe's knowledge of the dataset told him that the data in successive transmissions would change very little. This was confirmed in the data by observing forty successive transmissions and noting that 80% of the 24 bits representing the blood glucose data were identical. (min 31)

Radcliffe then used a replay attack on the packet, which disabled the CGM. It perceived interpreted this as error and asked him to re-calibrate the device. He tried to attack it using Denial of Service, which disrupted data transmission. A simplified introduction to replay attack can be found in the works cited, titled "YardStick One Replay Attack Lab." (Fletcher,) Radcliffe also identified injection attack as a risk. It allows invalid data to be received.

Although Radcliffe, currently employed by Boston Scientific, first publicly discussed Medtronic insulin pump and Dexcom CGM hacking in 2011, his work has been continued by a number of interested parties and researchers. Three years after the Radcliff presentation, "Security and Privacy for Implantable Medical Devices," was published and provided more current information on pacemaker and diabetes devices (they provide the illustration of pump communication packet structure in figure 21). The most recent publication was last year on the Decom G4, titled "Breaking (and Fixing) a Widely Used Continuous Glucose Monitoring System." By this time, the G4 is far less widely used than it was and has been replaced by several more recent models. The G4, a traditional embedded system, is becoming less common and newer systems are increasingly incorporate mobile technology. There are common complaints regarding shortened sensor life and hardware failure in newer models. (cauthren, 2018; Mila, 2018) There have been no studies published on newer models, and as new studies are still being published on the older G4, it remains the only model fit for full exploration.

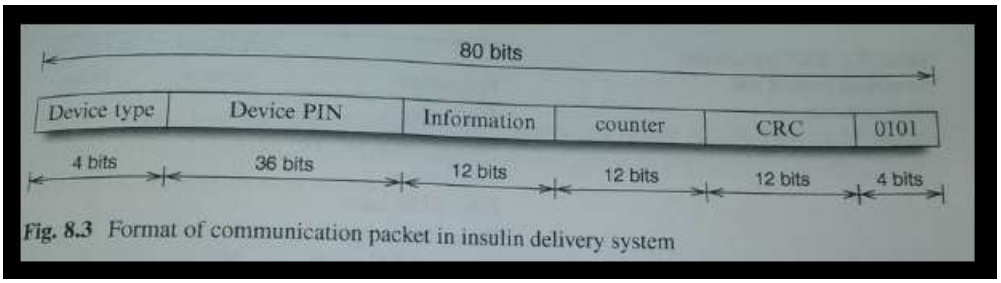


Figure 21. Medtronic insulin pump communication packet (Burleson & Carrara, 2014)

Extension of the Radcliffe Findings

The paper, "Breaking (and Fixing) a Widely Used Continuous Glucose Monitoring System," is from 2017 and identifies risks in addition to those identified by Radcliffe. They summarize their findings as follows,

> **"We practically demonstrate a series of security issues in this device that enable, amongst others, the tracking of a user and the forging of incorrect sensor readings. The attacks can be carried out at minimal cost using software-defined radio and low-cost RF chipsets. Finally, we devise and practically implement an efficient protocol based on best practices and well-known crypto algorithms to mitigate the weaknesses we discovered."** (Reverberi & Oswald, 2016)

The authors identify one freeware solution for converting Dexcom communications to Bluetooth on Github (although this is no longer maintained and control has been transferred to NightScout). Hardware requirements and complete documentation is listed on his GitHub page. (https://github.com/StephenBlackWasAlreadyTaken/xDrip/blob/gh-pages/hardware_setup.md) Since the authors could not gain access to the firmware, so they instead used Software Defined Radio that relied on GNU radio and HackRF to analyze communication protocols. I was introduced to SDR through the conference proceedings and labs conducted at Wild West Hackin' Fest 2018, including "Building a Small and Flexible Wireless Exfiltration Box with SDR," by Paul Clark. (2017 wild west hackin' fest labs. 2017; Clark, Oct 26, 2018) Transmitter tear down and packet structure are shown in figures 22 and 23. The researchers successfully brute forced the Dexcom Checksum algorithm to obtain the packets shown in

figure 24. Their analysis showed that no cryptography mechanisms are implemented. “With the knowledge of the CRC algorithm used on the Dexcom protocol level, an adversary can easily generate a valid packet with freely chosen contents.” (Reverberi & Oswald, 2016)

Figure 22. Dexcom G4 Transmitter (Reverberi & Oswald, 2016)

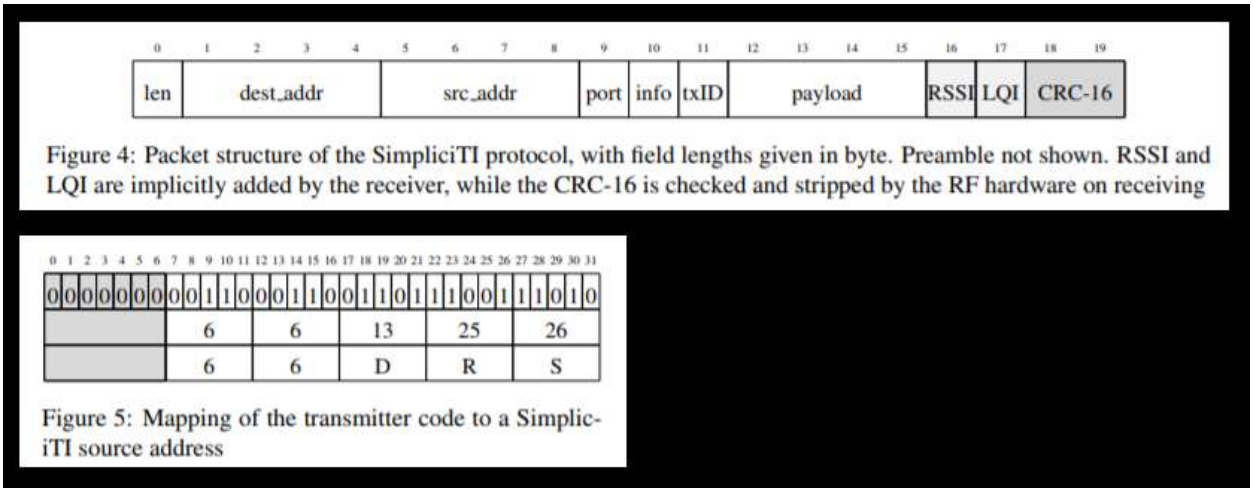


Figure 23. Dexcom G4 packet structure is defined by a Texas Instruments protocol called SimpliciTI (Reverberi & Oswald, 2016)

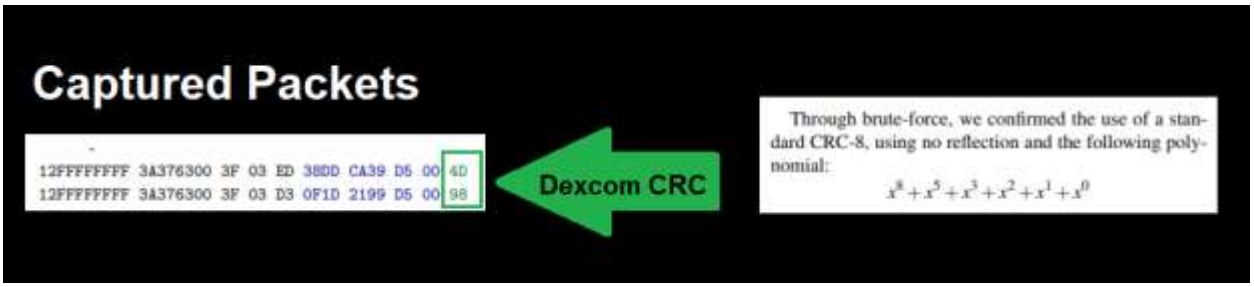


Figure 24. Packets captured from the Dexcom G4 (Reverberi & Oswald, 2016)

The first attack mounted against the Dexcom was a jamming attack. It resulted in no data being received by the receiver, which displayed a loss of transmission error warning to the user. (Reverberi & Oswald, 2016) This is a subtle error notification that the user may be unlikely to notice during day tot day activities. It is the same error that notifies the user when they step out of range of the receiver. Some diabetics refer to it as a "compression error," because it can occur when a user is sleeping on the transmitter or when it is unable to transmit through heavy blankets.

Next, they launched a replay attack and attempted to send invalid readings to the receiver. They note that, "Replaying old packets (and thus old sensor readings) to the receiver is not trivial because transmitter and receiver work in a synchronized time window with aligned transmission counter txID. Therefore, it is essential to send a correctly formed packet exactly before the original transmitter." (Reverberi & Oswald, 2016, p6) They write that,

> **"It is sufficient to transmit only on the first frequency because, if the packet is well constructed, the receiver accepts it and just discards the other frequencies. In a similar way, we could also generate completely new packets accepted by the receiver: after creating the desired, freely chosen contents, the Dexcom CRC is computed, and the frame is wrapped in a SimpliciTI packet with correct source address and txID. This packet is then transmitted aligned with the receiving window as described for the replay attack."** (Reverberi & Oswald, 2016, p6)

A view of the receiver is shown in figure 25 as the outlined attacks are underway.

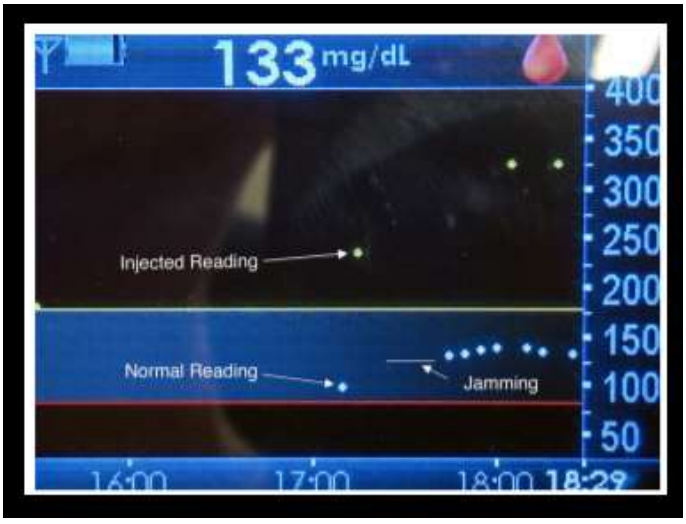


Figure 25. Jamming and replay attack against Dexcom G4 (Reverberi & Oswald, 2016, p6)

The authors continue with an attack that forces the user to replace the sensor and transmitter hardware. They misquote the price of a new transmitter as $200. Transmitter market value is $1,000 out of pocket. Sensor price is correctly estimated at $90, or roughly $30 per day, when the hardware is not used beyond the lifetime recommended by the FDA. Of these, an unexpected transmitter failure is the more catastrophic for a user because the high cost and time to ship replacement hardware makes it inevitable that a user will not be able to use the device for some days.

They also launch a denial of service attack that is based on, "(slowly) misaligning the receiving time window. Since the receiver has to account for clock drift and inaccuracies in the transmitter's and its own oscillators, it has to continuously calibrate the receive window based on the prior transmissions." The injection of many invalid packets leads the clock to become unsynchronized and forces the user to reboot the receiver. This guarantees a loss of data for at least 2 hours while the device is re-calibrated. This attack is illustrated in figure 26. Lastly, the researchers developed a mobile application to allow for "wardriving" that enabled them to search for diabetics within their geographic location and launch the previously outlined attacks. They included functionality to locate a given Dexcom within 36 square meters.

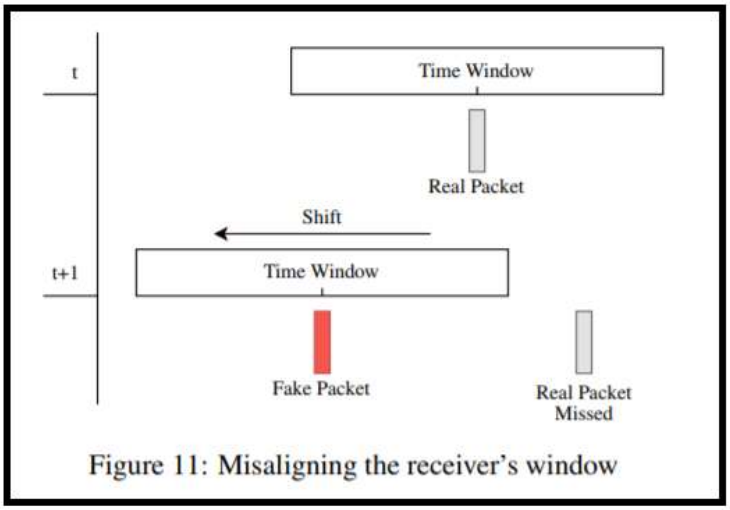


Figure 26. Timing attack against Dexcom G4 (Reverberi & Oswald, 2016, p7)

In order to overcome these vulnerabilities, the recommendation is to improve the communication protocol. "To this end, we make use of the hardware AES engine provided by the CC2510/11 SoC (used in the transmitter), which executes a single-block AES-128 in 40 clock cycles (780 µs at the default clock of 26 MHz)." (Reverberi & Oswald, 2016, p8) "Our protocol furthermore provides: 1. encryption of privacy-relevant data (src addr, glucose values) using AES-128 in Counter (CTR) mode, 2. authentication of the full frame with AES-128 as CMAC, 3. unique, random keys per transmitter, and 4. a random choice of frequencies to mitigate narrowband jamming (to an extent)." (Reverberi & Oswald, 2016, p8) This paper was provided to Dexcom in 2016, order to adhere to the principal of responsible disclosure and Dexcom responded that, "…in 2015, the next generation Dexcom G5 Mobile CGM (G5) system was introduced to the market with further cybersecurity enhancements." (Reverberi & Oswald, 2016, p8)

White Hat communities organized by patients

Many devices from a variety of industries use unencrypted RF communication. (3G and 4G smartphone security; 2017) Dexcom CGM and Medtronic Minimed insulin pumps are included in this group. NIST assigns a medium (5.3) vulnerability score in its National Vulnerabilities Database (CVE-2018-10634) to the following Medtronic devices, "Medtronic MMT 508 MiniMed insulin pump, 522 / MMT - 722 Paradigm REAL-TIME, 523 / MMT - 723 Paradigm Revel, 523K / MMT - 723K Paradigm Revel, and 551 / MMT - 751 MiniMed 530G," associated with RF, "…communications between the pump and wireless accessories…transmitted in cleartext." (CVE-2018-10634 detail.2018) The Omnipod insulin pump that I currently use, however, is not known to have been exploited.

While vulnerabilities in diabetes devices have been realized since at least 2011 (discussed by Jay Radcliffe at the 2011 BlackHat Conference proceeding, "Hacking Medical Devices for Fun and Insulin: Breaking the Human SCADA System"), reassessment of risk seems to be increasing device vulnerability scores over time. According to Newman, the identical researchers who filed vulnerably disclosures against Medtronic Pacemakers during summer of 2018 (Billy Rios, Jesse Young, and Jonathan Butts of Whitescope LLC) also amplified warnings against Medtronic insulin pumps,

> "**Multiple Medtronic I(n)sulin Pumps are prone to an authentication-bypass vulnerability and an information-disclosure vulnerability. Attackers may exploit these issues to gain unauthorized access to the affected device or to obtain sensitive information that may aid in launching further attacks."** (Multiple Medtronic insulin pumps authentication bypass and information disclosure vulnerabilities.2018)

Research by Rios and Butts seem to have increased the vulnerability assessment score from 2.9 to 5.3 on 08/13/2018. (CVE-2018-10634 detail.2018) As device manufacturers move towards integrating systems and extending functionality into IoT and the cloud, we might expect vulnerability research to further increase risk assessment scores.

While security researchers expose device vulnerabilities, patient communities are busy exploiting identified vulnerabilities in order to integrate multiple devices and extend functionality at a rate not possible for medical device manufacturers. Opensource communities have far less legal liability and do not require FDA oversight. Thus, they have become drivers for development of commercial systems, who incorporate work from the freeware community into commercial systems.

Dexcom & Corporate Patient Advocacy

Dexcom shifted the paradigm of diabetes by being the first reliable mainstream continuous blood glucose monitor (CGM). It is arguably still the *only* CGM because of hardware superiority. But, this is also a result of corporate patient advocacy, which has undoubtably translated into a vastly larger market share. Diabetics have collected their own blood glucose data for 60 years. It has always required blood, metaphorical sweat, and the metaphorical tears of great financial expenditure. It is possible that efforts, by Dexcom, to lock up painfully gathered personal medical data into a proprietary software, was never going to be an easy sell to consumers. Software may have placed unmanageable burden on the company, who's primary product success was one of hardware. The incorporation of a software system and whatever analytic tools a doctor or patient may demand represents a can of worms because there is so much variability between patients. It is inconceivable that Dexcom could have, as a single company,

developed adequate analysis tools for the dataset because there was not yet any established medical paradigm for working with continuous blood glucose data. Dexcom granted open access to the data collected by their device and gave a nod to the software community to move forward in whatever direction development demanded. This catapulted use of their system into almost every possible hardware or software device that has existed since Dexcom's inception. Dexcom data can be exported as .xml or .csv. It can also be accessed through the Dexcom API. (Welcome to developer.dexcom.2018) It can be captured through RF and Bluetooth. The convention of Dexcom's open data standard stands in stark contrast to all other diabetes device on the market which use proprietary software systems to isolate device collected data.

Proprietary software does not lend itself to integration with other devices and software systems. Patient frustration, as a result, has led to many devices as being labeled 'insecure,' as opposed to 'open.' Insecurity, in turn, fueled white hat hacking by the diabetes community. I don't believe that any member of our community has ever been charged, nor does anyone seem to fear charges, from exploiting vulnerabilities in proprietary personal medical devices. I believe that ship has sailed. This is where we see some influence of the freeware paradigm, native to software, extending into hardware. The diabetic patient community has been granted open access to the blood glucose data system and, unofficially, granted access to whatever proprietary medical device data that we are able to obtain for ourselves. This seems reasonable, in that, *I* as a patient, should not be excluded from accessing *my own* medical data.

Nightscout & Mobile Development

Early development in the Nightscout project began when Dana Lewis, a type 1, was having severe low blood sugars overnight while she was sleeping. Her boyfriend, an engineer, modified an early model of Dexcom CGM to transmit data to his cell phone. This required modification in the $6,000 device's external casing, which invalidated the warrantee, and involved some financial risk. Despite the risk, type III diabetics (the nickname given to diabetic friends and family members who play supportive roles in managing the illness), particularly the parents of young type 1 diabetic children, wanted long range data access. Parents desperately wanted data access while their children were at school and adopted the device modifications. This was particularly important for the parents of young children who may lack the ability to recognize or communicate adequately about their condition.

As popularity of the project grew, they adopted the slogan #weAreNotWaiting (a reference to not waiting for FDA approval or development in commercial systems) and opensource functionality was eventually adopted by commercial systems like the Apple watch and a wider variety of cell phones. (clarky07, 2015; Farnsworth, 2014) Dexcom data was incorporated into many commercial software applications like Glooko for remote monitoring. (Wells, 2018) Glooko, and many other software applications, are now reaching into the somewhat precarious arena of determining medication dosages for type IIs, in partnership with the FDA. (Dehaaff, 2018) While commercial software systems abound, it is worthwhile to remember that the first remote monitoring application for diabetic data was enabled because Dexcom exercised enough patient advocacy to allow it to happen and that it was built by an opensource community of diabetics, and caregivers, who recognized a need. The Nightscout project is an organization mobilized from within the patient community of type 1's, 2's, and type 3's. Major

negotiation and advocacy work was required by diabetics in order to FDA approval to occur. (Nightscout contributors, 2016; Comstock, 2016) Nightscout represents a major confrontation between a patient community and the FDA in order to force policy changes. Debate continues in discussions on closed loop systems. (Aihie, 2016) These debates demonstrate a fascinating intersection between the desire for physical security, by patients, and wariness regarding device security by the FDA.

OpenOmni & Proprietary System Hacking

While Dexcom's open data policies have spurred development and forced the FDA to accommodate patient population requests for higher device risk to accommodate higher physical safety that results from better management of a chronic health condition, extending that development further has faced significant setback. Some diabetics hope to construct highly individualized artificial pancreas algorithms, given a wealth of data that might make that possible today. Everyone has access to the best hardware for data collection. However, diabetic patients lack access to the best proprietary hardware for delivering medication. Some believe this is necessary in order to successfully develop closed loop systems, although I believe there are simple ways around current restrictions. Many other diabetics are not interested in closed loop systems, but merely want to integrate their Dexcom data and insulin pump data into one data set. This is possible for some systems, but not for the Omnipod insulin pump because Omnipod builds a very secure proprietary system that diabetics have been unable to infiltrate.

The OpenOmni project attempts to take advantage of security flaws in order to gain control over one of the best insulin pumps on the market. OpenOmni is currently offering a $45,000 bounty of the Omnipod pump. This is shown in figure 27. (Bounty sourcing the creation of an omnipod plugin for the openaps project.)

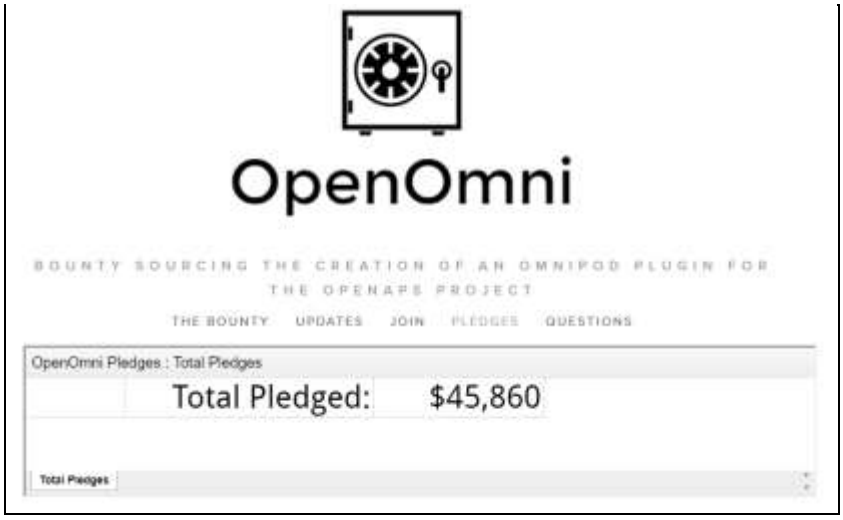


Figure 27. Bounty offered for the hacking of an Omnipod insulin pump (Bounty sourcing the creation of an omnipod plugin for the openaps project.)

OpenOmni has completed most of the work necessary to analyze communication protocols which were described earlier in the paper. Their work was extended by Dr Sergei Skorobogatov. He reiterates the point at which development has gotten stuck. It was originally posted on the OpenOmni slack page (accessible by invitation here: https://omniapsslack.azurewebsites.net/ or by members of the slack page here: https://omniaps.slack.com/threads/). Skorobogatov writes,

> **"The OpenAPS community performed a great job of analyzing the wireless communication between the PDM and Pod [12]. For the purpose of this teardown there is no need in expanding this area. The communication is based on FSK-2 modulation with 433.9MHz carrier frequency and Manchester encoding. Data bit rate is 40625 baud. Each packet has CRC checksum which was successfully reproduced by OpenAPS team. However, part of the communication protocol involves some random bytes which they were unable to emulate. Hence, there was the need for more detailed analysis of the Pod's SoC."** (Skorobogatov, 2017, page 6)

OpenOmni began investigating communications between the "pod" (aka pump actuator) and Omnipod receiver using RFCat. Information regarding this can be found on their GitHub Wiki at this link: https://github.com/openaps/openomni/wiki. The python toolkit for communicating with the Omnipod through RFCat is found here: https://github.com/winemug/omnipy. They report the following,

> **"We have figured out the RF modulation and packet/message encoding. We are now working on decoding the meaning of the bytes in the body for each of the Message Types.**
>
> **Device drivers for Rileylink are currently being developed with use of this documentation:**
>
> 1. **Rileylink branch Omnikit for using the pump with Loop**
> 2. **RileylinkAAPS branch dev_omnikit for using the pump with AndoidAPS"**

Packet captures are also listed on their wiki page (https://github.com/openaps/openomni/wiki/Packet-Captures). They report concern about the communication changes predicted for newer models,

> **"bk1113 [6:15 AM]**
>
> **Blake, I think that we will know more once the new pods available. The communication is switching from radio to Bluetooth but hopefully the protocol is the same. That would mean phone+pump+cgm only."**

OpenAPS & Closed Loop Development

Patients play a significant role, today, in the determination of what treatment tools are available to us. This stands in significant contrast to the 1960s when blood glucose machines were available only to doctors. Doctor Richard Bernstien describes difficulty, as an engineer with diabetes who was interested in the data system but prohibited from purchasing a manual blood glucose machine in the 1960s. He writes in his book that in order to have access to tools that everyone can buy over the counter today, his wife had to buy one on his behalf because she was, conveniently, a doctor. He felt that, as merely an engineer, no one in the medical community would take him seriously, so he was forced to become a doctor. I am pretty certain that Bernstein is a pretty regular contributor to one of our diabetic online communities, but he goes by "Richard," and won't tell me the title of his book. That's why I think its him. Today, there is a much larger place for engineering in the development of technical solutions for medical

treatment. Although I don't have room in this paper to describe the engineering that surrounds closed loop solutions, they are being dealt with, primarily, by open source communities of software engineers through OpenAPS and other similar organizations. Software engineers have really built a place for themselves within the medical field over the past decade. They continue to change the paradigm of the medical establishment and this can be seen to be spreading into paradigm change for other traditional limbs of the medical establishment, like drug manufacturing. An example of this is the Open Insulin Project (http://openinsulin.org/).

Hardware Investigation by Skorobatov

Sergei Skorobogatov summarizes his recent work in a short video interview regarding his presentation at the Hardware.io Conference in 2017. (*Hardwear.io | Dr. Sergei Skorobogatov | pentester academy.* 2017) Skorobogatov is a hardware security researcher at the University of Cambridge. He presented exploitations of: 1.) The iPhone 5C using NAND mirroring; 2.) Car keyfobs and, 3.) Omnipod insulin pumps. (Skorobogatov, Oct 26, 2017) An excellent paper by Skorobogatov is titled, "Deep dip teardown of tubeless insulin pump," and describes the challenges confronted. A 15 minute summary of this paper is presented at minute 34 of his 2017 lecture at The Hardware.io Conference in The Hauge (www.youtube.com/watch?v=KhGI8h4ODxI&t=971s).

Figure 28 shows Skorobogatov's trace of the integrated circuit connections to the mechanical hardware. (Skorobogatov, 2017) I mimicked his work, here, by disassembling the plastic pod using a table sander and inspecting the processor type. The processor identified by Skorobgatov was a System on Chip, SC9S08ER48CHP, and, "It was found that the chip has 48kB of Flash and was fabricated with 0.25µm process." (Skorobogatov, 2017, p6)

I identified the same processor type during SEIS 740 (Real Time Systems & Applications) last semester. (figure 29) Skorobgatov was unable to identify any documentation for the integrated circuit. Intuition suggested to him, that it was an unmarked, 8-bit Freescale microcontroller. Freescale was purchased by NXP in December of 2015. (Shah, 2015) NXP is where Professor Kruse, from SEIS 740, currently works. It is a chip manufacturer with a local office in Bloomington, Minnesota. (Worldwide locations.) This, it occurs to me now, may have been why he wasn't a giant fan of this paper idea last semester. He is also an ex-employee of Medtronic and so this may not have been a super comfortable paper topic for him and led to his judgment that such a paper was, in fact, illegal. Professor Cheung also tended towards declarations that study of this type is "illegal," but added a disclaimer that he simply could not imagine any value in investigating the innerworkings of medical devices, unless it was for nefarious purposes. I was unable to adequately explain. My endocrinologist, Dr Tressler, also placed a vote on the side of "illegal," although he certainly understands the patient motivation that leads to this type of study. OpenOmni addresses the issue first on their Questions page with the following,

"Is this legal? I think so. Maybe. But I'm not a lawyer. I'm hoping that the combination of a patient/owner derived solution in combination with the October issued Exception to DCMA for Medical Devices will offer some protection. It's entirely possible it doesn't, but I'm perfectly willing to take that risk. It's got to be less risky than a continued pattern of me dosing my child with insulin at 1:00 AM. And 3:45 AM..." (Some questions you might have.)

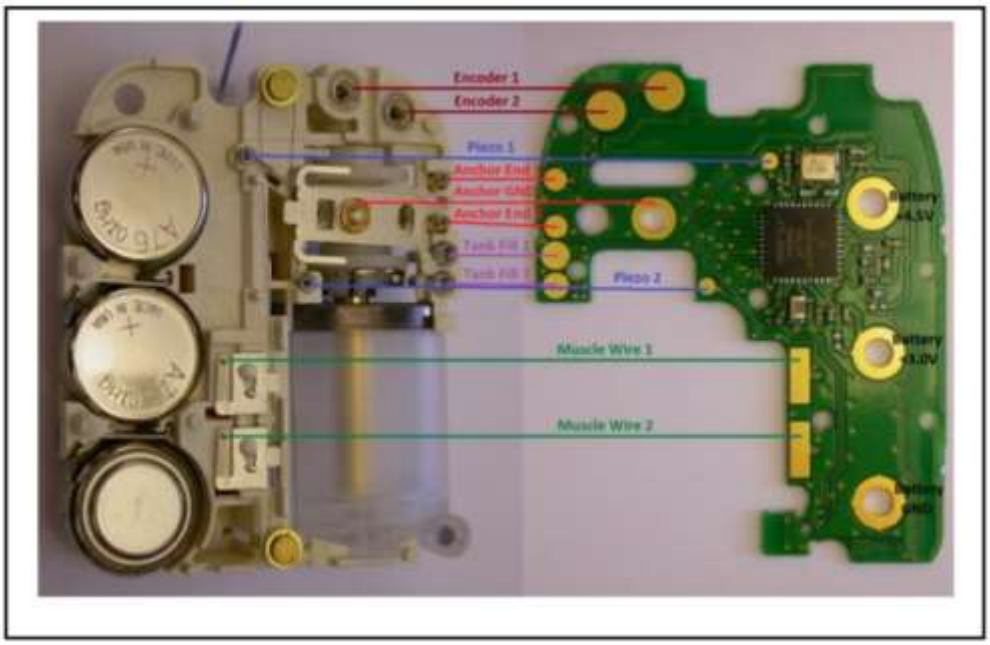


Figure 28. Trace of integrated circuit connections to the mechanical hardware (Skorobogatov, 2017, p5)

Figure 29. Identification of the same processor type

I can only say, in regard to the legality of freeware development and hardware exploitation in this context, that where the law becomes unclear, things are best decided by a jury. In that way, nothing is illegal until a jury says it is. I find it very difficult to believe that a jury would convict a parent for trying to do their best to care for their child. Since that has been a wellspring for development in diabetes (which

emerges as a pediatric condition in 10% of cases), we may have more legal slack than other chronic illness groups and we can assist others by setting some precedence in this context. If I ever had to make a case in court, I would, personally, make use of fundamental and long-standing conventions of medical ethics. Also, there is almost a decade of Medtronic insulin pump hacking. Pacemaker hacking has an even longer history. During that time, to my knowledge, there have never been any related convictions. Although it's true that there are more formalized procedures today about how device exploitations are disclosed. Skorobgatov is researching US medical device companies from a location outside the US, providing him with some level of protection. However, he concludes his paper by writing,

> **"Future work will involve further analysis of the disassembled code to understand the communication protocol. This work will be carried out by several communities (OpenAPS, OpenOmni, Nightscout Foundation and Loop project [28]) to improve the life of Type 1 diabetes patients. Any further findings, achievements and solutions will be presented by them."** (Skorobogatov, 2017, p10)

This may be an indication that he feels that he has reached the practical conclusion of what he can do, without getting into legal hot water with Omnipod. Hardware disassembly might be tolerated, but software disassembly seems to be the straw that breaks the coding camel's back. That said, I continue with my summary of Skorobogatov's recent hardware hack.

To construct a circuit diagram, he traced connections between components by following the wires on the Printed Circuit Board (PCB) in two different ways. The second being preferable to the first, due to its simplicity and low cost. First, he X-rayed the board (figure 30). Second, he de-soldered it and cleaned it with a solvent. He was then able to begin drawing a circuit diagram (Figure 31) after visual inspection. He measured resistors, capacitors, and inductors using an LCR. (How to use an LCR meter.2010)

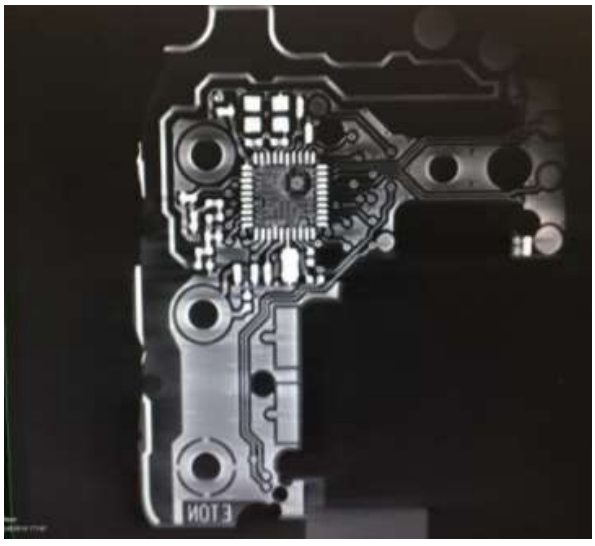


Figure 30. X-ray image of the board

Device communications were inspected using spectrum analysis. (figure 32) He referenced work on device communications by the OpenOmni community that were previously summarized in this paper. Skorobogatov notes that additional work must be done in order to further understand device communications, "… part of the communication protocol involves some random bytes which they

(meaning OpenOmni) were unable to emulate. Hence, there was the need for more detailed analysis of the Pod's SoC." (Skorobogatov, 2017, p6)

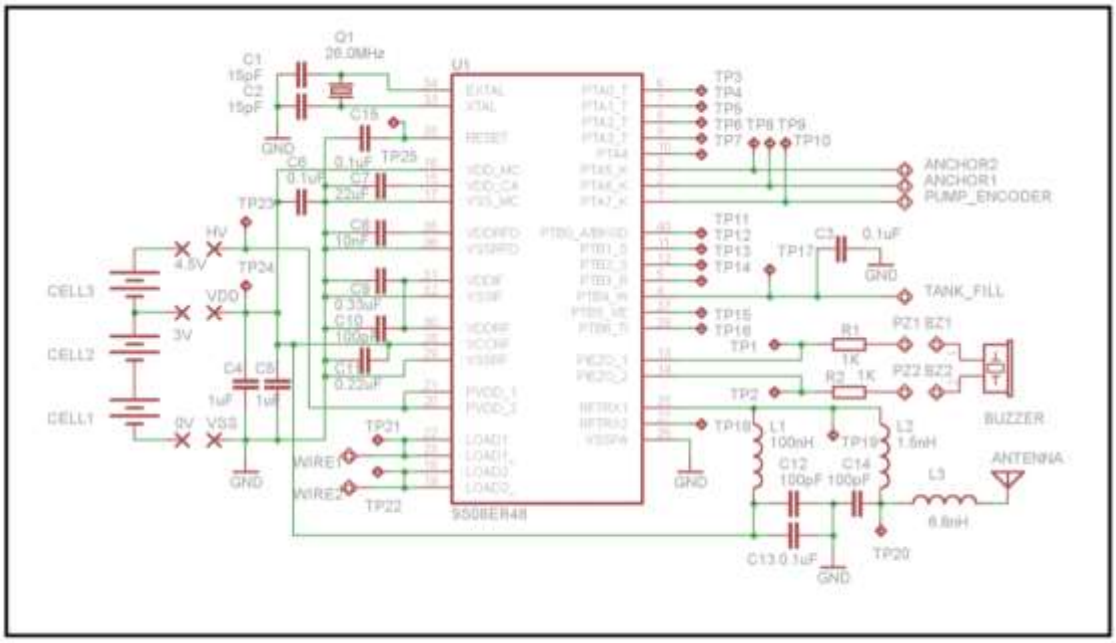


Figure 31. Circuit diagram of a pod (Skorobogatov, 2017, p8)

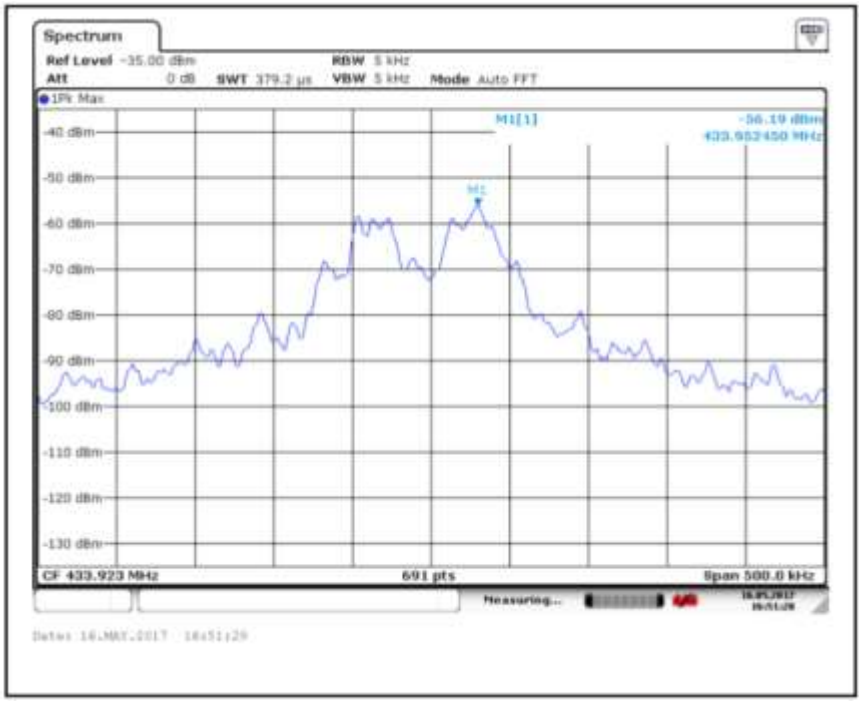


Figure 32. Spectrum Analysis of Device Communications in the 433.8MHz to 434.0MHz frequency range (Skorobogatov, 2017, p6)

Skorobogatov extended OpenOmni's work by contributing analysis of the hardware by 'tearing down,' the Integrated Circuit (IC) and identifying pin connections. "The voltages on all SoC pins during the device operation and static parameters of the I/O pins were measured on the core IC of the device." (Skorobogatov, 2017, p6) Results were cumulated in Table 2 of his paper. He states that, "This analysis allowed narrowing down the number of pins suitable for Debug interface connection." (Skorobogatov, 2017, p6) Pins 38 and 40 were identified as Reset and Debug pins. To accomplish this, Skorobogatov identified universal programmers that could be used with this particular SOC and used them with as oscilloscope. (p7) They were the Elnec BeeProg2 has MC9S08ER48 and SC9S08ER48. He was required to make payment to the manufacturer in order to do this, "If anyone wants to use ElnecBeeProg2 programmer to Read and Write this chip he has to pay Elnec a few hundred Euros." (p7)

He next investigated the internal of the chip by de-processing it. He states that the, "Backside image gives more information about the fabrication process and memory sizes which were confirmed as being 0.25µm, five metal layers, 48kB of Flash memory and 4kB of RAM. Half of the chip surface is taken by RF communication module." He utilized free tools (CodeWarrior) provided by the chip manufacturer to gain, "…crucial information about the CPU special registers and configuration." (Skorobogatov, 2017, p8)

Next, he used information gathered about the debug pin (pin #40) to attempt to gain access to the firmware. It was his conclusion that, ". The security protection in this chip was not possible to defeat with known non-invasive or semi-invasive methods." (Skorobogatov, 2017, p8) After exploring some failed options, he used a,

> **"Recently introduced direct Flash and EEPROM extraction methods using SEM (Scanning Electron Microscopy) could be used for firmware extraction [23]. However, in order to improve the image quality some additional techniques were used with the help from an industrial collaborator [24]. The example of the resulted image is presented in Figure 20 with brighter areas corresponding to the charged cells."** (Skorobogatov, 2017, p9)

His results from microscopy are shown here in figure 33. He performed some image processing and then used Matlab to produce a HEX file of the firmware. A 10 minute video of his firmware extraction and verification is posted at: https://www.youtube.com/watch?v=YK6aa4ojl7M&feature=youtu.be. Skorobogatov acknowledges that, "This paper does not aim at the code or firmware analysis." (p9) But, does recommend a number of tools for continuing his work. In particular, he recommends IDA Pro from HexRays as a tool for code disassembly. He recommends running the code on a CPU emulator, as opposed to a debugger. And writes that, "Although HCS08 core does not have JTAG interface, it does have a Background Debug mode. It can be used not only to look at the current state of the internal memory and CPU registers but to also enforce hardware breakpoints either on specific value of the program counter or on a certain memory or register condition." (p10) Skorobogatov concludes his paper with reflections on embedded device security,

> **"One of the possible countermeasures against firmware extraction could be in use of the memory encryption. However, embedded memory is not particularly suitable for strong encryption. This is because unlike external memory it is fetched at random addresses. That poses a big challenge for 8-bit and 16-bit CPU cores [27]. The solution could be in implementing at least 128-bit virtual memory array from which individual bytes are fetched. However, any additional buffers between the memory array and the CPU will increase latency."** (p10)

He outlines the needs for future development in figure 34, taken from his slide presentation to Hardwear.io.

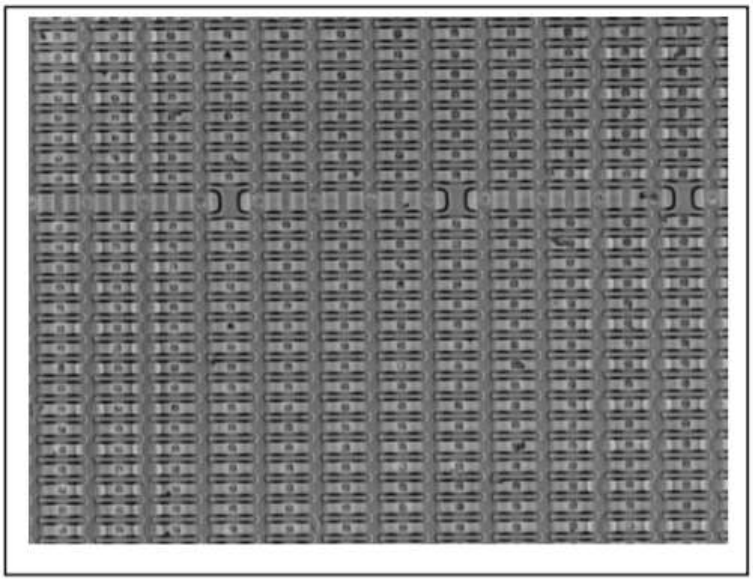


Figure 33. SEM image of the flash cells in the SoC (Skorobogatov, 2017, p9)

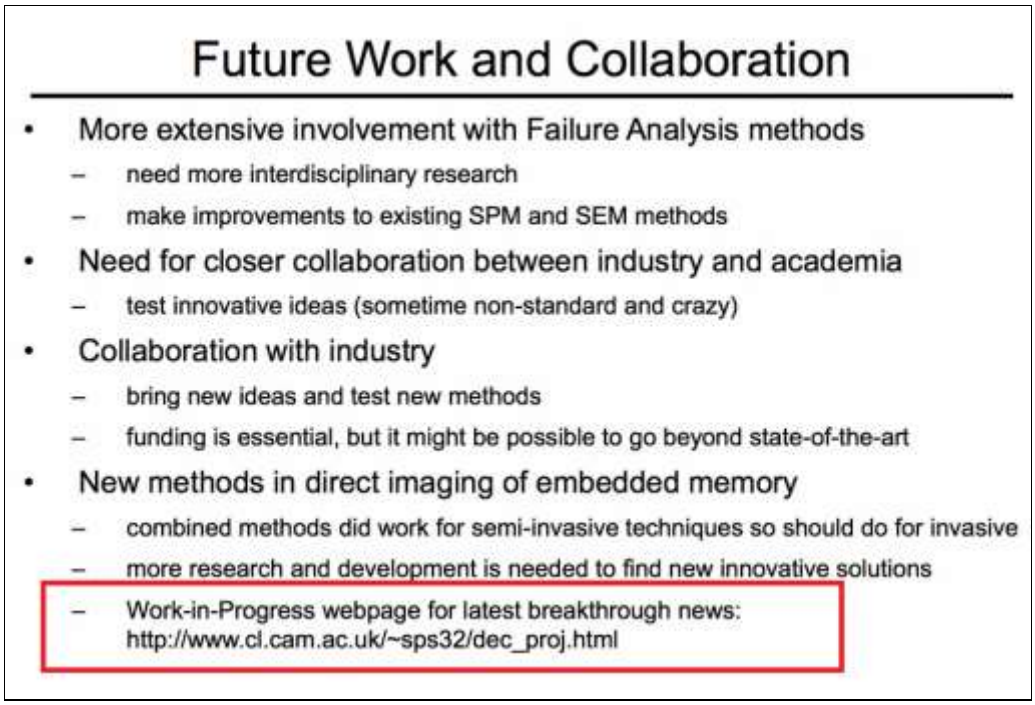


Figure 34. Needs for future development (Skorobogatov, Oct 26, 2017)

Potential for Black Hat Attacks

Individuals with pacemakers and diabetics with insulin pumps are often lumped into the same classification because we both rely on critical systems. Pacemakers reap huge benefits from being able to download software updates. It prevents a patient from enduring an additional surgery to replace the implanted device with an updated model. This year, however, researchers demonstrated how they could access the Medtronic Carelink 2090 pacemaker remotely, through hospital software, and use it to deliver electrical shocks capable of killing a patient with a pacemaker. Research was conducted by Billy Rios and Johnathan Butts, who specialize in cyber security and critical infrastructure. (Episode 33: Dr. jonathan butts; billy rios on cyber security, public safety and the layers of defense.2018)

This circumstance has significant implications for diabetic devices which, while not implanted, are moving increasingly to incorporate internet connectivity. The 508 Medtronic insulin pump has incorporated keefab-like technology to deliver insulin boluses remotely, although from a limited range. (Benedict, Keyes, & Sauls, 2004) While there is a long history of high profile pacemaker hacking – Dick Cheney had the wireless communications disabled in his pacemaker in 2007, new efforts to incorporate longer range communications over the internet may contribute risk, "…the latest variation on the terrifying theme depends not on manipulating radio commands, as many previous attacks have, but on malware installed directly on an implanted pacemaker." (Cheney's heart device altered to prevent assassination.2013; Newman,)

I found it challenging to gain technical information regarding the newest pacemaker hack. This may be due to liability concerns on the part of the researchers, who could not, ethically, disclose vulnerabilities until those vulnerabilities were resolved. Disclosure would put patients at risk. However, there were also concrete concerns regarding litigation,

> **"St. Jude Medical…opted to sue the researchers and investors who claimed the med-tech company's in-office programmers and at-home bedside monitors were surprisingly vulnerable to malicious computer hacking. Abbott Laboratories, which acquired St. Jude in 2017, quietly settled that litigation last month after issuing a series of software updates and vulnerability disclosures."** (Carlson, 2018)

Information was initially disclosed to Medtronic and several other companies, including Boston Scientific and St Jude, who responded in a variety of ways. "Since 2016, all three U.S. makers of pacemakers and defibrillators, all with major operations in Minnesota, have had cybersecurity warnings issued for the machines used by doctors to program and test implanted heart devices." (Carlson, 2018, para 3) Details of Rios and Butt's research was announced in Wired Magazine, prior to their public presentation, "Understanding and Exploiting Implanted Medical Devices," at the Black Hat Conference on August 9th, 2018. (Butts & Rios, August 9, 2018) That article summarizes their concerns,

> **"At Black Hat, Rios and Butts will demonstrate a series of vulnerabilities in how pacemaker programmers connect to Medtronic's software delivery network. The attack also capitalizes on a lack of "digital code signing"—a way of cryptographically validating the legitimacy and integrity of software—to install tainted updates that let an attacker control the programmers, and then spread to implanted pacemakers.**
>
> **"If you just code sign, all these issues go away, but for some reason they refuse to do that," Rios says. "We've proven that a competitor actually has these mitigations in place already. They make pacemakers as well, their programmer literally uses the same operating system [as Medtronic's], and they have implemented code signing. So that's what we recommend for Medtronic and we gave that**

> **data to the FDA." The programmers run the Windows XP operating system. (Yes, Windows XP.)"**
> (Newman, para 7 and 8)

Details regarding the progression of negotiations between the researchers, Medtronic, and the FDA are included in Appendix C.

Since the FDA and Medtronic have resolved the vulnerability reported by Butts and Rios, I expect more information to be released. In an effort to confirm and further asses the noted vulnerability, I inquired with Richard Thieme, a regular speaker at Black Hat, during his lecture at the University of St Thomas Minneapolis campus. He sent me the email correspondence in figure 35 regarding the pacemaker hack and provided information that it involved, "…software over HTTP (no SSL) and no signatures."

When I asked Thieme if I ought to worry about the integrity of my medical devices while attending an upcoming computer hacking conference, he replied, "I don't think anyone would be stupid enough to hack a medical device." I admit that I did not feel particularly reassured by this statement. It has been my experience that it is best *not* to put artificial limits on the depth of human stupidity or malice, particularly in matters of life and death. People with visible vulnerabilities quite often become the target of exploitation because they represent 'easy targets.' Diabetics have always enjoyed some security by obscurity in that our physiological vulnerabilities do not always obviously and visibly announce themselves. Endocrinological function and diabetes technologies are not well understood, even by family, police, or hospital staff. While this presents a multitude of frustrating circumstances, it also provides a layer of protection that people with other illnesses do not enjoy. I had nagging concerns about how a crowd of random individuals might 'poke at' an unsecured wireless signal, given that they were interested in doing so.

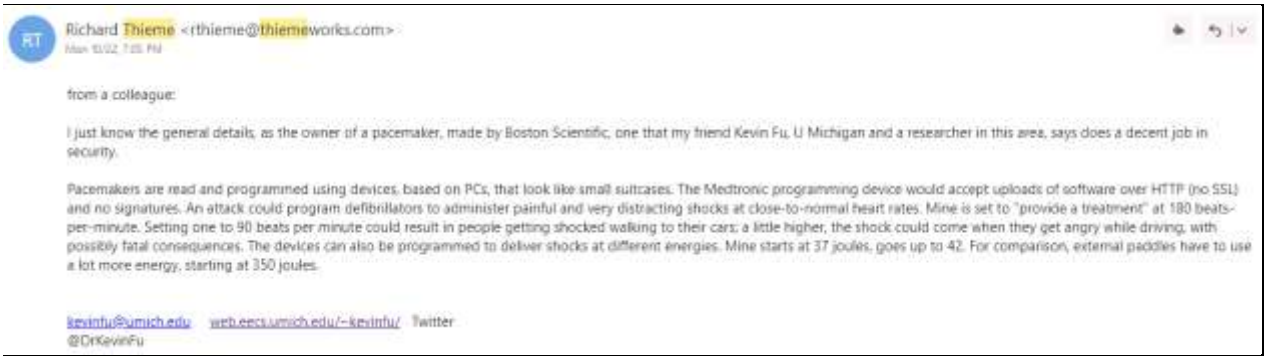


Figure 35. Communication forwarded from Richard Thieme

I inquired at the Wild West Hackin' fest since there was a presence of Black Hat Conference attendees. I asked the question, "If I went to the Black Hat conference, where there are a bunch of bored computer hackers, would I need to worry about compromise of my personal medical devices?" A friendly, conversational man was smoking cigarettes with a group from the conference and was clearly interested in my question. He had attended Black Hat with some friends from work, and one woman had asked the same question as I. They had all hoped to enter the Wireless Village, but she was apprehensive and held

back. She was an epileptic and used an implanted device to help with seizure control. It was a device used for Vagus Nerve Stimulation, similar to the one pictured in figure 36.

Figure 36. Vagus Nerve Stimulation (Vagus nerve stimulation (VNS).)

The vagus nerve is one of the more interesting nerves in the body - ripe with legend, words of warning, and the subject of exciting personal stories from Emergency Medicine, due to its “black magic-like” behavior. If a paramedic accidentally touches it while inserting an endotracheal tube to help a patient breathe, heart rate will immediately slow. The smoking man could see I was interested and continued his story with animated enthusiasm. Surrounding conversations trailed off as the smoking man now clearly controlled the most interesting conversation and a resulting smoking stage from which he performed, “She asked the men at the entrance to The Village if it was safe for her to go inside. They weren’t entirely sure, so they said that they would find out.” Several smokers extinguished and lit up a second cigarette, intent on listening. “They returned with some other people who looked at her device and asked her about it. Then, they all sat down together and sure enough, after some time, they were able to make the device deliver electrical impulses from her mobile phone, completely bypassing the controller. It was *really* neat!”

Conversations then moved to Black Hat and the Wireless Village. I was unable to gain any additional information about the pacemaker hack, but felt reassured that valuable information had been relayed to me. I learned that all the electronic slot machines in a Deadwood casino had been attacked several weeks prior and that the casino had been forced to close its doors. The amount of revenue lost was described as “incalcuable.” Then the smoking stage was overtaken by an amicable ‘Denial of Service Dog,’ dressed in a camofluge vest, pockets packed with hardware. He looked a little apprehensive as he

was identified as the culprit that had led to so much frusteration from hotel guests. But, he relaxed as people petted him and reasured him that he was, in fact, a *very good* Denial of Service dog.

Device Manufactuers

I next attended an ISACA meeting at the Plymouth, MN headquarters for Abbott, the recent aquirers of the St Jude pacemaker hacked by Rios and Butts. A speaker on behalf of Abbot discussed some of the difficulties in medical device development, in particular how they impact security. They passed around several models of a pacemaker, including those shown in figure 37, 38, and 39.

The ATLAS 2VR Model V-193 was recalled in 2008 for software defects. (Class 2 device recall ATLAS VR model V193.2008) Newest models of the ASSURITY are called the ASSURITY MRI and are safe to use in an MRI machine, which uses a "…superconducting magnet to align hydrogen atoms in the body; then excites the atoms with radio frequency (RF) energy from the transmitting RF coil. As the atoms return to equilibrium, energy is released in the form of radio waves which are recorded by the receiving RF coil." (*Introduction to magnetic resonance imaging (MRI).*2017) It also allows for remote communicates via the Merlin.net software platform. Merlin.net allows, "Clinicians (to) upload and manage patient-device data by accessing a web interface using the Internet and a compatible browser." (Featured remote care products.2016) Another device passed around for inspection was the St Jude Confirm Rx DM 3500 Implantable Cardiac Monitor (ICM). (Confirm rx™ insertable cardiac monitor.2018) It began communicating with smart phones this year after the FCC ceded to Nightscout, Dexcom, and diabetic lobbying to allow cell phone communication with medical devices.

Simple things can quickly become complicated within the context of medical devices. Take as an example, authentication. We are very familiar with authentication mechanism for traditional devices. However, if a patient is brought into the Emergency Room unconscious and is unable to recite their password so that doctors can access their device, we have a terrible problem. Simple logging of events is often prohibitively difficult because continuous logging requires so much battery power that it will drain away the life expectancy of a pacemaker. (Perspectives on medical devices;)

Abbott listed the regulatory entities that oversee medical devices as: Homeland Security, FDA, Industrial Control and Cyber Emergency Response Team, National Health-ISAC, NIST, Medical Device Innovation Safety & Security Consortium, Center for Internet Security, Department of Commerce, and the Department of Health and Human Services. In order to request that the FDA set formal premarket standards, Abbott and The Chertoff Group published a whitepaper, titled "Why Medical Device Manufacturers Must Lead on Cyber Security in an Increasingly Connected Healthcare System." This led the FDA to publish, this year, its revised Pre-Market Guidelines and Bill of Materials, in which medical manufacturers are asked to list software products, like operating systems, that are embedded inside their devices. This "soft report" requires them to list services and protocols. This is intended to help hospitals identify vulnerabilities in the innerworkings of their devices and apply patches before safety is compromised. (Kolbasuk McGee, 2018) Without these changes it was very dangerous for a hospital to use third party products because they were on the hook for any data loss or security breaches but hadn't the information necessary to independently verify device and software security on their own. FDAs new Guidelines, most notably, address specific concerns on internet connectivity. They classify devices as tier 1 (internet reliant) or tier 2 (having no internet connectivity) and puts a set of standards in place for

each tier. (Perspectives on medical devices;) Much effort to lead hospital infrastructure security has been started by Mayo, who Abbott is working with. Mayo is hiring full time hackers to attack their hospital network which consists of thousands of devices and countless attack vectors.

Figure 37. ATLAS VR Model V193 (Perspectives on medical devices)

Figure 38. St Jude Medical ASSURITY Pacemaker

Figure 39. St Jude Confirm Rx DM 3500 (Confirm rx™ insertable cardiac monitor.2018)

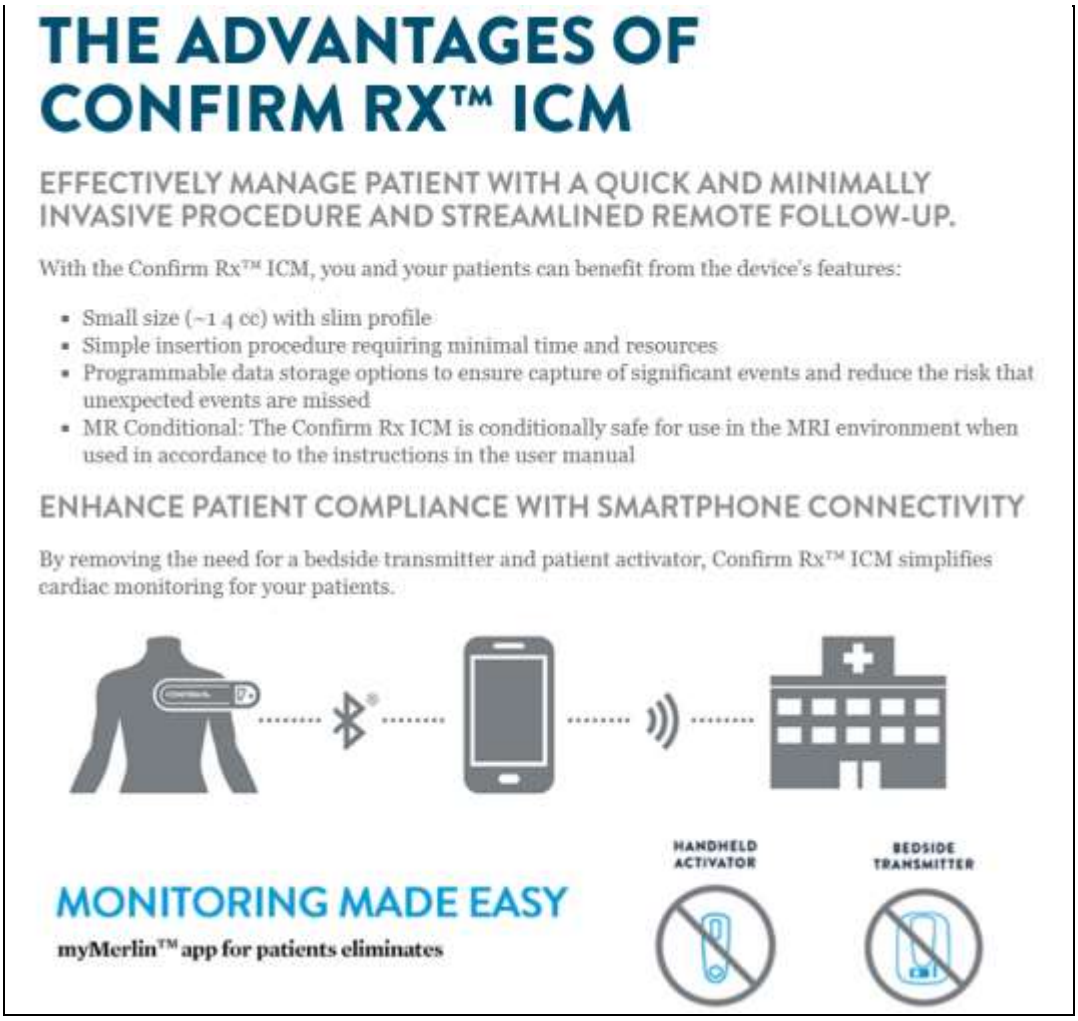


Figure 40. Info regarding the Confirm RX ICM

Conclusion

In conclusion, I was a little disappointed to see how much was entailed in writing a simple summary of significant technical publications on Diabetes medical technology. Regardless, I think that this paper represents a complete accounting of all that is publicly known about proprietary diabetes systems at this time. However, what is known reflects older systems, and there is very little published on new systems. That means that diabetics must continue to 'guinea pig' new devices.

Research publications are so slow to market that patients perform experiential testing of equipment and publish the defects. While we are fortunate to have online communities to widely disseminate and discuss our concerns (as these did not always exist), I continue to view attempts to upgrade medical equipment as a somewhat harrowing experience in the best of circumstances. Given how long it takes hardware analysis of proprietary systems to be published and given that software analysis might take just as long, or longer as the systems increase in complexity, it seems impractical for software to emerge from anywhere other than the freeware community where patients have open access to the algorithms. This is the only solution that provides patients with the bare minimum standard required to fulfill basic medical ethics – informed consent.

Basic medical ethics require that a patient be informed of risks before consenting to any treatment. They must, alternatively, be informed of all risks to which they are subject by refusing a particular treatment. As system complexity increases, risk increases for patients who do not understand their devices and information systems. I believe that there are currently far more diabetes devices being sold than there are diabetic patients that understand how to operate them. Risks of new devices are often unknown until patient communities use them and spread word.

As software systems begin to exert influence in addition to what we have now, this ought to represent concern for our community. There is urgency as more and more players enter the game and develop more and more systems that require testing and real-world feedback by our community. Patients are becoming overburdened by what they need to understand in order to survive using their technology. I think that is well illustrated here by the length of this paper, which did not even touch on newer devices. However, this hopefully provides me with a fundamental jumping off point from which I can evaluate new devices in a more technically informed manner and spread word to my community if I encounter dangerous circumstances.

All the effort it takes to understand something complex is useless unless you can explain it to someone else. I was able to explain, using metaphor (a common strategy in software), during the MedFuse Conference, one of the biggest mistakes our chronic illness community has made in regards to new technology in the past ten years. It reflects concern that I had upon seeing Missing Person posters for a St Thomas student, Dan Zamlen, who passed away in 2009. I would guess he used a Medtronic insulin pump without a Dexcom CGM. It was common for Doctors to prescribe an insulin pump without a sensor, then. The metaphor that I use to explain this mistake centers around misunderstanding what a technology can and cannot do. Flying a plane without instrumentation might be possible. But, sometimes you fly over water, or you fly over water at night, and it is very difficult to orient yourself without instrumentation to correct you before it's too late. JFK Jr. died this way in 1999. A plane is a critical system with complex controls. Diabetics aren't allowed to fly planes or drive semitrucks, in general, but we do operate critical systems. Instrumentation helps provide feedback regarding the

quality of your assumptions. Using an insulin pump without a Dexcom CGM is like flying a plane without instrumentation. It can be done, but you might crash if you need to fly through a new, disorienting environment. For this reason, a high quality CGM ought to be regarded as a prerequisite to an insulin pump, otherwise the pump has a high likelihood of doing harm for a variety of reasons.

I am uncomfortable relying on algorithms that I don't understand, with regard to medical systems. It is akin to me not being able possess informed consent regarding the risks for a particular device. This is an issue inherent in proprietary systems because they prohibit inspection of the underlying algorithms in order to protect their product. It is questionable, to me, whether a Doctor can ethically recommend a device for which he is not aware of the innerworkings of the software. I think that this gives freeware closed loop systems an edge that is difficult to overcome for commercial enterprises. Diabetics must develop and build the solutions we need, now that it is in the hands of software, predictive analytics, and closed loop systems. There is simply no other alternative available.

Appendix A. Figures supporting the Keeloq keefab Lab

- HackRF One - A relatively inexpensive Software Defined Radio (SDR) capable of half duplex transmit and receive operation within the range of 0-6GHz.
- Rtl-sdr dongle - An inexpensive receive-only software defined radio.
- Yardstick One - A sub-1GHz digital wireless transmitter device.
- GQRX - An open source SDR receiver software.
- Rfcat - An open source SDR transmitter software capable of generating an ASK OOK PWM signal.
- Audacity - An open source audio editing program that can be used to inspect the target waveform.

Figure A1. Hardware and software required for the Keeloq Fab attack (Fletcher, Jan 10, 2018)

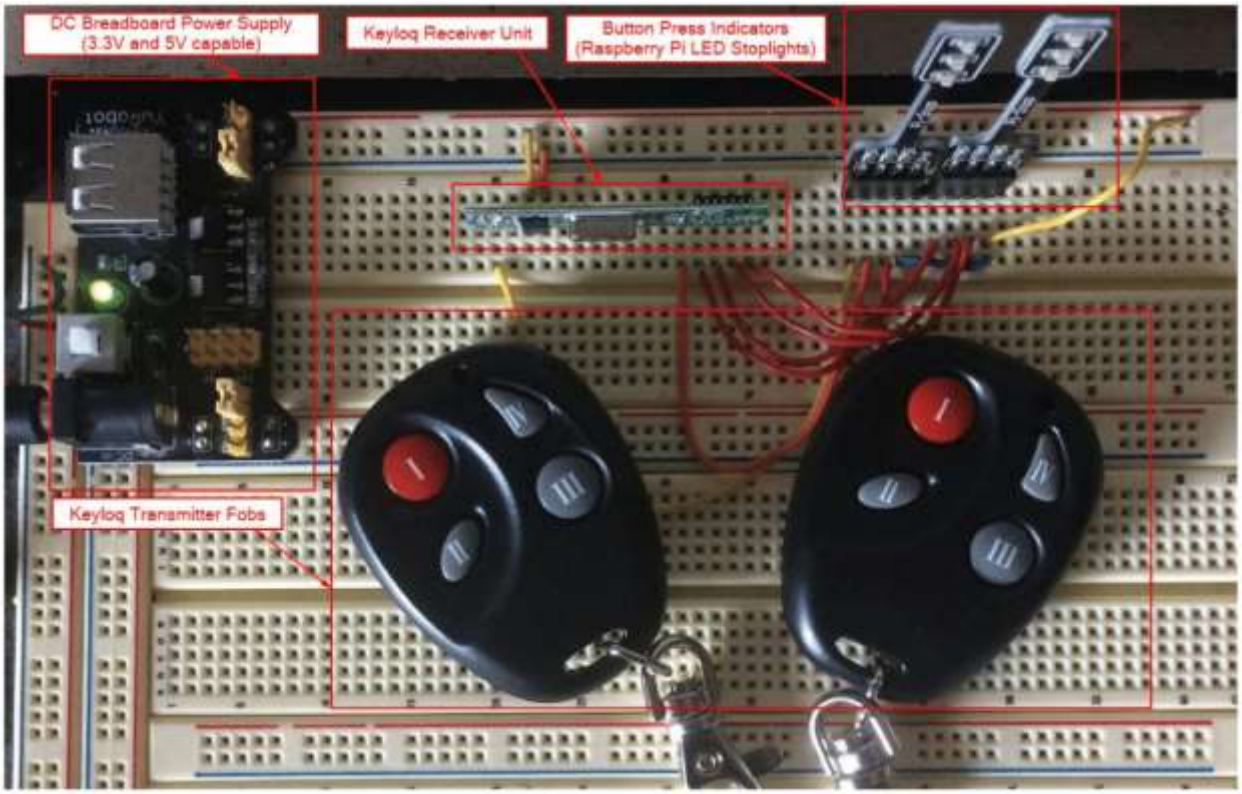


Figure A2. Hardware Setup for the Keeloq Fab attack Lab – part a (Fletcher, Jan 10, 2018)

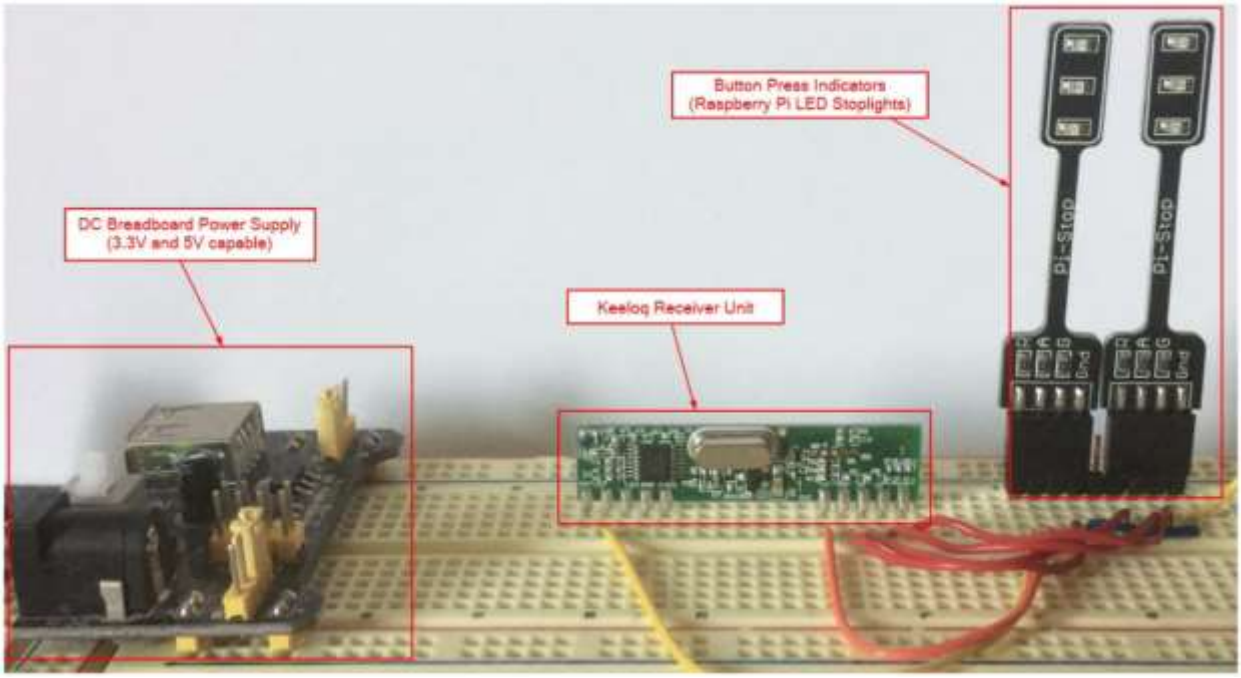


Figure A3. Hardware Setup for the Keeloq Fab attack Lab – part b (Fletcher, Jan 10, 2018)

MICROCHIP

HCS200

KEELOQ® Code Hopping Encoder

FEATURES

Security

- Programmable 28-bit serial number
- Programmable 64-bit crypt key
- Each transmission is unique
- 66-bit transmission code length
- 32-bit hopping code
- 28-bit serial number, 4-bit button status, low battery indicator transmitted
- Crypt keys are read protected

Operating

- 3.5–13.0V operation
- Three button inputs - seven functions available
- Selectable baud rate
- Automatic code word completion
- Low battery signal transmitted to receiver
- Non-volatile synchronization data

Other

- Easy to use programming interface
- On-chip EEPROM
- On-chip oscillator and timing components
- Button inputs have internal pull-down resistors
- Low external component cost

Typical Applications

The HCS200 is ideal for Remote Keyless Entry (RKE) applications. These applications include:

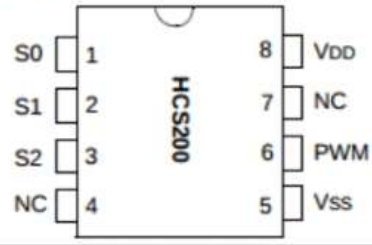
- Fixed code replacement
- Automotive RKE systems
- Automotive alarm systems
- Automotive immobilizers
- Gate and garage door openers
- Identity tokens
- Burglar alarm systems

DESCRIPTION

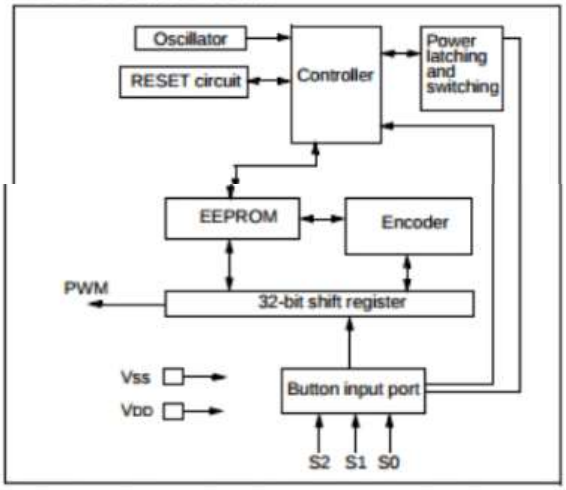
The HCS200 from Microchip Technology Inc. is a code hopping encoder designed primarily for Remote Keyless Entry (RKE) systems. The device utilizes the KEELOQ® code hopping technology, incorporating high security, a small package outline and low cost. The HCS200 is a perfect replacement of fixed code devices in unidirectional remote keyless entry systems and access control systems.

PACKAGE TYPES

PDIP, SOIC



BLOCK DIAGRAM



The HCS200 operates over a wide voltage range of 3.5 volts to 13.0 volts and has three button inputs in an 8-pin configuration. This allows the system designer the freedom to implement up to seven functions. The

Figure A4. Data sheet for keefab internals

Appendix B. Details of the Radcliffe Hack

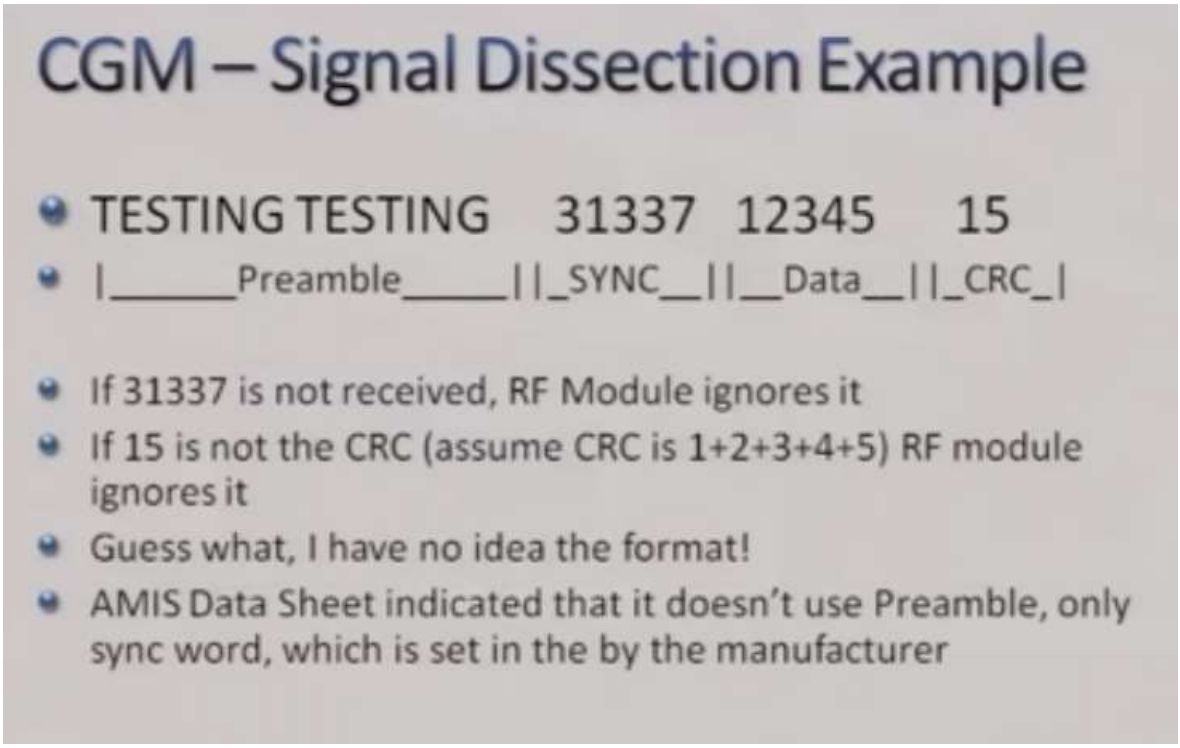


Figure B1. Signal Dissection (Radcliffe, Minute 26)

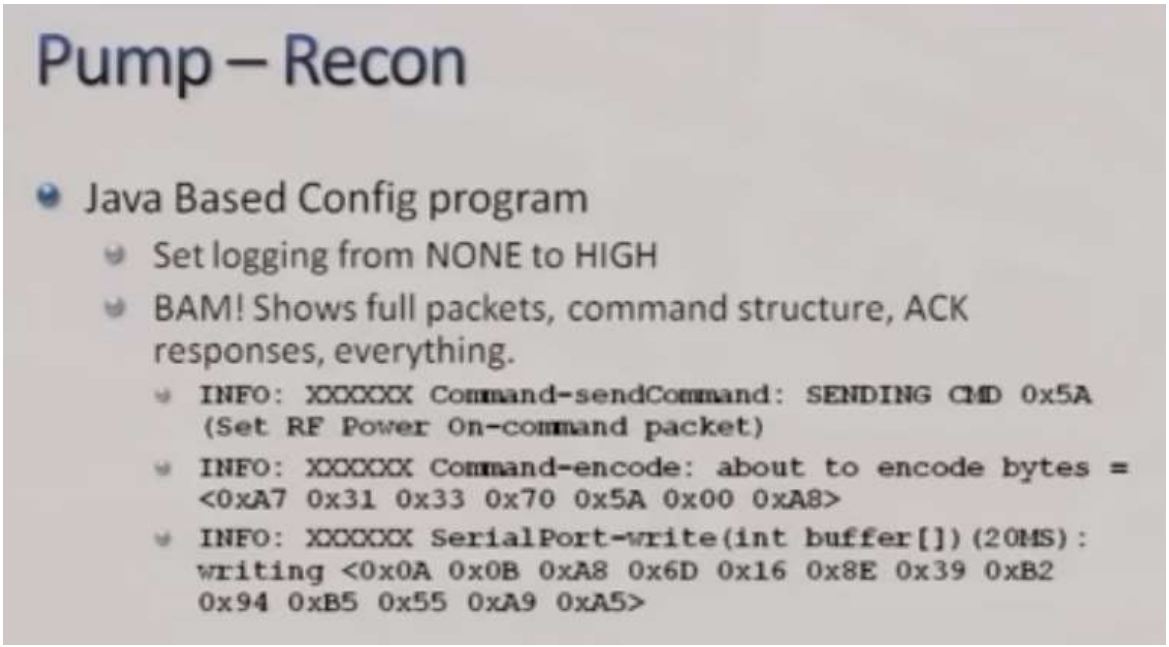


Figure B2. Packet retrieval (Radcliffe, 2011)

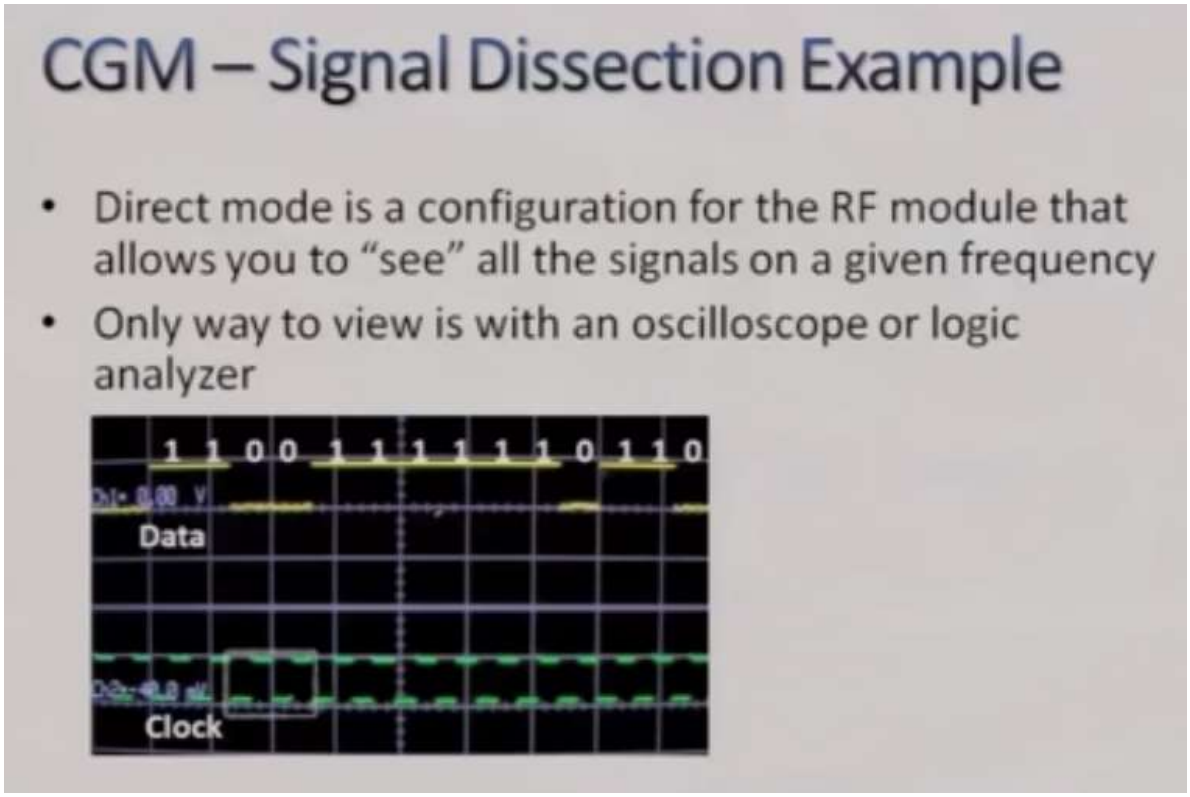


Figure B3. Signal Dissection, part ii (Radcliffe, min 27)

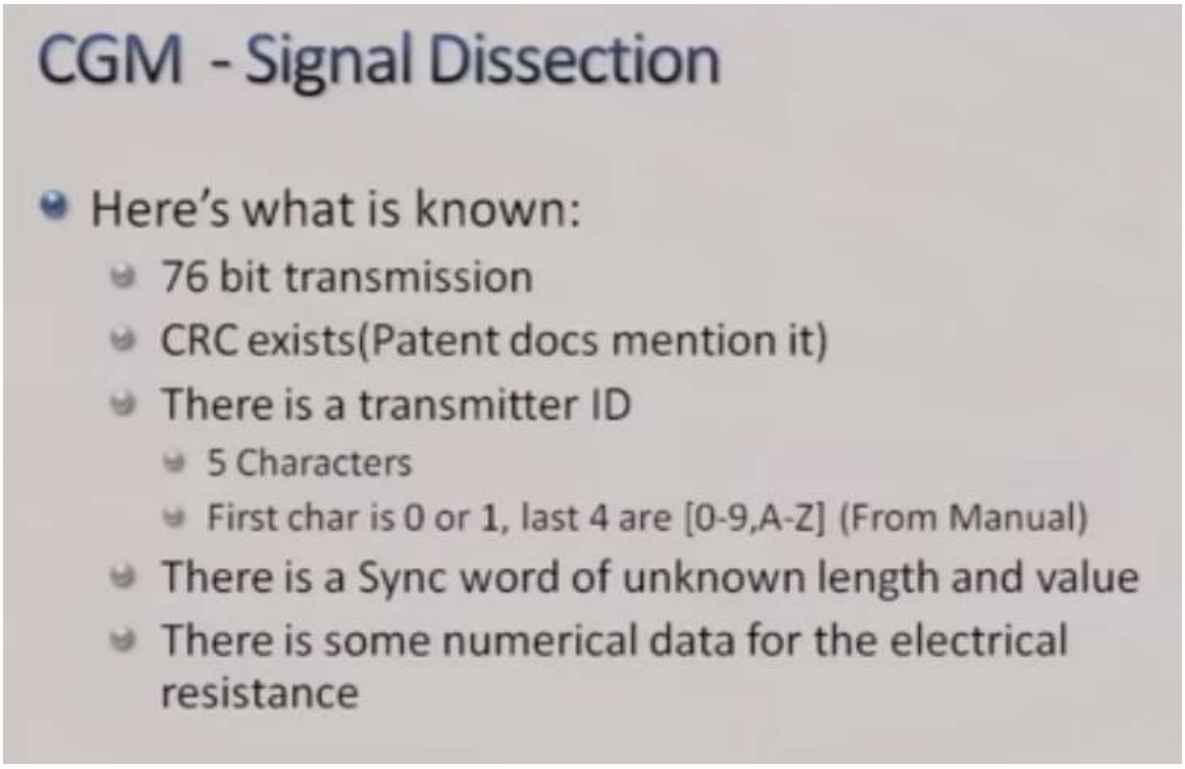


Figure B4. Signal Dissection, part iii (Radcliffe, 2011)

Appendix C. Details of the Medtronic 2090 Pacemaker Vulnerability

The Advisory (ICSMA-18-058-01) for the Medtronic 2090 Carelink Programmer Vulnerabilities (Update B) can be viewed at the ICAMA-18-058-01 Us Cert Advisory. This vulnerability raised concerns that pacemakers could be accessed and operated remotely by unauthorized actors. This was a particularly interesting advisory because I could trace the progression of the investigation through local and national sources over the past year, up until very recently when the matter was resolved by NCCIC. It was also interesting because pacemakers, like insulin pumps, are considered critical medical device systems. A Star Tribune article from August 15th reported that, "The five Medtronic security alerts published this year involve vulnerabilities in machines that are supposed to communicate with patients' implanted heart devices, neurostimulators, or body-worn insulin pumps."

The original US-CERT Advisory was filed February 27, 2018 by researchers Billy Rios and Jonathan Butts, who had submitted their concerns to Medtronic 10 months earlier. Medtronic published a responding Security Bulletin that same day. Medtronic wrote, "The risks are controlled, and residual risk is acceptable." (SECURITY BULLETIN, Medtronic) Medtronic stated in the Mitigations section of the advisory that they would, "…not be issuing a product update," for identified risks.

The Star Tribune reported in April of 2018, "…Saying that such a hack is possible in a controlled laboratory setting and executing it in the real world are different things, skeptics say." The article cites,

"…Dr. Kevin Wheelan, the chief of staff at Baylor Scott & White Heart and Vascular Hospital in Dallas who has worked as an investigator on Medtronic clinical studies, (and who) said such risks seem remote. "The combination of events that would be required are almost impossible to envision other than in a staged scenario where someone was trying to prove that possibility that could occur," Wheelan said. "None of us consider that a real-world clinical threat."

However, on October 11, 2018 The NCCIC updated its advisory for the Carelink Programmer Vulnerability (Update B). The Mitigations section of the advisory was revised to read, "After additional review and risk evaluation of the affected products, Medtronic has disabled the networked-based software update mechanism, including both the VPN and HTTP subservices, as an immediate security mitigation." Risks were described as including, "Improper restriction of communication channel to intended endpoints CWE-923." The advisory describes vulnerabilities to malicious updates and man-in-the-middle attacks. Investigation concluded that the device, "…insufficiently relied on the security of the VPN." The Common Vulnerabilities and Exposures Score was increased to 7.1, indicating a High risk vulnerability. This was revised upward from the previous scores, indicating a Medium risk vulnerability, 4.8 to 4.9. These previous scores were associated with previously acknowledged vulnerabilities related to storing passwords in a recoverable format and, "…directory traversal vulnerabilities that could allow an attacker to read files…"

References

FCC rules for unlicensed wireless equipment operating in the ISM bands. Retrieved from http://afar.net/tutorials/fcc-rules/

2017 wild west hackin' fest labs. (2017). (2017). Paper presented at the Wild West Hackin' Fest, Retrieved from www.youtube.com/watch?v=KQMXT56OMnI

3G and 4G smartphone security; (2017). Retrieved from https://www.redcom.com/3g-4g-smartphone-security/

433 MHz vs. 2.4 GHz –comparison. Retrieved from www.smartbridge-tech.com/2.4%20vs%20433.pdf

Aihie, I. (2016). FDA CDRH webinar: A dialogue between the diabetes community and FDA; (). Retrieved from www.fda.gov/downloads/MedicalDevices/NewsEvents/WorkshopsConferences/UCM506113.pdf

Barnaby jack. (2018). Retrieved from https://en.wikipedia.org/wiki/Barnaby_Jack

Benedict, B., Keyes, R., & Sauls, C. (2004). The insulin pump as murder weapon: A case report. The American Journal of Forensic Medicine and Pathology, (Volume 25), 159-160. doi:10.1097/01.paf.0000127383.69760.72

Board on Physics and Astronomy. (2015). A strategy for active remote sensing amid increased demand for radio spectrum . 500 Fifth Street, NW Washington, DC 20001: National Academies Press.

Bounty sourcing the creation of an omnipod plugin for the openaps project. Retrieved from www.openomni.org/pledges/

Bourdena, A., George Kormentzas, & Skianis, C.Real-time TVWS trading based on a centralized CR network architecture. Paper presented at the 2011 IEEE GLOBECOM Workshops (GC Wkshps), doi:10.1109/GLOCOMW.2011.6162600 Retrieved from https://ieeexplore.ieee.org/document/6162600

Burleson, W., & Carrara, S. (2014). Security and privacy for implantable medical devices. New York: Springer.

Butts, J., & Rios, B. (August 9, 2018). (August 9, 2018). Understanding and exploiting implanted medical devices. Paper presented at the Black Hat Conference, Retrieved from www.blackhat.com/us-18/briefings/schedule/index.html#understanding-and-exploiting-implanted-medical-devices-11733

Carlson, J. (2018, Mar 10,). Hackers take aim at common medical devices. Star Tribune Retrieved from www.startribune.com/hackers-take-aim-at-common-medical-devices/476415023/

cauthren. (2018). Dexcom G6 failing before 10 days. Retrieved from https://forum.tudiabetes.org/t/dexcom-g6-failing-before-10-days/71578

Cheney's heart device altered to prevent assassination. (2013). Retrieved from www.youtube.com/watch?v=X19Y3SwvCqM

Clark, P. (Oct 26, 2018). (Oct 26, 2018). Building a small and flexible wireless exfiltration box with SDR Paper presented at the Wild West Hackin' Fest, Retrieved from https://wwhf18.sched.com/event/FoAh/building-a-small-and-flexible-wireless-exfiltration-box-with-sdr

Class 2 device recall ATLAS VR model V193. (2008). Retrieved from www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfres/res.cfm?id=68597

Comparing wireless communication protocols. (2018). Retrieved from www.anixter.com/en_mx/resources/literature/techbriefs/comparing-wireless-communication-protocols.html

Comstock, J. (2016). In JAMA editorial, nightscout opens up about dealings with FDA. Retrieved from www.mobihealthnews.com/content/jama-editorial-nightscout-opens-about-dealings-fda

Confirm rx™ insertable cardiac monitor. (2018). Retrieved from www.sjm.com/en/professionals/featured-products/electrophysiology/recording-and-monitoring/implantable-cardiac-monitor/confirm-rx-insertable-cardiac-monitor/tech-specs

Dehaaff, M. (2018). Glooko's mobile insulin dosing system – MIDS – is FDA cleared! Retrieved from www.glooko.com/2018/02/glookos-mobile-insulin-dosing-system-mids-fda-cleared/

DexCom opens data platform and launches developer program to fuel diabetes app innovation. (2017). Retrieved from https://provider.dexcom.com/industry-news/dexcom-opens-data-platform-and-launches-developer-program-fuel-diabetes-app-innovation

Eisenbarth, T., & Kasper, T. (2008). Messing around with messing around with garage doors - breaking KeeLoq with power analysis. Unpublished manuscript. Retrieved from https://www.emsec.ruhr-uni-bochum.de/media/crypto/attachments/files/2011/05/KeeLoq25C3_opt.pdf

Episode 33: Dr. jonathan butts & billy rios on cyber security, public safety & the layers of defense. (2018). Retrieved from https://shows.pippa.io/5ac7388bca218bc36c067efc/episode33

FAIR HISTORY & ARCHIVES. Retrieved from www.mnstatefair.org/about-the-fair/history/

Featured remote care products. (2016). Retrieved from www.sjm.com/en/professionals/featured-products/cardiac-rhythm-management/remote-care

Fletcher, D.YardStick one replay attack lab. Paper presented at the Wild West Hackin' Fest, Retrieved from www.dropbox.com/s/cd4kk0407i4q5lh/Wireless%20Doorbell%20Replay%20Attack%20Lab.pdf?dl=0

Frenzel, L. E. (2008). Short-range radios enable wireless everything. Retrieved from www.electronicdesign.com/communications/short-range-radios-enable-wireless-everything

Frequency-shift keying. (2018). Retrieved from https://en.wikipedia.org/wiki/Frequency-shift_keying

Frequently asked questions. Retrieved from www.myomnipod.com/podder-support/faq

Introduction to magnetic resonance imaging (MRI). (2017). ().Texas Instruments. Retrieved from www.ti.com/lit/an/sbaa248/sbaa248.pdf

ISM band. (2018). Retrieved from https://en.wikipedia.org/wiki/ISM_band

Jeffries, A. (2014). Famous hacker barnaby jack's death ruled a drug overdose. Retrieved from www.theverge.com/2014/1/3/5270352/famous-hacker-barnaby-jacks-death-was-a-drug-overdose

Kolbasuk McGee, M. (2018). FDA calls for 'cybersecurity bill of materials' for devices. Retrieved from www.inforisktoday.com/fda-calls-for-cybersecurity-bill-materials-for-devices-a-11628

Mila. (2018). Our dexcom G6 experiences. Retrieved from https://forum.tudiabetes.org/t/our-dexcom-g6-experiences/68655/24

MiniMed™ 640G system user guide. (). 18000 Devonshire St Northridge, CA 91325: Medtronic. Retrieved from www.medtronic.com/content/dam/medtronic-com/de-de/hcp/documents/diabetes/download-page/insulinpumpe/MiniMed-640G-UserGuide-mmol-englisch.pdf

Mortensen, H.Internet of diabetes. Paper presented at the Medfuse 2018, Retrieved from www.medfuse.io/home/

Nelson, T. (2018). New chief leading minnesota state fair police force. Retrieved from www.mprnews.org/story/2018/09/01/new-chief-leading-minnesota-state-fair-police-force

Nightscout contributors. (2016). Nightscout FDA presubmission ().FDA. Retrieved from https://media.readthedocs.org/pdf/nightscout-fda-presubmission-01/latest/nightscout-fda-presubmission-01.pdf

Perspectives on medical devices; Paper presented at the ISACA Presentation,

Microsoft (Producer), & Rayanchu Shravan (Director). (2012, March 12,). Understanding wireless interference in the unlicensed band. [Video/DVD] University of Wisconsin-Madison: YouTube. Retrieved from https://www.microsoft.com/en-us/research/video/understanding-wireless-interference-in-the-unlicensed-band/

Rayanchu, S., Patro, A., & Banerjee, S. (2011). (2011). Airshark: Detecting non-WiFi RF devices using commodity WiFi hardware. Paper presented at the Imc'11,

Rayanchu, S., Banerjee, S., & Patro, A.Airshark: Detecting non-WiFi RF devices using commodity WiFi hardware. Paper presented at the Imc'11, Retrieved from http://pages.cs.wisc.edu/~suman/pubs/airshark.pdf

Reverberi, L., & Oswald, D. (2016). Breaking (and fixing) a widely used continuous glucose monitoring system. Retrieved from www.usenix.org/system/files/conference/woot17/woot17-paper-reverberi.pdf

Skorobogatov, S. (Oct 26, 2017). (Oct 26, 2017). Hardwear.io 2017:- Challenging real world targets by dr sergei skorobogatov. Paper presented at the Hardwear.Io,

Some questions you might have. Retrieved from http://www.openomni.org/questions/

StephenBlackWasAlreadyTaken/wixel-xDrip. Retrieved from https://github.com/StephenBlackWasAlreadyTaken/wixel-xDrip

United states frequency allocations radio frequency spectrum. (2003). ().US Dept of Commerce. Retrieved from www.ntia.doc.gov/files/ntia/publications/2003-allochrt.pdf

Vagus nerve stimulation (VNS). Retrieved from www.epilepsy.com/learn/treating-seizures-and-epilepsy/devices/vagus-nerve-stimulation-vns

Why medical device manufacturers must lead on cybersecurity in an increasingly connected healthcare system. (2018). (). Washington DC: Retrieved from http://dam.abbott.com/en-us/documents/pdfs/cybersecurity/Chertoff_Abbott_WhitePaper_FINAL.pdf