



BLE Smart Tags and Sales in Physical Clothing Retail Markets

Yu-Hsuan Yeh

Background & Motivation

- Online shopping rise
 - Clear item information
 - Low cost -> Low price
- Physical clothing store
 - Touch
 - Try-on

RFID?

- ✓ auto inventory
- ✗ special, expensive reader
- ✗ benefit only retailers

BLE Beacon?

- ✓ active advertising
- ✓ received by cellphone
- ✗ fix data format
- ✗ need a server/database
- ✗ benefit only customer

Goal

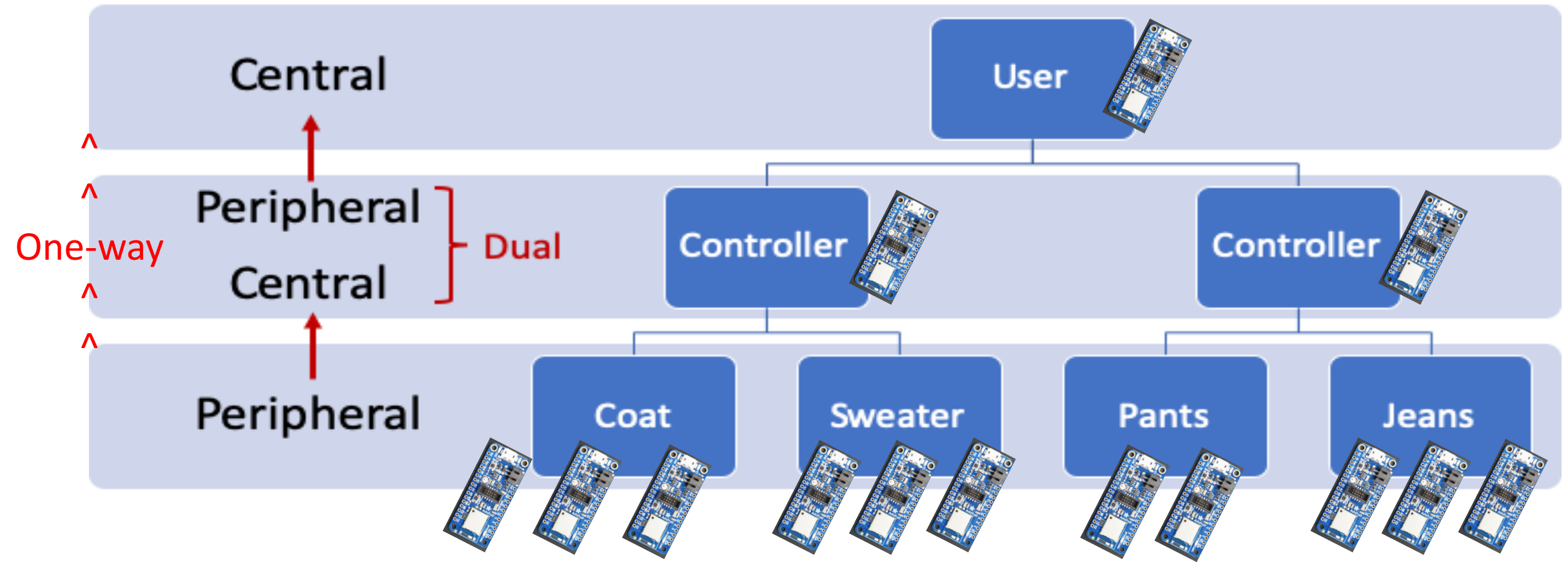
- Use BLE advertising and passive scanning mode
- Decentralized architecture
 - Clothing data is scattered on each tag
- Integrated system for both customers and retailers
- Active information advertisement
 - New data format for clothing
- Inventory automation



Approach

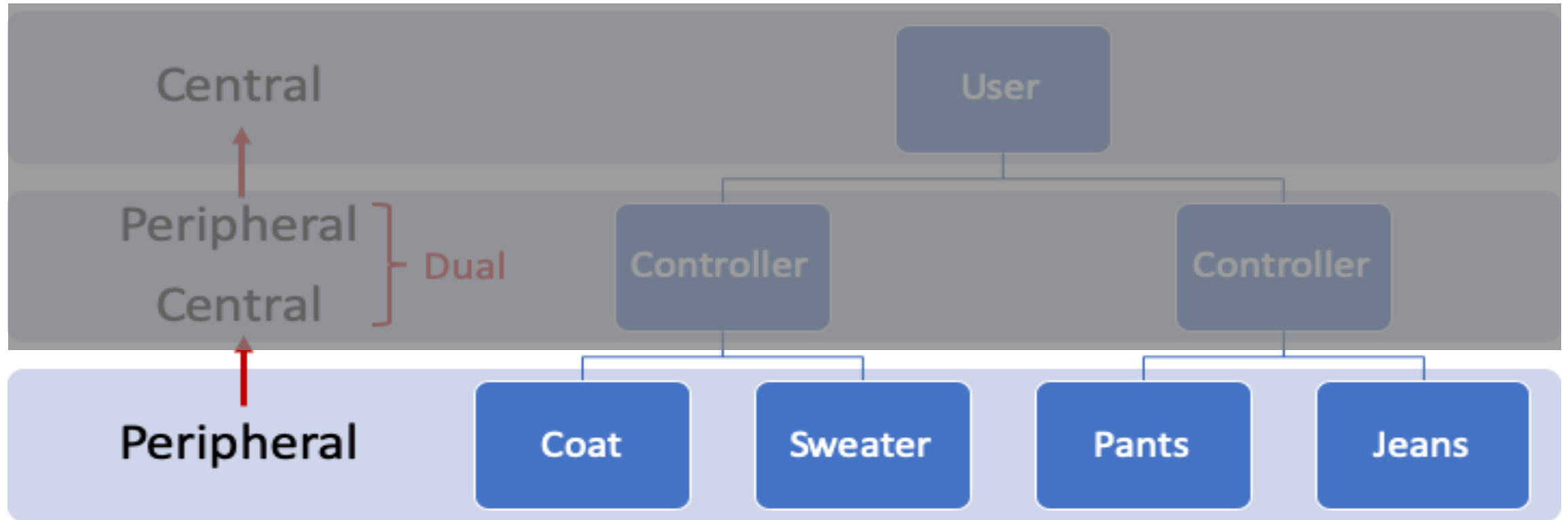
Communication Architecture

- Peripheral = Advertiser = Transmitter
- Central = Scanner = Receiver



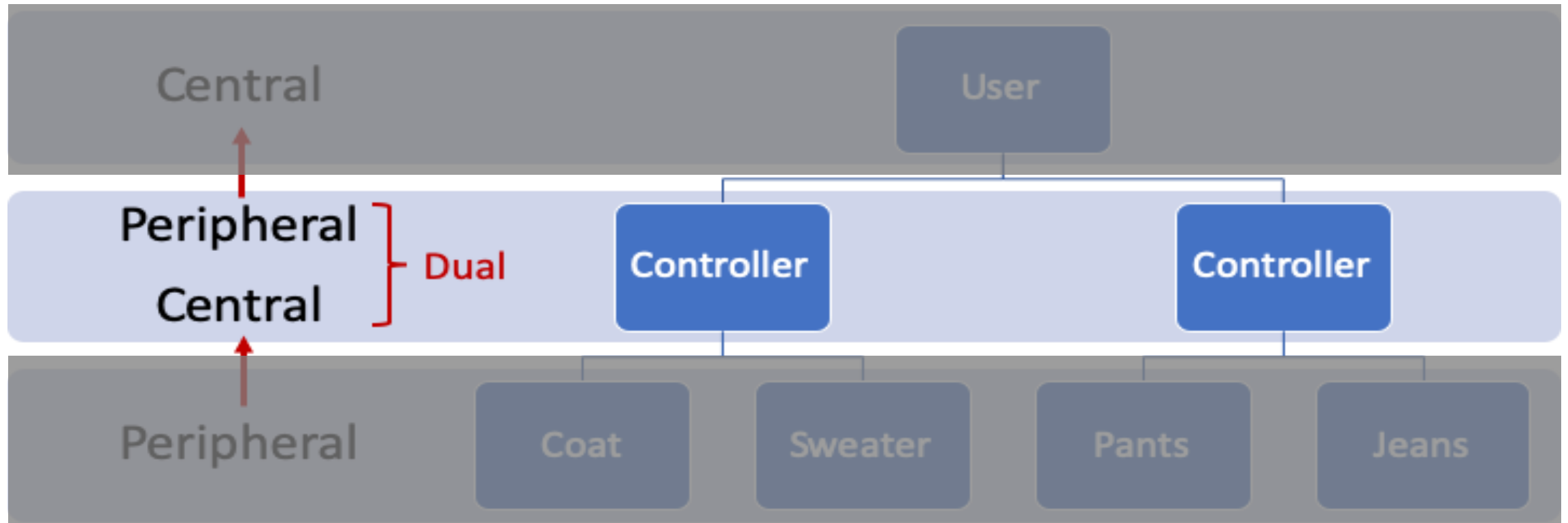
Communication Architecture

- Advertise encoded clothing information
- Specify which controller to receive



Communication Architecture

- Integrate received clothing info, stored in catalog
- Advertise catalog sequentially to user



Address Format

- Encoded in **Clothing** ADV **Address**

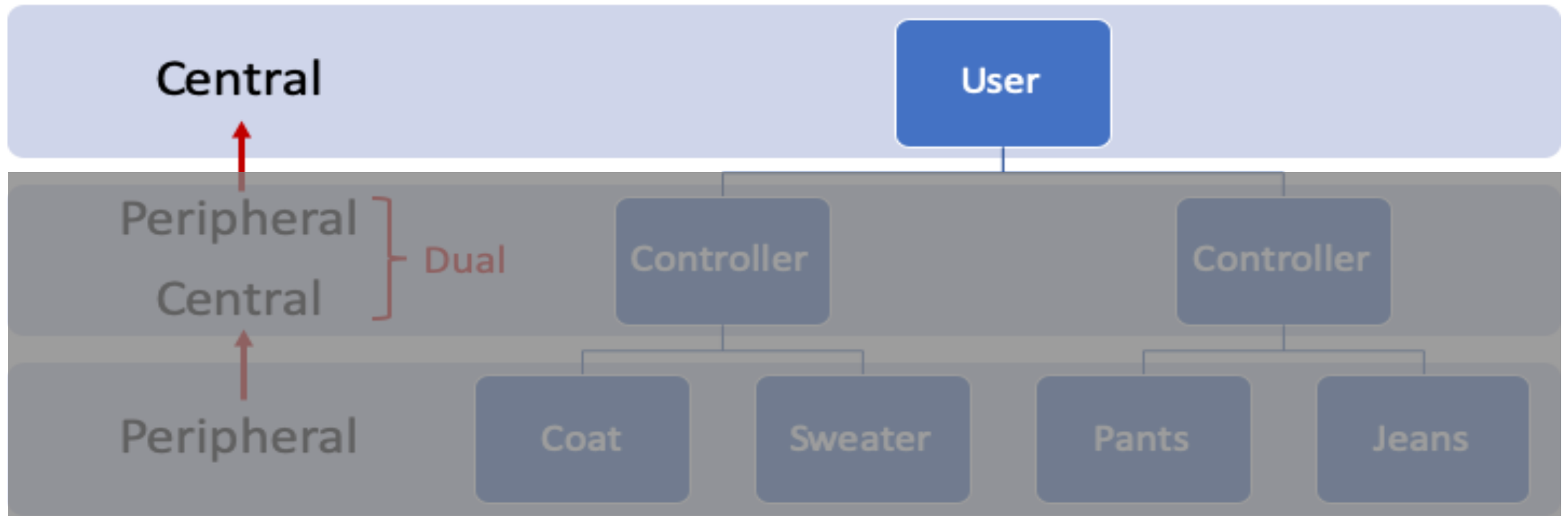
Bit	4	2	2	2	2	4	4	4	2	6	8	8		
Address	C	1		7		6	3	0	8	F	0	B	0	3
Content	cannot modify	receiver	sender	group	season	category	size	color	promo	price(x3+5)	item ID		instance ID	
Example	-	ctrl 0	item	female	winter	coat	M	red	XMAS	50	item No.11		the third coat in stock	

- Encoded in **Controller** ADV **Name**

Bit	4	2	2	2	2	4	4	4	2	6	8	8
Address	C	0	7	6	3	0	8	F	0	B	0	A
Content	cannot modify	receiver	sender	group	season	category	size	color	promo	price(x3+5)	item ID	# instance in stock
Example	-	user	ctrl 0	female	winter	coat	M	red	XMAS	50	item No.11	total 10 coats in stock

Communication Architecture

- Can be customer or retailer
- Receive same catalog, utilize differently with corresponding App

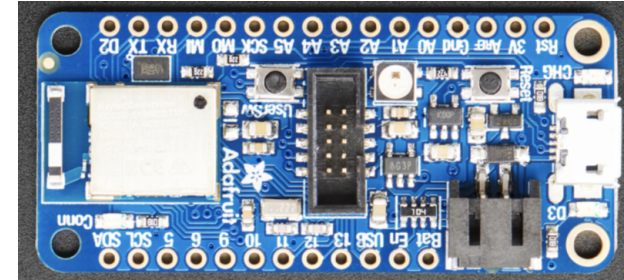




Implementation Detail

Environment

- Feather nRF52840 Express
- Arduino IDE + Adafruit Bluefruit library
 - CircuitPython ?



Modify Adv Pkt Content in Runtime

Goal:

Add sequence num
and address, adv
1pkt /sec

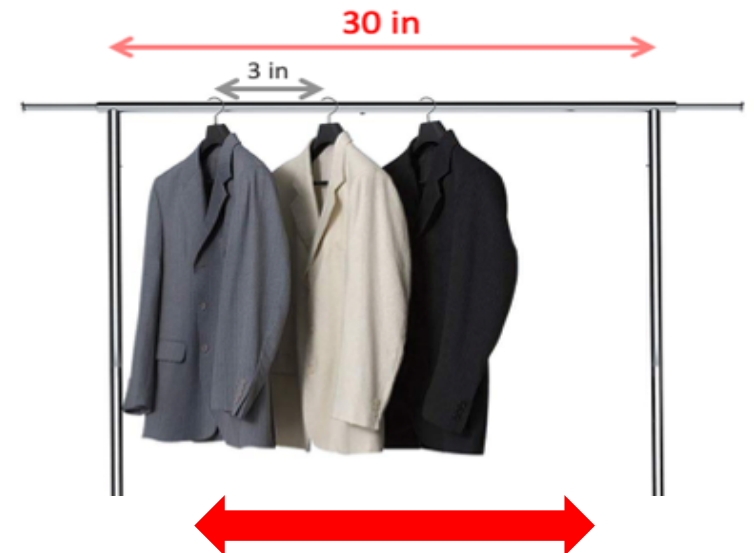
- Stop Adv
- Clear Data
- Attach to Name slot as constant string

```
// In Bluefruit lib: BLEAdvertising.cpp
ble_gap_adv_params_t adv_para = {
    .max_adv_evts = 1,
    .channel_mask = { 0, 0, 0, 0, 0 } // {7-0,15-8,23-16,31-24,39-32}
};

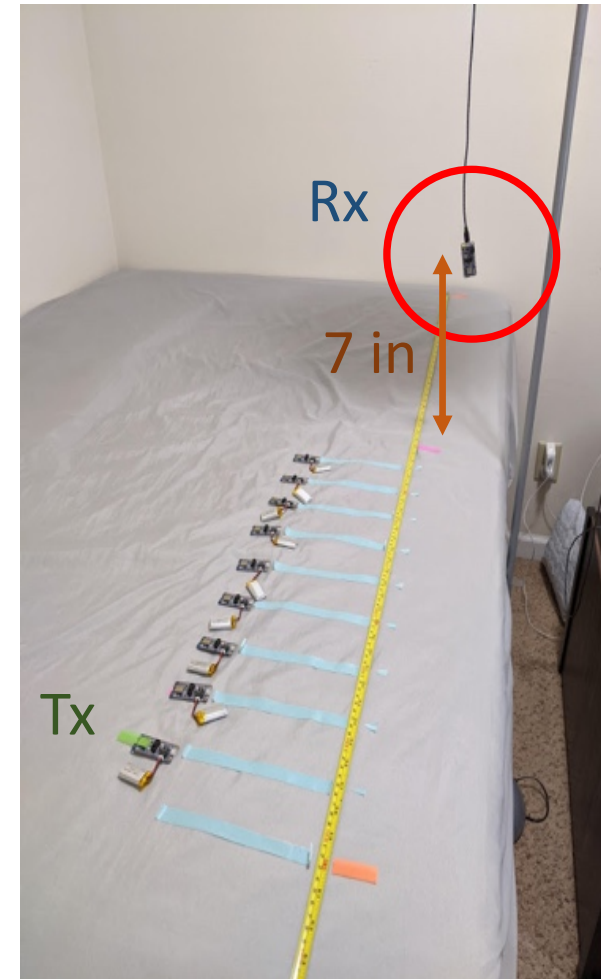
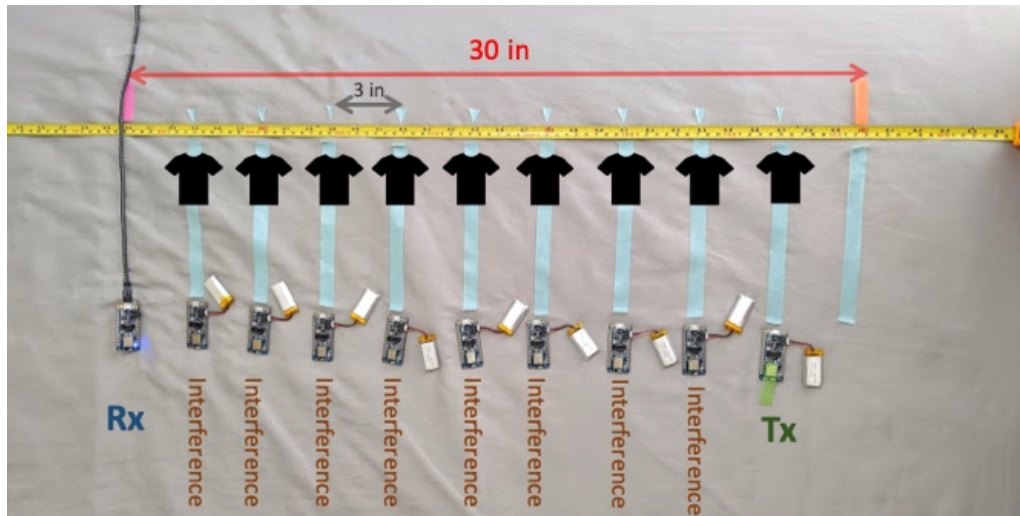
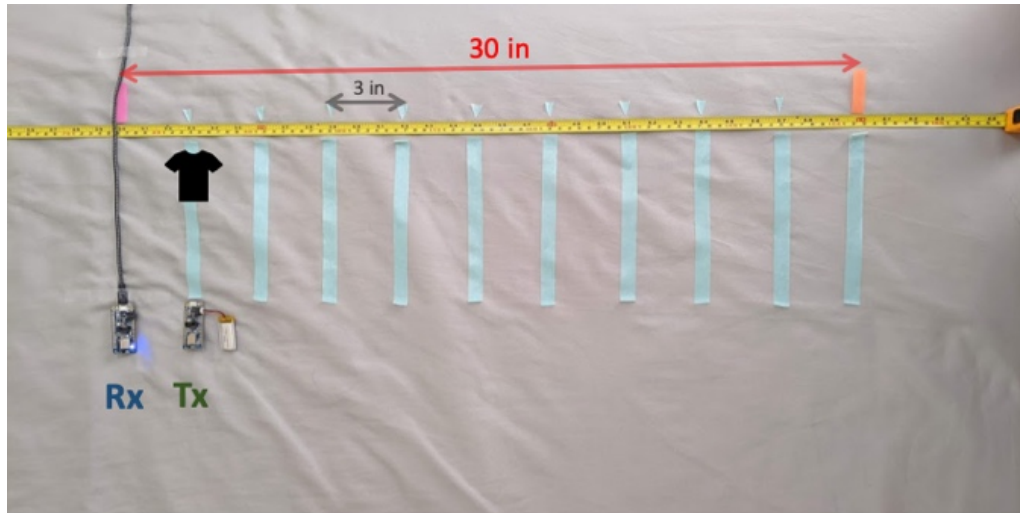
if (evt -> evt.gap_evt.params.adv_set_terminated.reason ==
BLE_GAP_EVT_ADV_SET_TERMINATED_REASON_LIMIT_REACHED){
    if (_stop_cb) ada_callback(NULL, 0, _stop_cb); // invoke stop callback
}

// In main function
void adv_stop_callback(void) {
    Bluefruit.Advertising.clearData();
    // attach seq num in name string
    sprintf(newName, "BFItem%7d", ++seq_num); // string length must be fixed
    Bluefruit.setName((char const *) newName);
    Bluefruit.Advertising.addName();
    delay(1000); // millisec
    Bluefruit.Advertising.start();
}
```

Simulate High Device Density Problem



Simulate High Device Density Problem





Evaluation

Influence Factors

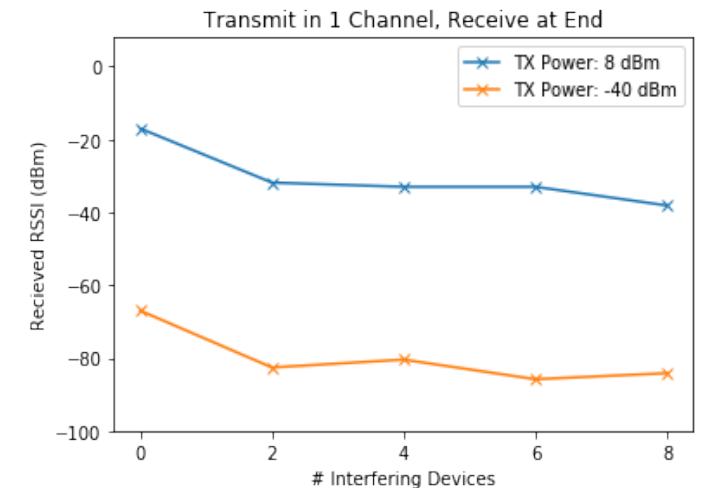
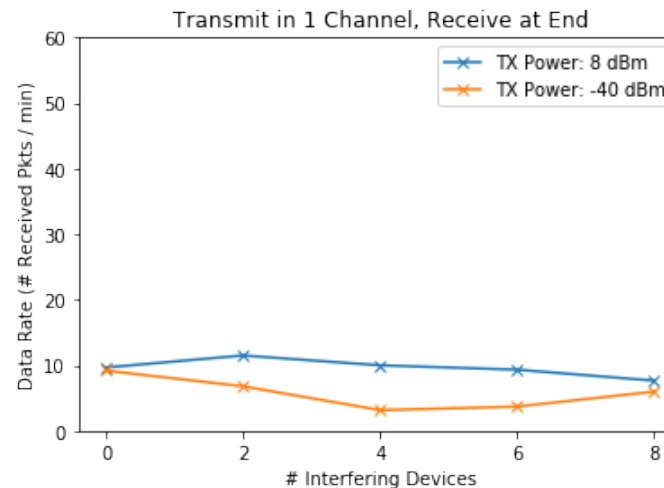
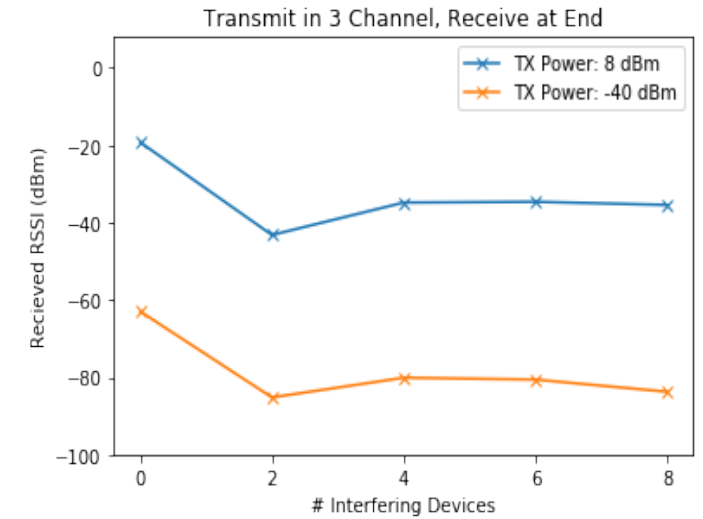
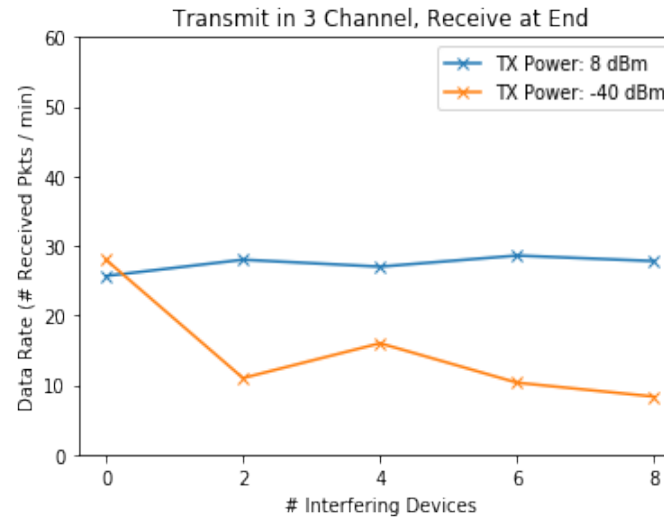
- Number of channels
 - 37, 38, 39
 - 37
- Receiver position
 - End
 - Side
- TX power
 - -40 dBm
 - 8 dBm
- Number of interference devices
 - [0, 2, 4, 6, 8]

Metrics

- Average Data Rate
 - Send 1 pkt / sec
 - # pkts received in 1 min
 - Min = 0, Max = 60
- Average RSSI

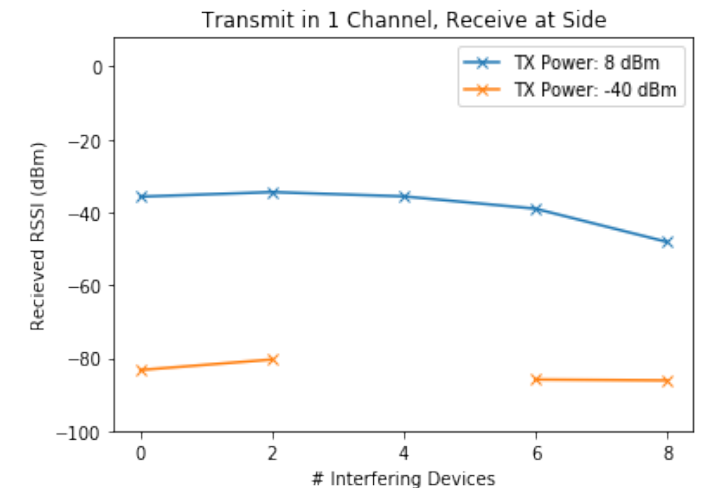
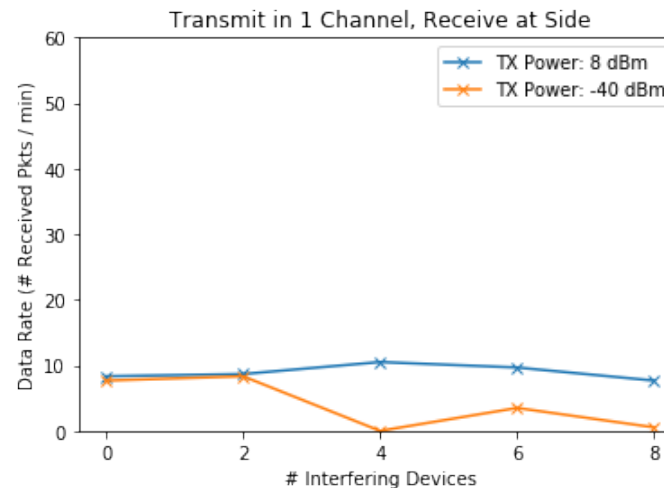
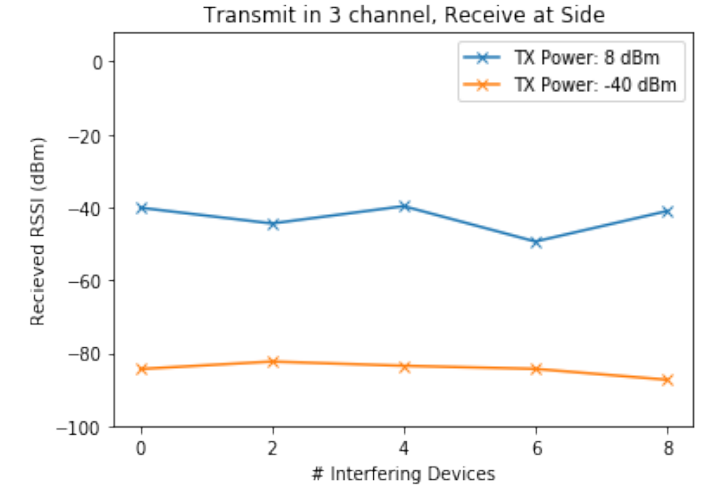
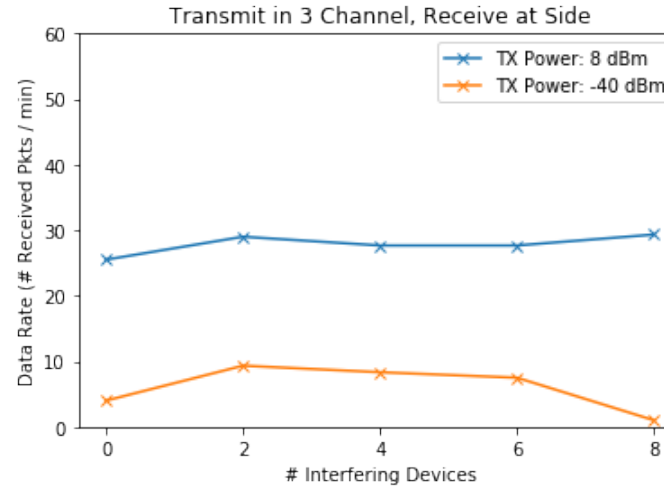
Number of Channels

- All receive at end
- Higher tx power is more robust to interferences
- Data rate is proportional to # channels



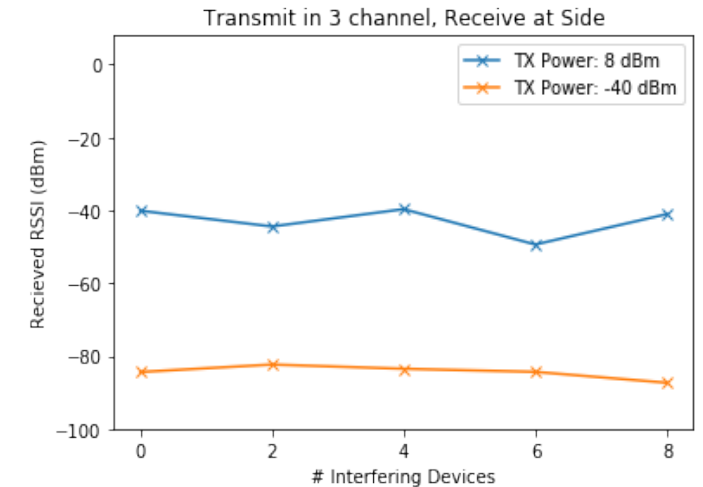
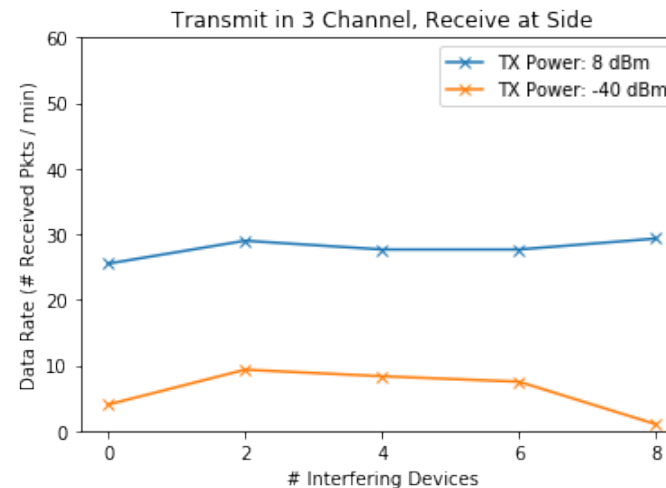
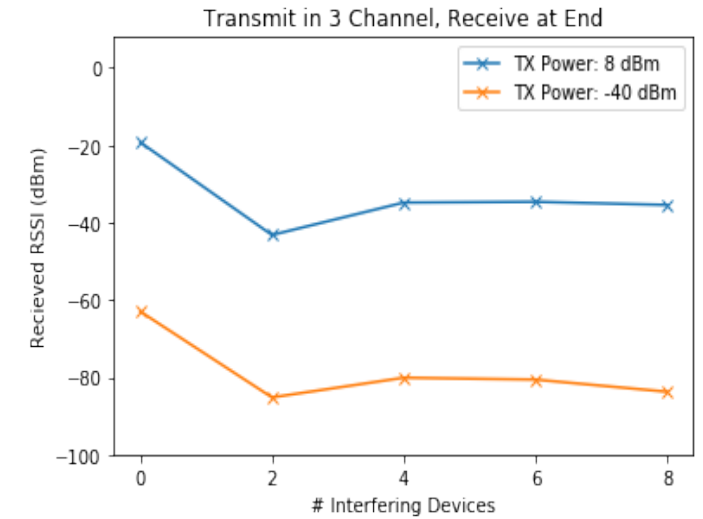
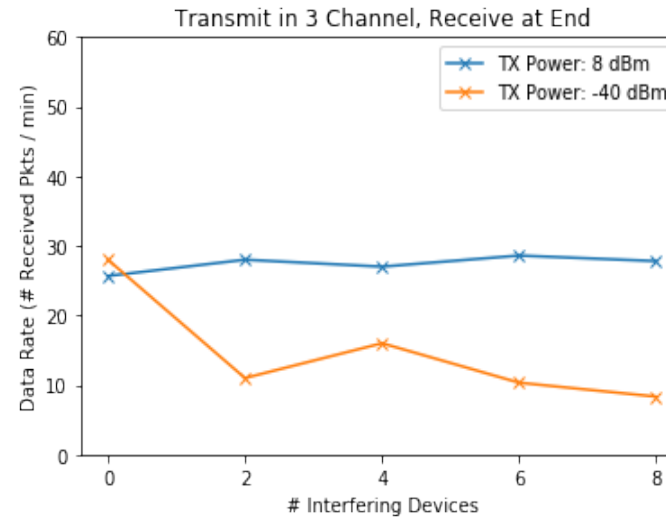
Number of Channels

- All receive at side
- Same result



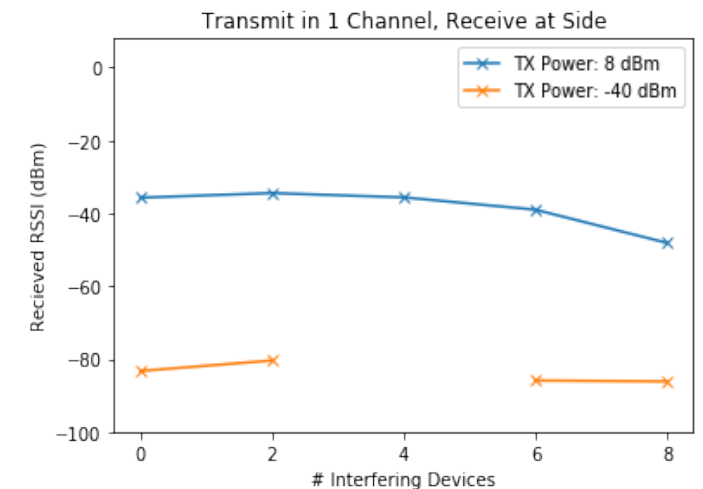
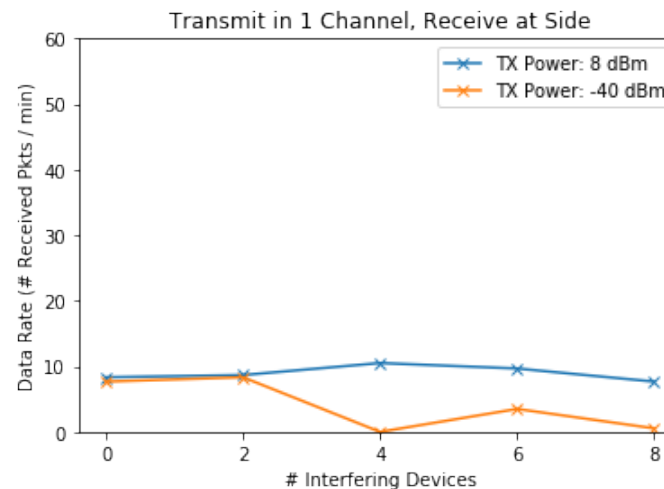
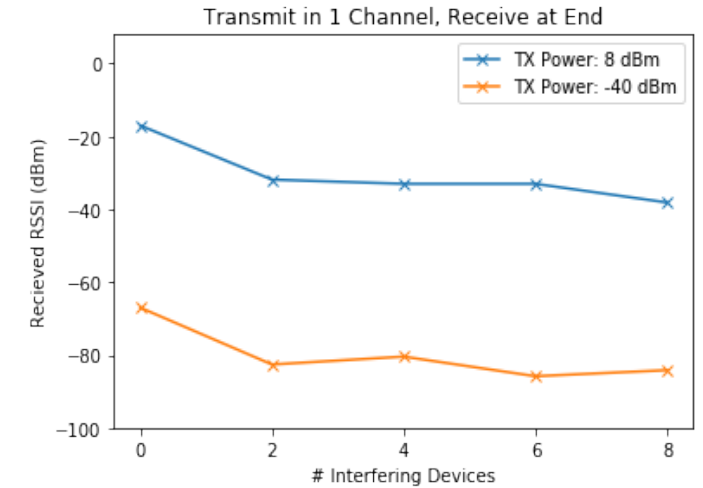
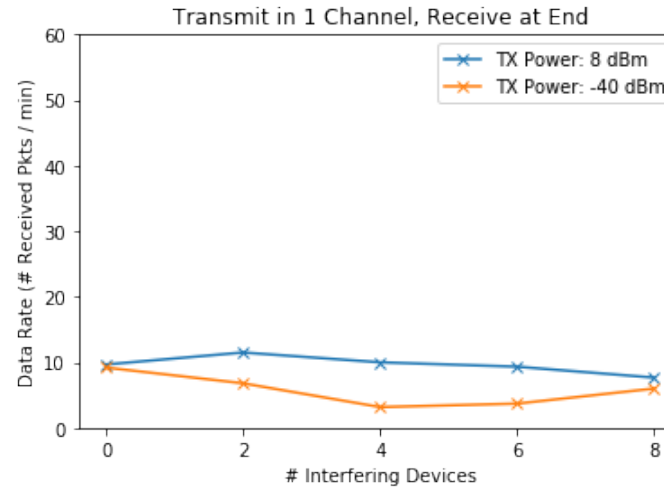
Receiver Position

- All transmit in ch37-39
- Closer to receiver, higher data rate and RSSI



Receiver Position

- All transmit in ch37
- Data rate is low, influence is not obvious
- Closer to receiver, higher RSSI



The image features a large, dark blue, irregular shape that resembles a splatter or a blot of ink. This shape is centered on a white background and has several smaller, lighter blue splatters around its edges. The word "Demo" is written in a white, sans-serif font, centered within the dark blue shape.

Demo

Two Same Orange T-shirts

F1:50:21:CF:01:03

```
RECEIVER maskRECEIVER = CTRL_TSHIRT;  
SENDER maskSENDER = ITEM;  
GROUP maskGROUP = FEMALE;  
SEASON maskSEASON = SUMMER;  
CATA maskCATA = TSHIRT;  
SIZE maskSIZE = S;  
PROMO maskPROMO = BOGO;  
uint8_t price = 50;  
COLOR maskCOLOR = ORANGE;  
uint8_t itemID = 1;  
uint8_t instanceID = 3;
```

E1:50:21:CF:01:05

```
RECEIVER maskRECEIVER = CTRL_TSHIRT;  
SENDER maskSENDER = ITEM;  
GROUP maskGROUP = FEMALE;  
SEASON maskSEASON = SUMMER;  
CATA maskCATA = TSHIRT;  
SIZE maskSIZE = S;  
PROMO maskPROMO = BOGO;  
uint8_t price = 50;  
COLOR maskCOLOR = ORANGE;  
uint8_t itemID = 1;  
uint8_t instanceID = 5;
```

White & Gray T-shirts

F1:90:13:8F:02:01

```
RECEIVER maskRECEIVER = CTRL_TSHIRT;  
SENDER maskSENDER = ITEM;  
GROUP maskGROUP = YOUTH;  
SEASON maskSEASON = SUMMER;  
CATA maskCATA = TSHIRT;  
SIZE maskSIZE = XS;  
PROMO maskPROMO = XMAS;  
uint8_t price = 50;  
COLOR maskCOLOR = WHITE;  
uint8_t itemID = 2;  
uint8_t instanceID = 1;
```

E1:D0:42:8F:06:01

```
RECEIVER maskRECEIVER = CTRL_TSHIRT;  
SENDER maskSENDER = ITEM;  
GROUP maskGROUP = KID;  
SEASON maskSEASON = SUMMER;  
CATA maskCATA = TSHIRT;  
SIZE maskSIZE = L;  
PROMO maskPROMO = XMAS;  
uint8_t price = 50;  
COLOR maskCOLOR = GRAY;  
uint8_t itemID = 6;  
uint8_t instanceID = 1;
```

Black & Blue Shorts

C5:51:34:CF:01:01

```
RECEIVER maskRECEIVER = CTRL_SHORTS;  
SENDER maskSENDER = ITEM;  
GROUP maskGROUP = FEMALE;  
SEASON maskSEASON = SUMMER;  
CATA maskCATA = SHORTS;  
SIZE maskSIZE = M;  
PROMO maskPROMO = BOGO;  
uint8_t price = 50;  
COLOR maskCOLOR = BLACK;  
uint8_t itemID = 1;  
uint8_t instanceID = 1;
```

E5:51:37:4F:07:01

```
RECEIVER maskRECEIVER = CTRL_SHORTS;  
SENDER maskSENDER = ITEM;  
GROUP maskGROUP = FEMALE;  
SEASON maskSEASON = SUMMER;  
CATA maskCATA = SHORTS;  
SIZE maskSIZE = M;  
PROMO maskPROMO = NEWMEMBER;  
uint8_t price = 50;  
COLOR maskCOLOR = BLUE;  
uint8_t itemID = 7;  
uint8_t instanceID = 1;
```




Conclusion

Challenge & Future

- If clothes are moved frequently...
- Checkout automation
- App on phone
- Active scanning
- Work as a scanner, extract price from address
- Combine with ML prediction