0222

【実証実験（大学内）】

2/27にある養殖場で実証実験前のテストとして, 大学内でデータの送受信を実施.

以下の場所にデバイスを設置し, 院生部屋にあるセンサ端末(Enddevice)からK棟前のCoordinatorまでデータ送信を行った.

院生部屋 → アルベルト先生部屋の前 → 廊下の窓際 → 南棟入り口前の道路 →K棟

結果として, データをEnddeviceからCoordinatorで受信することができた.

以下はSnifferで得た詳細なデータ情報である.

グラフィカル ユーザー インターフェイス, テキスト, アプリケーション

中程度の精度で自動的に生成された説明

以下はCoordinatorで受信したデータである（Enddeviceのshort addressとセンサデータ）

新聞の記事のスクリーンショット

低い精度で自動的に生成された説明

〇IEEE 802.15.4 Data の部分

Extended Sourseとして, 00:1b:c5:01:22:03:7c:72 は

Coordinatorに送信する一つ手前の「南棟入り口前の道路」に設置してあったデバイスのMACアドレスと一致した.

〇ZigBee Network Layer Data の部分

Extended Sourseとして,　00:1b:c5:01:22:01:6b:dd は,

「院生部屋」にあるセンサ端末(Enddevice)のMACアドレスと一致していた.

Short addressは0x28dcだった.

　今回の実証実験前テストでは, 以下の２点を明らかにすることができた.

・センサ端末であるEnddeviceからCoordinatorに送信していることを確認.

・どのデバイスで生成されたデータがどこのデバイスを経由して送信されてきたのかを確認.

また以下の２点の問題点も判明した.

・デバイス間の距離が遠くなると, データの受信速度が低下する. データが抜けている, などといったデータの照らし合わせは行っていない.

・「APP：vCheckStackEvent : unhandled event 29」といったエラー文が数回出現した. エラーの詳細は今後調査する.

【センサデータ送受信のプログラム】

〇以下のように　センサ → Enddevice　は16進数でセンサデータを正確に受信していることが分かる.

Enddeviceでは, RxByte[i] = rxByte;　のようにセンサからのデータ(rxByte)をRxByte[]の配列に格納している.

携帯電話の画面のスクリーンショット

自動的に生成された説明



またEnddevice → Coordinatorでも途中まで16進数のセンサデータを受信, 表示できている.

テキスト

自動的に生成された説明



流れとして

16進数データが格納されているRxByte[]

　↓

ビット演算

　↓

16進数のセンサデータが２進数として変換

　↓

変数SensorDataに代入

　↓

Coordinatorに送信

↓

10進数として表示

以下がビット演算をするコードである.

テキスト

自動的に生成された説明

RxByte[]には, ８つの16進数データが格納されている.

結果

Enddeviceでは, SennsorData : ffffffff

Coordinatorでは, データが表示されなかった.

次に以下のようにした.

テキスト

自動的に生成された説明

結果　センサデータ「0x0023dbf97e6abce43f」に対して

Enddeviceでは, SensorData : 23dbf97e

Coordinatorでは, 何も表示されなかった.

ここでも, 途中までしか表示されないという問題になった.

また, Coordinatorに送受信できてない理由として,

・PDUM\_u16APduInstanceWriteNBO関数

・PDUM\_u16APduInstanceReadNBO関数

の引数はa/x08が原因かもしれない. ｂに変更すべきか.