ToFセンサーボード(MM-S50MV)

更新履歴

版数	更新内容	更新日
0.2	Refsを追加した。	2023/01/12
0.1	初版を0.1版とする。	2022/05/16

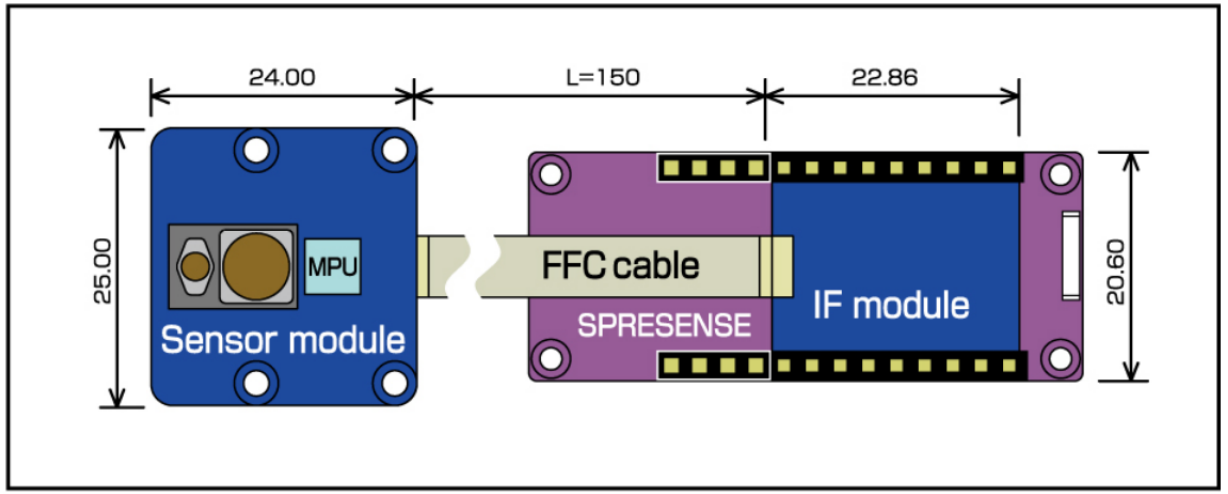
目次

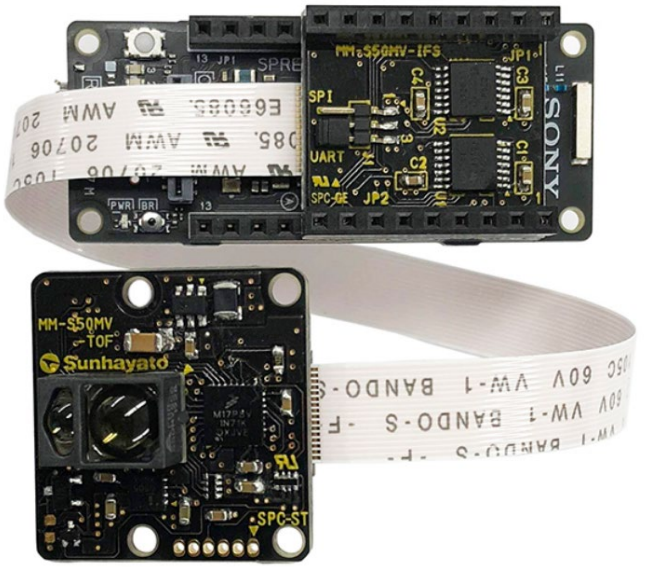
1. ToFセンサーボード（SPRESENSE用）概要
2. ToFセンサーがどのように測距しているのか
3. ToFセンサーボードはどんなデータ取得できるか
4. spidemo_Aを動作確認する(Sunhayato社提供したサンプル)
5. こちらで作ったサンプルの紹介
    1. Spresense + ToFセンサーボード + Host PC システム構成図
    2. ToFセンサーボードを動作確認するには？
    3. ToFセンサーボードから取得したデータを保存する
6. 距離データの性能と精度について
    1. 距離データの高速サンプリングについて
    2. 距離データの精度について：4mまでの検証結果
    3. 使用環境について検証結果(身近な環境のものを使っています)：
7. 3Dの距離データについて
    1. 3Dの距離データの確認方法について：一部pixelデータのみ有効
8. 気になる点纏め：
    1. サンプルはライブラリ化されていないため、わかりずらいです。
    2. 3Dの距離データを取得処理がないです。
    3. 照度のデータを取得処理がないです。
    4. 動作確認するときに、ToFモジュールが発熱することがあります。手で触ると、熱が感じられる程度です。
    5. SPI通信の場合、3D距離データの有効pixelの判定基準が記載されていません。
    6. SPI通信のDemo動作確認できますが、UART通信のDemoがないので、確認できません。
    7. ELTRES(クリエイティブジャパン)ボードと同時に使用できません。

ToFセンサーボード（SPRESENSE用）概要

1. ToF(Time of Flight)センサーボード「MM-S50MV」はレーザーと複数の受光素子により最大10mまでの高精度3D距離測定と高速サンプリングが可能です。
2. レーザー光を照射するため暗い場所でも使用が可能で、複数のセンサーを同時に近接使用してもお互いに干渉することはありません。
    1. ※詳細は https://www.sunhayato.co.jp/material2/ett09/item_1187

概略寸法

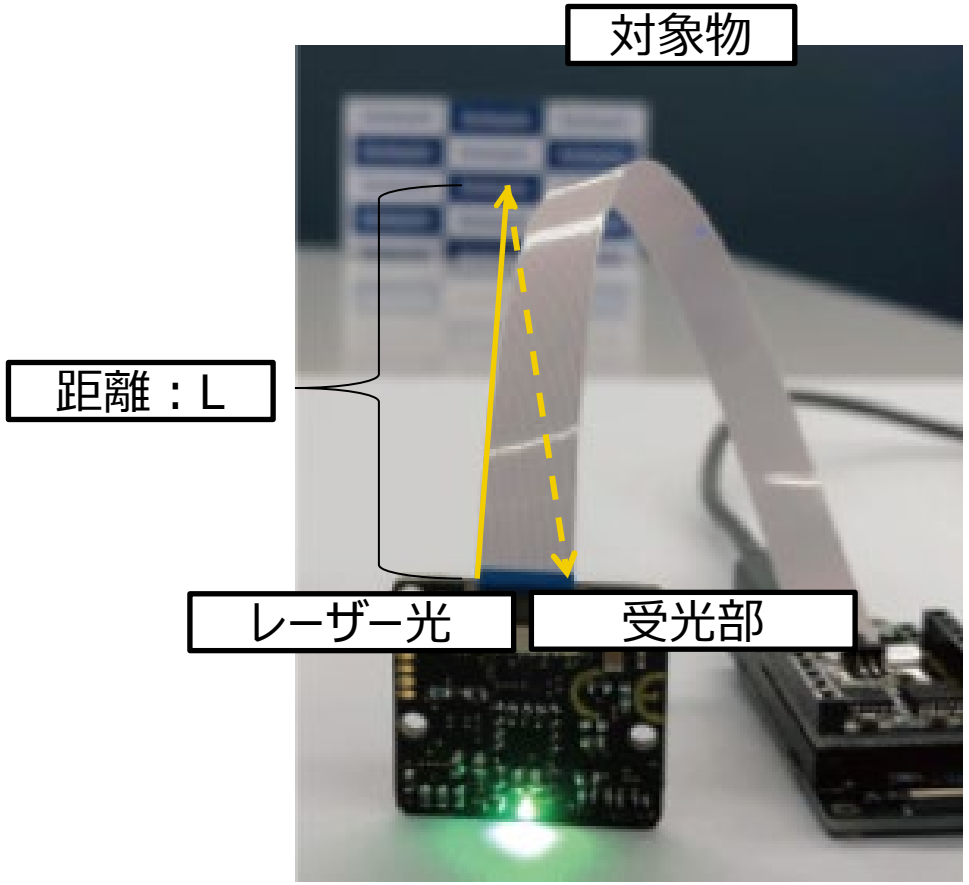




[Refs]
https://www.marutsu.co.jp/pc/i/1558728/

ToFセンサーがどのように測距しているのか

- 原理
  - 対象物に光を照射し、反射光が戻ってくるまでの時間を 計測することで対象物までの距離を求める。
- 距離　$L = ½ * C * T_{tof}$
  - C　：光の速度
  - Ttof：光の飛行時間(往復)



[Refs]
https://www.marutsu.co.jp/pc/i/1558728/

ToFセンサーボードはどんなデータ取得できるか

- 距離
  - 単位：m
  - 型：int32_t
  - 範囲：-512~511.9999998
  - 1Dデータ：数1個
  - 3Dデータ：数4×8個
- 光量(照度)
  - 単位：lux
  - 型：uint16_t
  - 範囲：0~4095.9375
    - (実際該当ToFを使ってみたら、1 kLuxぐらい場合が多い)
  - 1Dデータ：数1個
  - 3Dデータ：数4×8個

★[SPI 通信]

データ形式(256byte)　※ビッグエンディアン形式

マジック(0xe9)	シーケンス ID1	reserved	reserved	データ本体	パディング	シーケンス ID2
(1byte)	(1byte)	(1byte)	(1byte)			(1byte)

1D の距離(1)	1D の光量(1)	3D の距離(4×8)	3D の光量(4×8)
[0] (32bit)	[0] (16bit)	[0,0],[0,1],[0,2],[0,3], [1,0],[1,1],[1,2],[1,3], ・ ・ ・ ・ [7,0],[7,1],[7,2],[7,3] (32bit×32)	[0,0],[0,1],[0,2],[0,3], [1,0],[1,1],[1,2],[1,3], ・ ・ ・ ・ [7,0],[7,1],[7,2],[7,3] (16bit×32)

距離(32bit)	:	符号(1bit)	整数部(9bit)	小数部(22bit)

距離(m)＝符号×(整数部＋小数部×238.42×10^{-9}[1/2^{22}])　範囲:−512～511.9999998

光量(16bit)	:	整数部(12bit)	小数部(4bit)

光量＝整数部＋小数部×62.5×10^{-3} [1/2^{4}]　範囲:0～4095.9375

データ・ブロックの有効の有無

シーケンス ID1 ＝ シーケンス ID2:データ・ブロック有効

シーケンス ID1 ≠ シーケンス ID2:データ・ブロック無効　※準備ができていないブロックです。

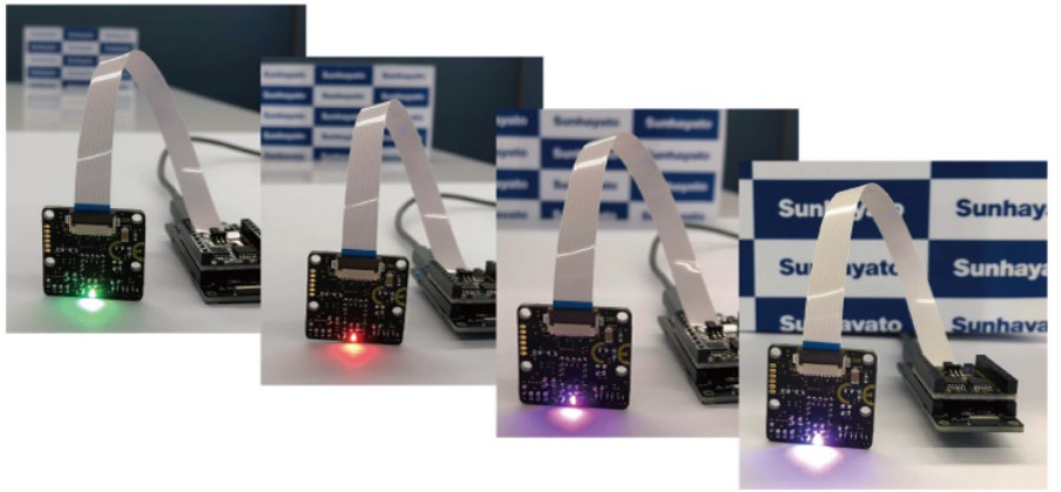
[Refs]
https://shop.sunhayato.co.jp/products/sample-program-mm-s50mv

spidemo_Aを動作確認する(Sunhayato社提供したサンプル)

- 対象との距離に応じてLED発光色が変化するプログラム
- このサンプルを動作確認とソースコードを確認したら、下記のことがわかりました。
  1. 対象との1Dの距離を使ってLED発光色を変化させていること。
  2. サンプルはライブラリ化されていないため、わかりずらいこと(気になる点1) 。
  3. 3Dの距離データを取得処理がないこと(気になる点2)。
  4. 照度データを取得処理がないこと(気になる点3) 。
  5. 動作確認するときに、ToFモジュールが発熱することがあります。
     手で触ると、熱量が感じられる程度(気になる点4) 。
- 上記の気になる点1~3について
  1. 誰でも簡単に使えるように、ライブラリ化して対応しました。
  2. 3Dの距離データを検証したいため、取得処理を追加対応しました。
  3. 照度データの使い方を検証したいため、追加対応しました。

プログラム例

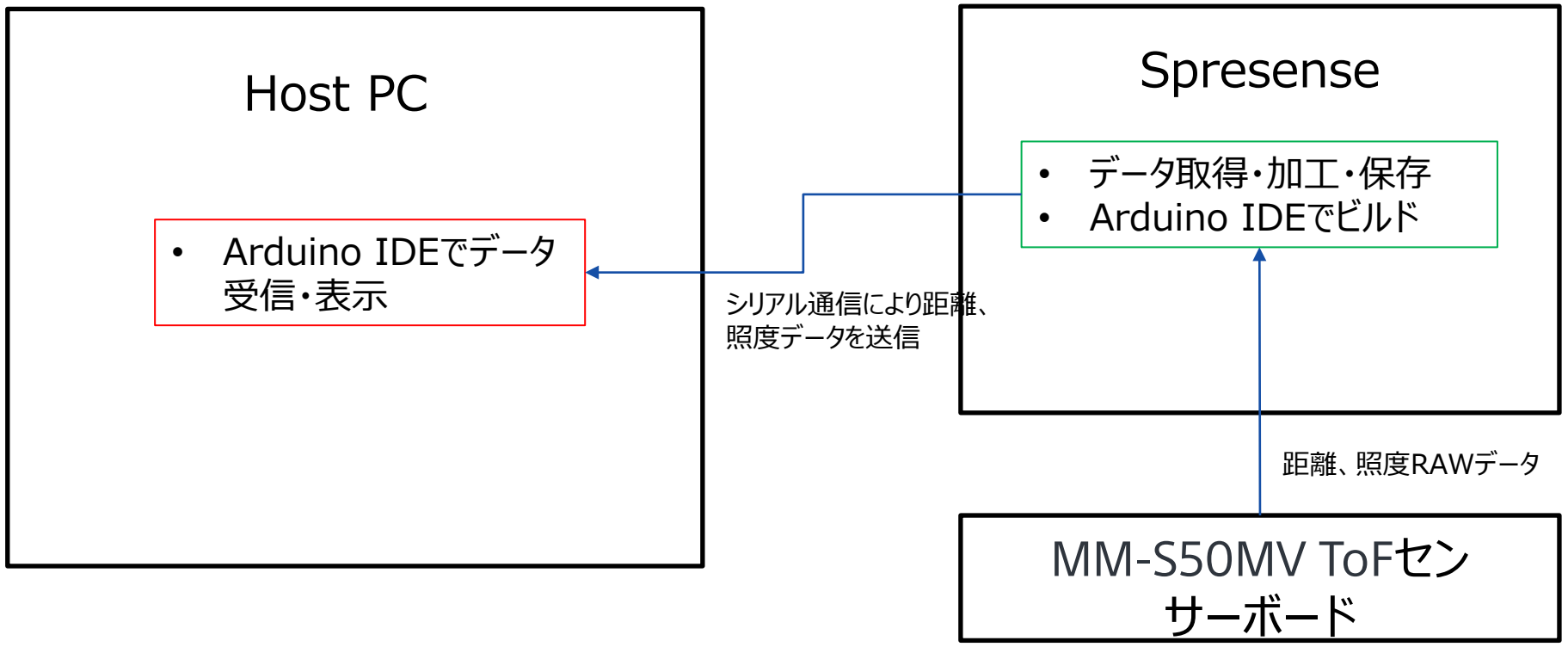
対象との距離に応じてLED発光色が変化するプログラム



[Refs]
https://www.marutsu.co.jp/pc/i/1558728/

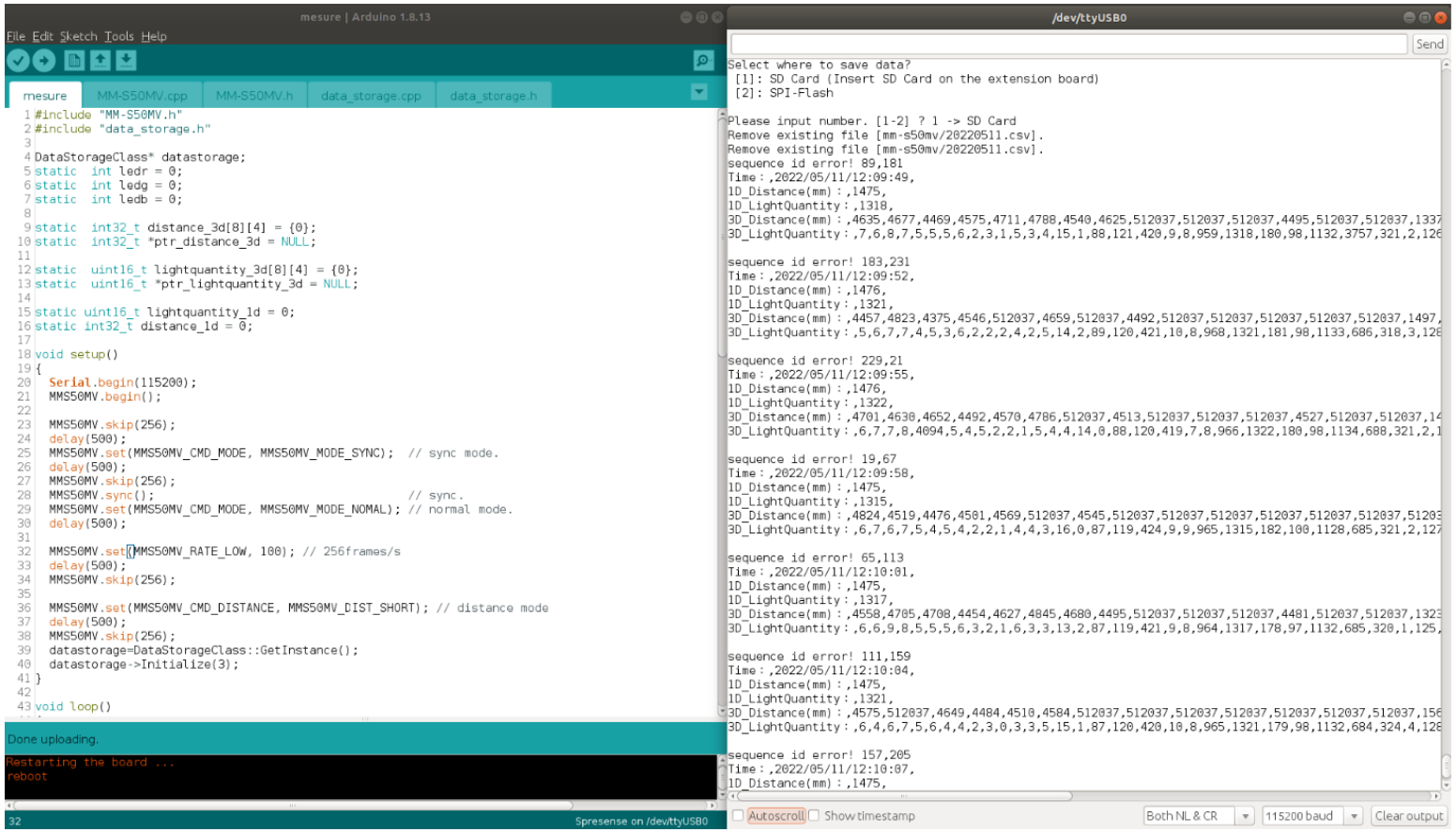
Spresense + ToFセンサーボード + Host PC システム構成図

1. 概略図



ToFセンサーボードを動作確認するには？

1. **環境**
    1. PC
        1. Ubuntu 18.04
        2. Arduino IDE:v1.8.13
    2. Spresense Arduino:v2.6.0
    3. Spresense Main Board
    4. Spresense 拡張ボード
    5. ToFセンサーボード（SPRESENSE用）（MM-S50MV）
2. **セットアップ**
    1. Spresense Arduino スタートガイドに記載の手順に従って環境を構築する
       ※Spresense Arduino環境インストール済みの場合は実施不要
3. **ビルド方法**
    1. Arduinoソースコードビルド方法を参照して、ToFのExampleをDown Loadして、Arduino IDEで開いてマイコンボードに書き込む ボタンをクリックして、スケッチのコンパイルと書き込みを行います。
    2. スケッチの書き込みが完了するまで待ちます。
    3. スケッチの書き込みが完了すると自動的にリセットがかかってプログラムが起動されます。



[Refs]
https://github.com/SonySemiconductorSolutions/ssup-spresense/tree/main/Arduino/MM-S50MV/measure

ToFセンサーボードから取得したデータを保存する

ToF_MMS50MV

説明

ToF_MMS50MVセンサーから1d、3dの距離と照度データを取得して、 FlashメモリとSDカードのどちらに下記のようなcsvの形式で保存するサンプルとなります。 データ取得間隔は秒単位で設定できます。

Header	Value
Time：	2022/04/28/17:28:04
1D_Distance(mm)：	1759
1D_LightQuantity：	954
3D_Distance(mm)：	4898,4717,4986,4806,5009,512037,4637,4640,512037,5188,512037,4732,512037,512037,1842,512037, 1766,1759,1773,512037,1399,1779,1759,1766,-510275,1756,1736,1752,512037,1743,1787,1626
3D_LightQuantity：	7,9,7,9,8,4,6,5,20,6,0,5,2,0,7,3,50,70,259,3,7,666,954,116,71,836,523,236,0,95,8,20

[Refs]
https://github.com/SonySemiconductorSolutions/ssup-spresense/tree/main/Arduino/MM-S50MV/measure

距離データの高速サンプリングについて

1. マイコン、データストリームモード、使用ピクセル数などの要素により、高速サンプリング周波数は3 kHzがサポートされてあります。

   ※AFBR-S50MV85G-DS105.pdfのP1

   – Frame rates of up to 3 kHz are supported, depending on the microcontroller, data streaming mode, and number of pixels used.

2. 右記はSpresenseを使って、1Dの距離データを30000回を取得したときに時間を計測しました。計測時間は10秒かかったため、サンプリング周波数は3 kHzであることを確認しました。

[log]

1. now time:3s
2. ……(396)
3. Stop streaming
4. end time:13s
5. diff time:10s
6. FPS:3000
7. 1476,
8. 1474,1477,1479,1476,1474,1477,1479,1476,1478,1477,1479,1476,1478,1477,1479,1476,1478,1477,1479,1476,
9. 1478,1475,1479,1476,1478,1475,1479,1476,1478,1475,1476,1478,1475,1475,1476,1478,1475,1475,1476,1478,
10. ……(30000個データ)

距離データの精度について：4mまでの検証結果

1. 1m測定結果：
    1. 1D_Distance(mm)：,999,
    2. 1D_LightQuantity：,990,

1. 2m測定結果：
    1. 1D_Distance(mm)：,2008,
    2. 1D_LightQuantity：,1087,

2. 3m測定結果：
    1. 1D_Distance(mm)：,3003,
    2. 1D_LightQuantity：,1457,

3. 4m測定結果：
    1. 1D_Distance(mm)：,3998,
    2. 1D_LightQuantity：,1272,

使用環境について検証結果(身近な環境のものを使っています)：

1. 夜暗い環境で天井に向いての測定結果(トンネル中によりもっと暗い、目がギリギリ見えるぐらい程度)：特に問題なし
    1. 1D_Distance(mm)：,1474,
    2. 1D_LightQuantity：,1338,
2. 昼間明るい環境で天井に向いての測定結果：特に問題なし
    1. 1D_Distance(mm)：,1469,
    2. 1D_LightQuantity：,970,
3. カーテンに向いての測定結果：特に問題なし
    1. 1D_Distance(mm)：,488,
    2. 1D_LightQuantity：,1467,
4. クローゼットに向いての測定結果：特に問題なし
    1. 1D_Distance(mm)：,1156,
    2. 1D_LightQuantity：,1446,
5. Note PC Displayに向いての測定結果：特に問題なし
    1. 1D_Distance(mm)：,540,
    2. 1D_LightQuantity：,1129,
6. ガラスに向いての測定結果：特に問題なし
    1. 1D_Distance(mm)：,425,
    2. 1D_LightQuantity：,1394,
7. 青い空に向いての測定結果：NG、距離は無限大のため、測定できない(予想通り)
    1. 1D_Distance(mm)：,225,
    2. 1D_LightQuantity：,116,

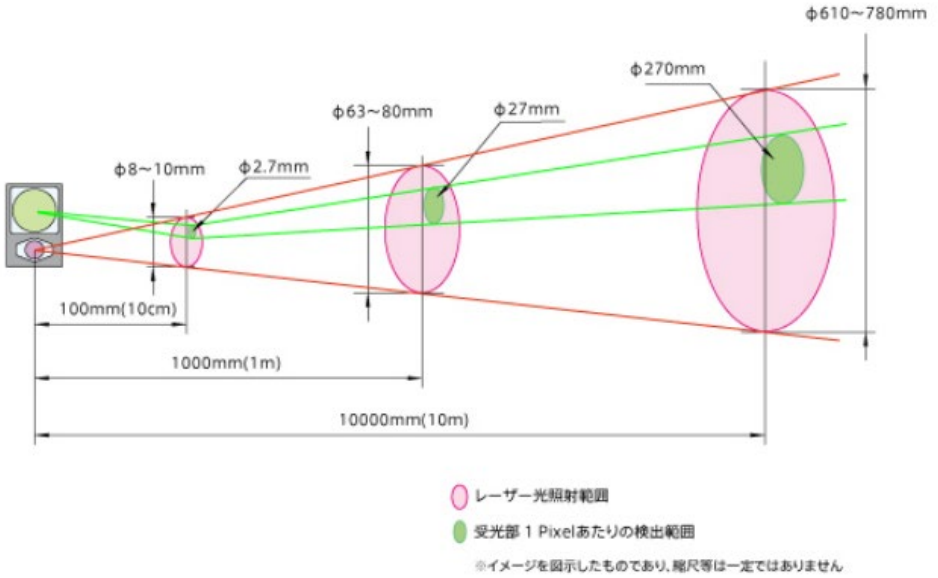
3Dの距離データの確認方法について：一部pixelデータのみ有効

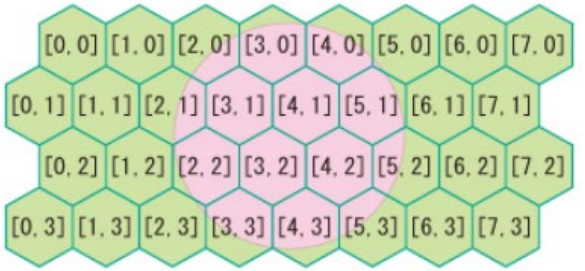
1. レーザー光及びセンサーpixelの関係イメージ
2. 下記は実際取ったデータの例(実測値1465mm)：

1D_Distance(mm):,1472,
1D_LightQuantity:,944,
3D_Distance(mm):,

512037,5074,4778,512037,	//[0,0]~[0,3]
512037,512037,512037,512037,	//[1,0]~[1,3]
512037,512037,512037,512037,	//[2,0]~[2,3]
512037,512037,1421,512037,	//[3,0]~[3,3]
1461,1418,1467,1760,	//[4,0]~[4,3]
1254,1466,1464,1468,	//[5,0]~[5,3]
1478,1483,1474,1479,	//[6,0]~[6,3]
512037,1501,1486,1552,	//[7,0]~[7,3]

3D_LightQuantity:, 2,5,5,3,
2,4,4,4,
2,0,0,3,
0,4,11,1,
53,73,280,5,
9,676,944,114,
63,788,470,220,
1,79,12,20,





レーザー光及びセンサー pixelの関係イメージ

[Refs]
https://shop.sunhayato.co.jp/products/sample-program-mm-s50mv

気になる点纏め：

1. サンプルはライブラリ化されていないため、わかりずらいです。
2. 3Dの距離データを取得処理がないです。
3. 照度のデータを取得処理がないです。
4. 動作確認するときに、ToFモジュールが発熱することがあります。手で触ると、熱が感じられる程度です。
5. SPI通信の場合、3D距離データの有効pixelの判定基準が記載されていません。
6. SPI通信のDemo動作確認できますが、UART通信のDemoがないので、確認できません。
7. ELTRES(クリエイティブジャパン)ボードと同時に使用できません。

気になる点1： Demoのソースコードはライブラリ化されていないため分かりにくい

下記の例は1Dの距離データを取る時、ソースコード変更前後Image.

変更前(ライブラリ化していない)

```
1.  spicommand(0x10, 0); // 256frames/s
2.  for (;;) {
3.      int magic0 = spigetb();
4.      int seq0 = spigetb();
5.      spiskip(2);   /* ver h/l */
6.      int range = spigetw();
7.
8.      spiskip(256 - 9 - 5 * 3);
9.
10.     int seq1 = spigetb();
11.     if (magic0 != 0xe9)
12.       continue;
13.     if (seq0 != seq1)
14.       continue;
15.     if (oldseq < 0)
16.       ;
17.     else if (((oldseq - seq0) & 0x80) == 0)
18.       continue;
19.     oldseq = seq0;
20.     int dis = range / (0x400000 / 1000);
```

変更後(ライブラリ化済み)

```
1.  int dis = MMS50MV.get1d();
```

気になる点2： 3Dの距離データを取得処理がないです。

1. SPI通信のDemoは1Dの距離を取得処理がありますが、3Dの距離データを取得する場合はソースコードを実装必要です(今回は対応しました。そしてライブラリ化済み)。

実装イメージ(ライブラリ化済み)

```
1. Int32_t dis3d[8][4] = {0};
2. MMS50MV.get3d(dis3d);
```

気になる点3：照度のデータを取得処理がないです。

1. SPI通信のDemoは1Dの距離を取得処理がありますが、照度データを取得する場合はソースコードを実装必要です(今回は対応しました。そしてライブラリ化済み)。

実装イメージ(ライブラリ化済み)

1. 1D照度データを取得処理：
    1. uint16_t light_1d = 0;
    2. light_1d = MMS50MV.get1p(light);
2. 3D照度データを取得処理：
    1. uint16_t light_3d[8][4] = {0};
    2. MMS50MV.get3p(light);

気になる点6： UART通信のDemoがないので、確認できません。

1. SPI通信のDemo動作確認できますが、UART通信のDemoがないので、確認できません。
2. 仕様書により、UART通信により下記のような構造のデータが出力されますが、そのDemoが提供されていないです。

★[UART 通信]

下記のような構造のデータが出力されます。

exsample データ

→1243→44.852419→0→1.626752→998.06→ 7→0000|0000|0000|0000|1110|0110|0110|0000→511.000000→4.324787→511.000000→511.000000→511.000000→
511.000000→511.000000→511.000000→511.000000→511.000000→511.000000→511.000000→1.775832→511.000000→1.674620→511.000000→1.609840→
1.629318→1.635115→1.794137→1.484615→1.623412→1.628170→1.647884→1.623507→1.621020→1.644051→1.644969→511.000000→1.644816→1.788122→
1.749969→4.81→5.00→3.06→4.19→4.63→4.31→3.19→3.69→3.06→1.81→1.81→3.38→11.44→4.75→20.06→3.19→270.19→232.25→459.44→9.69→ 14.06→
998.06→941.06→149.94→186.13→942.56→435.25→170.94→4.06→152.69→14.31→18.94→0b00100100→0b00000100→0b001001 00→0b00100100→
0b00100100→0b00100100→0b00100100→0b01100100→0b00100100→0b01100100→0b00100100→0b00100100→0b00000100→0b00100100→0b00000100→
0b00100100→0b00000000→0b00000000→0b00000000→0b00000100→0b00000100→0b00000000→0b00000000→0b00000100→0b00000100→0b00000000→
0b00000000→0b00000100→0b00100100→0b00000100→0b00000100→0b00000100→22→4.0000→High→54.56→MediumHigh→42→2883.50→2895.81→3648.13→
2054.75→33.69→A-Frame→True→False→False CR LF

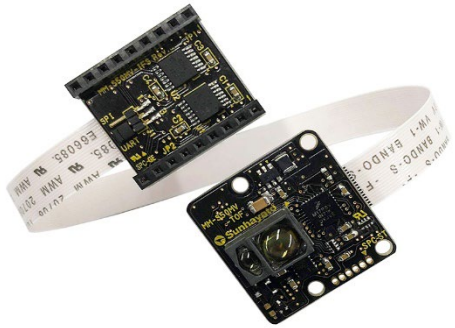
→ :Tab コード(0x08), CR :CR コード(0x0D), LF :LF コード(0x0A)

※CR,CF までが一つのデータ・ブロックになります

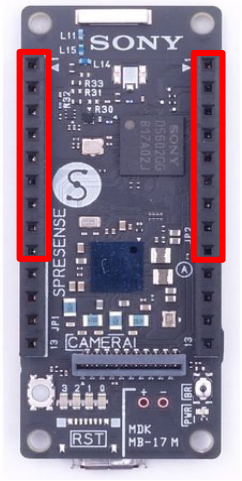
[Refs]
https://shop.sunhayato.co.jp/products/sample-program-mm-s50mv

気になる点7： ELTRES(クリエイティブジャパン)ボードと同時に使用できない

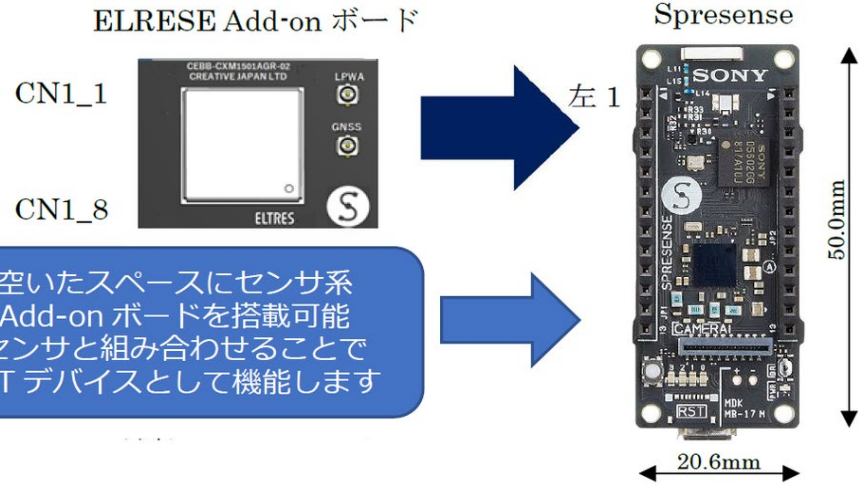
距離データをCLIPサーバーに表示(ToFとELTRES(クリエイティブジャパン)ボードが物理接続衝突のため、確認不可)



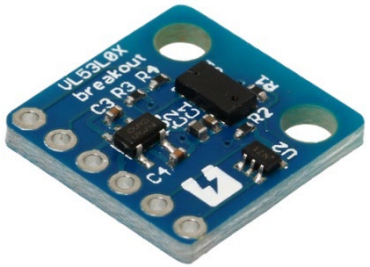
MM-S50MV	CXD-5602
GND	GND
UART2_TX	UART2_TX
UART2_RX	UART2_RX
UART2_RTS	UART2_RTS
UART2_CTS	UART2_CTS
NC	I2S0_BCK
NC	I2S0_LRCK
SPI5_CS_X	SPI5_CS_X
SPI5_BCK	SPI5_SCK



CXD-5602	MM-S50MV
XRST_PIN_1.8V	XRST
1.8V	1.8V
EXT_VDD	3.7V(4V)
EMMC_DATA3	NC
EMMC_DATA2	NC
I2S0_DATA_IN	NC
I2S0_DATA_OUT	NC
SPI5_MISO	SPI5_MISO
SPI5_MOSI	SPI5_MOSI



※下記のVL53L0X搭載Time-of-Flight距離センサとELTRES(クリエイティブジャパン)基板は同時に使用可能



[Refs]
https://developer.sony.com/ja/develop/spresense/specifications
https://shop.sunhayato.co.jp/products/mm-s50mv
https://prtimes.jp/main/html/rd/p/000000067.000064534.html
https://www.switch-science.com/products/2894/