



アプリケーションノート

# 多点圧力センサ及び駆動システム



株式会社リコー

産業技術総合研究所

2023.3

# 目次

1. 概要.....	1
2. ハード仕様.....	1
2.1 システム構成 .....	1
2.2 センサ仕様.....	3
2.3 駆動回路仕様.....	3
3. データ通信仕様.....	4
3.1 I/F.....	4
3.2 通信制御.....	5
3.2.1 基本制御フロー .....	5
3.2.2 各処理のフロー .....	6
3.3 通信データ構造 .....	9
3.3.1 データパケット構造.....	9
3.3.2 データペイロード形式.....	10
3.3.3 制御例 .....	11
3.4 サンプルマクロ (Tera Term) .....	12
3.4.1 ターミナル設定.....	12
3.4.2 動作コマンド .....	12
3.4.3 データ表示.....	13
4. アプリケーションソフト .....	14
4.1 機能 .....	14
4.2 画面表示例.....	14

## 1. 概要

形状自由度の高いフレキシブルな多点圧力センサは、ロボットハンド等に装着してロボットに触覚を付与する、あるいはグローブ等に装着してヒトの作業・動作をデジタル化することを可能とするデバイスです。

本アプリケーションノートでは、高い空間分解能を有する多点圧力センサ及び駆動システムの構成とデータハンドリングについて解説します。

## 2. ハード仕様

### 2.1 システム構成

本システムは、圧力センサシートとフレキシブル配線からなる多点圧力センサ、ドライバーIC を搭載した COF (Chip On Film) 及び駆動ボードを主要パーツとしており、適宜、中継基板、変換基板及び中継ケーブルを用いて相互接続を行っています。以下に構成例を示します。

図1は8X8=64点の測定点を有する圧力センサを用いた場合の構成で、COFにはドライバーIC1個(写真裏面)が搭載されています。駆動方式は、X(走査線)-Y(データ線)マトリクス線の線順次駆動です。ロボットハンドへの装着をはじめ、広範な活用が可能な構成です。

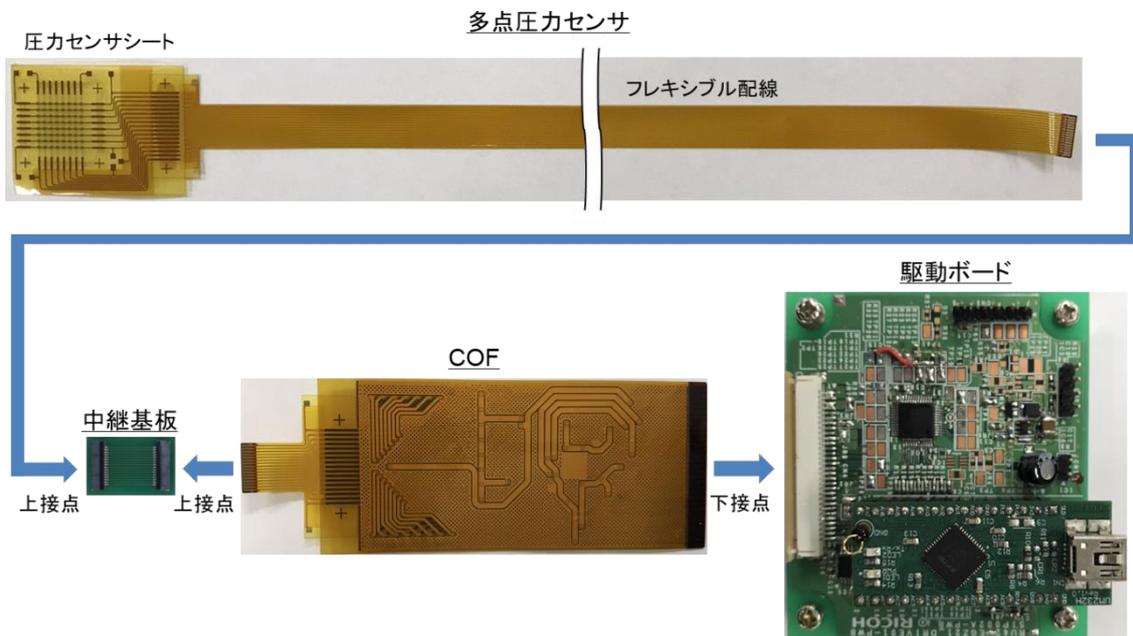


図1 64点センサを用いた構成

図 2 は  $4 \times 4 = 16$  点の測定点を有する圧力センサ 4 セット(合計 64 点の測定点)を用いた場合の構成(センサは 1 セットのみを図示)で、 $4 \times 4$  マトリクス 4 セットを変換基板の配線設計によって、 $8 \times 8$  マトリクスに変換しています。これにより、図 1 と同じ駆動回路を使用することができます。グローブに装着して指先の触圧や把持圧等を検知するのに適した構成となっています。この多点圧力センサをグローブに簡易装着した状態を図 3 に示します。フレキシブル配線の代わりに伸縮配線(オプション・非図示)を用いることにより、さらに良好な装着感を得ることができます。

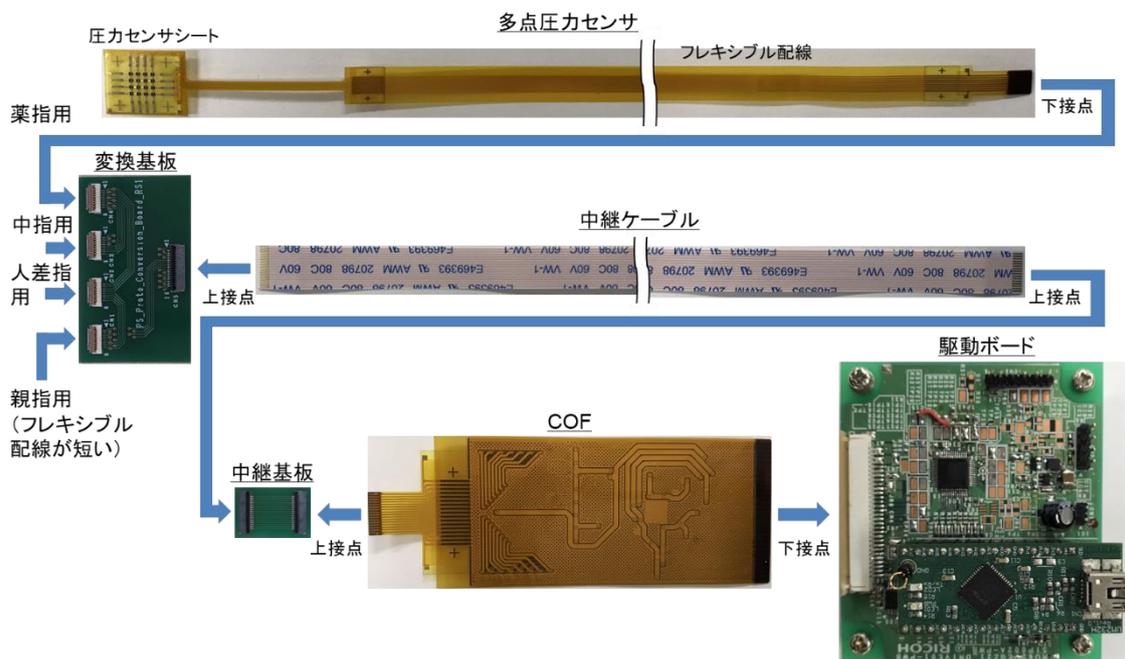


図 2 16 点センサ X4 セットを用いた構成



図 3 多点圧力センサのグローブへの簡易装着状態

## 2.2 センサ仕様

圧力センサシートは、フィルム基板にマトリクス状に配置された上下電極間に、印加圧力によって抵抗が変化する感圧層を形成した構造で、上下電極の交点が測定点になります。圧力センサシートには中継基板または変換基板に接続可能なフレキシブル配線が接合されています。センサの仕様を表 1 に示します。これ以外も設計変更により対応可能です。

表 1 センサの仕様

項目	規定値
センサシート厚さ	約0.1mm
測定圧力範囲	3kPa~300kPa
空間分解能	1.2mm
測定点数 (Total)	64点 (最大1024点)
経時安定性 (常温常湿)	出力変化±5%以下 (4ヶ月)

## 2.3 駆動回路仕様

駆動回路は、ドライバーIC を搭載した COF と駆動ボードからなり、線順次駆動により各点の感圧抵抗変化を電流値として読み取った後に A/D 変換を行い、感圧値 (bit 値) として出力します。駆動ボードの基本仕様を表 2 に示します。駆動センサ素子数は上記センサの測定点数と同義です。図 1 及び図 2 に示した駆動ボードは外部 I/F として USB モジュール (UM232H: FTDI 社) を実装したのですが、その他の I/F にも対応可能です。

表 2 駆動ボードの基本仕様

項目	規定値/機能	備考
駆動ボードサイズ	60mmX70mm	
電源電圧	DC5V	USB給電または外部入力
駆動センサ素子数	8X8=64 (最大32X32=1024)	
フレームレート	100Hz	USB-I/F、64素子駆動時
消費電力	300mW以下	
外部I/F	USB	USB2.0 (12Mbps)
	BLE	BLE4.1
	UART	有線

### 3. データ通信仕様

#### 3.1 I/F

駆動ボード上の CPU-I/F に用意されるシリアル通信機能(UART)を USB モジュールにより変換します。モードとして以下の 2 種類が用意されます。

1. 仮想 COM ポートモード / COM ポート(UART)として認識
2. 専用ドライバーモード / USB デバイス(1 エンドポイント)として認識

USB モジュールの設定により上記を排他的に使用することが可能です。

各モードで使用するドライバーを表 3 に示します。詳細資料はメーカーサイトより入手ください。

[メーカー資料の一例]

FTDT FTD2xx.DLL

FTD2XX Drivers Windows 2.12.28(x64) 2017-08-30

D2xx\_Programmer's\_Guide(FT\_000071)

表 3 各モードで使用するドライバー

ホスト側	ドライバー	I/Fモード
仮想COMポート	VCP Drivers	FT245 Asynchronous FIFOモード
専用ドライバー	D2XX Drivers	FT245 Asynchronous FIFOモード

仮想 COM ポート使用時の設定を表 4 に示します。ボーレートの変更(USB モジュールの設定変更含む)は可能です。

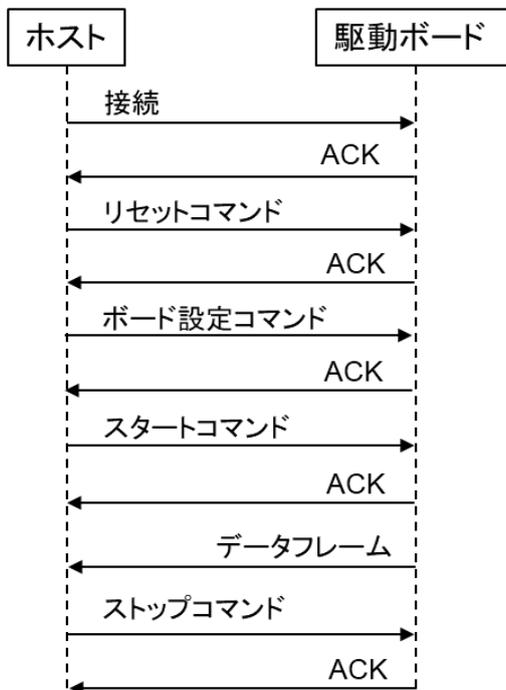
表 4 仮想 COM ポート時の設定

ボーレート	データビット	パリティ	ストップビット	フロー制御	電圧	備考
115200	8	なし	1	なし	3.3V	テスト用

## 3.2 通信制御

### 3.2.1 基本制御フロー

通信全体のシーケンスを図 4 に示します。ホストデバイスと駆動ボードが通信を開始する段階（通信開始フロー）とデータを受信する段階（データ受信フロー）に分けられます。



注) ストップコマンドに対する ACK 送信のタイミングはセンサデータ送信終了後

図 4 通信全体のシーケンス

### 3.2.2 各処理のフロー

リセット処理、ボード設定処理、処理スタート、データ受信処理、表示処理、ストップ処理及びエラー処理のフローを図 5～図 11 に示します。

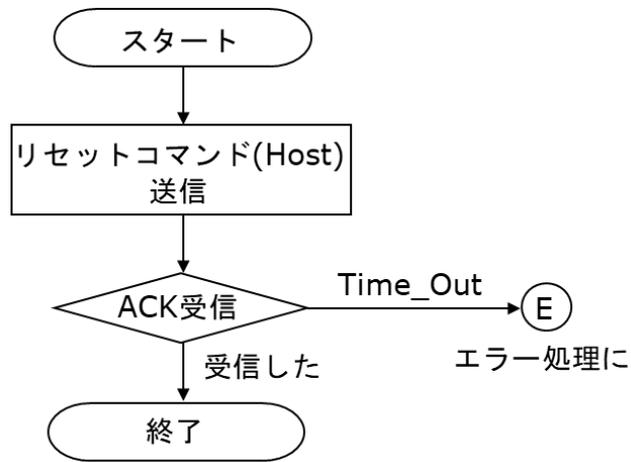


図 5 リセット処理フロー

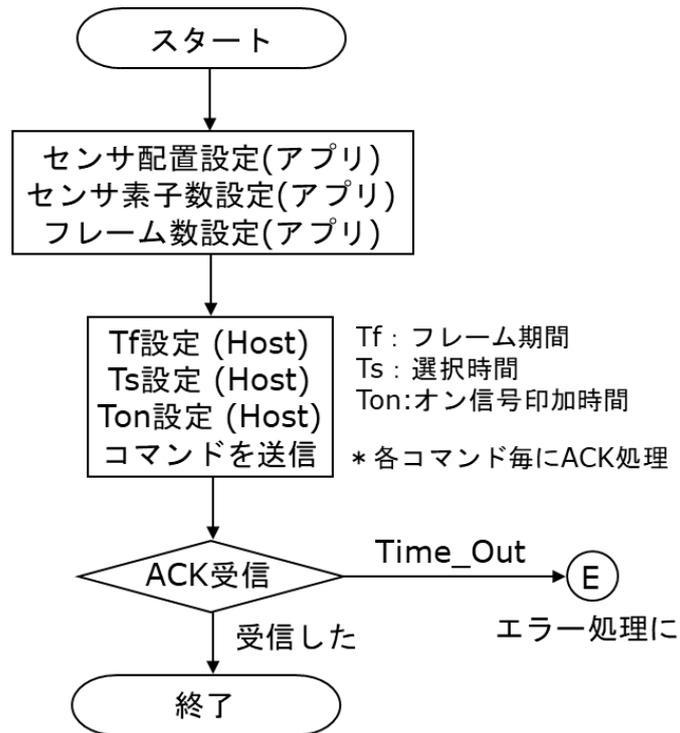


図 6 ボード設定処理フロー

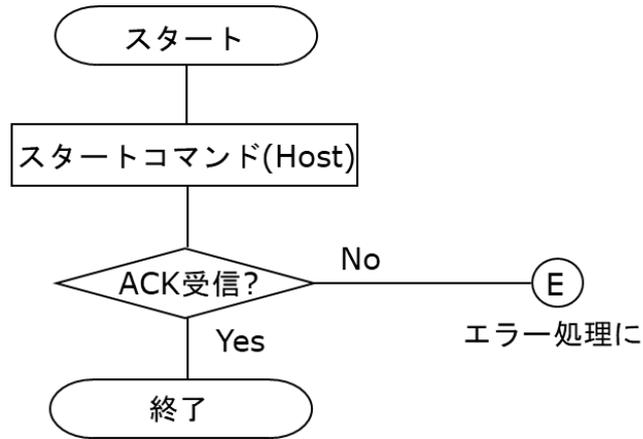


図7 処理スタートフロー

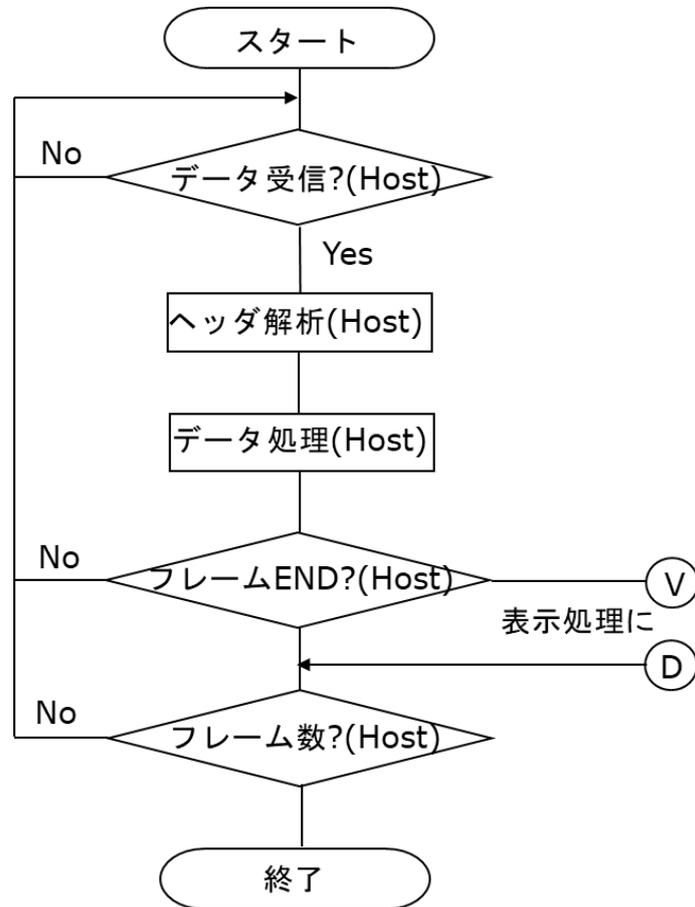


図8 データ受信処理フロー

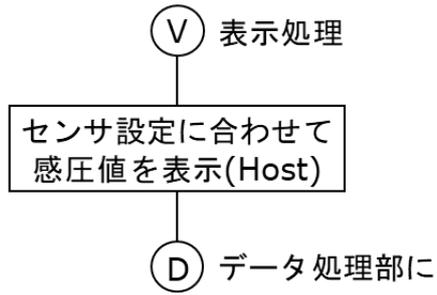


図9 表示処理フロー

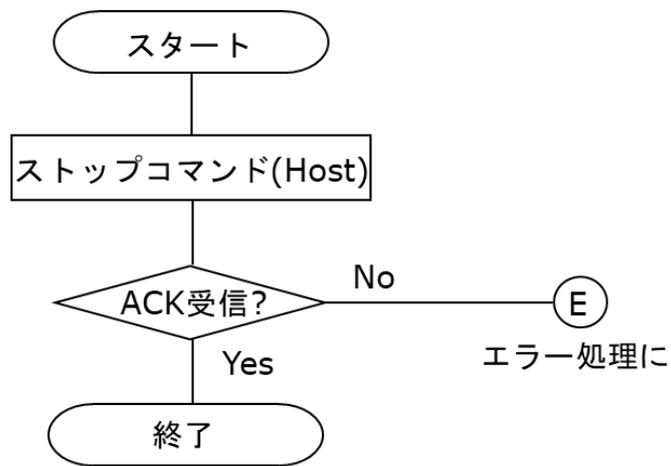


図10 ストップ処理フロー

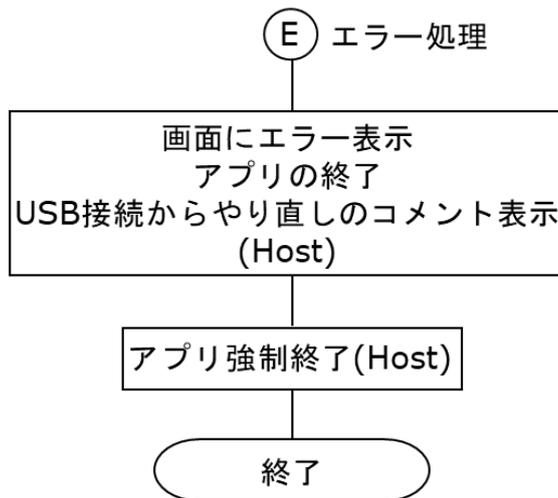


図11 エラー処理フロー

### 3.3 通信データ構造

#### 3.3.1 データパケット構造

パケットを構成する項目と byte 数を表 5 に示します。データがない場合にはデータ長、データペイロードはなくなり、ヘッダ+コマンド+パケット END で構成されます。

表 5 パケット各項目の byte 数

項目	byte数
ヘッダ	2
コマンド	1
データ長	1
データペイロード	データ長で指定されたbyte数
パケットEND(フッタ)	1

各項目の値を表 6 に示します。なお、パケットを連続でホストから送信する場合(駆動ボードから ACK 処理を行わない場合)のパケット(コマンド)間隔は 30ms 以上としています。

表 6 各項目の値

項目	値	備考	
ヘッダ	0xFF_0x00	ヘッダは2byte	
コマンド	ACK	0x06	
	エラー	0x07	(暫定)
	リセット	0x10	
	Tf	0x11	データペイロード(形式2): 3.3.2参照
	Ts	0x12	データペイロード(形式2): 3.3.2参照
	Ton	0x13	データペイロード(形式2): 3.3.2参照
	Block_X	0x14	データペイロード(形式1): 3.3.2参照
	Block_Y	0x15	データペイロード(形式1): 3.3.2参照
	Data	0x16	データペイロード(形式3): 3.3.2参照
	スタート	0xFA	
ストップ	0xFB		
データ長	01~FF	標準は65(41h)byte(センサ番号1byte+センサ素子64byte)	
パケットEND	0xFF		

### 3.3.2 データペイロード形式

データペイロード(形式 1、形式 2 及び形式 3)の構成を表 7～表 9 に示します。

表 7 データペイロード(形式 1)

データペイロード
1byte 数値
データ(1byte)

表 8 データペイロード(形式 2)

データペイロード	
1byte 先頭2Bitが単位指定Bit 00:ms, 01:us, 10:ns, 11:予備	1byte 数値
データ(2byte)	

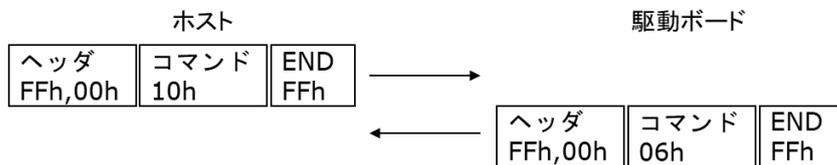
表 9 データペイロード(形式 3)

データペイロード	
1byte センサ番号	64byte 64センサ素子の感圧値
データ(65byte)	

### 3.3.3 制御例

前記のルールに基づく制御例を図 12 に示します。

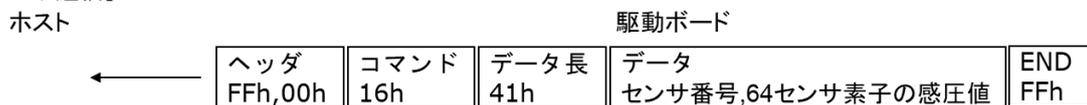
[リセットコマンド]



[Tfコマンド]



[データ送信]



[エラー時(暫定)]

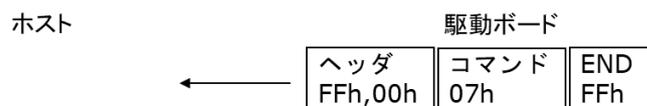


図 12 制御例

### 3.4 サンプルマクロ (Tera Term)

#### 3.4.1 ターミナル設定

通信データはすべてバイナリコードのため、ターミナル設定をバイナリモードに切り替えます。

```
;Tera Term マクロコマンド バイナリモード  
setdebug 2
```

#### 3.4.2 動作コマンド

```
;Reset  
Send $FF $00 $10 $FF  
mpause 1000  
  
;Tf コマンド 250ms  
;Send $FF $00 $11 $01 $05 $FF  
Send $FF $00 $11 $02 $00 $FA $FF  
mpause 1000  
  
;Ts コマンド  
Send $FF $00 $12 $02 $00 $05 $FF  
mpause 1000  
  
;Ton コマンド  
Send $FF $00 $13 $02 $00 $05 $FF  
mpause 1000  
  
;Block_X コマンド  
Send $FF $00 $14 $01 $01 $FF  
mpause 1000  
  
;Block_Y コマンド  
Send $FF $00 $15 $01 $01 $FF  
mpause 1000  
  
;Start  
Send $FF $00 $FA $FF
```

注) Tera Term では ACK コード受信処理を行わないため、コマンド間隔を持たせています。

### 3.4.3 データ表示

スタートコマンド後に受信データが Tera Term のコンソール上に表示されます。CR\_CF 処理などはないため改行は行われません。

[表示例]

```
FF 00 16 41 01 00 00 00 00 00 00 00 00 0E 08 05 05 06 06 09 05 0E 08 06 04 07 12 08 04 ...
```

センサ素子データ(感圧値)を 8X8 マトリクス形式にするために、便宜的に改行を行うと下記のようになります。

```
FF 00 16 41 01          ; ヘッダーコード
00 00 00 00 00 00 00 00  -+
0E 08 05 05 06 06 09 05  |
0E 08 06 04 07 12 08 04  |
0F 08 05 05 06 06 08 04  | センサ素子データ
0F 0A 05 04 06 06 08 04  | 8x8 マトリクス 64byte
0F 08 04 05 05 07 09 04  |
0F 08 05 05 07 06 09 09  |
10 08 05 05 06 06 08 04  -+
FF                      ; END コード
```

## 4. アプリケーションソフト

### 4.1 機能

ホストとしてローカル PC を使用する場合は、視覚効果を高めた表示機能を有する専用アプリケーションソフト(PressureSensorController)の利用が可能です。主な機能を表 10 に示します。

表 10 主な機能

項目	機能
設定	センサ情報設定、通信方式設定、データ種別設定
表示	テキスト表示、2D分布画像表示
データ処理	物理量変換処理、ノイズ除去フィルタ処理、簡易統計処理
付加機能(プラグイン)	物体判定機能、分割表示(グローブへの装着想定)機能

### 4.2 画面表示例

図 13 に画面表示例を示します。図 1 の構成で、センサシート上に球体を載せた時の感圧値(bit 値)分布を表しています。左が 8X8 センサ素子の感圧値(bit 値)テキスト表示、右が 2D 分布画像表示です。ノイズ除去フィルタ処理として、メディアンフィルタ(レベル 1)処理及びガウシアンフィルタ(レベル 1)処理を行っています。また、センサ設置状態に合わせるために 2D 表示分布画像を上下反転させています(表示反転・回転機能使用)。

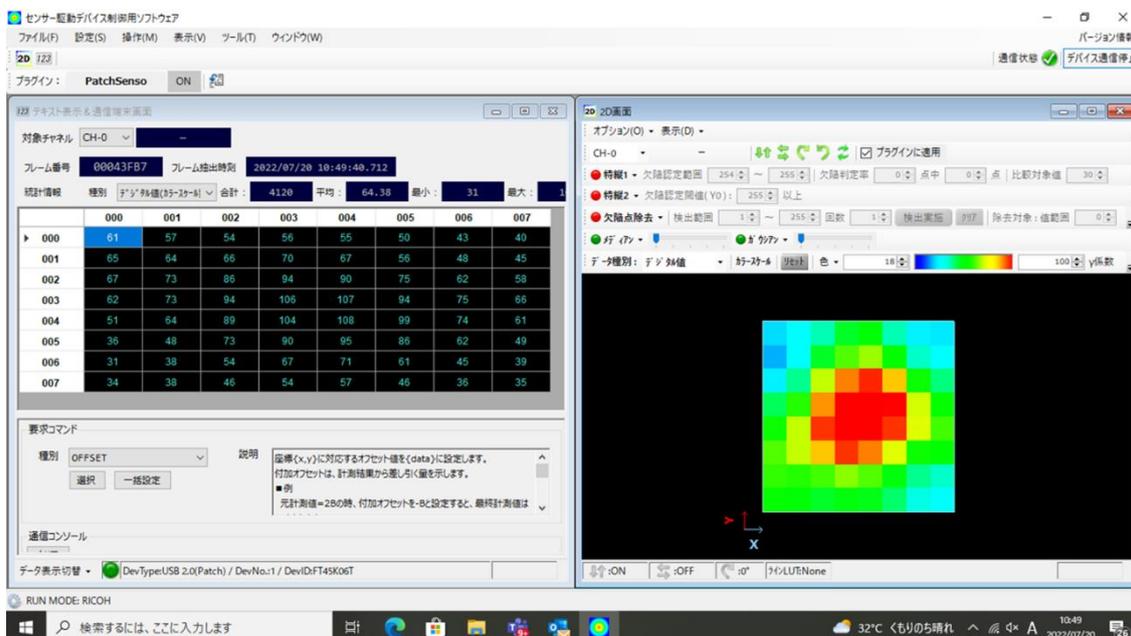


図 13 画面表示例