

パラメータファイルとは

以前、「171 データロガーのファイル構成」で 説明した**収録条件**が パラメータファイルに相当します。今回は 右の171の動画で説明した **収録条件**を用いて パラメータファイルを扱うテストプログラムを パソコン、マイコンの双方で作成します。

まず、パソコン側で、用意する機能を上げると

- ① パラメータのデータ入力、修正機能
- ② パラメータファイルの 書き出し
- ③ パラメータファイルの 読み込み
- ④ パラメータを マイコンに 送信して
　　マイコン側で SDカードに書き出しそる
- ⑤ マイコンの パラメータを 受信する

と、なります。

まずは、パソコン側のプログラム開発を行います。
171の動画にて **収録条件の 構造体型宣言**を作成してますので それを、パラメータの構造として使用します。

収録条件

ID code	Ver No.
収録タイトル	63文字
コメント	80文字
収録開始 年月日	Y-M-D-yo
収録開始 時分秒	H:M:S.ss
収録チャネル数	3
サンプルレイト (Hz)	100
ch. 0	トリガ判定レベル
ON	トリガ判定カウント
ch. 1	トリガ判定レベル
ON	トリガ判定カウント
ch. 2	トリガ判定レベル
ON	トリガ判定カウント
ch. 3	トリガ判定レベル
OFF	トリガ判定カウント
収録時間(秒)	180
収録サンプル数	18000
予備エリア	

パラメータファイル(収録条件)の 構造体宣言

```
typedef struct {
    int id, ver; // IDコード、バージョン番号
    char ttl[64]; // タイトル文字列
    char cmt[81]; // コメント文字列
    char ccn; // チャネル数
    short sps; // サンプルレイト
    DATE_S dts; // 収録開始 日付
    TIME_S tms; // 収録開始 時刻
    TRIGGER tg_0; // ch. 0 トリガ設定
    TRIGGER tg_1; // ch. 1 トリガ設定
    TRIGGER tg_2; // ch. 2 トリガ設定
    TRIGGER tg_3; // ch. 3 トリガ設定
    int rtm; // 収録時間
    int rsn; // 収録サンプル数
    char yobi[52]; // 予備
} HEAD_REC; // パラメータ構造体
```

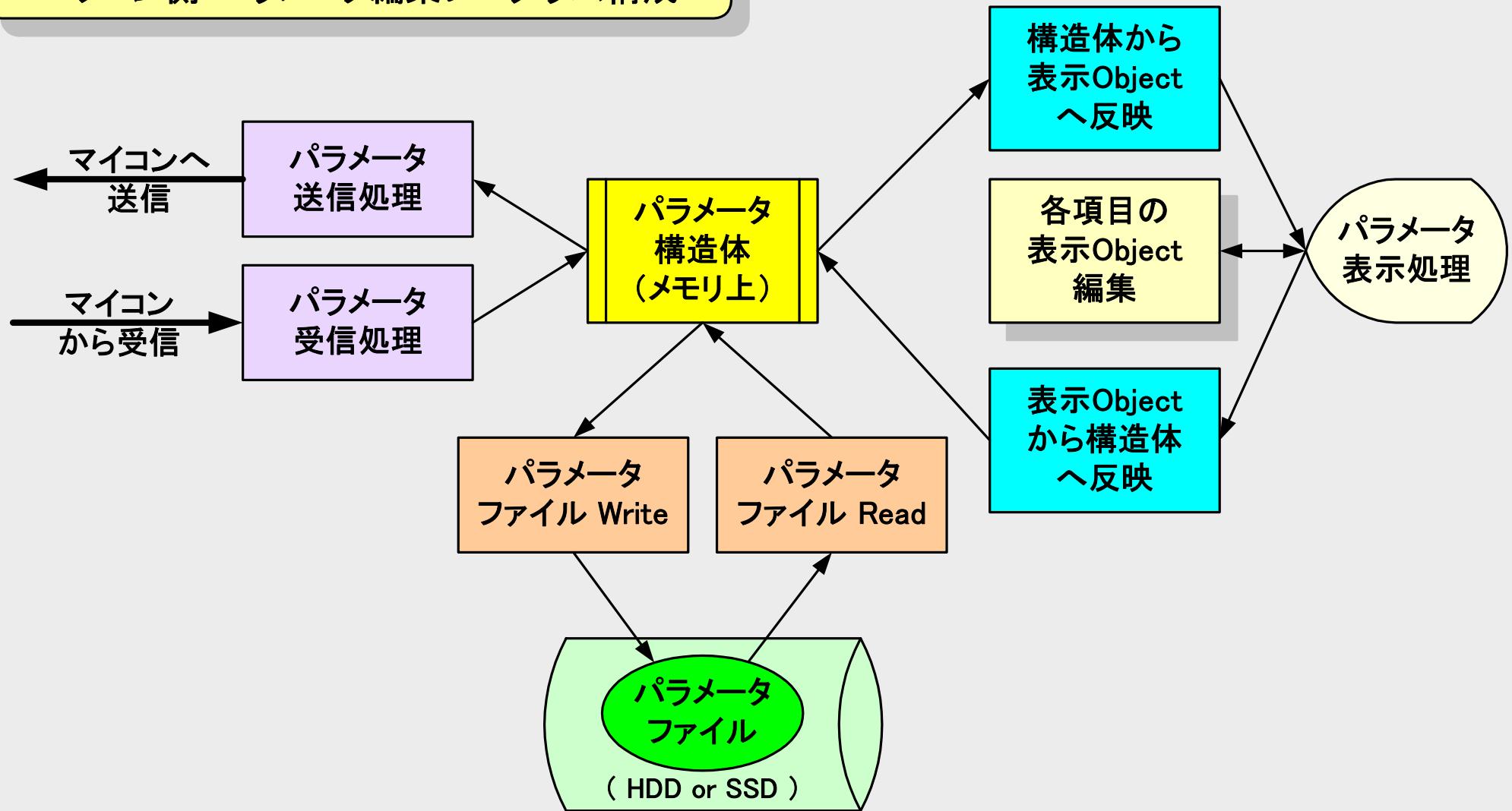
前の物と比べ HEAD_REC構造体の内容を若干変更しました。

```
typedef struct {
    char y; // 西暦 下2桁 年
    char m; // 月
    char d; // 日
    char yo; // 曜日 ( これは任意 )
} DATE_S; // 日付 構造体

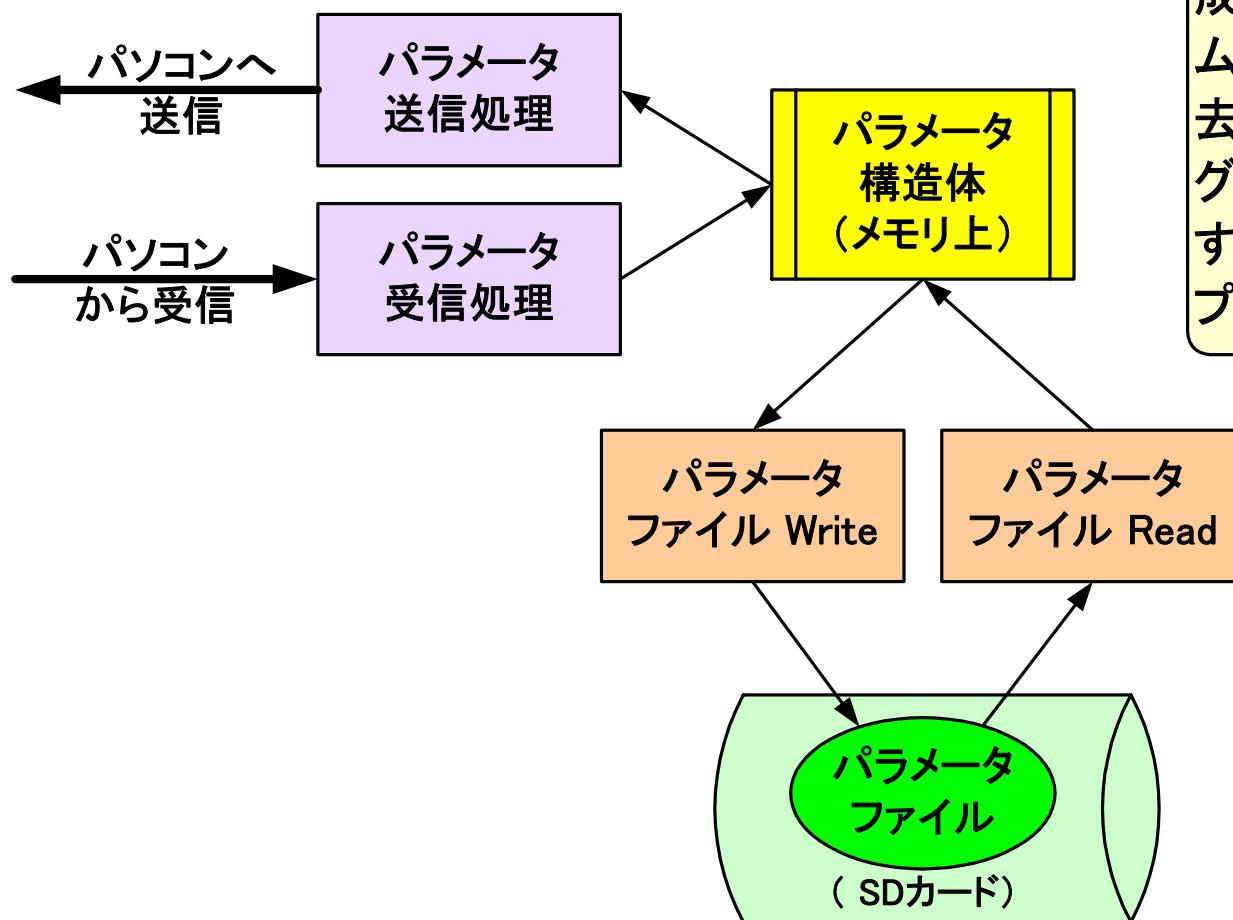
typedef struct {
    char h; // 時
    char m; // 分
    char s; // 秒
    char ss; // 1/100秒 (必要な時のみ使用)
} TIME_S; // 時刻 構造体

typedef struct {
    char sw; // 使用有無 選択
    char jg; // チャネルの トリガ判定フラグ
    char cnt; // トリガ判定カウント
    char yb1; // 予備.1
    short lvl; // トリガ判定レベル(2byte整数)
    short yb2; // 予備.2
} TRIGGER; // 各チャネルの トリガ構造体
```

パソコン側 パラメータ編集プログラム構成



マイコン側 パラメータ操作プログラム構成



マイコン側 パラメータ操作プログラム構成は、パソコン側 パラメータ編集プログラム構成から、GUIの画面編集処理を取り去った構成になっています。全体のプログラムの構成は、このような構成になります。後は、このブロック図を意識してプログラム開発を行います。

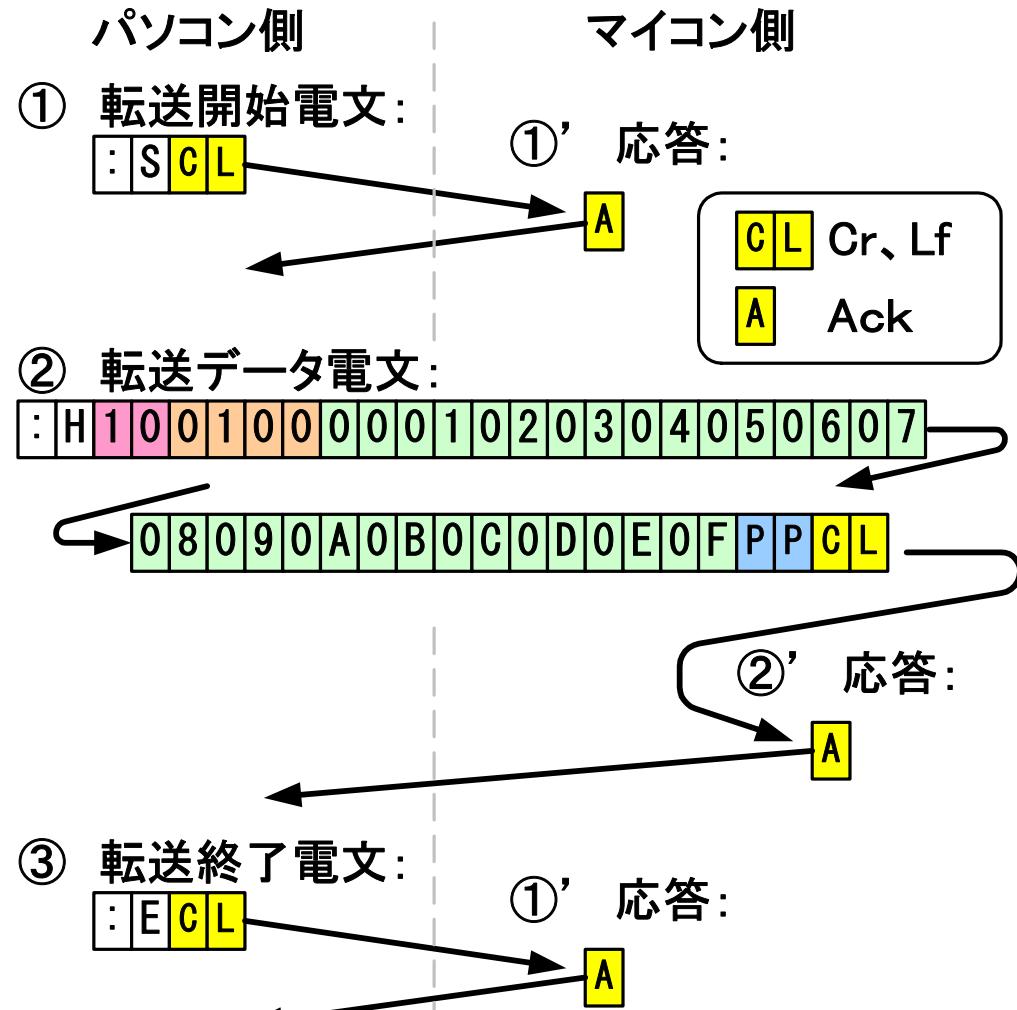
パソコン、マイコン間 データ通信

パソコン単体で行うパラメータの表示、編集そして、読み出し、書き込みのプログラムは出来ました。

後は、マイコンとのパラメータデータのやり取りです。これを行うには、パソコン、マイコン間で行う伝送制御手順を決めておく必要があります。今回は大雑把に電文のフォーマットとやり取りするシーケンス(手順)を決めておく事にします。シリアル通信を行う場合、デバッグ時確認しやすい事もあり、1行の文字列データ(最後にCrLfコードを付ける)をやり取りする場合が多いです。

パラメータを転送する時は、内部にバイナリコードも含まれているので、転送するデータ量は倍に増えますがHexフォーマット的な文字列で転送する事にします。

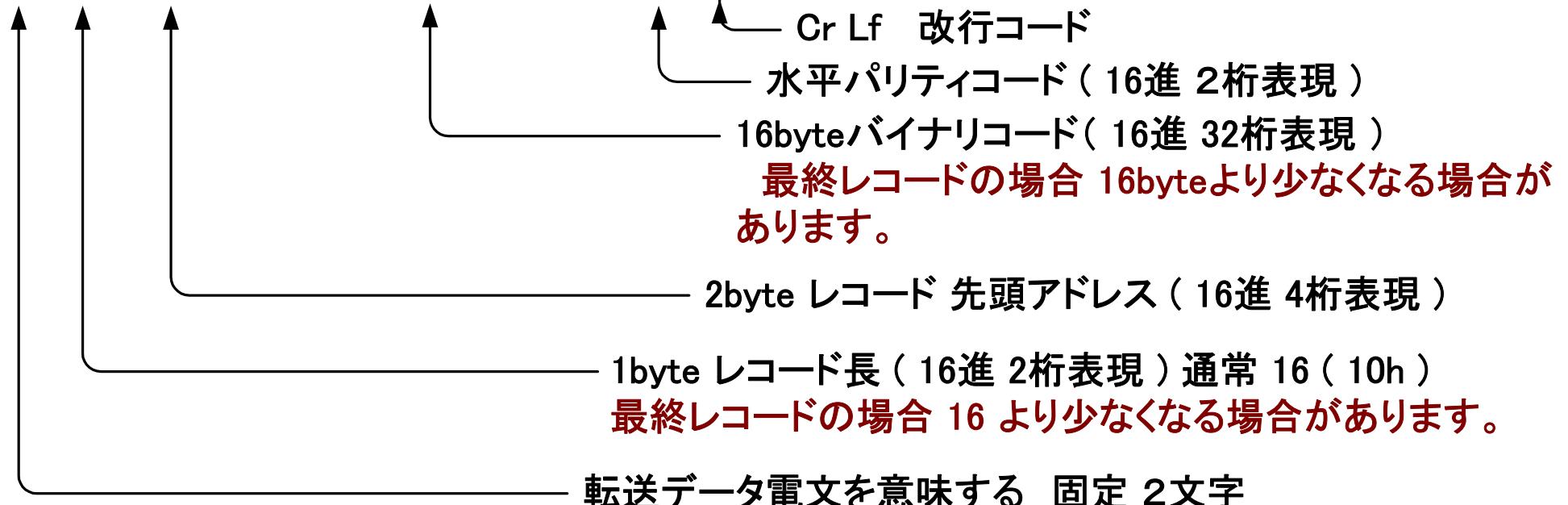
伝送手順



転送データ電文の内容

転送データ電文:

: H 1 0 0 1 0 0 0 0 0 1 0 2 0 3) 0 E 0 F P P C L



パラメータ編集 | パラメータ転送

ID (Hex8桁)

01234567

Ver (Hex8桁)

1234ABCD

タイトル

あいうえお

コメント

Comment string.

タイトルは Max 63文字、コメントは Max 80文字です。長すぎる文字列は Max値で切り捨てます。

サンプルレイト Hz

- 1
- 20
- 2
- 50
- 5
- 100
- 10

収録時間

[Sec]

ch.0 トリガ

 On/Off

600

レベル カウント

ch.1 トリガ

 On/Off

600

レベル カウント

ch.2 トリガ

 On/Off

600

レベル カウント

ch.3 トリガ

 On/Off

変更の反映 格納

