

開発情報(開発者支援キット[DK]付属 CD-ROM)の概要

2024 年 5 月 10 日 第 3.1.1 版

株式会社アートファイネックス

1. はじめに

本書は、弊社 UHF 920MHz 帯リーダ・ライタ(以下、リーダ・ライタとします)を使用し、ソフトウェア開発を行なうための開発情報について説明するものです。

- ※ Windows、Microsoft Visual Studio、Microsoft Visual C++ .NET、Microsoft Visual Basic .NET、Microsoft Visual Basic 6.0、Microsoft Visual C# .NET は、米国 Microsoft Corp.の登録商標です。
- ※ その他、商品名及び製品名などは一般に各社の商標または登録商標です。

2. リーダ・ライタ種別

リーダ・ライタは、以下の 2 種を用意しております。

表 2-1

名称	使用プロトコル	対象機種
ART Finex UHF リーダ・ライタ CB ファミリ	ART Finex リーダ・ライタ プロトコル UHF 版	[CB ファミリ SP モデル] UP-200-J2、UP4-200-J2、UP8-200-J2 他 [CB ファミリ SU モデル] UP-1000-J2、UP2-1000、UP4-1000-J2、 UP8-1000-J2、UP-250-J2、UP2-250、UP4- 250-J2、UP8-250-J2 他 [CB ファミリ AR モデル] UXA250-4、UXA250-8
ART Finex UHF リーダ・ライタ f ファミリ	URW-SP プロトコル	[f ファミリ SP モデル] PRM92J30CE-S、UP-200f-J2、UT-200f-J2 UPen-200f-J2 他 [f ファミリ SensorTag モデル] PRM92J30CE-S(2)、UCS-200f-J2、 UC-200f-J2

開発情報は、インターフェース拡張オプションの有無によって区別できます。

- [CB ファミリ]： インターフェース拡張オプションが付属されている製品群
- [f ファミリ]： インターフェース拡張オプションがない製品群

同一システムで、様々なインターフェースをご利用になる場合や、将来的なインターフェース拡張を予定している場合には、[CB ファミリ]をお勧め致します。組み込み利用など、特定のインターフェースでご利用になる場合には、[f ファミリ]をご利用ください。

また、インターフェース拡張オプション[CB ファミリ]をご利用になることにより、以下の機能が拡張されます。

- 4つのインターフェース(RS232C、USB、LAN、無線 LAN)
 - ※ 製品により、標準搭載インターフェースは異なります。
- リーダ・ライタが自律的に RFID タグ認識変化状態を検知し、上位へ通知する「タグ移動時通知」機能
- 拡張基板の汎用 IO ポート(12 本)
 - ※ 製品により、利用可能なポート数は異なります。
- UHF 帯アンテナ切替器
 - ※ 拡張基板の汎用 IO ポートを利用します。

3. 開発手法

開発情報 CD-ROM には、以下のものが同梱されています。

- プロトコル仕様書
- Windows 向け開発用ライブラリ(API)、及び仕様書
- サンプルソース
- Windows 用サンプルアプリケーション

リーダ・ライタを使用したソフトウェア開発の手法として、リーダ・ライタプロトコルを使用した開発手法と、Windows 向け開発用ライブラリ(API)を使用した開発手法の二つがあります。以下に、それぞれの場合において必要なドキュメント、及びツールの説明を記述します。

3.1. リーダ・ライタプロトコルを使用した開発手法

使用するドキュメント、及びツール

- プロトコル仕様書
- サンプルアプリケーション(IC タグライタ)

プロトコル仕様書は、ご利用になるリーダ・ライタによって、異なる種類(URW-SP プロトコル、ART Finex リーダ・ライタプロトコル UHF 版)の仕様書がございます。仕様書には、リーダ・ライタを制御するためのコマンド定義が記載されています。

※ URW-SP プロトコル、ART Finex リーダ・ライタプロトコル UHF 版には、互換性がございません。

※ リーダ・ライタに対応するプロトコルは、表 1-1 にてご確認ください。

それぞれのコマンドは、サンプルアプリケーション(IC タグライタ)にて確認することができます。メイン画面のメニュー [Help]-[Cmd Test] から、定義されたコマンドをリーダ・ライタに送信し、受信したコマンドが確認できます。

サンプルアプリケーションは Windows 10、Windows 11 に対応しています。それ以外の環境で開発される場合は、ターミナルソフト等を使用してコマンドの確認を行なってください。

3.2. 開発用ライブラリ(API)を使用した開発手法

使用するドキュメント、及びツール

- 開発用ライブラリ(API)、及び仕様書
- サンプルソース

開発用ライブラリ(API)は、定義されている各関数をプログラム内で呼び出すことにより、プロトコル仕様書に記載されているコマンド定義を意識することなくリーダ・ライタを制御し、RFID タグへのアクセス、リーダ・ライタパラメータの設定・取得等を行うことができます。

開発用ライブラリ(API)を使用することで、これらの機能を実装したアプリケーションを容易に開発することができます。開発用ライブラリ(API)の関数、及びファミリ、モデル毎の対応・非対応につきましては、「別紙：開発用ライブラリ(API)関数一覧」をご参照ください。

開発用ライブラリ(API)は、Visual Studio(Visual C++、Visual C# .NET、Visual Basic .NET)、Visual Basic 6.0 で使用することができ Windows 10、Windows 11 に対応しています。

以下に、開発用ライブラリ(API)を使って、RFID タグの EPC を取得し、表示する簡単な例を示します(C++)。

```
TypeCTagID_T    tagID[5];
unsigned char    Count = 5;

int iCommNo = so_CommSetup(255, 1, 19200);
if(iCommNo > 0) {
    int iResult = su_ReadTagID_Single(iCommNo, tagID, &Count, 0);
    if(iResult == OK) {
        for(int i = 0; i < Count; i++) {
            for(int j = 0; j < tagID[i].EPC_Length; j++) {
                printf("%02X", tagID[i].EPC[j]);
            }
            printf("\n");
        }
    }
    so_CommDelete(iCommNo);
}
```

また、同梱されているサンプルソースは、開発用ライブラリ(API)を使った、RFID タグの EPC の取得、RFID タグのメモリの読み取り、書き込みを行うアプリケーションのソースコード群です。これらのソースコード群を参照することで、実装方法を確認することができます。

サンプルソースは、Visual Studio 2015(Visual C++、Visual C#、Visual Basic .NET)、Visual Basic 6.0 で作成されたものをそれぞれ同梱しています。

4. USB ドライバ

USB ドライバは、FTDI 社の HP から最新のものをダウンロードしてご利用いただけます。

Windows 10、Windows 11 に対応しております。また、Linux、Mac OS 用の USB ドライバもございます。

5. 開発情報 CD-ROM 同梱アプリケーション

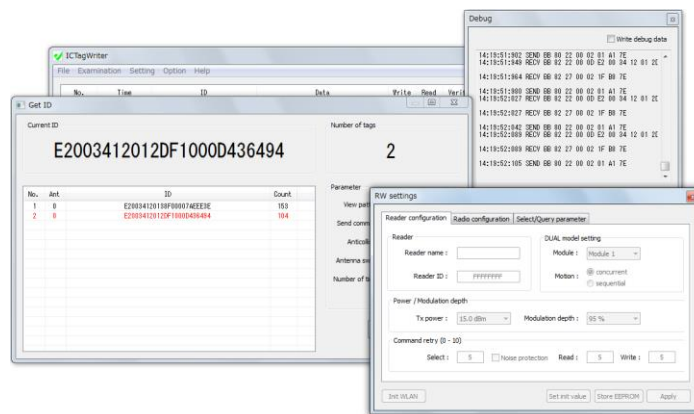
開発情報 CD-ROM には、下の Windows 用アプリケーションが同梱されています。

- IC タグライター

5.1. IC タグライター

IC タグライターは、RFID タグのエンコード、リーダ・ライター設定、タグとの通信確認、メモリアクセス等、リーダ・ライタのほぼ全ての機能が確認できるアプリケーションです。また、タグ情報をシステムなどへ取りこむ方法として、指定時間間隔でタグの認識状況の変化情報を記載したLOGファイルを生成する機能や、タグの認識状況の変化時にファイルを生成する機能などがあります。

RFID タグのエンコード方法については、「UHF 帯タグエンコードガイド」をご参照ください。



6. Android 向け開発情報

USB ホスト機能を有する Android 端末にて、USB インターフェースを持つリーダ・ライタを制御することができます。Android 向けの場合、FTDI 社の「JAVA D2XX for Android API」を使用し、リーダ・ライタプロトコルを送信関数に渡すことでリーダ・ライタにコマンドを送ることができます。Android 向けの開発情報の詳細につきましては、「Android 搭載ホスト向け開発情報について」をご参照ください。

7. ラズパイ向け開発情報

ラズパイでリーダ・ライタを制御することができます。標準 OS (Raspberry Pi OS) には、FTDI 社の USB ドライバが標準で入っているようです。リーダ・ライタプロトコルに従ってリーダ・ライタにコマンドを送ることができます。ラズパイ向けの制御方法の詳細につきましては、「Raspberry Pi で RFID リーダ・ライタを制御する例」をご参照ください。

以上

■開発環境

Microsoft Visual Studio (Visual C++, Visual Basic .NET, Visual C# .NET)

■対応OS

Microsoft Windows 10, Windows 11

■関数一覧

No.	関数名	説明	対応			
			fファミリ	CBファミリ	CBファミリ	CBファミリ
			SPモデル	SPモデル	SUモデル	ARモデル
ホストPCとの接続・切断						
1	so_CommSetup	接続	○	○	○	○
2	so_CommSetupIP	接続(IP)	○	○	○	○
3	so_CommDelete	切断	○	○	○	○
4	so_ResetReaderBeforeConnect	リセット	○	×	×	×
ユーティリティ						
1	so_SetReceiveTimeout	リーダー・ライタとの受信タイムアウト設定	○	○	○	○
2	so_SetCommandMonitor	リーダー・ライタとホスト間のコマンド表示	○	○	○	○
機器設定の変更・取得						
1	so_SpSetLineSpeed	回線速度設定(fファミリ)	○	×	×	×
2	so_SpGetLineSpeed	回線速度取得(fファミリ)	○	×	×	×
3	so_SoSetLineSpeed	回線速度設定(CBファミリ)	×	○	○	○
4	so_SoGetLineSpeed	回線速度取得(CBファミリ)	×	○	○	○
5	so_GetReaderInformation	機器情報取得	○	○	○	○
6	so_GetSPVersion	ファームウェアバージョン取得	○	○	○	○
7	so_SetReaderID	機器ID設定	○	○	○	○
8	so_GetReaderID	機器ID取得	○	○	○	○
9	so_SetReaderName	機器名称設定	×	○	○	○
10	so_GetReaderName	機器名称取得	×	○	○	○
11	so_GetReaderType	機器種別(モデル)取得	×	○	○	○
12	so_SetEEPROM	EEPROMに現在の設定を保存	○	○	○	○
13	so_SetStorageLocation	機器設定変更時の書込先(EEPROM/ROM)の設定	×	○	×	×
14	so_GetStorageLocation	機器設定変更時の書込先(EEPROM/ROM)の取得	×	○	×	×
15	so_SetPowerOnParameter	電源投入時の制御設定(無線LAN設定の初期化)	×	○	○	○
16	so_GetPowerOnParameter	電源投入時の制御取得(無線LAN設定の初期化)	×	○	○	○
機器制御						
1	so_ResetReader	機器のリセット	○	○	○	○
2	so_ResetReaderAnyTime	機器の強制リセット	○	○	○	○
3	so_SpTransmit	任意コマンドの送信(fファミリ)	○	×	×	×
4	so_SoTransmit	任意コマンドの送信(CBファミリ)	×	○	○	○
タグ取得パラメータ						
1	so_SetCurrentRFchannel	周波数チャンネル設定	○	○	○	○
2	so_GetCurrentRFchannel	周波数チャンネル取得	○	○	○	○
3	so_SetTxPAPower	RF出力値設定	○	○	○	○
4	so_GetTxPower	RF出力値取得	○	○	○	○
5	so_SetFrequencyHopping	周波数ホッピング有効/無効設定	○	○	○	○
6	so_GetFrequencyHopping	周波数ホッピング有効/無効取得	○	○	○	○
7	so_SetHoppingSequence	周波数ホッピングの種類を設定	○	○	×	×
8	so_GetHoppingSequence	周波数ホッピングの種類を取得	○	○	×	×
9	so_SetTypeSelectParameters	Select/パラメータ設定	○	○	×	×
10	so_GetTypeSelectParameters	Select/パラメータ取得	○	○	×	×
11	so_SetTypeQueryParameters	Queryパラメータ設定	○	○	×	×
12	so_GetTypeQueryParameters	Queryパラメータ取得	○	○	×	×
13	so_SetCommandRetry	コマンドリトライの設定	○	○	×	×
14	so_GetCommandRetry	コマンドリトライの取得	○	○	×	×
15	so_SetRegion	リージョン(地域)設定	○	○	×	×
16	so_GetCurrentRegion	リージョン(地域)取得	○	○	×	×
17	so_SetSelectBankSetting	タグ特定メモリバンク(EPC or TID)設定	○	○	×	×
18	so_GetSelectBankSetting	タグ特定メモリバンク(EPC or TID)取得	○	○	×	×
19	so_GetRSSI	受信電波強度(RSSI)取得	○	○	×	×
20	so_ScanRSSI	全チャンネルの受信電波強度(RSSI)の取得	○	○	×	×
21	so_SetQuery	QueryパラメータのQ値設定	×	×	○	○
22	so_GetQuery	QueryパラメータのQ値取得	×	×	○	○
23	so_SetDwellTime	送信時間設定	×	×	○	○
24	so_GetDwellTime	送信時間取得	×	×	○	○
25	so_SetRoundCount	カウント数設定	×	×	○	○
26	so_GetRoundCount	カウント数取得	×	×	○	○
27	so_SetRssiFilter	タグのRSSIフィルタリング条件設定	×	×	○	○
28	so_GetRssiFilter	タグのRSSIフィルタリング条件取得	×	×	○	○
29	so_SetModulation	変調方式設定	×	×	○	○
30	so_GetModulation	変調方式取得	×	×	○	○

No.	関数名	説明	対応			
			ファミリ	CBファミリ	CBファミリ	CBファミリ
			SPモデル	SPモデル	SUモデル	ARモデル
タグ操作						
1	su_ReadTagID_Single	タグID取得	○	○	○	○
2	su_ReadTagID_Single_ForManyTag	タグID取得(多数タグ用)	×	○	○	○
3	su_ReadTagID_Multiple	連続タグID取得	○	○	○	○
4	so_StopReadTagID_Multiple	連続タグID取得停止	○	○	○	○
5	su_ReadTagID_Multiple_ForManyTag	連続タグID取得(多数タグ用)	×	○	○	○
6	so_ReadTagID_Multiple_Stock	連続タグID取得(溜めこみ)	×	○	×	×
7	su_ReadTagID_Multiple_Stock_ForManyTag	連続タグID取得(溜めこみ)(多数タグ用)	×	○	○	○
8	so_ReadTagID_Multiple_Nomerge	連続タグID取得(マージなし用)	×	×	○	○
9	so_GetTagRssi	タグID+RSSI取得	○	○	○	○
10	so_GetTagRssi_ForManyTag	タグID+RSSI取得(多数タグ用)	×	○	○	○
11	so_ReadTagID_Memory	EPCと指定したメモリを同時に取得	×	×	×	○
12	so_ReadTagMemory	タグメモリ読み取り	○	○	○	○
13	su_WriteTagData	タグメモリ書き込み	○	○	○	○
14	so_BlockWriteTagData	タグメモリ書き込み(BlockWriteコマンド)	○	○	×	×
15	so_WriteTagData_CustomVerify	タグメモリ書き込み(ベリファイ動作変更)	○	×	×	×
16	so_KillTag	タグ使用不可	○	○	○	○
17	so_LockTag	タグロック	○	○	○	○
18	so_BlockPermalock	タグ恒久ロック(BlockPermaLockコマンド)	○	×	×	×
IOポート操作						
1	so_SpSetIOPortState	リーダ・ライタモジュールIOポート状態設定	○	×	×	×
2	so_SpGetIOPortState	リーダ・ライタモジュールIOポート状態取得	○	×	×	×
3	so_SpSetIOPortValue	リーダ・ライタモジュールIOポート設定	○	×	×	×
4	so_SpGetIOPortValue	リーダ・ライタモジュールIOポート取得	○	×	×	×
5	so_SoSetIOPortDirection	IOポート入出力設定	×	○	○	○
6	so_SoGetIOPortDirection	IOポート入出力取得	×	○	○	○
7	so_SoOutputIOPort	IOポート出力	×	○	○	○
8	so_SoInputIOPort	IOポート入力	×	○	○	○
9	so_SetDefaultOutputIOPortValue	IOポート出力切替値設定	×	○	○	○
10	so_GetDefaultOutputIOPortValue	IOポート出力切替値取得	×	○	○	○
11	so_SoControlIOPort	ワンショットIOポート制御	×	○	×	×
タグ移動時通知(動作モード)						
1	so_StartOperatingMode	タグ移動時通知(動作モード)開始	×	○	○	○
2	so_SendOperatingMode	タグ移動時通知(動作モード)開始(VB6用)	×	○	○	○
3	so_RecvOperatingMode	タグ移動時通知(動作モード)データ取得(VB6用)	×	○	○	○
4	so_StopOperatingMode	タグ移動時通知(動作モード)停止	×	○	○	○
5	so_GetOperatingModeState	タグ移動時通知(動作モード)状態取得	×	○	○	○
6	so_SetOperatingParameter	タグ移動時通知(動作モード)パラメータ設定	×	○	○	○
7	so_GetOperatingParameter	タグ移動時通知(動作モード)パラメータ取得	×	○	○	○
IO入力変化時通知						
1	so_StartInputIOMonitorMode	IO入力変化時通知開始	×	○	○	○
2	so_StopInputIOMonitorMode	IO入力変化時通知停止	×	○	○	○
3	so_GetInputIOMonitorModeState	IO入力変化時通知状態取得	×	○	○	○
4	so_SetInputIOMonitorParameter	IO入力変化時通知パラメータ設定	×	○	○	○
5	so_GetInputIOMonitorParameter	IO入力変化時通知パラメータ取得	×	○	○	○
アンテナ切替						
1	so_SetAntennaSwitchState	アンテナ切替器有効・無効設定	×	○	○	○
2	so_GetAntennaSwitchState	アンテナ切替器有効・無効取得	×	○	○	○
3	so_SetAntennaSwitchNum	動作アンテナポート設定	×	○	○	○
4	so_GetAntennaSwitchNum	動作アンテナポート取得	×	○	○	○
5	so_SetAntennaPort	動作アンテナポート設定(UP2用)	×	×	○	×
6	so_GetAntennaPort	動作アンテナポート取得(UP2用)	×	×	○	×
7	so_SetTxPowerAnt	各アンテナポートのRF出力値を設定	×	×	○	○
8	so_GetTxPowerAnt	各アンテナポートのRF出力値を取得	×	×	○	○
9	so_SetSessionAnt	各アンテナポートのセッション値を設定	×	×	○	○
10	so_GetSessionAnt	各アンテナポートのセッション値を取得	×	×	○	○
11	so_SetSessionState	各アンテナポートのセッション値の有効/無効の設定	×	×	○	○
12	so_GetSessionState	各アンテナポートのセッション値の有効/無効の取得	×	×	○	○

○: 利用可能 ×: 利用不可

■制御プロトコル一覧

No.	機能	対応			
		fファミリ	CBファミリ	CBファミリ	CBファミリ
		SPモデル	SPモデル	SUモデル	ARモデル
1	タグID取得	○	○	○	○
2	連続タグID取得	○	○	○※5	○※5
3	タグ情報通知(動作モード)	×	○	○	○
4	連続タグID取得停止	○	○	×	×
5	タグメモリ読み取り	○	○	○	○
6	タグメモリ書き込み	○	○	○	○
7	タグ使用不可	○	○	○	○
8	タグロック	○	○	○	○
9	タグ恒久ロック(BlockPermaLock)	○	×	×	×
10	リーダーバージョン取得	○	○	○	○
11	リーダー情報取得	○	○	○	○
12	リージョン取得	○	○	×	×
13	リージョン設定	○	○	×	×
14	出力取得	○	○	○	○
15	出力設定	○	○	○	○
16	RFチャンネル取得	○	○	○	○
17	RFチャンネル設定	○	○	○	○
18	周波数ホッピング取得/設定	○	○	○※6	○
19	ホッピング有無取得/設定	○	○	○	○
20	コマンドリトライ取得/設定	○	○	×	×
21	Select/パラメータ取得	○	○	×	○
22	Select/パラメータ設定	○	○	×	○
23	Query/パラメータ取得	○	○	×	×
24	Query/パラメータ設定	○	○	×	×
25	EEPROM状態変更	○	○	○	○
26	モジュール回線速度取得/設定	○	×	×	×
27	IOポート取得/設定	○	○	○	○
28	IOポート状態取得/設定	○	○	○	○
29	IOポート出力切替値取得/設定	○	○	○	○
30	RSSI取得	○	○	×	×
31	RSSIスキャン	○	○	×	×
32	タグRSSI取得	○	○	○	○
33	タグ特定メモリバンク(EPC or TID)設定	○	○	×	×
34	タグ特定メモリバンク(EPC or TID)取得	○	○	×	×
35	タグメモリ書き込み(BlockWriteコマンド)	○	○	○	○
36	リーダーID取得	○	○	○	○
37	リーダーID設定	○	○	○	○
38	省電力モード	○	×	×	×
39	リセット	○	○	○	○
40	強制リセット	○	○	○	○
41	回線速度設定	×	○	○	○
42	回線速度取得	×	○	○	○
43	機器種別取得	×	○	○	○
44	ワンショットIO制御	×	○	×	×
45	動作モード設定	×	○	○	○
46	動作モード取得	×	○	○	○
47	動作パラメータ設定	×	○	○	○
48	動作パラメータ取得	×	○	○	○
49	IO入力監視パラメータ設定	×	○	○	○
50	IO入力監視パラメータ取得	×	○	○	○
51	IO入力監視モード設定	×	○	○	○
52	IO入力監視モード取得	×	○	○	○
53	IO入力通知	×	○	○	○
54	コマンドパラメータ制御設定	×	○	○	○
55	コマンドパラメータ制御取得	×	○	○	○
56	PowerOn/パラメータ設定	×	○	○	○

No.	機能	対応			
		fファミリ	CBファミリ	CBファミリ	CBファミリ
		SPモデル	SPモデル	SUモデル	ARモデル
57	リーダ名設定	×	○	○	○
58	リーダ名取得	×	○	○	○
59	アンテナ切替	×	○※3	○※3	○
60	アンテナ切替(UP2用)	×	×	○※4	×
61	アンテナ切替値取得	×	○※3	○※3	○
62	アンテナ設定	×	○※3	○※3	○
63	アンテナ取得	×	○※3	○※3	○
64	タグID取得(多数タグ用)	×	○	○	○
65	TagRSSI取得(多数タグ用)	×	○	○	○
66	Q値設定	×	×	○	○
67	Q値取得	×	×	○	○
68	送信時間設定	×	×	○	○
69	送信時間取得	×	×	○	○
70	カウント数設定	×	×	○	○
71	カウント数取得	×	×	○	○
72	Read Sensor Tag	×	×	×	×
73	タグID取得(マージなし用)	×	×	○	○
74	EPC とメモリをリード	×	×	×	○
75	セッション有効/無効設定	×	×	○	○
76	セッション有効/無効取得	×	×	○	○
77	RSSIフィルタリング条件設定	×	×	○	○
78	RSSIフィルタリング条件取得	×	×	○	○
79	RSSIフィルタリング範囲値設定	×	×	○	○
80	RSSIフィルタリング範囲値取得	×	×	○	○
81	アンテナごとの送信出力設定	×	×	○	○
82	アンテナごとの送信出力取得	×	×	○	○
83	アンテナごとのセッション設定	×	×	○	○
84	アンテナごとのセッション取得	×	×	○	○
85	ファームウェアタイプ取得	×	×	○	○

※1

○:利用可能 ×:利用不可

※2

※3 アンテナ切替器利用時のみ対応しています。

※4 UP2-250、UP2-1000 利用時のみ対応しています。

※5 取得回数の範囲は 1~100、通知モードは一括通知のみ対応しています。

※6 UPx-250 のみ対応しています。