

# sakura.io モジュール(LTE)

## SCM-LTE-01 / SCM-LTE-01 Rev.B

### コマンドリファレンス

#### 概要

sakura.io モジュール(LTE)は、さくらインターネット株式会社が提供する「sakura.io」の sakura.io プラットフォームに、LTE 網を通じて接続するための専用通信モジュールです。本モジュールをユーザの装置に組み込んでホスト MCU と接続すると、簡単なコマンドだけで sakura.io プラットフォームとデータの送受を容易に行うことができます。また、デジタル入出力と ADC 入力も備えており、MCU がないユーザの装置でも本モジュールを用いて単純なデータの送受であれば行うことも可能です。

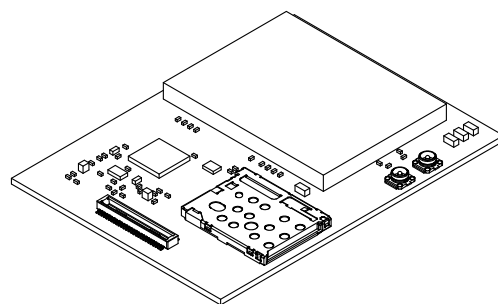


Figure 1

#### 本資料について

- ユーザ機器の MPU が sakura.io モジュールと通信するためのコマンドの仕様を記載しています。
- ユーザ機器の設計の際には、本資料のほか、データシート／アプリケーションノートを十分に確認して設計を行ってください。
- 各コマンド使用の際には実行エラー処理などを適宜行い、異常状態を含め、十分な評価を行ってください。
- 本資料は改良のため予告なく全体、または一部を変更する場合があります。あらかじめご了承ください。

## 目次

概要 .....	1
本資料について .....	1
目次 .....	2
代表的構成例.....	4
ブロック図.....	4
制御インターフェース.....	5
概要.....	5
SPI インターフェース.....	5
I <sup>2</sup> C インターフェース.....	6
UART インターフェース(AT コマンドインターフェース) .....	7
GPIO モード .....	8
省電力制御 .....	10
Active モード/DeepSleep モード選択 .....	10
省電力モード設定 .....	10
時刻同期 .....	10
概要.....	10
データキュー .....	11
概要.....	11
時刻情報の付与 .....	11
キューの大きさ.....	12
送信キューの構造 .....	13
受信キューの構造 .....	15
コマンドシンタックス .....	16

コマンドの構成 .....	16
要求のシンタックス .....	16
応答のシンタックス .....	16
実行結果 .....	17
コマンド定義 .....	18
一般コマンド .....	18
データ送信コマンド .....	23
データ受信コマンド .....	37
ファイル受信コマンド .....	41
管理コマンド .....	49
制約事項 .....	59
ソフトウェアライセンス .....	60
注意事項 .....	61
用途の限定 .....	61
屋外での使用 .....	61
コネクタ .....	61
設置環境 .....	61
改訂履歴 .....	62

## 代表的構成例

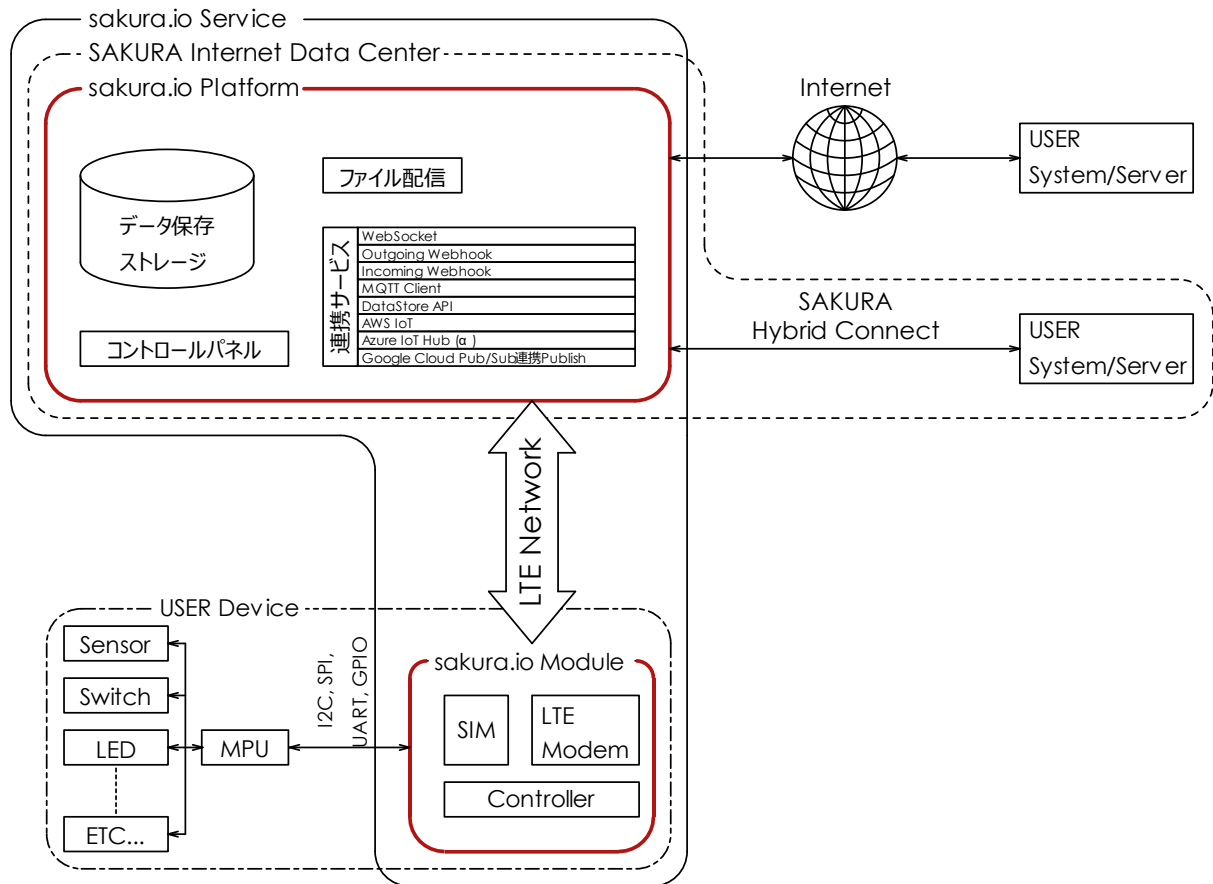


Figure 2

## ブロック図

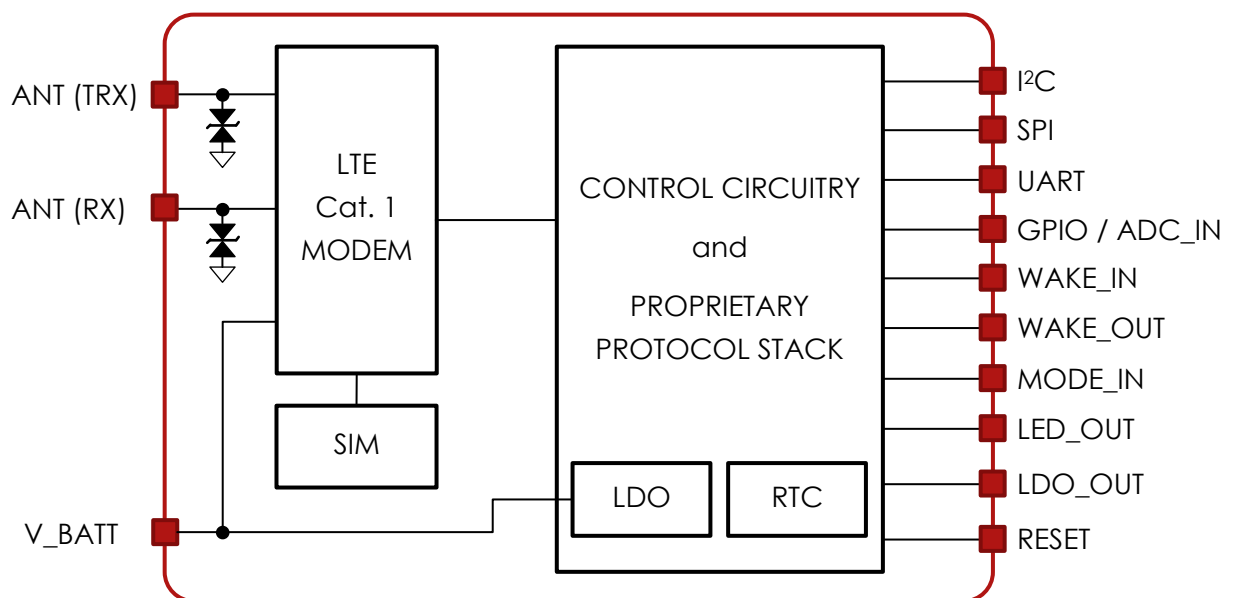


Figure 3

## 制御インターフェース

### 概要

本モジュールの制御は、I<sup>2</sup>C インタフェース、SPI インタフェース、または UART インタフェースに、ユーザがコマンドを与えることによって行うことができます。また、既存システムにおいて使用されているモデムのようなレガシーデバイスを、本モジュールにより置き換えることを念頭に置いた互換レイヤとして、UART インタフェースでは AT コマンドインタフェースを提供します。

本モジュールは、起動時における MODE\_IN ピンのレベルに応じて、スレーブモードまたは GPIO モードのいずれかで動作します。MODE\_IN ピンのレベルを L にして本モジュールを起動させた場合本モジュールは GPIO モードとして動作し、H にして本モジュールを起動させた場合本モジュールはスレーブモードとして動作します。

スレーブモードで動作しているときは、本モジュールは I<sup>2</sup>C スレーブデバイスおよび SPI スレーブデバイスとして振る舞うほか、UART 入出力ピンを通じて AT コマンドインタフェースが利用できます。

スレーブモードで動作しているとき、どのインタフェースを用いても本モジュールは同一のコマンドにより制御することができ、利用できる機能にも違いはありません。ただし、同一のモジュールにおいて、2 つ以上の異なるインタフェースを混在して使用した場合の動作は未定義です。いずれか 1 つのインタフェースのみを使用してください。

GPIO モードで動作している場合は、I<sup>2</sup>C、SPI、UART インタフェースは使用できません。

### SPI インタフェース

SPI インタフェースは、本モジュールがスレーブモードで動作しているときのみ利用可能です。本モジュールは SPI スレーブデバイスとして動作し、動作モードは、CPOL=0、CPHA=0、ビットオーダーは MSB first、コマンドは先頭バイトから順に送付、最大 350kbit/s です。SPI\_CS ピンをアサートすると、ユーザからのコマンド受け付けを開始します。書き込まれたコマンドを解釈すると、応答を返します。応答を完全に読み出すまでは、SPI\_CS をデアサートしないでください。コマンドの書き込み中に SPI\_CS をデアサートすると、当該コマンドの実行は中止されます。応答の読み出し中に SPI\_CS をデアサートすると、応答内容は失われます。

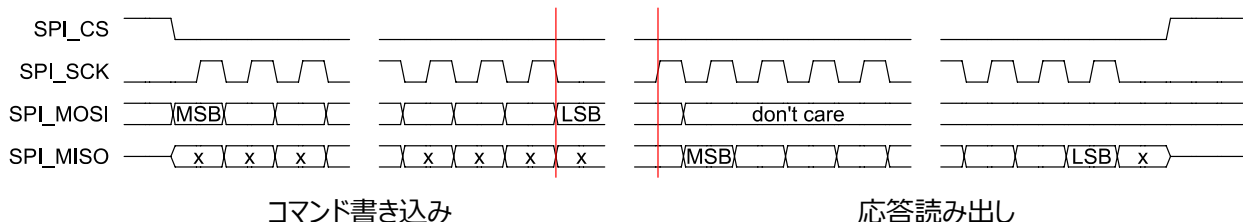


Figure 4

## I<sup>2</sup>C インタフェース

I<sup>2</sup>C インタフェースは、本モジュールがスレーブモードで動作しているときのみ利用可能です。本 I<sup>2</sup>C インタフェースは Standard-mode で動作しており、最大 100kbit/s での通信が可能です。本モジュール内でプルアップ抵抗は内蔵していません。本モジュールの I<sup>2</sup>C デバイスアドレスは 7 ビットで 4Fh です。デバイスアドレスは変更することはできません。

本モジュールへのコマンド発行は、ユーザ機器のホスト MCU からライトサイクルでコマンドの要求シーケンスを送信します。本モジュールは、ユーザからのコマンド受け付けて処理を開始します。受け取ったコマンドを解釈すると、応答シーケンスを返しますので、ユーザ機器のホスト MCU からリードサイクルで読みだしてください。この時応答をすべて読み出すまでは、ストップコンディションを与えないでください。

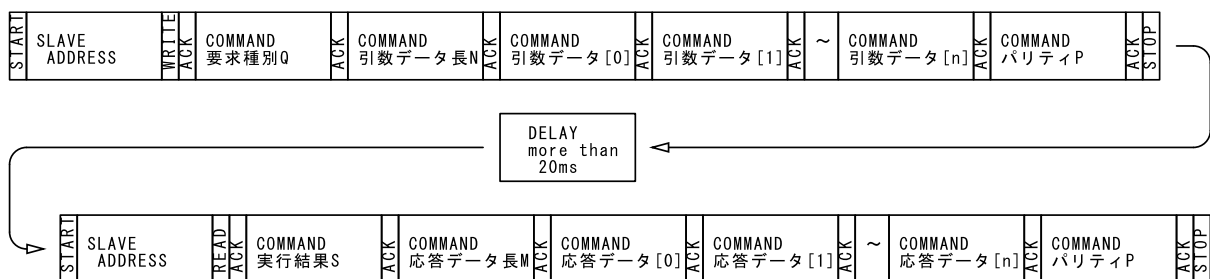


Figure 5

## UART インタフェース(AT コマンドインタフェース)

UART インタフェースでは本モジュールの制御用に AT コマンドインタフェースを提供しています。

AT コマンドインタフェースは、AT コマンドに準じた独自の文字列コマンドにより本モジュールを制御するためのインタフェースです。AT コマンドインタフェースは、本モジュールがスレーブモードで動作しているときのみ利用可能です。このインタフェースは、AT コマンドによりモデムや他社製通信モジュールなどと接続する既存システムとの互換レイヤとして提供するものであり、新規設計での使用は推奨されません。

ユーザからの AT コマンドの書き込みは、本モジュールの UART\_RX に対して行います。コマンドに対する応答は、本モジュールの UART\_TX から出力されます。UART\_RX, UART\_TX ともに、データ 8 ビット、パリティなし、ストップビット 1 ビットの調歩同期信号であり、ボーレートは 115200 ボーで固定です。これらの設定は、変更することができません。

AT コマンドは、「AT\*CMD=」の文字列に続き、コマンドシンタックスの章で定義する「要求」を 16 進の文字列として表したもので、およびこれに続くデリミタから構成されます。デリミタは CR (0Dh) または LF (0Ah) であり、いずれかを受信した時点で書き込まれた AT コマンドを解釈します。AT コマンドに対する応答は、「\*CMD:」の文字列に続き、コマンドシンタックスの章で定義する「応答」を 16 進数文字列として表したもので、および CR LF を返します。続けて、成功の場合には「OK」、AT コマンドになんらかのエラーがあった場合には「ERROR」の文字列に続き、CR LF を返します。「OK」「ERROR」はあくまでも AT コマンドとしての OK/エラーで、コマンドの実行結果としてのものではありません。

### 要求例

AT\*CMD=030003 [LF]

A	T	*	C	M	D	=	COMMAND 要求種別 Dh	COMMAND 要求種別 Dl	COMMAND 引数長 Nh	COMMAND 引数長 Nl	COMMAND 引数 [0] D0h	COMMAND 引数 [0] D0l	COMMAND 引数 [1] D1h	COMMAND 引数 [1] D1l	...	COMMAND 引数 [n] Dnh	COMMAND 引数 [n] Dnl	COMMAND パリティ Ph	COMMAND パリティ Pl	[CR or LF]
---	---	---	---	---	---	---	-----------------------	-----------------------	----------------------	----------------------	--------------------------	--------------------------	--------------------------	--------------------------	-----	--------------------------	--------------------------	-----------------------	-----------------------	------------------

Figure 6

### 応答例

\*CMD:0108543732BD58010000BC [CR] [LF]

OK [CR] [LF]

*	C	M	D	=	COMMAND 実行結果 Sh	COMMAND 実行結果 Sl	COMMAND 応答長 Mh	COMMAND 応答長 Ml	COMMAND 応答 [0] D0h	COMMAND 応答 [0] D0l	COMMAND 応答 [1] D1h	COMMAND 応答 [1] D1l	...	COMMAND 応答 [m] Dmh	COMMAND 応答 [m] Dml	COMMAND パリティ Ph	COMMAND パリティ Pl	[CR]	[LF]	O	K	[CR]	[LF]
---	---	---	---	---	-----------------------	-----------------------	----------------------	----------------------	--------------------------	--------------------------	--------------------------	--------------------------	-----	--------------------------	--------------------------	-----------------------	-----------------------	------	------	---	---	------	------

Figure 7

## GPIO モード

GPIO モードでは、GPIO\_0~GPIO\_3 の各ピンを入力ピンまたは出力ピンとして、また ADC0,1 をアナログ値入力として動作させることができます。各 GPIO ピンの動作モードはピンごとに独立して設定することができ、設定値は sakura.io プラットフォーム側からの要求によって変更することが可能です。本モジュールの起動時、GPIO\_0~GPIO\_3 の各ピンは入力モードで動作します。

## 入力モード

GPIO ピンを入力モードにした場合、いずれかのピンに印加されている信号レベルが変化した場合に全ピンに関して自動的にメッセージを生成し、sakura.io プラットフォームへデータを送信します。変化を検出するのは GPIO ピンのみで、ADC ピンのアナログ値は変動しても検出されません。

sakura.io プラットフォームに送信されるデータは、GPIO はそれぞれピン番号・入出力設定・変化後の信号レベル・変化した日時、ADC はそれぞれピン番号・計測アナログ値・サンプルした日時が、含まれます。

信号変化後の sakura.io プラットフォームへの到達時間や、信号変化間隔通りの sakura.io プラットフォームへの到達の保証はされません。また、連続した変化信号に対しては一部しか変化を検出できないことがあります。最終的な信号値と sakura.io プラットフォームへ送信した最終データは一致しない可能性があります。

入力データはある程度キューイングされますが、通信回線状態やキューの開き具合によるデータ破棄がありうるため、確実な到達は保証されません。またデータの到達を確認する手段は用意されていません。必要な場合はユーザ側での信号の送受で実現するか、信号上でデータが到達しなかった場合にも問題が生じないようにユーザ機器を設計する必要があります。

## 出力モード

ピンを出力モードにした場合、sakura.io プラットフォーム側からの要求に応じてピンの出力レベルを変化させることができます。出力される値は H と L のいずれかとなり、メッセージを受信した時点でピンの出力値が変更されます。その際、sakura.io プラットフォーム側からのメッセージを受け取った後、現在の全ピンの状態を sakura.io プラットフォームへ返送します。

sakura.io プラットフォームに送信されるデータは、GPIO はそれぞれピン番号・入出力設定・変化後の信号レベル・変化した日時、ADC はそれぞれピン番号・計測アナログ値・サンプルした日時が、含まれます。



sakura.io プラットフォーム側でデータセット後の到達時間や、データセット間隔通りの到達間隔の保証はありません。また、連続的な sakura.io プラットフォームからのデータセットに対して GPIO 出力信号の変化の間隔は未定義です。

通信回線状態やキューの開き具合によるデータ破棄がありうるため、確実な到達は保証されません。またデータの到達を確認する手段は用意されていません。必要な場合はユーザ側での信号の送受で実現するか、信号上でデータが到達しなかった場合にも問題が生じないように、ユーザ側で設計する必要があります。

GPIO ピンで LED 等のデバイス を駆動する場合はそのまま駆動はできませんので、ドライブ回路を設計頂きご利用ください。

## 省電力制御

### Activeモード/DeepSleepモード選択

WAKE\_IN 端子の状態により Activeモードと DeepSleepモードを選択することができます。

	WAKE_IN 端子	説明
Activeモード	Hi	モジュールを動作させます。 WAKE_IN を Lo にすると現在の処理内容、状況を問わず即 DeepSleepモードへ移行します。この際に処理中内容等は破棄されます。
DeepSleepモード	Lo	モジュールの機能を停止し消費電力を抑えます。 WAKE_IN を Hi にするとモジュールが再起動し、Activeモードへ移行します。

Table 1

### 省電力モード設定

(※ファームウェア v1.4.0 で追加)

Activeモード時に選択された省電力モードに従い、電力制御を行います。

	説明	通信可否
省電力制御無効モード	省電力制御を行いません。(デフォルト)	可
自動スリープモード	通信可能な状態で待機消費電力を削減します。 通信状態によって、RUNモードと SLEEPモードの間を自動的に遷移します。 電力制御の為、コマンド処理時に無効時よりも遅延が発生します。また、接続状態、電波強度の情報更新がリアルタイムではなく、10分に一度、もしくはデータ送受が発生したタイミングになります。	可

Table 2

## 時刻同期

### 概要

本モジュールは sakura.io プラットフォームと時刻の同期を行い現在時刻を保持します。

下記のタイミングで時刻の同期は行われます。

- ・起動、リセット復帰、DeepSleep 復帰後、sakura.io プラットフォームに接続したとき
- ・プラットフォーム接続後一時間に一回

## データキュー

### 概要

本モジュールには、本モジュールと、sakura.io プラットフォーム間で授受されるデータを一時的に保持するための記憶領域として、送信キューと、受信キューが内蔵されています。

送信キューは、ユーザ機器から sakura.io プラットフォームに送信したいデータ（以下、上りデータと呼びます）を受け付けるためのキューで、ユーザ機器は上りデータを随時追加することができます。ユーザ機器からの送信要求でキューに溜まったデータを sakura.io プラットフォームに送信することができます。

受信キューは、sakura.io プラットフォームから、デバイスに対して送達されたデータ（以下、下りデータと呼びます）を一時的に保持するためのキューで、ユーザ機器は、ここから下りデータを読み出すことができます。

### 時刻情報の付与

キューが介在することにより、データの伝達要求が発生した時刻と、当該データが実際に送達される時刻には、予測不可能な差異が生じ得ます。この解消を目的として、デバイスと、sakura.io プラットフォーム間で授受されるすべてのデータには、時刻情報が付与されます。

上りデータに関しては、データが本モジュールの送信キューに追加されたときから、sakura.io プラットフォームへの送達を開始するときまでの時間が自動的に管理され、キューによる遅延にかかわらず、当該データがいつ発生したものなのかを sakura.io プラットフォーム側で知ることができます。また、データを本モジュールの送信キューに追加した時点ですでに既知の遅延があった場合に備え、キューへのデータ追加の際に任意のオフセット時間を指定することもできます。

下りデータに関しては、sakura.io プラットフォームから本モジュールにデータが到着してから、当該データをユーザ機器が実際に受け取るまでの時間が自動的に管理され、キューによる遅延にかかわらず、いつ到着したものなのかをユーザ機器側で知ることができます。

## キューの大きさ

キューの大きさは、Figure 3 キューの最大保持数のとおりです。

1つのデータは、

- Ch : 1 バイト
- Type : 1 バイト
- Data : 8 バイト
- Time : 8 バイト

の 18 バイトで構成されます。

ただし、内部処理の状況により、キューイングできる見かけ上のデータ数がキューの最大数よりも少なくなる場合があります。

## キューの最大保持数

送信キュー	32 データ
受信キュー	32 データ

Table 3

## 送信キューの構造

送信キューは、図のような構成で実装されています。

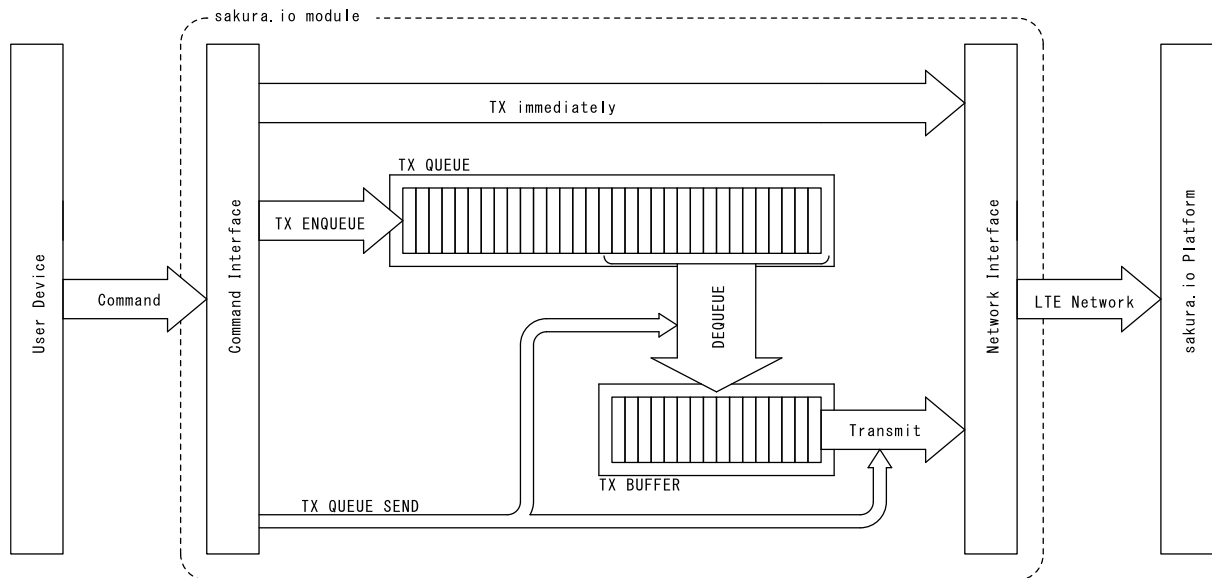


Figure 8

ユーザ機器から sakura.io プラットフォーム側へのデータ送信を行う場合には、本モジュールのコマンドインタフェースを通じて、Tx enqueue コマンドを使って送信キューへのデータ追加を要求することができます。送信キューに空きがある場合は、当該データはキューに追加されます。送信キューに空きがない場合は、追加要求はエラーとなりますが、すでにキューイングされているデータには影響ありません。

送信キューに溜められたデータは、即座には sakura.io プラットフォーム側には送信されず、明示的に Tx queue send コマンドを与えたときに送信が試みられます。Tx queue send コマンドが実行されると、まずこの時点でキューに入っている全データを送信対象に指定します。これ以降にキューに追加されたデータは今回のコマンドでは送信されません。送信処理は最大 16 個までのデータをひとまとめにして sakura.io プラットフォームへの送信が行われます。まず送信キューから最大 16 個のデータを取り出して、一旦送信用にバッファに確保し、ひとつのセットとして sakura.io プラットフォームへ送信を行います。正常に送信完了すると、17 個目以降のデータについて同様に送信処理を行い、送信対象にしたデータをすべて送信完了すると送信処理は終了します。

送信が失敗した場合には送信用のバッファにデータは確保したまま送信処理は終了し、Get Tx status で状態を取得すると送信エラーが応答される状態になります。この状態で Tx queue send コマンドを再度発行するとキューからはデータを取らずに送信用のバッファにあるデータの再送を試みます。これを繰り返すことで再度送信リトライが可能となります。

送信処理が送信キューから最大 16 個のデータを取り出して一旦送信用にバッファに確保した段階で送信キューには空きができるので、送信処理が完了していないタイミングでも、最大 16 個のデータをキューに追加することができます。

なんらかの理由により、送信キューに滞留しているデータを送信せずに消去したい場合には、送信キュー削除コマンドにより行うことができます。また、送信キューの記憶域は揮発性であり、電源を遮断、リセット、DeepSleep モードへの移行を行った場合には保持されません。

ユーザ機器から sakura.io プラットフォーム側へのデータ送信経路の例外として、送信キューを通さずに即座に sakura.io プラットフォーム側へのデータ送信を試みる、Tx immediately コマンドも用意されています。Tx immediately コマンドと、送信キューは併用することができ、Tx immediately コマンドの実行は、送信キューの状態には影響を与えません。

## 受信キューの構造

受信キューは、図のような構造で実装されています。

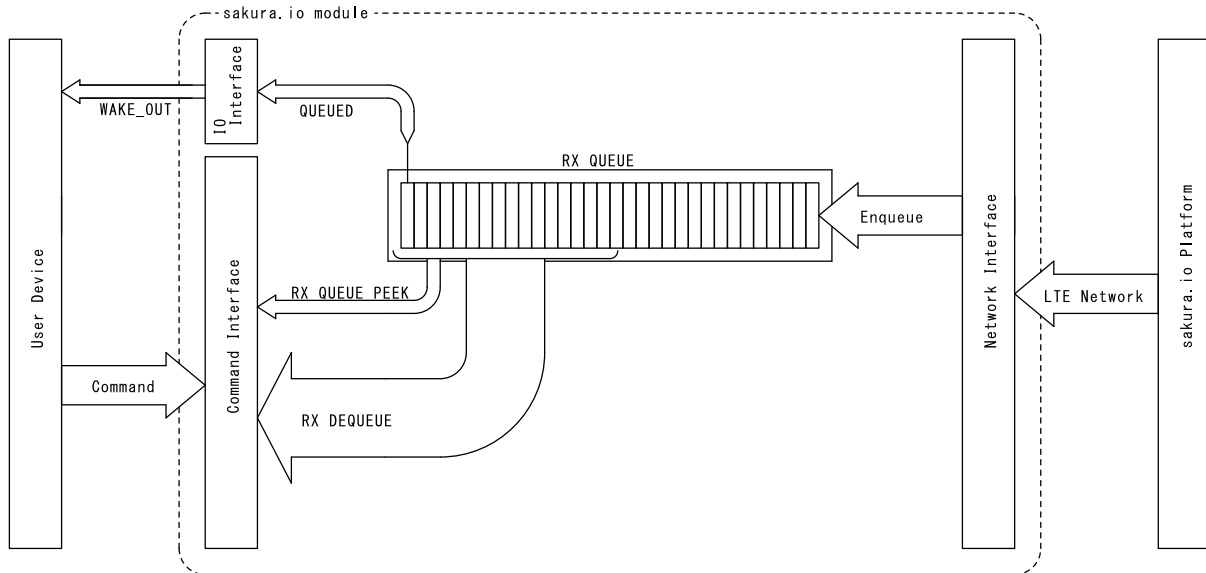


Figure 9

sakura.io プラットフォームからデバイスに対して送達されたデータは、本モジュールの受信キューに蓄積されます。

受信キューにデータがあるとき、ユーザ機器が本モジュールに対して受信読み出しコマンドを与えると、受信キューの先頭にあるデータが返され、当該データは受信キューから削除されます。受信キューからの削除を希望しない場合は、受信キューの状態に影響を与えずに、受信キューの先頭にあるデータを参照のみするコマンドを使用することができます。受信キューに空きがない場合は、sakura.io プラットフォーム側からの送達データは失われますが、すでにキューイングされているデータには影響ありません。

なんらかの理由により、受信キューに滞留しているデータを受信せずに消去したい場合には、受信キュー削除コマンドにより行うことができます。また、受信キューの記憶域は揮発性であり、電源を遮断、リセット、Deepsleep モードへの移行を行った場合には保持されません。

受信キューに 1 つ以上のデータがある場合は、WAKE\_OUT 信号がアサートされ続けます。受信キューのデータが空になると、デアサートされます。この出力は、ユーザ機器の MCU に対し、sakura.io プラットフォーム側から到達したデータが受信キューに存在することを知らせる信号として、またユーザ機器の MCU がスリープ状態にある場合には、ウェイクアップさせるための信号として使用することができます。

## コマンドシンタックス

### コマンドの構成

SPI/I2C/UART 問わず同じコマンドシンタックスを使用します。

コマンドは、「要求」と「応答」から構成されます。ユーザが、何らかの結果を期待して本モジュールに対して「要求」を与えると、本モジュールは、当該要求を処理した結果としてユーザに対して「応答」を返します。

要求は、1 バイトの要求種別と、1 バイトの引数データ長、および 0 バイト以上の引数データから構成される可変長のバイト列です。

応答は、1 バイトの実行結果と、1 バイトの応答データ長、そして 0 バイト以上の応答データから構成される可変長のバイト列です。

要求および応答のシンタックスは、以下のとおりです。

### 要求のシンタックス

要求種別	引数データ長	引数データ				パリティ
Q	N	A (0)	A (1)	...	A (n-1)	P

Figure 10

### 応答のシンタックス

実行結果	応答データ長	応答データ				パリティ
S	M	D (0)	D (1)	...	D (m-1)	P

Figure 11

引数データまたは応答データが多バイトで構成されるデータ型（整数または浮動小数点の値）である場合は、リトルバイトエンディアンとして扱い、下位バイトから順にバスへ送信します。パリティ P の計算方法は、要求であれば Q から A<sub>(N-1)</sub>までの、応答であれば S から D<sub>(M-1)</sub>までの値を 1 バイトで排他的論理和を取った値とします。

例：  $Q \oplus N \oplus A_{(0)} \oplus A_{(1)} \oplus \dots \oplus A_{(N-1)} = P$



## 実行結果

応答バイト列のうち実行結果 S は、要求の処理結果を示します。S が取り得る値とその大まかな意味については、Table 4 のとおりです。詳細な応答の意味については各コマンドの項にて説明します。

S の値が 01h の場合は、与えられた要求種別に対して規定されている応答データ長を返します。それ以外の場  
合については、S に続き、応答データ長 00h、パリティ P の 3 バイト固定で応答します。

S の値	意味
00h	予約
01h	要求のシンタックスに異常がなく要求が正常に処理されたことを示します。
02h	与えられた要求のパリティ値が異常であったことを示します。
03h	与えられた要求の要求種別が未定義であったことを示します。
04h	与えられた要求にその他のエラーがあったことを示します。これには、引数の長さの異常、範囲異常などが含まれます。
05h	与えられた要求を実行したものの、結果としてエラーになったことを示します。
06h	アンロックコマンドによるロック解除が行われていない状態でロック解除が必要なコマンドが発行されたことを示します。
07h	要求された操作が既に進行中であり、新たに操作を開始することができないことを示します。

**Table 4**

## コマンド定義

この項では、コマンドの要求シンタックスと応答シンタックスを定義します。

### 一般コマンド

#### 接続状態の取得

Get connection status				
要求	要求種別Q	引数データ長N	バリティP	
	01h	00h	p	
応答	実行結果S	応答データ長M	応答データD	バリティP
	s	01h	D	p

Figure 12

本モジュールの sakura.io プラットフォームへの接続状態を応答します。

応答データ D は以下の値を取ります。

D 値	意味
00h	LTE 網接続後の sakura.io プラットフォームへの接続処理中です。
01h	圏外、モジュール未登録、失効、SIM 抜け、などの LTE 網に接続できないエラーの際に発生します。 圏外の場合は提供エリアに復帰次第自動的に再接続処理が行われます。圏外でないことが確認できているものの、この値になる場合は、エラー要因を取り除き、再度接続を試みてください。
02h	接続エラー。LTE 網には繋がるものの sakura.io プラットフォームの認証 NG になった状態です。sakura.io プラットフォームへのモジュールの登録を確認してください。
03h	意図しない切断が発生しました。 本モジュールのリセットを行い、再度接続処理を試みてください。
80h	エラーなし。sakura.io プラットフォームへ接続完了しデータが送受できる状態です。

Table 5

ファームウェアバージョン v1.4.0 現在、sakura.io コントロールパネルへ本モジュールが登録されていないことによる接続失敗は圏外(01h)として扱われます。

モジュール登録後、認証できるようになるまで多少時間かかることがあります。しばらく時間をおいて再度試してみてください。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 接続状態の正常性を表すものではありません。

02h	与えられた要求のパリティ値が異常であったことを示します。 このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

**Table 6**

## 電界強度の取得

### Get signal quality

要求	要求種別Q	引数データ長N	パリティP	
	02h	00h	p	
応答	実行結果S	応答データ長M	応答データD	パリティP
	s	01h	D	p

Figure 13

本モジュールが接続している無線基地局からの受信品質を応答します。応答データ D の値は、そのときの受信信号品質に応じて、以下の値を取ります。

D 値	意味
00h	圏外 モジュール未登録や失効などによるものも含まれます。 LTE 電波状況や、モジュールの登録状況を確認して再接続してください。
01h	非常に弱い
02h	弱い
03h	中くらい
04h	強い
05h	非常に強い

Table 7

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 受信品質状態の正常性を表すものではありません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 8

## 現在時刻の取得

Get date time										
要求	要求種別Q	引数データ長N	パリティP							
	03h	00h	p							
応答	実行結果S	応答データ長M	応答データD							パリティP
	s	08h	D0[LSB]	D1	D2	D3	D4	D5	D6	D7[MSB]

Figure 14

応答データDとして、本モジュールが有している現在時刻を、64ビットのUNIX時刻形式（1970年1月1日00時00分00秒UTCを起点とした秒数）の1000倍（ミリ秒単位）で応答します。

実行結果Sは以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	エラー。時刻が sakura.io プラットフォームと同期していないときこの値を返します。 このとき応答データはなく M=00h となります。
06h	使用しません
07h	使用しません

Table 9

## エコーバックテスト

Echo back test											
要求	要求種別Q	引数データ長N	引数データD								パリティP
	0Fh	n	D0	...	...	...	...	...	...	Dn-1	p
応答	実行結果S	応答データ長M	応答データD								パリティP
	s	n	D0	...	...	...	...	...	...	Dn-1	p

Figure 15

要求にて与えられた任意長の引数データ D を、そのまま応答する試験用のコマンドです。引数データ長 n は、少なくとも 1 としてください。このコマンドを実行しても、本モジュールの動作には影響を与えません。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 エコーバックされるデータの正常性を表すものではありません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 10

## データ送信コマンド

### 送信キューへの追加

Tx enqueue											
Time offset付											
要求	要求種別Q	引数データ長N	引数データ							パリティP	
	20h	12h	ch	type	データD			Time Offset			
			ch	type	D0	...	D7	T0	...	T7	p
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								
Time offsetなし											
要求	要求種別Q	引数データ長N	引数データ				パリティP				
	20h	0Ah	ch	type	データD						
			ch	type	D0	...	D7	p			
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								

Figure 16

sakura.io プラットフォームへ送信するデータを、本モジュールの送信キューに追加するためのコマンドです。引数 ch は、ユーザが任意に割り当てられるチャンネル番号で、00h から 7Fh までの任意の値を使用することができます。80h 以上の値を指定することはできません。sakura.io プラットフォーム側でこのチャンネル番号とデータが紐づけられて取得できるので、送信データの分類分けなどユーザの目的に合わせて使用可能です。type は、送信するデータの型を指定するための型指定子で、以下のいずれかの値とする必要があります。

データD型	type 値
符号あり 32bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	69h
符号なし 32bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	49h
符号あり 64bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	6Ch
符号なし 64bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	4Ch
32bit 浮動小数点数 IEEE754 binary32 フォーマットで格納します。 sakura.io プラットフォーム到着時に 10 進の浮動小数点テキスト表記に変換されます。 完全な IEEE754 binary32 精度が必要な場合は「8 バイトの配列」での送信をお勧めします。	66h
64bit 浮動小数点数 IEEE754 binary64 フォーマットで格納します。 sakura.io プラットフォーム到着時に 10 進の浮動小数点テキスト表記に変換されます。 完全な IEEE754 binary64 精度が必要な場合は「8 バイトの配列」での送信をお勧めします。	64h

<p>8 バイトの配列</p> <p>sakura.io プラットフォーム到着時に 8 個連続した 16 進文字列に表記に変換されます。</p> <p>例 :</p> <p><math>D_{(0)} = 01h, D_{(1)} = 02h, D_{(2)} = 03h, D_{(3)} = 04h, D_{(4)} = 05h, D_{(5)} = 06h, D_{(6)} = 07h, D_{(7)} = 08h</math> の時、 "0102030405060708"</p>	62h
---	-----

**Table 11**

引数 D には、64bit 型データの値を、下位バイトから順に 8 バイト格納します。

32bit 型の値を格納する場合には、上位に 32bit 付加して 64bit データとし、64bit 型データと同様に格納する必要があります。上位 32bit の値は don't care として扱われます。

また、8 バイト配列型の場合も同様に、必要データサイズが 8 バイト未満であってもダミーデータを付加し、8 バイトのデータとして送る必要があります。

D (64 ビット値)							
$D_{(0)}$ [LSB]	$D_{(1)}$	$D_{(2)}$	$D_{(3)}$	$D_{(4)}$	$D_{(5)}$	$D_{(6)}$	$D_{(7)}$ [MSB]

**Figure 17**

D (32 ビット値)				don't care			
$D_{(0)}$ [LSB]	$D_{(1)}$	$D_{(2)}$	$D_{(3)}$ [MSB]	X	X	X	X

**Figure 18**

引数 T には、当該データが発生した時刻を、このコマンドが実行された時点基準する 1 [ms] 単位の相対時間として指定することができます。

例として、T の値が 2,500 (09C4h) であるときは、当該データは、コマンドが実行された瞬間を起点として 2.5 [s] 過去に発生したものであることを意味します。モジュールは Tx queue send コマンド発行後、各データ毎に T にキューでの滞留時間を加えた合計時間 T' を計算しデータとセットで sakura.io プラットフォームへ送信します。sakura.io プラットフォームではデータの到着時間から T' を引いた時間をデータの発生時間として保持します。しかし、ネットワーク回線上での送信遅延や、LTE 圏外からの再接続処理での遅延などは補正されませんので、不特定時間のずれが発生します。確実な時間が必要な場合には別途ユーザ側で時間管理データを送受するなど補償処理を行ってください。

T の最大値は、7,776,000,000 (1 CF7C 5800h) で、最大で 90 日まで過去の情報を扱うことができます。

ただし、sakura.io プラットフォーム側の仕様として、T の値と送信キューでの滞留時間を合計した結果、データ発生時間が sakura.io プラットフォーム到着時間よりも 90 日を超える過去となったデータは受け容れを行わずに破棄します。90 日の判定は sakura.io プラットフォーム側でのみ行われます。破棄されたかどうかはモジュール側からは検出できません。90 日超の T をセットしたり、キュー滞留で 90 日超えた場合には、sakura.io プラットフォームでのデータ蓄積は行われませんが、本モジュールからはエラー無くデータの送信が行われてポイントの消費が発生しますので注意してください。

Time offset (64 ビット値)							
$T_{(0)}$ [LSB]	$T_{(1)}$	$T_{(2)}$	$T_{(3)}$	$T_{(4)}$	$T_{(5)}$	$T_{(6)}$	$T_{(7)}$ [MSB]



**Figure 19**

Tは省略することが可能で、その場合はTの値として0が指定されたものとみなされます。また、Tを省略した場合のNは0Ahとなります。Tの値として、範囲外の値が与えられた場合の動作は未定義です。また、未来の時刻を与えることはできません。

キューがオーバーフローした場合には、データ追加要求はエラーとなります。その場合、実行結果Sとして05hを返します。事前にGet Tx queue lengthでキューの空きを確認することをお勧めします。

Tx queue sendコマンドでデータ送信中にTx enqueueした場合、送られるかキューに残るかはタイミング次第になります。確実な送信の為に、キューの空きの確認と、送信ステータスの確認を行ってください。

**データ例**

符号あり 32bit 整数

		引数データ										
要求	要求種別Q	引数データ長N	ch	type	符号あり 32bit 整数D			Time Offset			パリティP	
	20h	12h	ch	69h	D0	...	D7	T0	...	T7	p	
応答	実行結果S	応答データ長M	パリティP	D4~D7はダミーデータで埋める								
	s	00h	P									

**Figure 20**

符号あり 32bit 整数

		引数データ						
要求	要求種別Q	引数データ長N	ch	type	符号あり 32bit 整数D			パリティP
	20h	0Ah	ch	69h	D0	...	D7	p
応答	実行結果S	応答データ長M	パリティP	D4~D7はダミーデータで埋める				
	s	00h	p					

**Figure 21**

符号なし 32bit 整数

		引数データ										
要求	要求種別Q	引数データ長N	ch	type	符号なし 32bit 整数D			Time Offset			パリティP	
	20h	12h	ch	49h	D0	...	D7	T0	...	T7	pph	
応答	実行結果S	応答データ長M	パリティP	D4~D7はダミーデータで埋める								
	s	00h	pph									

**Figure 22**

符号なし 32bit 整数

		引数データ						
要求	要求種別Q	引数データ長N	ch	type	符号なし 32bit 整数D			パリティP
	20h	0Ah	ch	49h	D0	...	D7	p
応答	実行結果S	応答データ長M	パリティP	D4~D7はダミーデータで埋める				
	s	00h	p					

**Figure 23**

符号あり 64bit 整数

		引数データ										
要求	要求種別Q	引数データ長N	ch	type	符号あり 64bit 整数D			Time Offset			パリティP	
	20h	12h	ch	6Ch	D0	...	D7	T0	...	T7	p	
応答	実行結果S	応答データ長M	パリティP									
	s	00h	p									

**Figure 24**

符号あり 64bit 整数

		引数データ						
要求	要求種別Q	引数データ長N	ch	type	符号あり 64bit 整数D			パリティP
	20h	0Ah	ch	6Ch	D0	...	D7	p
応答	実行結果S	応答データ長M	パリティP					
	s	00h	p					

Figure 25

符号あり 64bit 整数

		引数データ									
要求	要求種別Q	引数データ長N	ch	type	符号あり 64bit 整数D			Time Offset			パリティP
	20h	12h	ch	4Ch	D0	...	D7	T0	...	T7	p
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								

Figure 26

符号あり 64bit 整数

		引数データ						
要求	要求種別Q	引数データ長N	ch	type	符号あり 64bit 整数D			パリティP
	20h	0Ah	ch	4Ch	D0	...	D7	p
応答	実行結果S	応答データ長M	パリティP					
	s	00h	p					

Figure 27

32bit 浮動小数点

		引数データ									
要求	要求種別Q	引数データ長N	ch	type	32bit 浮動小数点D			Time Offset			パリティP
	20h	12h	ch	66h	D0	...	D7	T0	...	T7	p
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								

D4～D7はダミーデータで埋める

Figure 28

32bit 浮動小数点

		引数データ						
要求	要求種別Q	引数データ長N	ch	type	32bit 浮動小数点D			パリティP
	20h	0Ah	ch	66h	D0	...	D7	p
応答	実行結果S	応答データ長M	パリティP					
	s	00h	p					

D4～D7はダミーデータで埋める

Figure 29

64bit 浮動小数点

		引数データ									
要求	要求種別Q	引数データ長N	ch	type	64bit 浮動小数点D			Time Offset			パリティP
	20h	12h	ch	64h	D0	...	D7	T0	...	T7	p
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								

Figure 30

64bit 浮動小数点

		引数データ						
要求	要求種別Q	引数データ長N	ch	type	64bit 浮動小数点D			パリティP
	20h	0Ah	ch	64h	D0	...	D7	p
応答	実行結果S	応答データ長M	パリティP					
	s	00h	p					

Figure 31

8byte配列		引数データ									
Time offset付											
要求	要求種別Q	引数データ長N	ch	type	8byte配列D			Time Offset			パリティP
	20h	12h	ch	62h	D0	...	D7	T0	...	T7	p
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								

Figure 32

8byte配列		引数データ									
Time offsetなし											
要求	要求種別Q	引数データ長N	ch	type	8byte配列D			パリティP			
	20h	0Ah	ch	62h	D0	...	D7	p			
応答	実行結果S	応答データ長M	パリティP								
	s	00h	p								

Figure 33

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 キューにデータが正しく入れることができた状態です。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常(Ch, Type 値の間違い)などその 他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	コマンドの実行結果がエラーです。キューがオーバーフローしたことを示します。 このとき応答データはなく M=00h となります。
06h	使用しません
07h	使用しません

Table 12

## データの即時送信

Tx immediately

Time offset付																					
要求	要求種別Q	引数データ長N	ch	type	引数データ1			引数データ2			...	引数データn			Time Offset	パリティP					
21h	(0Ah x n) + 08h		ch	type	D0	...	D7	ch	type	D0	...	D7	ch	type	D0	...	D7	T0	...	T7	p
応答	実行結果S	応答データ長M	パリティP																		
s	00h	p																			

Time offsetなし																					
要求	要求種別Q	引数データ長N	ch	type	引数データ1			引数データ2			...	引数データn			パリティP						
21h	0Ah x n		ch	type	D0	...	D7	ch	type	D0	...	D7	ch	type	D0	...	D7				p
応答	実行結果S	応答データ長M	パリティP																		
s	00h	p																			

Figure 34

sakura.io プラットフォームへ送信するデータを、送信キューを通さずに即座に送信するためのコマンドです。このコマンドは送信キューには影響せず、キューイング済みのデータがある場合はそのまま保持されます。複数のデータを最大 16 個まで連結して同時に送信可能です。連続して送信する場合、Time Offset 値は一つだけつけることができ、これを付けた場合連続した最大 16 個のデータは同一の Time Offset 値のデータとして扱われます。異なる Time Offset 値を付ける必要がある場合はコマンドを分ける必要があります。

引数 ch は、ユーザが任意に割り当てられるチャンネル番号で、00h から 7Fh までの任意の値を使用することができます。80h 以上の値を指定することはできません。sakura.io プラットフォーム側でこのチャンネル番号とデータが紐づけられて取得できるので、送信データの分類分けなどユーザの目的に合わせて使用可能です。

type は、送信するデータの型を指定するための型指定子で、以下のいずれかの値とする必要があります。

D 型	type 値
符号あり 32bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	69h
符号なし 32bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	49h
符号あり 64bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	6Ch
符号なし 64bit 整数 sakura.io プラットフォーム到着時に 10 進のテキスト表記に変換されます。	4Ch
32bit 浮動小数点数 IEEE754 binary32 フォーマットで格納します。 sakura.io プラットフォーム到着時に 10 進の浮動小数点テキスト表記に変換されます。 完全な IEEE754 binary32 精度が必要な場合は「8 バイトの配列」で送信をお勧めします。	66h
64bit 浮動小数点数 IEEE754 binary64 フォーマットで格納します。 sakura.io プラットフォーム到着時に 10 進の浮動小数点テキスト表記に変換されます。 完全な IEEE754 binary64 精度が必要な場合は「8 バイトの配列」で送信をお勧めします。	64h

<p>8 バイトの配列</p> <p>sakura.io プラットフォーム到着時に 8 個連続した 16 進文字列に変換されます。</p> <p>例 :</p> <p><math>D_{(0)} = 01h, D_{(1)} = 02h, D_{(2)} = 03h, D_{(3)} = 04h, D_{(4)} = 05h, D_{(5)} = 06h, D_{(6)} = 07h, D_{(7)} = 08h</math> の時、 "0102030405060708"</p>	62h
--	-----

**Table 13**

引数 D には、64bit 型データの値を、下位バイトから順に 8 バイト格納します。

32bit 型の値を格納する場合には、上位に 32bit 付加して 64bit データとし、64bit 型データと同様に格納する必要があります。上位 32bit の値は don't care として扱われます。

また、8 バイト配列型の場合も同様に、必要データサイズが 8 バイト未満であってもダミーデータを付加し、8 バイトのデータとして送る必要があります。

D (64 ビット値)							
$D_{(0)}$ [LSB]	$D_{(1)}$	$D_{(2)}$	$D_{(3)}$	$D_{(4)}$	$D_{(5)}$	$D_{(6)}$	$D_{(7)}$ [MSB]

**Figure 35**

D (32 ビット値)				don't care			
$D_{(0)}$ [LSB]	$D_{(1)}$	$D_{(2)}$	$D_{(3)}$ [MSB]	X	X	X	X

**Figure 36**

[ch, type, Data]のセットは連続して n=16 回まで繰り返すことができます。それぞれ、別の ch、type、Data を指定することができます。TimeOffset だけは、連続データすべてで共通の Offset ととして、一つだけ最後に引数 T で付加することができます。引数 T は省略可能で、その場合は 0 がセットされたものとみなします。

引数 T には、当該データが発生した時刻を、このコマンドが実行された時点を基準する 1 [ms] 単位の相対時間として指定することができます。

例として、T の値が 2,500 (09C4h) であるときは、当該データは、コマンドが実行された瞬間を起点として 2.5 [s] 過去に発生したものであることを意味します。モジュールは Tx immediately コマンド発行後、各データ毎に T に sakura.io プラットフォームへ送信されるまでの滞留時間を加えた合計時間 T' を計算しデータとセットで sakura.io プラットフォームへ送信します。sakura.io プラットフォームではデータの到着時間から T' を引いた時間をデータの発生時間として保持します。しかし、ネットワーク回線上での送信遅延や、LTE 圏外からの再接続処理での遅延などは補正されませんので、不特定時間のずれが発生します。確実な時間が必要な場合には別途ユーザ側で時間管理データを送受するなど補償処理を行ってください。

T の最大値は、7,776,000,000 (1 CF7C 5800h) で、最大で 90 日まで過去の情報を扱うことができます。

ただし、sakura.io プラットフォーム側の仕様として、T の値と sakura.io プラットフォームへ送信されるまでの滞留時間を合計した結果、データ発生時間が 90 日より過去となったデータは受け容れを行わずに破棄します。90 日の判定は sakura.io プラットフォーム側でのみ行われます。破棄されたかどうかはモジュール側からは検出できません。90 日超の T をセットしたり、滞留で 90 日超えた場合には、sakura.io プラットフォームでのデータ蓄積は行われませんが、データの送信が行われてポイントの消費が発生しますので注意してください。

Time offset (64 ビット値)							
T <sub>(0)</sub> [LSB]	T <sub>(1)</sub>	T <sub>(2)</sub>	T <sub>(3)</sub>	T <sub>(4)</sub>	T <sub>(5)</sub>	T <sub>(6)</sub>	T <sub>(7)</sub> [MSB]

**Figure 37**

T の値として、範囲外の値が与えられた場合の動作は未定義です。また、未来の時刻を与えることはできません。

Tx immediately でのデータ送信が成功したかどうかを Get Tx status で必ず確認してください。  
Get Tx status で失敗となっている場合、Tx immediately コマンドのデータは破棄されています。

Tx immediately の後、送信が完了する前に Tx immediately を発行すると後者のコマンドは、実行結果 S=07h が返りデータは送られません。Get Tx status で前の Tx immediately が完了したことを確認してから、新しい Tx immediately を発行してください。

新しい Tx immediately の発行が成功するまでは、Get Tx status では直前の Tx immediately のステータスが保持されます。

Tx queue send コマンドでデータ送信中に Tx immediately を発行すると、現在送信処理中のデータの送信完了後に、Tx immediately のデータが送られます。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 データの送信が正常に行われたかどうかは表していません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常(Ch, Type 値の間違い)などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	Busy 状態です。ひとつ前に実行された Tx immediately が完了していない状態です。

**Table 14**

## 送信キュー状態取得

Get Tx queue length

要求	要求種別Q	引数データ長N	パリティP	
	22h	00h	p	
応答	実行結果S	応答データ長M	応答データ	パリティP
	s	02h	available	queued

Figure 38

送信キューの状態を取得します。

available: 追加可能なデータの個数です。0 の場合には、データを追加することはできません。

queued: キューイングされているデータの個数です。0 の場合には、キューは空であることを示します。

(available + queued) の値はキュー全体の大きさを示しますが、キューの内部処理状況により合計値は変動する可能性があり、値が一定になることは保証されません。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 キューの状態が取得できた状態です。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 15

## 送信キュークリア

### Tx queue flush

要求	要求種別Q	引数データ長N	パリティP
	23h	00h	p
応答	実行結果S	応答データ長M	パリティP
	s	00h	p

Figure 39

送信キュー内のデータを sakura.io プラットフォームへ送信せずにすべて破棄し、キューを空にします。送信中状態になっているキューから取り出し済みのデータは破棄されず送信を継続します。ただし、送信がエラーとなり送信用バッファで待機しているデータについては破棄されます。送信中状態になっているキューから取り出し済みのデータが Tx queue flush 実行後にエラーとなった場合は破棄されず送信用バッファに確保され次回の Tx queue send で再送が試みられます。

キューが空の時の実行でも実行結果は成功になります

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 キューのクリアが成功しています。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 16



## 送信キュー送信

Tx queue send

要求	要求種別Q	引数データ長N	パリティP
	24h	00h	p
応答	実行結果S	応答データ長M	パリティP
	s	00h	p

Figure 40

送信キューのデータを、すべて sakura.io プラットフォーム側へ送信します。この要求を与えると、キューからのデータ送信処理が開始されます。コマンド実行後即応答返ってきますが、これは送信の完了を意味しているものではなく、モジュール内で引き続き送信作業が行われています。送信状態を知るには Get Tx status を使用します。

このコマンドが発行されると、まずこの時点でキューに入っている全データを送信対象に指定します。これ以降にキューに追加されたデータは今回のコマンドでは送信されません。送信処理は最大 16 個までのデータをひとまとめにして sakura.io プラットフォームへの送信が行われます。まず送信キューから最大 16 個のデータを取り出して、一旦送信用にバッファに確保し、ひとつのセットとして sakura.io プラットフォームへ送信を行います。正常に送信完了すると、17 個目以降のデータについて同様に送信処理を行い、送信対象にしたデータをすべて送信完了すると送信処理は終了します。

送信が失敗した場合には送信用のバッファにデータは確保したまま送信処理は終了し、Get Tx status で状態を取得すると送信エラーが応答される状態になります。この状態で Tx queue send コマンドを再度発行するとキューからはデータを取らずに送信用のバッファにあるデータの再送を試みます。これを繰り返すことで再度送信リトライが可能となります。

Tx queue send コマンドが活動している間は Get Tx status コマンドの応答として 01h(送信中)のステータスが応答されます。

送信が正常に終了すると、Get Tx status で 00h が応答されます。

キューが空の時に発行した Tx queue send コマンドは成功します。

また、Tx queue send コマンドで送信処理中に Tx queue send コマンドを発行した場合には S=07h(busy)を応答し処理されません。Get Tx status で状態を確認してからのコマンド発行をお勧めします。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 送信処理が開始されたことを示します。正常な送信完了を意味するものではありません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。

03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	Busy。以前の Tx queue send コマンドでの送信が処理中です。

**Table 17**

## 送信ステータス取得

Get Tx status

要求	要求種別Q	引数データ長N	パリティP		
	25h	00h	p		
応答	実行結果S	応答データ長M	応答データ		パリティP
	s	02h	D0	D1	p

Figure 41

直前に実行した、Tx queue send コマンドと、Tx immediately コマンドの送信状態を取得します。  
D<sub>(0)</sub>には Tx queue send コマンドによる送信の状態、D<sub>(1)</sub>には Tx immediately コマンドによる送信の状態が示されます。

D <sub>(0)</sub> 値	意味
00h	Tx queue send コマンドによる送信が起動後一度も行われていない、 Tx queue send コマンドによる送信が成功済み、 Tx queue flush コマンドで送信用バッファも空になった時、 のいずれかを表します。
01h	Tx queue send コマンドによる送信処理が実行中です。
02h	Tx queue send コマンドによる送信が失敗し、処理が終了した状態です。この時、キューとは別の送信用バッファに失敗時の送信データは保持しています。この状態で Tx queue send コマンドを発行すると送信用バッファのデータを再送することができます。 また、この状態で Tx queue flush コマンドを実行するとキューのデータだけでなく、送信用バッファで保持しているデータも消すことができます。

Table 18

D <sub>(1)</sub> 値	意味
00h	Tx immediately コマンドによる送信が起動後一度も行われていない、 Tx immediately コマンドによる送信が成功済み、 のいずれかを表します。
01h	Tx immediately コマンドによる送信処理が実行中です。
02h	Tx queue send コマンドによる送信が失敗し、処理が終了した状態です。この時、Tx immediately で送信を試みたデータは破棄されています。

Table 19

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 状態取得が正常に行われたことを意味します。

02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

**Table 20**

## データ受信コマンド

### 受信キュー取得

Rx dequeue											
要求	要求種別Q	引数データ長N	パリティP								
	30h	00h	p								
応答	実行結果S	応答データ長M	応答データ							パリティP	
	s	12h	ch	type	D		Time Offset			p	
			ch	type	D0	...	D7	T0	...	T7	p

Figure 42

受信キューから先頭のデータを一つ取り出して応答データとして返します。取り出したデータは、キューから削除されます。

応答データのうち、ch、type、およびDのシンタックスは、Tx enqueueの引数データと同一です。Tには、データがモジュールに到着してから本コマンドが実行されるまでの1 [ms] 単位の時間が格納されます。Tx enqueueのような時間のオフセットをsakura.ioプラットフォーム側から指定することはできません。

sakura.ioプラットフォーム側から送られたデータが、本モジュールに届いたときに化けるなどしてエラーとなっていた場合は破棄され受信キューには入りません。この破棄が発生したことを検出する手段は、sakura.ioプラットフォーム側からも本モジュール側からも用意されていません。確実な受信の為にはユーザ側で到達確認の情報をやり取りするなどの保障手段を講じてください。

キューが空の場合は、本コマンドは失敗し、sの値として05h、Mの値として00hを返します。

実行結果Sは以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 受信キューから正しくデータ取得できたことを示します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなくM=00hとなります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなくM=00hとなります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなくM=00hとなります。
05h	エラー。キューが空の時に帰します。 このとき応答データはなくM=00hとなります。
06h	使用しません
07h	使用しません

Table 21

## 受信キュー参照

Rx queue peek										
要求種別Q	引数データ長N	パリティP								
31h	00h	p								
実行結果S	応答データ長M	応答データ								パリティP
		ch	type	D			Time Offset			
s	12h	ch	type	D0	...	D7	T0	...	T7	p

**Figure 43**

受信キューから先頭のデータを一つを参照することができます。参照したデータは、キューにのこります。応答データのうち、ch, type, および D のシンタックスは、Tx enqueue の引数データと同一です。T には、データがモジュールに到着してから本コマンドが実行されるまでの 1 [ms] 単位の時間が格納されます。

Tx enqueue のような時間のオフセットを sakura.io プラットフォーム側から指定することはできません。

sakura.io プラットフォーム側から送られたデータが、本モジュールに届いたときに化けるなどしてエラーとなっていた場合は破棄され受信キューには入りません。この破棄が発生したことを検出する手段は、sakura.io プラットフォーム側からも本モジュール側からも用意されていません。確実な受信の為にはユーザ側で到達確認の情報をやり取りするなどの保障手段を講じてください。

キューが空の場合は、本コマンドは失敗し、s の値として 05h、M の値として 00h を返します。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 受信キューから正しくデータ取得できたことを示します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	エラー。キューが空の時に帰します。 このとき応答データはなく M=00h となります。
06h	使用しません
07h	使用しません

**Table 22**

## 受信キュー済み件数取得

### Get Rx queue length

要求	要求種別Q	引数データ長N	パリティP	
	32h	00h	p	
応答	実行結果S	応答データ長M	応答データ	パリティP
	s	02h	available queued	p

Figure 44

受信キューの状態を取得します。

available: 追加可能なデータの個数です。0 の場合には、新たなデータを受信することはできません。

queued: キューイングされているデータの個数です。0 の場合には、キューは空であることを示します。

(available + queued) の値はキュー全体の大きさを示しますが、キューの内部処理状況により合計値は変動する可能性があり、値が一定になることは保証されません。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 受信キュー件数が正しく取得できたことを示します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 23

## 受信キュークリア

Rx queue flush

要求	要求種別Q	引数データ長N	パリティP
	33h	00h	p
応答	実行結果S	応答データ長M	パリティP
	s	00h	p

Figure 45

受信キュー内のデータをすべて削除し、キューを空にします。  
キューが空の時に実行した場合も成功します。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 受信キューを正しく削除できたことを示します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 24



## ファイル受信コマンド

### ファイル受信開始

Start file download

要求	要求種別Q	引数データ長N	引数データ		パリティP
	40h	02h	ID0	ID1	p
応答	実行結果S	応答データ長M	パリティP		
	s	00h	p		

Figure 46

ファイル受信処理を開始します。引数 ID により、取得するファイル ID を指定します。ID は 16bit 符号なし整数で指定します。

このコマンドにより、ID に対応したファイルをインターネットよりダウンロードし、モジュールへ送信する準備をするよう sakura.io プラットフォームに対して指示が行われます。

実際に指定した ID のファイルが受信可能になったかどうかなどは Get file metadata コマンドで確認する必要があります。

Ver.1.4.3 以降は ID は 1～100 が指定できます。0 と 101 以降を指定すると s=04h が返ります。

Ver.1.2.1～1.4.2 は ID は 1～5 が指定できます。0 と 6 以降を指定すると s=04h が返ります。

Ver.1.2.0 以前はファイル受信機能をサポートしていません。s=03h が返ります。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 受信できるファイルがあるかなどを保証するものではありません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 ファイル受信機能をサポートしていない Ver. のときもこのエラーになります。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。ID の指定範囲の間違いでも発生しますが、Fw Ver.により発生条件が異なります。 このとき応答データはなく M=00h となります。
05h	エラー。ほかの Start file download により、ファイル受信が進行している状態です。 完了を待つか、キャンセルをする必要があります。 このとき応答データはなく M=00h となります。
06h	使用しません
07h	使用しません

Table 25

## ファイルメタ情報取得

Get file metadata													
要求	要求種別Q	引数データ長N	パリティP										
	41h	00h	p										
応答	実行結果S	応答データ長M	応答データ										
	s	11h	受信ステータスRs	合計サイズZ			タイムスタンプT			冗長符号C			パリティP
			r	Z0	...	Z3	T0	...	T7	C0	...	C3	p

Figure 47

受信するファイルのメタ情報を取得します。

このコマンドは、ファイル受信開始コマンドを実行したあと、ファイルの受信を開始するまでの間に実行することにより、受信するファイルサイズなどを取得できます。

応答データとして得られるメタ情報は、以下のとおりです。

Rs: ファイル受信リクエストステータス(sakura.io プラットフォームが、指定 URL から受信した際のステータス)

Z: ファイルの合計サイズ (32bit 符号なし整数で表すバイト数)

T: ファイルタイムスタンプ (64ビットの UNIX 時刻)

sakura.io プラットフォームが、指定 URL からファイルを取得したときの、HTTP ヘッダの Last-Modified を 64ビット UNIX 時刻に変換したものが入っています。

C: ファイル全体の冗長検査符号 (32bit 符号なし整数 : POSIX-CRC32)

Rs には、sakura.io プラットフォームがインターネット上のサーバに対してファイルのリクエストを行った結果が格納されます。Rs は、以下のいずれかの値となります。Rs におけるファイル受信エラーは、インターネット上のサーバから sakura.io プラットフォーム間で発生したエラーを意味します。

Rs 値	意味
00h	リクエスト成功。sakura.io プラットフォームからファイルサイズ、タイムスタンプ、冗長符号が正しく取得されています
01h	sakura.io コントロールパネルの指定 ID に対してファイル要求先 URL が登録されていないときです。Z,T,C の値は不定です。
02h	不正なリクエスト (要求されたファイル ID が許容範囲外) です。Z,T,C の値は不定です。
81h	sakura.io プラットフォームが指定の URL からダウンロードを試みたものの、ファイルが存在しなかった(HTTP 404) ときです。Z,T,C の値は不定です。
82h	sakura.io プラットフォームが指定の URL からダウンロードを試みたものの、サーバエラー (HTTP 500) が返ってきたときです。Z,T,C の値は不定です。
83h	sakura.io プラットフォームが指定の URL からダウンロードを試みたものの、リクエストの結果が不正な場合です。Z,T,C の値は不定です。以下のような場合が含まれます。 <ul style="list-style-type: none"> <li>● 応答に Content-Length ヘッダが含まれない(ファイルサイズ取得失敗)</li> <li>● ファイルサイズ制限超過(Max. 16MB)</li> <li>● 応答のペイロードが空</li> </ul>

Table 26

Start file download してない時、実行結果 S は不定となります。本コマンドは Start file download を行ってから実行してください。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 sakura.io プラットフォームからファイルメタに関する情報を取得できたことを意味します。あくまでも sakura.io プラットフォームから情報を取得できたことを表すもので、ファイルが無いなどのエラー状態も含まれます。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 ファイル受信機能をサポートしていない Ver. のときもこのエラーになります。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	Start file download 後、sakura.io プラットフォームがファイルメタ情報を準備している状態を示します。01h が返るまで、本コマンドで情報取得を繰り返してください。 圏外や、モジュール未登録時などでもこちらの状態になります。

**Table 27**

## ファイル受信ステータス取得

Get file download status

要求	要求種別Q	N	パリティP				
	42h	00h	p				
応答	実行結果S	応答データ長M	応答データ				パリティP
			受信ステータスRm	受信済みデータ長Z			
	s	05h	r	Z0	...	Z3	p

Figure 48

受信中ファイルの受信状態を取得します。ファイルの受信経過や完了の管理に使用します。  
 応答データとして得られるステータス情報は、以下のとおりです。

Rm: ファイル受信ステータス(本モジュールが sakura.io プラットフォームから受信したときのステータス)  
 Z: 受信済みデータ長 (32bit 符号なし整数で表すバイト数) モジュールからユーザ機器が受け取ったデータ量

Rm は、以下のいずれかの値となります。Rm であらわされるステータスは、 sakura.io プラットフォームから本モジュール間でのやり取りの状態を表します。

Rm 値	意味
00h	以下のいずれかの状態を表します ・Start file download が一度も発行されていない ・sakura.io プラットフォームから本モジュールへ全データが転送完了した状態。 (モジュールからユーザ機器に全データが渡ったことを表すものではありません。 ファイル受信用のバッファにデータが残っている場合も、本モジュールまで、 全データが届いた時点でこちらのステートに移行します。) ・ファイル受信中にキャンセルを発行した場合
01h	ファイル受信要求中。Start file download 発行後、sakura.io プラットフォームから本モジュールへファイルのメタデータが届いていない状態です。 Get file metadata コマンドで S=07h が返ってくる状態です。
02h	ファイル受信中。本モジュールが sakura.io プラットフォームからファイルのメタデータを取得してから、sakura.io プラットフォームから本モジュールへ全データが転送完了する直前までのステートになります。 本モジュールはファイルメタデータを取得できた時点で自動的に sakura.io プラットフォームから、ファイル受信用のバッファへダウンロードします。このファイル受信用のバッファの空きに寄らず完了直前までこのステータスを返します。
81h	ファイル受信エラー。 この状態で Get file metadata を実行すると Rs が 00h 以外の値が返る状態です。

Table 28

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 ファイル受信の正常性を示すものではありません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 ファイル受信機能をサポートしていない Ver. のときもこのエラーになります。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 29

## ファイル受信キャンセル

Cancel file download

要求	要求種別Q	引数データ長N	パリティP
	43h	00h	p
応答	実行結果S	応答データ長M	パリティP
	s	00h	p

Figure 49

ファイル受信を中止します。

受信中のバッファ内のデータは破棄され、ファイル受信関連の状態が初期化されます。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 ファイル受信機能をサポートしていない Ver. のときもこのエラーになります。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 30

## ファイル受信データ取得

Get file data					
要求	要求種別Q 44h	引数データ長N 01h	受信可能サイズRsize Rsize	パリティP p	
応答	実行結果S	応答データ長M	応答データ D		パリティP
	p	z	D0	...	Dz-1 p

**Figure 50**

受信中ファイルのデータを取得するためのコマンドです。

本モジュールは、指定された Rsize 以下で、ユーザ機器へむけて受信ファイルの先頭から応答データとして返送します。本コマンドを繰り返すたびに前回ユーザ機器へ未送信のところから順次データを応答します。応答データサイズは、最大データを指定された Rsize、最小で 0 の任意の値を取ります。このデータサイズは、sakura.io プラットフォームから本モジュールがデータを受け取っているバッファ量などで変化します。

Rsize: ユーザ機器側が一度に受け取れる応答データサイズを指定します。

最小 1byte、最大 255byte を指定することができます。(8bit 符号なし整数で表すバイト数)

M: sakura.io モジュールからユーザ機器に実際に送信されるデータサイズが応答されます。

0 以上、指定された Rsize 以下の値を取ります。(8bit 符号なし整数で表すバイト数)

ファイルの末尾まで受信できたかどうかは下記の二条件が同時に成り立っているかで判定できます

- Get file data コマンドで M=00h が返ってくる
- Get file Download Status コマンドで Rm=00h が返ってくる

ただし、通信エラーなどで一部データが喪失した場合などごく一部の条件で、Get file Download Status コマンドの結果がいつまでも Rm=00h にならないことがあります。その際は適宜ファイル受信をキャンセルし、やり直してください。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 ファイル受信機能をサポートしていない Ver. のときもこのエラーになります。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。

05h	<p>以下のいずれかの状態を表します</p> <ul style="list-style-type: none"> <li>・Start file download が発行されていない状態でのコマンド実行</li> <li>・Start file download 発行後、sakura.io プラットフォームから本モジュールへファイルのメタデータが届いていない状態でのコマンド実行</li> <li>・ユーザ機器が受信ファイルの全データを受け取った後のコマンド実行</li> <li>・ファイル受信キャンセル後のコマンド実行</li> </ul>
06h	使用しません
07h	使用しません

**Table 31**



## 管理コマンド

### 製品 ID の取得

#### Get product ID

要求	要求種別Q	引数データ長N	パリティP		
	A0h	00h	p		
応答	実行結果S	応答データ長M	応答データ		パリティP
	s	02h	D0	D1	p

Figure 51

本モジュールの製品 ID を応答します。

SCM-LTE-01 は 02h 00h、SCM-LTE-01 Rev.B は 03h 00h を返します。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 32

## 製品個体 ID の取得

### Get unique ID

要求	要求種別Q	引数データ長N	パリティP			
	A1h	00h	p			
応答	実行結果S	応答データ長M	D			パリティP
	s	0Ah	D0	...	D9	p

Figure 52

本モジュールの製品個体 ID を応答します。

製品個体 ID は個体ごとに異なる ID で、10 文字の ASCII 文字列です。

応答データの各文字は、21h から 7Eh までの、任意の値が出現します。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 33

## ファームウェアバージョンの取得

Get firmware version

要求	要求種別Q A2h	引数データ長N 00h	パリティP p			
応答	実行結果S s	応答データ長M m	D0	...	Dm-1	パリティP p

Figure 53

本モジュールのファームウェアバージョンを応答します。

バージョンを示す応答データは最大 32 文字の ASCII 文字列です。

応答データの各文字は、21h から 7Eh までの、任意の値が出現します。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 34

## 特殊コマンドアンロック

Unlock							
要求	要求種別Q	引数データ長N	引数データ				パリティP
	A8h	04h	53h	6Bh	72h	61h	p
応答	実行結果S	応答データ長M	パリティP				
	s	00h	p				

Figure 54

ファームウェア更新コマンドおよびソフトウェアリセットコマンドを実行するためのアンロックコマンドです。  
ファームウェア更新コマンドおよびソフトウェアリセットコマンドを実行する直前に、このコマンドを実行してください。

このコマンドの実行直後のコマンドのみロックが解除された状態で実行されます。

その次のコマンドからはロックがかかった状態になります。

Unlock コマンドを連続して発行した場合は最後の Unlock コマンドの直後のコマンドのみロックが解除された状態になります。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 35

## ファームウェア更新

### Firmware update

要求	要求種別Q	引数データ長N	パリティP
	A9h	00h	p
応答	実行結果S	応答データ長M	パリティP
	s	00h	p

Figure 55

ファームウェア更新処理を開始します。

このコマンドを実行するためには、直前に、特殊コマンドアンロックコマンドを実行する必要があります。

Sakura.io プラットフォームに用意されている最新ファームウェアに更新されます。バージョン指定はできません。

更新状況や、エラー状況は Get firmware update status で取得します。

正常にファームウェア更新が完了すると、本モジュールは自動的にリセットされ再起動します。

最新版にアップデートされたかどうか確認するには、もう一度 Firmware update してから、Get firmware update status コマンドでエラー01h が返ることで確認できます。

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 ファームウェア更新が正常に行われたことを意味しません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	Unlock コマンドでロックが解除されていないため、ファームウェア更新処理を開始できません。
06h	使用しません
07h	使用しません

Table 36

## ファームウェア更新状態取得

Get firmware update status

要求	要求種別Q	引数データ長N	パリティP	
	AAh	00h	p	
応答	実行結果S	応答データ長M	応答データ	パリティP
	s	01h	D	p

Figure 56

ファームウェア更新操作の状態を取得します。

D にエラーコードがセットされます。

エラーコードは以下の値を取ります。

D[7:0]の値	意味
00h	アップデート処理は行っていません。 また、前回実行のアップデート処理はありません。
01h	アップデート処理は行っていません。 また、前回実行のアップデート処理で既に最新版な為アップデート不要と判断されました。
02h	アップデート処理は行っていません。 また、前回実行のアップデート処理で、sakura.io プラットフォームから現在の最新ファームウェアのバージョン番号などの情報を取得に失敗しています。 回線状況などを確認してリトライしてください。
03h	アップデート処理は行っていません。 また、前回実行のアップデート処理で、現在より新しいファームウェアの存在を確認できたものの、ファームウェア自体のダウンロードが正常にできませんでした。 回線状況などを確認してリトライしてください。
04h	アップデート処理は行っていません。 また、前回実行のアップデート処理で、ファームウェア自体のダウンロードは成功したものの、書き込んだフラッシュ領域のデータのCRCをチェックに失敗しています。 回線状況などを確認してリトライしてください。
80h	現在アップデート処理中です。 Sakura.io プラットフォームからファームウェアデータを受信中はもちろん、回線の一次断、圏外、モジュール未登録時などでもこちらの状態になり、無限にリトライが繰り返されます。

Table 37

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。 ファームウェア更新が正常に行われたことを意味しません。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

**Table 38**

アップデート中に通信断などが発生したときの挙動は未定義です。

Firmware update コマンド発効後、100ms は正常な応答データが返らないため、Get firmware update status を発行しないでください。

## ソフトウェアリセット

### Software reset

要求	要求種別Q	引数データ長N	パリティP
	AFh	00h	p
応答	実行結果S	応答データ長M	パリティP
	s	00h	p

Figure 57

ソフトウェアリセットを行います。このコマンドが正常に実行された場合、即座にリセットが実行されます。応答は返していますが、基本的に即座にリセットが実行されるため、I2C や SPI からの応答の読み込みが遅れると既にリセット済みの場合があります。

その場合、ユーザ機器からは I2C や SPI の通信がエラーとしてみえる場合があります。

事前にアンロックコマンドを発行していない場合このコマンドは失敗します。

Update 中は発行しないでください

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	Unlock コマンドでロックが解除されていないため、リセットを開始できません。
06h	使用しません
07h	使用しません

Table 39



## 省電力モード設定コマンド

Set power save mode

要求	要求種別Q	引数データ長N	引数データ	パリティP
	B0h	01h	Pwr	p
応答	実行結果S	応答データ長M	パリティP	
	s	00h	p	

Figure 58

省電力モードの設定切り替えを行います。

Pwr: 省電力モード

Pwr は以下の値を取ります。

Pwr 値	意味
00h	省電力制御無効モード。 起動後、WAKE_IN 復帰後、リセット後のデフォルトモードです。
01h	自動スリープモード

Table 40

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 41

v1.4.0 よりも前のファームウェアでは実装されていません。この場合 S=03h が返ります。

## 省電力モード取得コマンド

Get power save mode

要求	要求種別Q	引数データ長N	パリティP	
	B1h	00h	p	
応答	実行結果S	応答データ長M	応答データ	パリティP
	s	01h	pwr	p

Figure 59

現在設定されている省電力モードを取得します。

Pwr: 省電力モード

Pwr は以下の値を取ります。

Pwr 値	意味
00h	省電力制御無効モード
01h	自動スリープモード

Table 42

実行結果 S は以下の値を取ります。

S 値	意味
00h	予約
01h	コマンド自体が正常に処理されたことを意味します。
02h	与えられた要求のパリティ値が異常であったことを示します このとき応答データはなく M=00h となります。
03h	与えられた要求の要求種別が未定義であったことを示します。 このとき応答データはなく M=00h となります。
04h	与えられた要求が、引数の長さの異常、数値範囲異常などその他のエラーがあったことを示します。 このとき応答データはなく M=00h となります。
05h	使用しません
06h	使用しません
07h	使用しません

Table 43

v1.4.0 よりも前のファームウェアでは実装されていません。この場合 S=03h が返ります。

## 制約事項

- ファイル受信コマンドは、v1.2.0 よりも前のファームウェアでは実装されていません。
- 省電力モード設定コマンド/省電力モード取得コマンドは、v1.4.0 よりも前のファームウェアでは実装されていません。

## ソフトウェアライセンス

この製品は、内部でオープンソースソフトウェアの lwIP を使用しています。そのライセンスは以下のとおりです。

lwIP is licenced under the BSD licence:

Copyright © 2001-2004 Swedish Institute of Computer Science.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## 注意事項

### 用途の限定

この製品は、特別に高い信頼性が要求され、その故障や誤作動が生命または身体に危害を及ぼす恐れ、膨大な財産損害を引き起こす恐れ、もしくは社会的に深刻な影響を及ぼす恐れのある用途に適用されることは意図されていません。購入者側の責任で以下を含む用途にこの製品を使用された場合、当社はいかなる責も負いかねますのでご了承ください。

原子力関連機器、航空宇宙機器、海底機器、電力制御機器、医療機器、輸送機器（自動車、列車、船舶等）、交通用信号機器、燃焼制御機器、各種安全関連機器、金融関連機器、昇降機器、その他これらの機器と同等の機器

### 屋外での使用

屋外で使用される場合や、防水性を要求されるような環境で使用される場合は、適切な防水構造の採用および結露対策を実施してください。

### コネクタ

コネクタ挿抜時には、過度の応力を加えないよう注意してください。また、搭載時には、応力が連続して加わらないよう注意してください。

### 設置環境

誤作動や、動作不良が予想されますので、以下のような状態または環境条件下では使用しないでください。

- (ア) 液体の中、または腐食性雰囲気のある場所
- (イ) 高温が連続する場所
- (ウ) 潮風を受ける場所
- (エ) 直射日光を受ける場所
- (オ) 急激な温度変化がある場所
- (カ) ヒーターやエアコンなどの風を直接受ける場所
- (キ) 強い振動がある場所
- (ク) 強い電磁波がある場所
- (ケ) 帯電または静電気が発生する場所
- (コ) その他、これらに準じる条件下

## 改訂履歴

日付	版	改訂内容
2020年1月31日	1.0.0	初版公開