3GIM の魅力

株式会社タブレイン 高本孝頼 2016/04

パート4:3 GIM を使ったクラウド連携

1. クラウド連携とは

IoT デバイスで取得したセンサ値は、ほとんどの場合、クラウド・サーバにアップすることとなります。このパート4でも、パート2で紹介した IoT デバイスを使って、温度センサ値をクラウド・サーバにアップする開発技術をご紹介します。

ここで利用するクラウドもフリーでサービスしている米国 AT & T が提供する M2X (URL: m2x.att.com)を紹介します。

まず今回のクラウド連携における全体フローを図1に紹介いたします。



図1. クラウドを利用した IoT システム開発事例

ここでは先に、m2xの登録方法とその利用についてご紹介し、その後 IoT デバイスから データをアップするプログラム(スケッチ)をご紹介し、さらにオプション機能として、 トリガー利用によるメール送信・ツイッター送信や、蓄積したデータ出力について説明し ていきます。

また、ここでのデータアップについては、フォーマットとして json 形式を使っています。 センサデータをアップする場合には、CSV や xml、それに json と3種類がほとんどですが、 現時点では、データが構造体で、かつタグが付けられ、さらにコンパクトであることでは、 json 形式が最も利用価値のあるものとなっています。単に1つのセンサだけだと、CSV で も構いませんが、今後多くの種類のセンサデータを蓄積していく場合には、json 形式をお 薦め致します。

(注意:本 m2x サイトは、英文サイトのみです)

2. クラウド・サーバ m2x の登録

ここで紹介する m2x は、つぎの URL を Web ブラウザで接続し、あらかじめ図 3 および 図 4 の画面でユーザ登録(SIGN UP)し、アカウントを取得しておいてください。



図2. m2x.att.comのメイン画面

0	AT&T Is T 5	ervices -	- 				
			Signup for yo	our	ATAT IOT Services	account	
			Signing up for an A1 of ATST lof Servic	87 ls7 H. INC	Services Account provides you will ucting N2X and Flow (free develop	traccesa to all er accounts).	
	ATAT	M2X	- e				
	ATAT	Plaw.	- a.		Million en ensenn : Presidente		
					Last move		
				Į,	phonese Off: Wardiamene 5 Intale over 100	Liamet Dr. Chinand	

図3. m2x.att.com/signup のユーザ登録サイト

この図4でユーザ登録の詳細項目を記載しました。この入力欄では、全て入力は英数字 で入力してください。

Last name	苗字(姓)
Username	ユーザ名
NOTE: Your username is unique	e, once set, it cannot be changed
Company	会社名
Phone	電話番号
Email	メールアドレス
Password	パスワード
Confirm Password	パスワード(確認)
Subscribe to the AT&T	M2X Newsletter

図4. m2xユーザ登録入力欄

全ての入力欄の記入が終わったら、「Create Account」(アカウント作成)ボタンを押して ください。直ぐにアカウントメールが入力したメールアドレスに送られ、図5の画面に切 り替わります。この画面がアカウントによってログインされた画面となります。



図5.m2xアカウント登録メール送信画面(初期画面)

それでは、このログインされた画面で、左上に表示されています「M2X」をクリックして、 各種設定画面に切り替えてください。

C Arat	NoTServices -					
M2X	Devices	© Detributions	() Deschloserete	<u>.</u>] • ?	r 🕑
Devices	1 Developer Acco	unt 🚳 0 of 10		Q Sent Deve	. A	barrend baard
(2) Gas	day Maran 🛛 👻					
G Al De	da New 🛛 👻 Ny Carl		No De	vices available yet.		
C Al De	ni ka ka v		No De	vices available yet.	sported devices.	
Al De	nda Maran 🗠		No De Get surred to eddrog your fo	vices available yet.	sported devices.	
C cha	de Maren e Frènce Dittorn La toot haven nitry Diffections		No De Get zazted by edding year fo	vices available yet. Stawice of by fining one of the sup Learn mate	sported devices.	

図6. 各種設定画面(デバイス選択画面)

ここで一旦、m2x での IoT デバイス設定のための 2 つの設定情報について説明いたします。 デバイス (Device):機器の設定(右画面の「Create Device」選択で設定:後述) ストリーム (Stream):デバイスに取りつくセンサ毎の設定

以下、この2つの設定を以下の画面で行っていきます。

Causer Name	And Contract Concerns
Level and	
Device Description (option)	Demothe your Davion.
Device Settal	+Q. \$256nie -
	An epidemionent intereffice for the database function constants any removables of the eccession transf, which Devices constants in a 2000/tachest much have a conjust Device Second .
Tage	Alter Fage to your Device
Visibility	Physic Device
	Pres can APP bank to change # pet have you share adds have # Device. 10.00mm to the man. #

図7.デバイス登録画面

	Martin State Streep	1
Triggers 💷	+ O () man // + 2x strant / an one littlants / w/2-4151467545693046	16-12 ¹
	Comment Affrent addressering Property.	
Add-Stream	Barrow De Vegers (g) Landen (g) Chen (g)	(WOurtes)
	Add in Streem	<streamid></streamid>
登録画面へ	Same I	
All	Bala Nov	Stream表示
OfficeTemp TC	1999 Type Concerning of Annual Statements of Annual	國文字系道訳
	Sectorization ()	中位設定
StreamIDは、入力した名前が		

図8.ストリーム(各センサ値)の登録設定

上記の2つ(デバイスとストリーム)の設定によって、以下の2つの ID および Key が IoT デバイスとセンサとの関係でプログラム記述で引用する必要があります。 deviceID:デバイス機器 ID(システム側設定:m2x では 10 個まで無償登録可能) x-m2x-key:1つのデバイス ID と関係づけられた m2x のキー(システム側設定) StreamID:センサ毎の ID(ユーザ側設定)

	f inttps://m2x.att.com	n/devices/dftc	- #1		12 - 1	2
36	IM E Datate		B PRIVA	ut di Patric samen		-
	FILIZE		deviceID		(100	J
A		0 <u>, 1999, 9</u>		🖉 1445	40.0055	
	toka moto:	atte	1	NV-logis conditionly	Device	
63	01063.7	иничных виресият.				
		renzioni	Ibd788978a1 Corr		Get Litzwy	
φ	west Hight Lockford	P research an etc.			API Documentation	
	No location available	2411				
			x-m2x-key			
entones treterro	API Keys API Ne	peer Log Trigger Log	Data Exports (0)	(1	Pl Chent Sheet	
in the second	API Keys API Her	Location (10) Charts	Dete Experts ())	Eng a Tart Maine	N Chert Sheet	
itreerro T	APT Krys APT Hee	Location (10) Charts	Corpuse ()	Lag street Maker	Pi Deet Sheet	

図9. デバイスとストリームの ID と Key

これらデバイスとストリームとの関係を再度、明確にしておきます。今回の IoT デバイス (Genuino101+3 GIM シールド+3 GIM) をデバイス ID として関係づけます。具体的

には、この IoT デバイスを m2x のデバイスと認識するだけです。図9 でのデバイス名は 「3GIM」としています。

この場合、自動的に deviceID と x-m2x-key が割り付けられます。つぎの StreamID は、IoT デバイスに取り付けた各センサを定義したもので、ストリーム(stream)表示名 とストリーム ID (StreamID) をユーザ側で設定しますが、StreamID の方がプログラム の中で関係します。

ここでの例としては、図8での IoT デバイスの温度センサを定義するとき、スケッチと 関連付け、しかも分かり易くするため、図9のようにストリーム表示名とストリーム ID を、 「3GIM_TEMP」と「TEMP」としておきます。プログラム(スケッチ)では、**StreamID** 「TEMP」のみが必要となります。

3. m2x へのデータアップ方法

つぎに m2x にセンサ値を送信する方法を説明いたします。具体的には、前述した deviceID および x-m2x-key、streamID を使って、以下のようなフォーマット(形式) によって、M2X クラウドにデータをアップします。

リスト①:m2x送信データ構造

\$WP http://api-m2x.att.com/v2/devices/deviceID/updates/
"{\$"values\$" : {\$" streamID \$" : [{ \$"timestamp\$" : \$"date-time\$" , \$"value\$" : \$"val\$"}]}} "
"Host: api-m2x.att.com\$r\$nX-M2X-KEY: x-m2x-key :\$r\$nContent-Type:application/ison\$r\$n"

ここで、「\$WP」は、3GIM による httpPOST コマンドとなります。(マニュアル参照) 次の「http://api-m2x.att.com/v2/devices/deviceID/updates/」はクラウドにアップする URL、 「"{\$"values\$": {\$"streamID\$": [{ \$"timestamp\$": \$"date-time\$", \$"value\$": \$"val\$"}]}}"」は ボディ部で、「"Host: api-m2x.att.com\$r\$nX-M2X-KEY:x-m2x-key:\$r\$nContent-Type:application/json\$r\$n"」は ヘッダー部となります。

なお、この中の「date-time」はセンサ値の取得日時で、「val」は、取得したセンサ値を変換した値(摂氏温度など)となります。つまりいずれも可変のデータとなります。

この「date-time」は、以下のようなフォーマットとします。

2016-04-20T01:23:45+09:00

上記は、「年-月-日 T 時:分:秒+時:分」となっていて、「年」は 4 桁、以下「月」、「日」、「時」、 「分」、「秒」は共に 2 桁の数字となります。また最後の「+時:分」は、国際時間の時差となり ます。日本時間だと、「+09:00」となります。ただ「+」がネット通信の際には特殊文字とな るため、URL コードの「%2B」に置き換えて記述します。(マニュアル参考、後述)

それからリスト①で示した**ボディ部**は、**json 形式**とし、以下内容となります。

{"values" : {"**streamID**" : [{"timestamp" : "**date-time**" , "value" : "**val**" }]}}

先のリスト①で「\$"」を使っているのは、3GIM での URL コード(特殊文字のみ)を指定しているためのもので、さらにこの内容をネット通信する場合には、「¥」を付けて、「¥\$¥"」と変換したコードを付ける必要があります。(以下の json 形式およびリスト②参照)

{¥\$¥"values¥\$¥" : {¥\$¥"streamID¥\$¥" : [{¥\$¥"timestamp¥\$¥" : ¥\$¥"date-time¥\$¥" , ¥\$¥ "value¥\$¥" : ¥\$¥"val¥\$¥" }]}} このように json 形式をネット通信で送信する場合、特殊文字の URL コード変換が必要で、 具体的には \$ コードや ¥ コードの挿入場所の理解が必要です。

さらに、これ全体を**ボディ部**として送るために、上記のリスト①のように前後に「"」(ダブ ルクォーテーション)で囲む必要があります。

4. IoT デバイスからの温度センサ値のクラウド・サーバアップ

それでは、既にパート2で紹介しました IoT デバイスを使って、温度センサの値を取得し、それを2分ごとに m2x クラウド・サーバにアップするスケッチを紹介しましょう。

以下のリスト②-1の中の2箇所「**deviceID**」と「**x-m2x-key**」を先に登録した m2x サイトから読み込んできて書き替えてください(カットアンドペースト)。この2ヶ所の変 更を行い、リスト②-1とリスト②-2からなるスケッチをコンパイルして、IoT デバイス (Geunino101)に書き込んで実行してください。これによって図*のように温度センサ値

がクラウドにアップされるようになります。

このリスト②-1の中のストリーム名「StreamID」は、あらかじめ定義した「TEMP」 に変更して記述しています。

リスト2-1:温度センサ値を3分ごとにm2xクラウドにアップするメインスケッチ



つぎにリスト2-2 について、紹介します。

```
// Get Date & Time (3GIM command)
// return --> string "2015-12-23T01:23:45%2B09:00"
// 2015-09-20T14:05:12.345Z\$\$", \$\$"value\$\": \$\$"32.0\$\$"]}\$\";
// + "2015-09-02T11:59:00%2B09:00&csv=TEMP01,2015-09-02T10:00:00%2B09:00LIGHT01,24.2"
String datetime() {
 String dtime;
 do {
    Serial1.println("$YT");
    while(!Serial1.available());
    dtime = Serial1.readStringUntil('¥n');
    Serial.println(dtime);
    delay(1000);
 } while ( !(dtime.indexOf("201") >0) );
 dtime.replace(" ","T"); dtime.replace("/","-");
 return(dtime.substring(7) + "$+09:00");
}
boolean _3Gsetup() {
 Serial.println("start _3Gsetup()");
 pinMode(7,OUTPUT);
 digitalWrite(7,HIGH); delay(100);
 digitalWrite(7,LOW); // 3G shield --> digitalWrite(7,HIGH);
 Serial1.begin(baudrate);
 //----- 3G module begin & connect ------
 String str;
 unsigned long tim = millis();
 do{ while(!Serial1.available());
      str=Serial1.readStringUntil('¥n');
 }while(!(str.indexOf("3GIM")>0) && (millis() - tim) <LIMITTIME);</pre>
  if( millis() -tim >= LIMITTIME) {
    return false;
 } else return true;
}
boolean _3G_WP(String command) {
 delay(10);
 Serial.println(command); // debug
 delay(10);
 Serial1.println(command);
 String rstr;
 unsigned long tim = millis(); // time set(ms)
 do{ while(!Serial1.available());
      rstr=Serial1.readStringUntil('¥n');
      Serial.println(rstr); //debug print....
 }while(!(rstr.indexOf("$WP=")==0) && (millis() - tim) <LIMITTIME);// $WP return check</pre>
  return (rstr.indexOf("$WP=")==0);
```

ここでの以下の3つの関数群について説明します。

_**3Gsetup** 関数: 3 GIM の初期接続設定を行う関数

_3G_WP 関数: httpPOST によるインターネット接続関数

引数は、「\$WP URL BODY HEADER」となります。(マニュアル参考)

datetime 関数:先に紹介したセンサ値取得日時で、3 GIM による日時よるフォーマット

から、m2xクラウド用に変換する関数



それでは、実際に実行した事例の結果を表示させてみましょう。

図 10. 出力グラフ事例

前に述べましたように、m2x での時間表示は、国際標準時(UTC)を用いて表示されま す。日本時間から比べると、9時間遅れていることから、表示された時間に9時間足した 時間が日本時間となります。

5. センサ値によるトリガー(メール送信・ツイート送信)の処理

m2x では、センサ値の動きを監視することができる仕組みがあり、その仕組みを使うことで、異常事態を知らせたりすることができるようになっています。

ここではセンサの値がある値(閾値)を超えた場合に、メールを送信する、またはツイ ート送信するものをご紹介しましょう。

例えば、ここでの温度センサの値が 30 度を超えた場合には、自分にメールを送信するようにしましょう。メール送信については、すでにパート2のリスト③「メール送信プログラム(sendmail.php)」をご紹介しました。このプログラムを使い、トリガー(温度が 30 度を超えた場合)によって、メールを自分に送信するものをご紹介しましょう。

このことで人手によっていつもチェック確認することなく、機械的に・自動的に、異常 を知らせるシステムとして利用できるようになります。

それでは、先の図9の中にある「Triggers」を選択します。(図11参照) その選択によって表示されたトリガー設定画面での内容を紹介いたします。







図12.トリガー(Triggers)詳細設定画面

ここで、「トリガー名」は任意に付けられますので、ここでは「temp over 30 degree」としておきます。次のストリームは、既に定義したストリーム表示名から選択できるようになっていますので、ここでは温度センサ値の「3 GIM_TEMP」を選択します。また比較演算は「>」を選択し、トリガー値は「30」を入力してください。 このことで、30 を超えた場合に、以下の URL を起動するようになります。 この URL は上記の「メール送信 URL」というところに記入してください。

http://******.webcrow.jp/sendmail.php?email=送信先アドレス&temp=over%2030C

ここでの sendmail.php の説明は、パート 2 をご覧ください。赤文字の 「over%2030C」は、メール送信される内容となります。

また、この部分を「ツイッタ―」によるものしたければ、以下の表記 URL を代わりに入れ てみてください。(トークンについては、パート 3 で紹介しています)

http://arduino-tweet.appspot.com/update?token=トークン&status=ツイート文

この場合には、温度が 30 度を超えた場合に、ツイートされます。ただ、同じ文章は 2 度以 上アップされない状況となります。

その他、トリガーの条件によっては、1度だけの URL 起動や、閾値を超えたりした場合とか、同じ状態でもある時間間隔ごとに URL 起動する場合があります。 ここでは、詳しく説明しませんが、条件等を変えて、起動してみてください。

6. クラウド蓄積データの CSV ファイル出力

m2x 上に溜めたデータは、任意に CSV ファイルで出力することができます。つまり指定 したデータを図 12 および図 13 のメニュー選択で、一時の CSV ファイルに出力し、ダウン ロードすることができます。



図13. データ出力 (Data Exports) 選択



図 14. データ出力(Data Exports) 設定画面

← → C & https://m2 http://m2	tx.att.com/devices/	aff.c	idata-exports	4 4 8
108.01		DOWNLOAD URL	013716	EXPERIS IN
	2040	(1111)//espect.421.41.41	이 Deta Export in Progress (working)	
1	. M	urty://apport.adu.st.w	You can check the status of your savey using the to been (INTERS CONTR. BTT. CONTRESS OF STORESTIN	
) must Use Privacy Policy Com	ant Us	Ĩ	Outo Export Completed	
UTE 4747 Intellement Property All rights 67. Gente cogni dels men propes an me attes manta an the anamig of their re	n hannan Iarrano el MAT postamación gentine conseto	-	CSV2P1	

図 15. データ出力 (Data Exports) 設定画面

上記の図 13、図 14、図 15 に沿って操作を行うことで、一時の CSV ファイルをダウンロードすることができます。また、その中身は、図 16 の内容で、時間とセンサ値のペアで出力することができます。

ファイル(F) ヘルプ(H)	編集(E) 1	式(0)	見示(∀)	
timestamp, 2016-04-19 2016-04-19 2016-04-19 2016-04-19 2016-04-19	TEMP 9T13:33:2 9T13:31:2 9T13:29:2 9T13:27:2 9T13:27:2	1.000+0 0.000+0 0.000+0 0.000+0 0.000+0	9:00,28.33 9:00,28.33 9:00,30.28 9:00,28.33 9:00,28.33	
2016-04-19 2016-04-19 2016-04-19 2016-04-19 2016-04-19 2016-04-19 2016-04-19	0T13:23:2 0T13:21:2 0T13:19:2 0T13:17:2 0T13:17:2 0T13:15:2 0T13:15:2	0.000+0 0.000+0 0.000+0 1.000+0 0.000+0 0.000+0 9.000+0	9:00.28.33 9:00.28.33 9:00.28.33 9:00.28.33 9:00.28.82 9:00.28.33 9:00.28.33	E
2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18 2016-04-18	T13:11:1 T13:09:1 T13:07:1 T13:05:1 T13:03:1 T13:03:1 T12:55:1 T12:55:1 T12:55:1 T12:55:1 T12:43:1 T12:43:1 T12:43:1 T12:43:1 T12:43:1 T12:43:1	8,000+0 9,000+0 9,000+0 9,000+0 9,000+0 9,000+0 9,000+0 9,000+0 9,000+0 8,000+0 8,000+0 8,000+0 8,000+0 8,000+0 8,000+0 8,000+0 8,000+0	9:00,22:33 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,27:84 9:00,28:33 9:00,28:32 9:00,28:30 9:00,28:30 9:00,28:30 9:00,29:30 9:00,29:30 9:00,29:30	
2016-04-19 2016-04-19 2016-04-19 2016-04-19	0T12:37:1 0T12:35:1 0T12:33:1 0T12:31:1	8,000+0 8,000+0 8,000+0 8,000+0 8,000+0	9:00.29.30 9:00.29.30 9:00.29.30 9:00.29.30 9:00.29.30	

図 16. m2x データ出力事例(CSV ファイル)

パート1:3GIMを使った開発事例

http://tabrain.jp/3GIM_V2.0/Part1%20Arduino%20and%203GIM.pdf

パート2:3GIM を使った技術資料(遠隔制御・メール送信など)

http://tabrain.jp/3GIM_V2.0/Part2%20Arduino%20and%203GIM.pdf

パート3:3 GIM を使ったツイート連携 http://tabrain.jp/3GIM_V2.0/Part3%20Arduino%20and%203GIM.pdf

以 上