

MX100/DARWIN 用 API ファーストステップガイド (API で MX100 用プログラムを作成するために)

本 API で MX100 用のプログラムを作成する場合、このマニュアルを最初にお読みください。

このマニュアルでは、MX100 用のプログラムを作成するための MX100 の基本的な事項について説明しています。本 API と MX100 については、下表のマニュアルを参照してください。これらのマニュアルは、それぞれの製品に付属の CD-ROM に収納されています。

マニュアル名	マニュアル No.
MX100/DARWIN 用 API ユーザーズマニュアル	IM MX190-01
MX100 データアキュイジションユニットユーザーズマニュアル	IM MX100-01

また、このマニュアルでは、API と拡張 API の違いを説明しています。拡張 API の特長をご理解の上、目的に合った API をご使用ください。

「MX100/DARWIN 用 API ユーザーズマニュアル」(IM MX190-01) に掲載されている以外のサンプルプログラムを弊社の Web ページで公開しています。「付録」をご覧ください。

目次

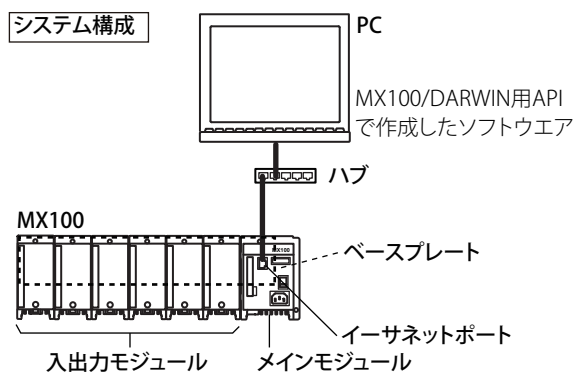
MX100 によるデータアキュイジションシステム	2
システム構成、モジュール番号、チャンネル番号.....	2
MX100 のデータと API の構造体の対応.....	3
API と拡張 API	4
拡張 API の特長.....	4
注意事項.....	4
データ収集機能 (FIFO)	5
MX100 のデータ収集機能 (FIFO).....	5
API または拡張 API によるデータ取得.....	6
付録	8
サンプルプログラムの掲載先.....	8
FAQ (Frequently Asked Questions) の掲載先.....	8

MX100 によるデータアキュイジションシステム

システム構成、モジュール番号、チャンネル番号

システム構成

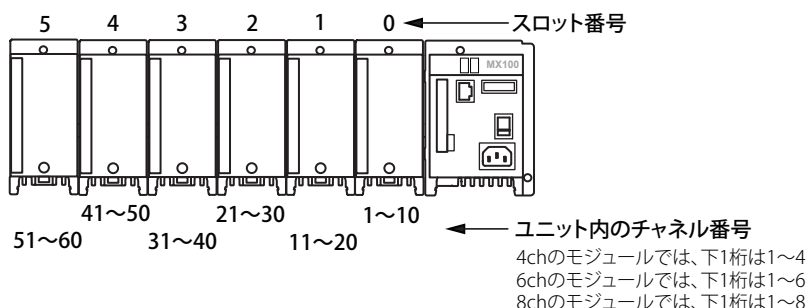
MX100 によるデータアキュイジションシステムは、データアキュイジションユニット（以下「MX100」と呼びます）、PC、およびネットワークへの接続機器で構成されます。ひとつの MX100 は、イーサネットポートを備えた「メインモジュール」、信号の入出力を行う「入出力モジュール」、およびそれらを装着し接続する「ベースプレート」で構成されます。



モジュール番号

入出力モジュールのモジュール番号は、そのモジュールを装着したスロットの番号です。スロット番号は、0、1、2、3、4、5です。複数スロットを占有するモジュールのモジュール番号は、占有するスロットのうち最も小さいスロット番号です。

例：スロット「2」、「3」、「4」を占有する「30ch 中速 DCV/TC/DI 入力モジュール」のモジュール番号は「2」です。



チャンネル番号

ユニット内のチャンネル番号でチャンネルを指定します。ユニット内のチャンネル番号は、そのモジュールが占有するスロット番号（ユニット上のスロット位置）と、そのモジュール内の何番目のチャンネルか（モジュール内の端子位置）によって決まります。

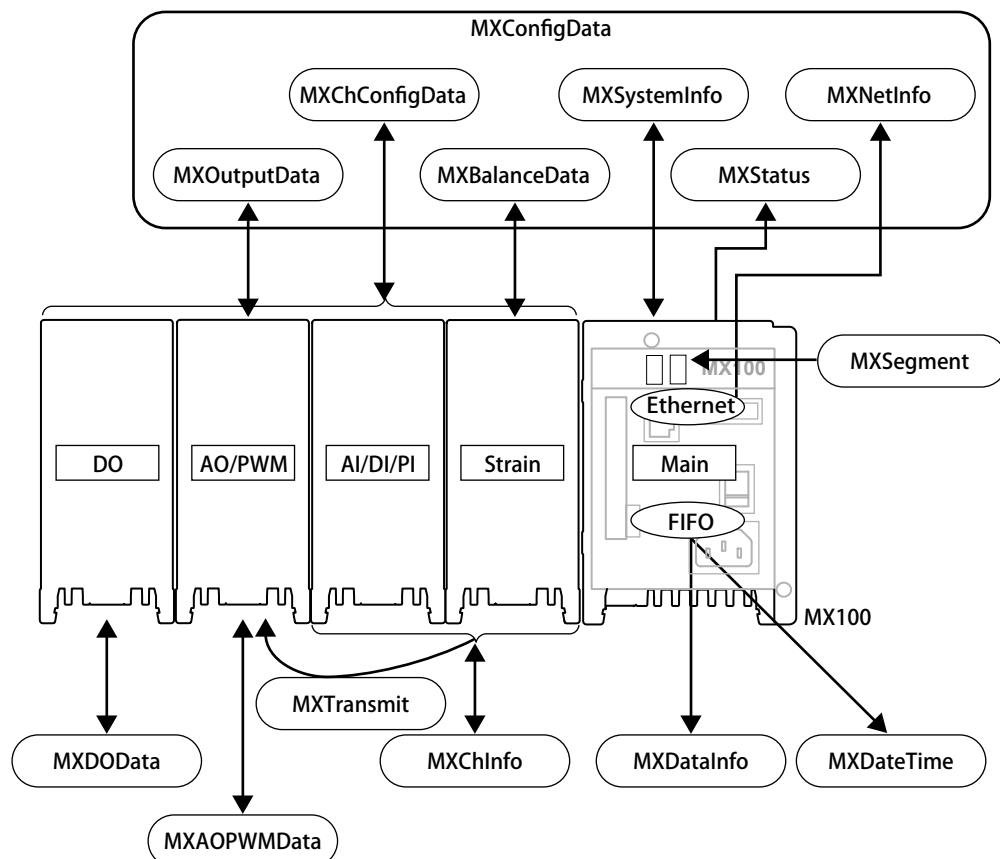
例：スロット番号3に装着したモジュールの、2番目のチャンネルのチャンネル番号は「32」です。

Note

弊社のソフトウェア、MX100 スタンダードやMXLogger では、チャンネル番号は「ユニット番号+ユニット内のチャンネル番号」の5桁で表記されますが、API ではユニットの IP アドレスを指定して通信するので、チャンネル番号としてユニット内のチャンネル番号だけを使用します。

MX100 のデータと API の構造体の対応

MX100 のデータを格納する構造体と対象のデータとの対応を示します。



MXSegmentなど :構造体名
 ← :矢印はデータの移動方向
 DO :デジタル出力モジュール
 AO :アナログ出力モジュール
 PWM :パルス幅出力モジュール
 AI :アナログ入力モジュール
 DI :デジタル入力モジュール
 PI :パルス入力モジュール
 Strain :ひずみ入力モジュール
 Main :メインモジュール
 Ethernet :Ethernet通信機能
 FIFO :データ収集機能(詳細は本書の5ページを参照してください)

各構造体については、「MX100/DARWIN 用 API ユーザーズマニュアル」(IMMX190-01)のそれぞれのページをご覧ください。

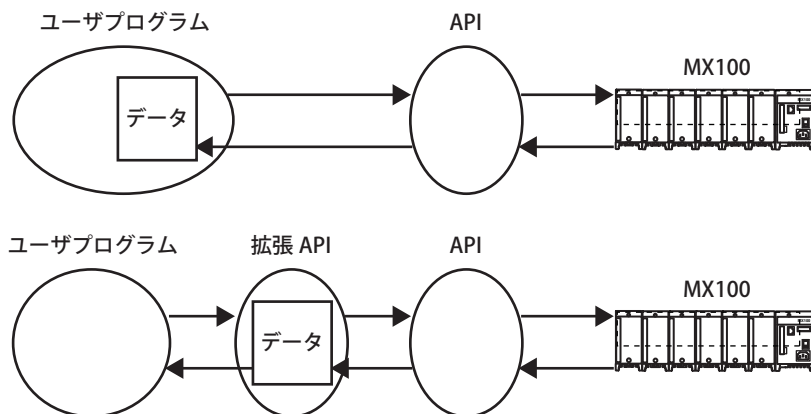
構造体	説明	参照先
MXAOPWMDData	出力モジュール (AO/PWM) の出力データです。	6-38 ページ
MXBalanceData	ひずみ入力モジュールの初期バランスデータです。	6-36 ページ
MXChConfigData	チャンネルの設定データです。	6-31 ページ
MXChInfo	測定チャンネル (入力モジュール) の表示用データです。	6-31 ページ
MXConfigData	全設定データです。	6-37 ページ
MXDataInfo	測定データのデータ値です。	6-28 ページ
MXDateTime	測定データの時刻情報です。	6-27 ページ
MXDODData	デジタル出力モジュール (DO) の出力データです。	6-38 ページ
MXNetInfo	通信設定データです。	6-36 ページ
MXOutputData	出力モジュール (AO/PWM) の設定データです。	6-37 ページ
MXSegment	7セグメント LED の表示データです。	6-38 ページ
MXStatus	システム状態データです。	6-35 ページ
MXSystemInfo	メインモジュールを含むシステム設定データです。	6-34 ページ
MXTransmit	入力モジュールから出力モジュールへの伝送出力データです。	6-38 ページ

API と拡張 API

拡張 API の特長

本ソフトウェアには、API と拡張 API の2つのソフトウェアがあります。拡張 API は、より簡単にプログラムを作成できるようにした API です。拡張 API は、いくつかの API の機能を組み合わせて、よりユーザフレンドリな機能として提供します。拡張 API は、API の上位 API に位置付けられており、API を呼び出して動作します。下図は2つの API の違いを説明したものです。API と拡張 API については、「MX100/DARWIN 用 API ユーザーズマニュアル」(IM MX190-01) の「1.2 ソフトウェアの構成と特長」にも説明が記載されています。

MX100 用の API と拡張 API



プログラムの単純化

API では、取得したデータの保存領域はユーザ側になります。そのため、メモリ量を含めたパフォーマンスは、ユーザプログラムで制御できます。一方拡張 API では、拡張 API がデータを保持し、管理します。そのため、オーバーヘッドが発生しますが、ユーザが領域を制御しなくてもよいので、プログラムが単純化できます。

データ操作の単純化

拡張 API では関数の戻り値や引数に、構造体を使用しません。

関数の単純化

拡張 API の関数は、

- ・ 計測器と通信を行い、計測器の状態を変更し、また、拡張 API が保持しているデータを変更する関数 (状態遷移関数)
 - ・ 拡張 API が保持しているデータを参照するための関数 (取得関数)
- の2種類に単純化されています。

対応プログラム言語の拡張

API	拡張 API
Visual C++	Visual C++
Visual C	Visual C
Visual Basic	Visual Basic
	Visual Basic.NET
	C#

注意事項

API と拡張 API の関数を混在して使用した場合、動作は保証できません。

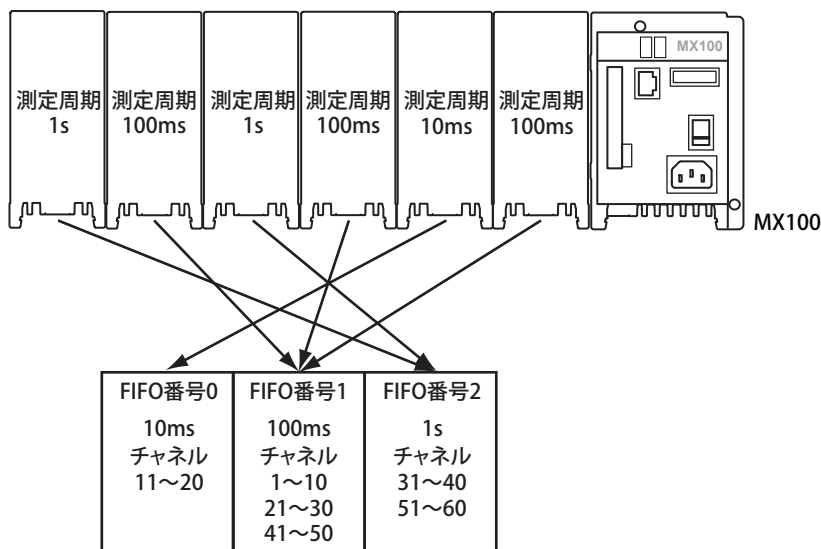
その他の拡張 API の注意事項については、「MX100/DARWIN 用 API ユーザーズマニュアル」(IM MX190-01) の 12-2 ページ「注意事項」をご覧ください。

データ収集機能 (FIFO)

MX100 のデータ収集機能 (FIFO)

測定周期と FIFO

MX100 ではモジュールごとに最大 3 つの測定周期を設定できます。測定周期ごとに、収集したデータを格納する領域が MX100 内に用意されます。この領域が FIFO(First In First Out) 方式のため、以下このデータ収集そのものを「FIFO」と呼びます。FIFO とは、領域の先頭からデータを書き込んでいき、領域の最後まで書き込むと、先頭から上書きしていく方式です。FIFO には測定周期の早い順に、「0」「1」「2」の番号が割り振られます。同じ測定周期に設定されたモジュール上のチャンネルの測定データは、同じ FIFO に格納されます。よって、FIFO 番号を指定して MX100 からデータを取得すると、同じ測定周期のチャンネルすべての測定データを取得することができます。



測定が設定されていないチャンネルは含まれません。

モジュールに設定できる測定周期は下表のとおりです。

モジュール	形名	測定周期の選択肢											
		10	50	100	200	500	1	2	5	10	20	30	60
		ms					s						
4ch 高速ユニバーサル入力モジュール	MX110-UNV-H04	○	○	○	○	○	○	○	○	○	○	○	○
10ch 中速ユニバーサル入力モジュール	MX110-UNV-M10	-	-	○	○	○	○	○	○	○	○	○	○
30ch 中速 DCV/TC/DI 入力モジュール	MX110-VTD-L30	-	-	-	-	○	○	○	○	○	○	○	○
6ch 中速 4 線式 RTD 抵抗入力モジュール	MX110-V4R-M06	-	-	○	○	○	○	○	○	○	○	○	○
4ch 中速ひずみ入力モジュール	MX112-B12-M04 MX112-B35-M04 MX112-NDI-M04	-	-	○	○	○	○	○	○	○	○	○	○
10ch 高速デジタル入力モジュール	MX115-D05-H10 MX115-D24-H10	○	○	○	○	○	○	○	○	○	○	○	○
10ch パルス入力モジュール	MX114-PLS-M10	-	-	○	○	○	○	○	○	○	○	○	○

○：選択可、-：選択不可

API または拡張 API によるデータ取得

API または拡張 API により、FIFO から測定データを取得する場合について説明します。

プログラムの流れ

MX100 の測定データを取得する場合、API と拡張 API ではそれぞれ下表のようなプログラムの流れになります。サンプルプログラムを参照してください。サンプルプログラムについては、本書の「付録」に記載の参照先をご覧ください。

API	機能	プログラムの流れ
	測定データの取得	1 FIFO を開始
		2 取得できるデータ範囲を FIFO から取得
		3 測定データの時刻情報を FIFO から取得
		4 測定データのデータ値を FIFO から取得
		5 FIFO を停止

2、3、4を繰り返すことにより、継続して FIFO から測定データを取得できます。

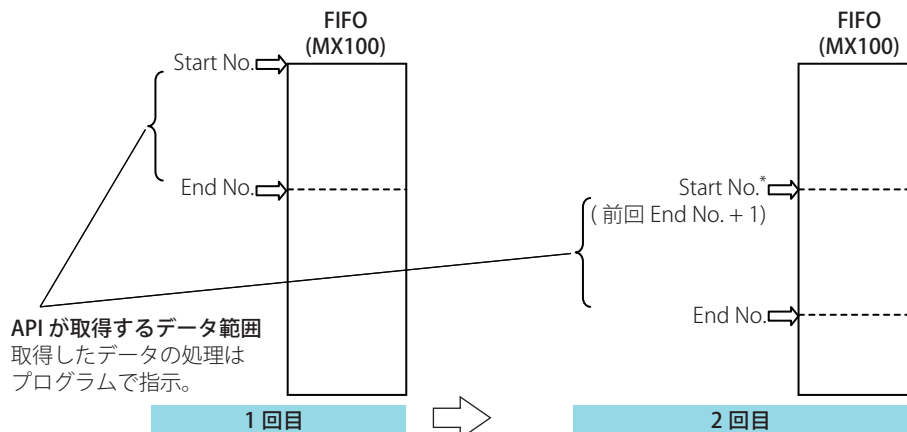
拡張 API	機能	プログラムの流れ
	測定データの取得	1 FIFO を開始
		2 計測点をひとつ進める。
		3 計測点の測定データを取得。
		4 FIFO を停止

2を実行するたびに、計測点をひとつ進めます。従って、2と3を繰り返すことにより、順番に測定データを取得できます。

API の場合

下図は、API が FIFO から測定データを取得するときの状態を示します（「プログラムの流れ」の表の「2」「3」「4」）。API は範囲を指定 (Start No. と End No.) して FIFO から測定データの時刻情報 (MXDateTime) とデータ値 (MXDataInfo) を取得します。

指定できる範囲は、データ範囲の取得関数で取得できます。



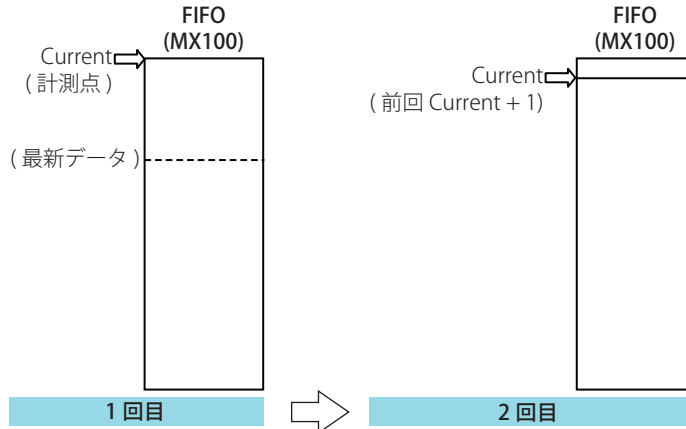
* 1回のデータ取得処理で、指定した範囲の最後のデータまで取得できないことがあります。その場合は、取得できた最後のデータ番号の次が Start No. になります。

使用する関数については、下表に記載の「MX100/DARWIN 用 API ユーザーズマニュアル」(IMMX190-01) の参照先をご覧ください。サンプルプログラムについては、本書の「付録」に記載の参照先をご覧ください。

言語	参照先
	取得するデータ範囲を取得、測定データの時刻情報を取得、測定データを取得
Visual C++	2-8 ページの「測定データの取得 (FIFO 指定)」
Visual C	3-4 ページの「測定データの取得 (FIFO 指定)」
Visual Basic	4-4 ページの「測定データの取得 (FIFO 指定)」

拡張 API の場合

下図は、拡張 API が測定データを取得するときの状態を示します（「プログラムの流れ」の表の「2」「3」）。瞬時値指定の場合は、最新データが Current になります。



計測点を進めるには、状態遷移関数の測定データ取得機能を使用します。Current データを読み出すには、各取得関数を使用します。

状態遷移関数の測定データ取得機能、取得関数については、下表に記載の「MX100/DARWIN 用 API ユーザーズマニュアル」(IMMX190-01) の参照先をご覧ください。サンプルプログラムについては、本書の「付録」に記載の参照先をご覧ください。

プログラム言語	参照先	
	状態遷移関数の取得機能	取得関数
Visual C++	12-8 ページの「取得機能」	12-9 ページ
Visual C	13-6 ページの「取得機能」	13-7 ページ
Visual Basic	14-6 ページの「取得機能」	14-7 ページ
Visual Basic.NET	15-6 ページの「取得機能」	15-7 ページ
C#	16-6 ページの「取得機能」	16-7 ページ

瞬時値によるデータ収集

瞬時値によるデータ収集の場合、FIFO 内の最新データを取得することになります。API は MX100 の測定データには最短 100ms でアクセスできます。従って、瞬時値を取得する場合、最速で 100ms ごとになります。それより速い測定周期の場合は、FIFO 番号を指定してデータを取得します。

FIFO のスタート/ストップ

MX100 からデータを取得するためには、FIFO がスタートしていることが必要です。API で FIFO をスタートするとすべての FIFO がスタートし、FIFO をストップすると、すべての FIFO がストップします。

データを取得するときの注意

バッファ内がデータでいっぱいになってしまう前に、PC から MX100 にアクセスしてデータを読み出すことが必要です。FIFO 用バッファの容量は、3 つの FIFO を合わせて 2M バイトです。たとえば、60 チャンネルのデータを測定周期 10ms で収集する場合、約 60 秒*でバッファがいっぱいになります。バッファがいっぱいになると、MX100 はバッファの先頭からデータを上書きします。従って、60 秒より短い周期で PC が MX100 からデータを取得することが必要です。実際の取得周期は、アプリケーションに合わせて決めます。

* バッファがいっぱいになるまでの時間の算出方法については、「MX100/DARWIN 用 API ユーザーズマニュアル」(IMMX190-01) の「付録 3 MX100 のタイムアウト値の算出」を参照してください。

約 3 分間アクセスがない場合、MX100 が通信を切断します。

付録

サンプルプログラムの掲載先

ユーザズマニュアル

CD-ROM に収納されている「MX100/DARWIN 用 API ユーザズマニュアル」(IMMX190-01) に、下表のサンプルプログラムが掲載されています。

API 用サンプルプログラム

プログラム言語	内容	参照先
Visual C++	測定データの取得	2-10 ページ
	設定データの取得 / 設定	2-13 ページ
Visual C	測定データの取得	3-7 ページ
	設定データの取得 / 設定	3-11 ページ
Visual Basic	測定データの取得	4-7 ページ
	設定データの取得 / 設定	4-10 ページ

拡張 API 用サンプルプログラム

プログラム言語	内容	参照先
Visual C++	測定データの取得	12-16 ページ
	設定データの読み出しと書き込み	12-18 ページ
Visual C	測定データの取得	13-13 ページ
	設定データの読み出しと書き込み	13-15 ページ
Visual Basic	測定データの取得	14-13 ページ
	設定データの読み出しと書き込み	14-15 ページ
Visual Basic.NET	測定データの取得	15-13 ページ
	設定データの読み出しと書き込み	15-15 ページ
C#	測定データの取得	16-13 ページ
	設定データの読み出しと書き込み	16-15 ページ

Web ページ

下記の URL に、サンプルプログラムが掲載されています。
<http://www.yokogawa.co.jp/ns/mx100/download/>

内容	プログラム言語	API	拡張 API
測定データを取得して波形表示する。	Visual C++	○	○
	Visual C	○	○
	Visual Basic	○	○
	Visual Basic.NET	—	○
	C#	—	○
2つのチャンネルの測定値を取得して、X-Y プロット表示する。	Visual Basic	○	○

○：サンプルプログラムあり。
—：サンプルプログラムなし（未対応）。

FAQ (Frequently Asked Questions) の掲載先

よくある質問と回答が、下記の URL に掲載されています。

<http://www.yokogawa.co.jp/ns/faq/daqmaster/ns-mxapi-faq-01.htm>