

WAGO-I/O-PRO CAA ライブラリ

DMX_01.lib

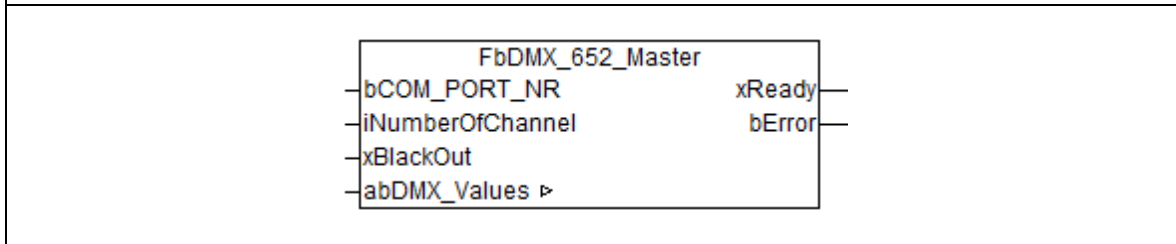
通信

DMX マスタブロック(FbDMX_652_Master)

WAGO-I/O-PRO V2.3 ライブラリ要素		
カテゴリ:	ビルディング	
名称:	FbDMX_652_Master	
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/> Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib	
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib	
適用:	リリースノート参照	
入力パラメータ:	データ型:	コメント:
xEnable	BOOL	ファンクションブロックを有効にする 初期設定=TRUE
bCOM_PORT_NR	BYTE	使用するシリアルポート番号 初期設定=2 1 → 内部サービスインターフェース 2 → 1 番目に接続されたシリアルモジュール 3 → 2 番目に接続されたシリアルモジュール
iNumberOfChannel	INT	送信チャンネル数 値の範囲: 1 – 254 初期設定: 45
xBlackOut	BOOL	TRUE → シャットダウンモード有効
入/出力パラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_MAX_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512
出力パラメータ:	データ型:	コメント:
xReady	BOOL	通信状態 TRUE=送信プロセスなし FALSE=送信プロセス有効
bError	BYTE	Sercom.lib 検出エラー 0x00 エラーなし 0x01 ファームウェアサポート外 0x02 COM ポート範囲外 0x03 ファンクションブロック、COM ポートインス

		<p>タンス未割当</p> <p>0x04 ファンクションブロック、COM ポート別インスタンス割当</p> <p>0x05 COM ポートが既に開放されている</p> <p>0x06 COM ポートが既に閉じられている</p> <p>0x07 開く COM ポートがない</p> <p>0x08 書込操作有効(COM1)</p> <p>0x09 送信パラメータ COM ポート未対応</p> <p>0x0A 通信ユニット設定読込不能</p> <p>0x0B ライブラリバージョン通信パラメータ未対応</p> <p>0x0C 通信ユニット識別不能</p> <p>0x0D 通信ユニットの FIFO メモリへのデータ書込エラー</p> <p>0x0E FIFO メモリ内容の送信がない(連続送信)</p> <p>0x0F 内部エラー</p>
--	--	---

図解:



機能内容:

ファンクションブロック **FbDMX_652_Master** は DMX 通信ラインにデータを送信するために使用されます。通信は RS-485 インターフェースモジュール 750-652 によって実行されます。このファンクションブロックは各シリアル機器に対してのみ使用されます。

データ列 **"abDMX_Values"** は DMX にて送信される値を含みます。各 DMX チャンネルがデータ列のインデックスです。DMX 信号は **"xEnable"** が TRUE にセットされると周期的に送信されます。

送信チャンネルの最大数は入力 **"iNumberOfChannel"** で制限することができます。

入力 **"xBlackOut"** は電源のシャットダウンモードを有効にします。

- **"xBlackOut" = TRUE** → 電源シャットダウンモード有効。すべての DMX チャンネルの値は 0 です。
- **"xBlackOut" = FALSE** → 電源シャットダウンモード無効。DMX レジスタ値は有効です。

フィールドバスコントローラは COM2 で始まる左側に接続されたシリアルモジュールのポート番号を検出および割り当します。コントローラのサービスインターフェースは常に COM1 です。希望するシリアルモジュールにファンクションブロックでアドレスを割り当てるためには、対応する番号(例. COM2 に対して"2")を入力"**bCOM_PORT_NR**"に定数として入力しなければなりません。

出力"**xReady**"はブロックが有効かどうか表示します。"**xReady**"が FALSE である限り、ファンクションブロックによりそれ以上のアクションはありません。

通信エラーの場合は、対応するエラーコードが"**bError**"に出力されます。

照明効果

RGB カラーミキサー(FbRGB_ColorMixer)

WAGO-I/O-PRO V2.3 ライブラリ要素		
カテゴリ:	ビルディング	
名称:	FbRGB_ColorMixer	
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/> Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib	
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib	
適用:	リリースノート参照	
入力パラメータ:	データ型:	コメント:
bValueRed	BYTE	DMX 値 チャンネル”赤”
bValueGreen	BYTE	DMX 値 チャンネル”緑”
bValueBlue	BYTE	DMX 値 チャンネル”青”
bValueIntensity	BYTE	DMX 値 チャンネル”明度”
iChannelRed	INT	チャンネル”赤”アドレス
iChannelGreen	INT	チャンネル”緑”アドレス
iChannelBlue	INT	チャンネル”青”アドレス
iChannelIntensity	INT	チャンネル”明度”アドレス
xWrite	BOOL	対応チャンネルに入力値をエッジ有効で書込む
xAutoWrite	BOOL	DMX 値が自動的に更新される
入/出力パラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_ MAX_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512
図解:		

```

FbRGB_ColourMixer
-bValueRed
-bValueGreen
-bValueBlue
-bValueIntensity
-iChannelRed
-iChannelGreen
-iChannelBlue
-iChannelIntensity
-xWrite
-xAutoWrite
-abDMX_Values ▶

```

機能内容:

ファンクションブロック **FbRGB_ColorMixer** は RGB カラーおよび明度を設定することができます。

それぞれのカラー要素は入力 **"bValueRed"**, **"bValueGreen"**, **"bValueBlue"**および**"bValueIntensity"**に割り当てられます。

相当する DMX チャンネルのアドレスは入力 **"iChannelRed"**, **"iChannelGreen"**, **"iChannelBlue"**および**"iChannelIntensity"**に割り当てられます。

入力 **"xWrite"**の立上りエッジで、値は DMX ラインに転送されます。

入力変数 **"xAutoWrite"**が TRUE にセットされると、入力 **"bValueRed"**, **"bValueGreen"**, **"bValueBlue"**および**"bValueIntensity"**は値の変化をモニタできます。値が変化するとすぐに、この変化が DMX ラインに転送されます。

ファンクションブロックは通信モジュールとの接続において使用されます。(6 ページ参照) 2 要素の同期はデータ列 **"abDMX_values"**によって有効にします。

したがって、通信モジュールとファンクションブロックは相互接続されている必要があります。この接続を通じて、ファンクションブロックの DMX ラインに値を書き込むことができます。

RGB カラーパレット保存(FbRGB_SaveColourPalette)

WAGO-I/O-PRO ライブラリ要素			
カテゴリ:	ビルディング		
名称:	FbRGB_SaveColourPalette		
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/>	Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib		
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib		
適用:	リリースノート参照		
入力パラメータ:	データ型:	コメント:	
bValueRed	BYTE	DMX 値 チャンネル”赤”	
bValueGreen	BYTE	DMX 値 チャンネル”緑”	
bValueBlue	BYTE	DMX 値 チャンネル”青”	
xColour_1	BOOL	エッジ有効で adwColourPalette[1]の色組合せ保存	
xColour_2	BOOL	エッジ有効で adwColourPalette[2]の色組合せ保存	
xColour_3	BOOL	エッジ有効で adwColourPalette[3]の色組合せ保存	
xColour_4	BOOL	エッジ有効で adwColourPalette[4]の色組合せ保存	
xColour_5	BOOL	エッジ有効で adwColourPalette[5]の色組合せ保存	
xColour_6	BOOL	エッジ有効で adwColourPalette[6]の色組合せ保存	
xColour_7	BOOL	エッジ有効で adwColourPalette[7]の色組合せ保存	
xColour_8	BOOL	エッジ有効で adwColourPalette[8]の色組合せ保存	
xColour_9	BOOL	エッジ有効で adwColourPalette[9]の色組合せ保存	
xColour_10	BOOL	エッジ有効で adwColourPalette[10]の色組合せ保存	
xReset	BOOL	エッジ有効で adwColourPalette の全内容を削除	

入/出力パラメータ:	データ型:	コメント:
adwColourPalette	ARRAY[1..10] of DWORD	色組合せのデータ列

図解:



機能内容:

ファンクションブロック **FbRGB_SaveColourPalette** で 10 種の色の組み合わせを保存することができます。

それぞれの色の組み合わせは入力 **"bValueRed"**, **"bValueGreen"** および **"bValueBlue"** により構成できます。

入力 **"xColour_1"** から **"xColour_10"** のエッジ有効で、相当するデータ列要素 **"adwColourPalette"** に色の組み合わせを保存します。

データ列 **"adwColourPalette"** に色の組み合わせが保存されます。表現方法は B(青)、G(緑)、R(赤)の順の 16 進数です。例えば、黄色はこの順番では値 **16#00FFFF** で白色は値 **16#FFFFFF** となります。

入力 **"xReset"** の立上りエッジでデータ列 **"adwColourPalette"** の内容を削除することができます。

注意:

入力 **"adwColourPalette"** 変数は **VAR RETAIN PERSISTENT** として定義すべきで、したがって色の組み合わせリストはコントローラのリセット後、ダウンロード後でさえそのままです。

RGB カラーパレット呼び出し(FbRGB_RecallColourPalette)

WAGO-I/O-PRO ライブラリ要素			
カテゴリ:	ビルディング		
名称:	FbRGB_RecallColourPalette		
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/>	Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib		
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib		
適用:	リリースノート参照		
入力パラメータ:	データ型:	コメント:	
adwColourPalette	ARRAY[1..10] of DWORD	色組合せのデータ列	
iChannelRed	INT	チャンネル"赤"のアドレス	
iChannelGreen	INT	チャンネル"緑"のアドレス	
iChannelBlue	INT	チャンネル"青"のアドレス	
xRecallColour_1	BOOL	adwColourPalette[1]の色組合せをエッジ有効で呼び出す	
xRecallColour_2	BOOL	adwColourPalette[2]の色組合せをエッジ有効で呼び出す	
xRecallColour_3	BOOL	adwColourPalette[3]の色組合せをエッジ有効で呼び出す	
xRecallColour_4	BOOL	adwColourPalette[4]の色組合せをエッジ有効で呼び出す	
xRecallColour_5	BOOL	adwColourPalette[5]の色組合せをエッジ有効で呼び出す	
xRecallColour_6	BOOL	adwColourPalette[6]の色組合せをエッジ有効で呼び出す	
xRecallColour_7	BOOL	adwColourPalette[7]の色組合せをエッジ有効で呼び出す	
xRecallColour_8	BOOL	adwColourPalette[8]の色組合せをエッジ有効で呼び出す	
xRecallColour_9	BOOL	adwColourPalette[9]の色組合せをエッジ有効で呼び出す	
xRecallColour_10	BOOL	adwColourPalette[10]の色組合せをエッジ有効で呼び出す	

入/出力パラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_MAX_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512

図解:



機能内容:

ファンクションブロック **FbRGB_RecallColourPalette** でデータ列 **adwColourPalette** から保存された色の組み合わせを呼び出すことができます。

入力 **adwColourPalette** はファンクションブロック **FbRGB_SaveColourPalette** に対して同じ変数に接続することや、保存された色の組み合わせを収容することができます。

表現方法は B(青)、G(緑)、R(赤)の順の 16 進数です。例えば、黄色はこの順番では値 **16#00FFFF** で白色は値 **16#FFFFFF** となります。

対応する DMX チャンネルのアドレスは入力 **iChannelRed**、**iChannelGreen** および **iChannelBlue** に割り当てられます。

入力 **xRecallColour_1** から **xRecallColour_10** のエッジ有効で、データ列 **adwColourPalette** の相当要素の色の組み合わせを呼び出します。

ファンクションブロックは通信モジュールとの接続において使用されます。(6 ページ参照) 2 要素の同期はデータ列 **abDMX_values** によって有効にします。したがって、通信モジュールとファンクションブロックは相互接続されている必要があります。この接続を通じて、ファンクションブロックの DMX ラインに値を書き込むことができます。

フェードシーケンス(FbDMX_FadeSequence)

WAGO-I/O-PRO ライブラリ要素		
カテゴリ:	ビルディング	
名称:	FbDMX_FadeSequence	
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/> Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib	
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib	
適用:	リリースノート参照	
入力パラメータ:	データ型:	コメント:
xEnable	BOOL	照明シーケンス周期有効
iChannel	INT	DMX チャンネルのアドレス
tPeriod	TIME	周期 最低: 1s 初期設定: 5s
bMaximumLevel	BYTE	最大 DMX チャンネル値 初期設定: 255
xTriangle	BOOL	三角関数 初期設定: TRUE
xSquare	BOOL	脈動信号
xSawtoothRise	BOOL	鋸波型上昇
xSawtoothFall	BOOL	鋸波型下降
入/出力パラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_MAX_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512
図解:		
<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <p style="text-align: center;">FbDMX_FadeSequence</p> <ul style="list-style-type: none"> -xEnable -iChannel -tPeriod -bMaximumLevel -xTriangle -xSquare -xSawtoothRise -xSawtoothFall -abDMX_Values ▶ </div>		

機能内容:

ファンクションブロック **FbDMX_FadeSequence** で照明シーケンス周期を生成します。"xEnable"でシーケンスは有効にされます。

入力 "tPeriod" は照明シーケンスの周期を定義します。

入力 "bMaximumValue" は照明シーケンスの最大到達値を定義します。

以下の変数が TRUE にセットされる場合、機能が生成されます:

1. "xTriangle"-三角関数
2. "xSquare"-脈動信号
3. "xSawtoothRise"-鋸波型上昇
4. "xSawtoothFall"-鋸波型下降

ファンクションブロックは通信モジュールとの接続において使用されます。(6 ページ参照) 2 要素の同期はデータ列 "abDMX_values" によって有効にします。したがって、通信モジュールとファンクションブロックは相互接続されている必要があります。この接続を通じて、ファンクションブロックの DMX ラインに値を書き込むことができます。

クロスフェード(FbDMX_CrossFadeSequence)

WAGO-I/O-PRO ライブラリ要素		
カテゴリ:	ビルディング	
名称:	FbDMX_CrossFadeSequence	
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/> Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib	
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib	
適用:	リリースノート参照	
入力パラメータ:	データ型:	コメント:
xEnable	BOOL	グラデーション有効
iChannelRed	INT	チャンネル"赤"のアドレス
iChannelGreen	INT	チャンネル"緑"のアドレス
iChannelBlue	INT	チャンネル"青"のアドレス
tDelay	TIME	送信周期 最低: 1s 初期設定: 5s
xToAndFro	BOOL	グラデーションアップ/ダウン
iNumberOfColours	INT	色彩変遷数 値範囲: 1-10 初期設定: 10
dwColour_1	DWORD	1. フェード
dwColour_2	DWORD	2. フェード
dwColour_3	DWORD	3. フェード
dwColour_4	DWORD	4. フェード
dwColour_5	DWORD	5. フェード
dwColour_6	DWORD	6. フェード
dwColour_7	DWORD	7. フェード
dwColour_8	DWORD	8. フェード
dwColour_9	DWORD	9. フェード
dwColour_10	DWORD	10. フェード
入/出力パラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_MA X_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512

図解:

```

FbRGB_CrossFadeSequence
-xEnable
-iChannelRed
-iChannelGreen
-iChannelBlue
-tDelay
-xToAndFro
-iNumberOfColours
-dwColour_1
-dwColour_2
-dwColour_3
-dwColour_4
-dwColour_5
-dwColour_6
-dwColour_7
-dwColour_8
-dwColour_9
-dwColour_10
-abDMX_Values ▶
    
```

機能内容:

ファンクションブロック **FbDMX_CrossFadeSequence** で動的な色彩グラデーションを生成します。変数 **"xEnable"** はファンクションブロックを有効にします。

対応する DMX チャンネルのアドレスは入力 **"iChannelRed"**, **"iChannelGreen"** および **"iChannelBlue"** に割り当てられます。

色彩間の変遷呼び出しはパラメータ **"tDelay"** により定義されます。

入力 **"dwColour_1"** から **"dwColour_10"** はグラデーションを構成することができます。表現方法は B(青)、G(緑)、R(赤)の順の 16 進数です。例えば、黄色はこの順番では値 **16#00FFFF** で白色は値 **16#FFFFFF** となります。

パラメータ **"iNumberOfColours"** で変遷色が定義されます。

入力 **"xToAndFro"** で TRUE は常に前後で実行されているグラデーションを有効にします。グラデーションが変遷色の最大数に達する後で、最初からやり直したい場合は FALSE を入力に設定する必要があります。

ファンクションブロックは通信モジュールとの接続において使用されます。(6 ページ参照) 2 要素の同期はデータ列 **"abDMX_values"** によって有効にします。したがって、通信モジュールとファンクションブロックは相互接続されている必要があります。この接続を通じて、ファンクションブロックの DMX ラインに値を書き込むことができます。

チェーサー(FbDMX_Chaser)

WAGO-I/O-PRO ライブラリ要素		
カテゴリ:	ビルディング	
名称:	FbDMX_Chaser	
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/> Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib	
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib	
適用:	リリースノート参照	
入力パラメータ:	データ型:	コメント:
xEnable	BOOL	ファンクションブロック有効
bChannelValue	BYTE	DMX チャンネル数
iStartChannel	INT	開始チャンネル 最小: 1 初期設定: 1
iEndChannel	INT	終了チャンネル 最小: 2 初期設定: 2
iOffset	INT	増分 最小: 1 初期設定: 1
tDelay	TIME	送信周期 最小: 50ms 初期設定: 1s
入/出カパラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_MAX_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512
図解:		
 <pre> graph TD subgraph FbDMX_Chaser xEnable bChannelValue iStartChannel iEndChannel iOffset tDelay abDMX_Values end </pre>		

機能内容:

ファンクションブロック **FbDMX_Chaser** は DMX チャンネルの予定数に対して DMX チャンネル(A)から異なる DMX チャンネル(B)に DMX 値をコピーします。チャンネル(A)の値は 0 にリセットされます。このファンクションブロックを使用すると、実行している照明効果を処理することができます。

変数 **"xEnable"** はファンクションブロックを有効にします。入力 **"xEnable"** の立下りエッジでこのブロックの DMX チャンネルよるすべての影響が 0 にリセットされます。

パラメータ **"bChannelValue"** にコピーする DMX 値が設定されます。

コピー操作を開始する DMX チャンネルは、入力 **"iStartChannel"** に割り当てます。DMX 値のコピーは DMX チャンネル **"iEndChannel"** で終了します。変数 **"iOffset"** は異なる DMX チャンネルへのコピーの増分を定義します。

パラメータ **"tDelay"** は各ステップ間でどのくらい待機するかを指定します。

ファンクションブロックは通信モジュールとの接続において使用されます。(6 ページ参照) 2 要素の同期はデータ列 **"abDMX_values"** によって有効にします。したがって、通信モジュールとファンクションブロックは相互接続されている必要があります。この接続を通じて、ファンクションブロックの DMX ラインに値を書き込むことができます。

チャンネル変更(FbDMX_ChangeChValues)

WAGO-I/O-PRO ライブラリ要素		
カテゴリ:	ビルディング	
名称:	FbDMX_ChangeChValues	
形式:	Function <input type="checkbox"/>	Function block <input checked="" type="checkbox"/> Program <input type="checkbox"/>
ライブラリ名:	DMX_01.lib	
使用ライブラリ:	SerComm.lib, Serial_Interface_01.lib, Util.lib	
適用:	リリースノート参照	
入力パラメータ:	データ型:	コメント:
xEnable	BOOL	ファンクションブロック有効
bChannelValue_1st	BYTE	DMX 値 1
iChannel_1st	INT	DMX チャンネルアドレス 1
bChannelValue_2nd	BYTE	DMX 値 2
iChannel_2nd	INT	DMX チャンネルアドレス 2
bChannelValue_3rd	BYTE	DMX 値 3
iChannel_3rd	INT	DMX チャンネルアドレス 3
bChannelValue_4th	BYTE	DMX 値 4
iChannel_4th	INT	DMX チャンネルアドレス 4
bChannelValue_5th	BYTE	DMX 値 5
iChannel_5th	INT	DMX チャンネルアドレス 5
入/出力パラメータ:	データ型:	コメント:
abDMX_Values	ARRAY[1..DMX_MAX_CH] of BYTE	DMX 値のデータ列 DMX_MAX_CH=512
出力パラメータ:	データ型:	コメント:
xReady	BOOL	通信状態 TRUE=ファンクションブロック・有効 FALSE=ファンクションブロック・ビジー
図解:		

```

FbDMX_ChangeChValues
-xEnable xReady
-bChannelValue_1st
-iChannel_1st
-bChannelValue_2nd
-iChannel_2nd
-bChannelValue_3rd
-iChannel_3rd
-bChannelValue_4th
-iChannel_4th
-bChannelValue_5th
-iChannel_5th
-abDMX_Values ▶

```

機能内容:

ファンクションブロック **FbDMX_ChangeChValue** では最大 5 つの DMX チャンネルの DMX 値をセットすることができます。変数"xEnable"でファンクションブロックを有効にします。

値の変更が入力"**bChannelValue_1st**"から"**bChannelValue_5th**"で指定する場合に DMX 値が送信されます。

5 つの DMX チャンネルのアドレスは入力"**bChannelValue_1st**" から"**bChannelValue_5th**"に割り当てられます。

ファンクションブロックは通信モジュールとの接続において使用されます。(6 ページ参照) 2 要素の同期はデータ列"**abDMX_values**"によって有効にします。したがって、通信モジュールとファンクションブロックは相互接続されている必要があります。この接続を通じて、ファンクションブロックの DMX ラインに値を書き込むことができます。

