

RL78 ファミリ

SIS を用いた MIDI 連動イルミネーション制御サンプルソフトウェア

要旨

本アプリケーションノートでは、MIDI インタフェース SIS (Software Integration System) モジュールを用いた MIDI 機器との通信制御の使用例を示します。

MIDI インタフェース SIS モジュールを利用して PC から流れてくる MIDI メロディ(=MIDI メッセージ)に合わせて、マトリクス LED の表示制御を行います。MIDI メッセージは音源モジュールに転送して演奏させることもできます。

対応規格は以下のとおりです。

- ・ MIDI 1.0

MIDI 規格の詳細については上記規格書を参照してください。

動作確認デバイス

RL78/G16

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

目次

1. 仕様	3
1.1 仕様概要	3
1.1.1 通信仕様	4
1.2 動作詳細	6
2. ハードウェア説明	9
2.1 ハードウェア構成	9
2.2 端子結線概略図	11
2.3 使用端子一覧	12
3. ソフトウェア説明	13
3.1 ソフトウェア環境	13
3.2 周辺機能の設定内容	13
3.3 オプション・バイトの設定一覧	18
3.4 マクロ一覧	19
3.5 定数一覧	20
3.6 変数一覧	21
3.7 関数一覧	23
3.8 関数仕様	24
3.9 フローチャート	27
3.9.1 メイン処理	27
3.9.2 ボリュームスイッチから表示対象の MIDI チャンネル決定処理	29
3.9.3 ボリュームスイッチ入力取得処理	30
3.9.4 マトリクス LED 表示処理	31
3.9.5 システムタイマ取得処理	33
3.9.6 システムタイマカウント処理	33
3.9.7 マトリクス LED データ表示サイズ決定処理	34
3.9.8 マトリクス LED データ送信処理	35
3.9.9 マトリクス LED データ送信完了チェック処理	36
3.9.10 TAU0_3 割り込み処理	36
3.9.11 AD 変換完了割り込み処理	37
3.9.12 マトリクス LED データ CSI 送信完了処理	37
3.9.13 UART0 送信完了処理	38
3.9.14 UART0 受信完了処理	38
4. DAW の設定方法	39
5. サンプルプログラム	41
6. 参考ドキュメント	42
改訂記録	44

1. 仕様

1.1 仕様概要

本サンプルプログラムでは、MIDIインタフェースSIS (Software Integration System) モジュールを使用して、MIDIメッセージ入出力機能の使用例を示します。

受信機能を利用してPCから流れてくるMIDIメロディ(=MIDIメッセージ)に合わせて、マトリクスLEDの表示制御を行います。MIDIメッセージは送信機能を利用してMIDIOUT端子から音源モジュールに転送（スルー）して演奏させることができます。

表 1-1に使用する周辺機能と用途を示し、図 1-1にサンプルプログラムの動作概要を示します。

表 1-1 使用する周辺機能と用途

周辺機能	用途
シリアル・インタフェース UART0 P03/TxD0、P04/RxD0	MIDI 機器との UART 通信
A/D コンバータ P05/ANI4	表示チャンネル選択用のボリュームスイッチ入力
シリアル・アレイ・ユニット CSI20 P13/SCK20、P16/GPIO、P15/SO20	マトリクス LED モジュールとの SPI 通信

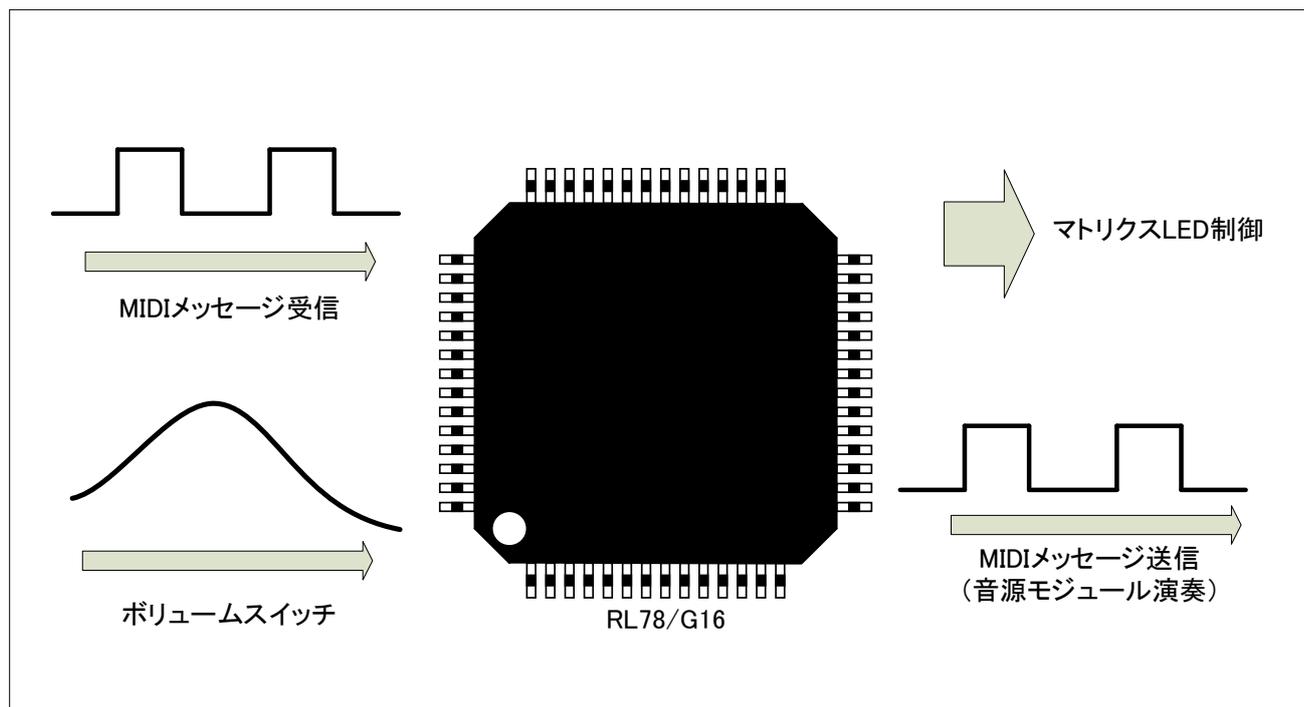


図 1-1 サンプルプログラムの動作概要

1.1.1 通信仕様

本サンプルプログラムにて使用する MIDI 規格のデータ構成について記載します。

MIDI データ列は図 1-2 に示すとおり、各バイトが 10 ビット（1 スタート・ビット、8 データ・ビット、1 ストップ・ビット）で送信される、31.25K ビット/ 秒の単方向非同期のビット列です。

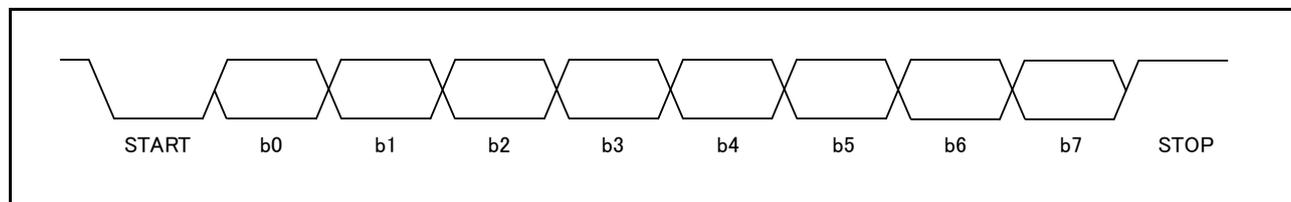


図 1-2 MIDI データ列

MIDI メッセージは図 1-3 に示すとおり、ステータスバイトとデータバイトで構成され、ステータスバイトによってチャンネルメッセージとシステムメッセージに大別されます。

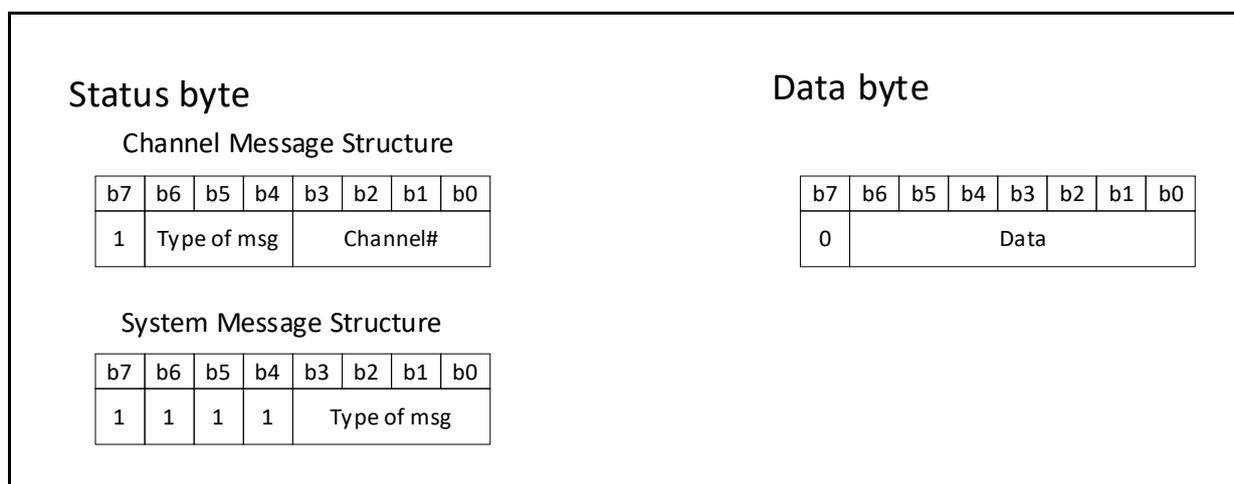


図 1-3 MIDI データ構造

本サンプルプログラムでマトリクス LED の制御に用いるチャンネルメッセージである NoteOn(打鍵)メッセージについて説明します。

NoteOn(打鍵)メッセージは図 1-4 に示すとおり、ステータスバイトとそれにつづく 2つのデータバイトで構成されます。

ステータスバイトにはチャンネル番号(Channel#)が含まれています。

データバイト 1 には音階を表すノート・ナンバー(Note#)が含まれています。

データバイト 2 には音の強さを表すベロシティ(Velocity)が含まれています。

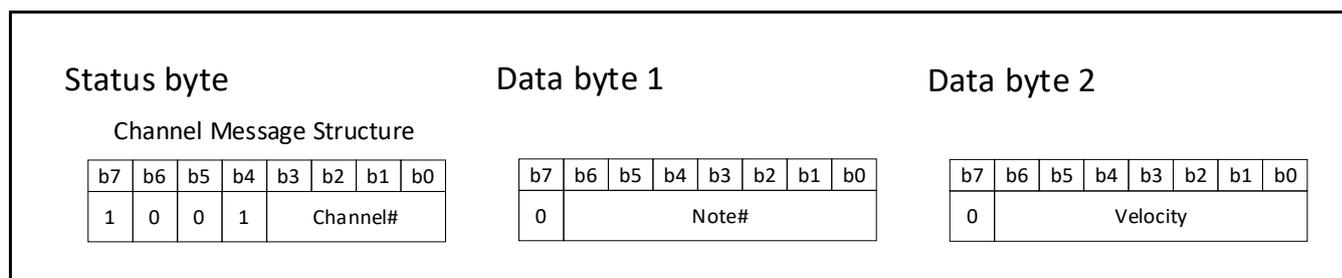


図 1-4 NoteOn(打鍵)メッセージ

1.2 動作詳細

本サンプルプログラムでは、PCから送信したMIDIメッセージを受け取りNoteOn(打鍵)メッセージであれば、その内容に従ってマトリクスLEDを光らせます。また、NoteOn(打鍵)メッセージを含むその他のMIDIメッセージは音源モジュールに転送して演奏させます。

PCはDAWから任意のMIDIファイルを再生することで、MIDIメッセージを送信します。

DAW : Digital Audio Workstation(デジタルオーディオワークステーション)

PC上で作曲や録音、ミックスなど幅広く制作を行うためのソフトウェアです。

(1) 表示対象チャンネルの設定、変更

- MIDI規格として1~16のチャンネル(和音の意味)があり、これをMIDIチャンネルと呼びます。
- 特定の1つのMIDIチャンネル、または全てのMIDIチャンネルを同時に表示することができます。(以降、特定の1つのMIDIチャンネルを表示するモードを“特定MIDIチャンネル表示モード”、すべてのMIDIチャンネルを同時に表示するモードを“全MIDIチャンネル同時表示モード”と呼びます)
- 表示する対象のMIDIチャンネルのことを“表示対象チャンネル”と呼びます。
- 上記のモードは、MIDIシールドのVolume(A0)の状態によって決定します。
- ユーザはVolume(A0)を回して表示対象チャンネルを変更することができます。(メロディ再生中も変更可能です)
- Volume(A0)を回すと、マトリクスLEDに表示対象チャンネル情報を1秒間表示します。
- “特定MIDIチャンネル表示モード”を選択した場合は“Ch 01” ~ “Ch 16”を表示します。
- “Ch 16”からさらにVolume(A0)を回すと、“全MIDIチャンネル同時表示モード”を選択できます。
- “全MIDIチャンネル同時表示モード”が選択された場合は”Ch All”と表示します。
- 表示対象チャンネル表示中のマトリクスLED表示内容は図1-5 MIDIチャンネル表示一覧のとおりです。(黒は消灯、赤と緑で点灯)



図 1-5 MIDIチャンネル表示一覧

(2) メロディ表示中(特定 MIDI チャンネル表示モード)のマトリクス LED 表示

- 受信した MIDI メッセージのうち、NoteOn(打鍵)メッセージに含まれる”音階”・”チャンネル番号”・”Velocity” 情報をもとにマトリクス LED を点灯します。すなわち音の鳴り始めのタイミングでマトリクス LED を点灯します。
- 音階(ド・レ・ミ・ファ・ソ・ラ・シ)ごとに色を図 1-6 のとおりに割り当てます。ピアノで例えると、白鍵の打鍵メッセージを受信した場合は 1 色で、黒鍵の打鍵メッセージを受信した場合は 2 色で表現しています。

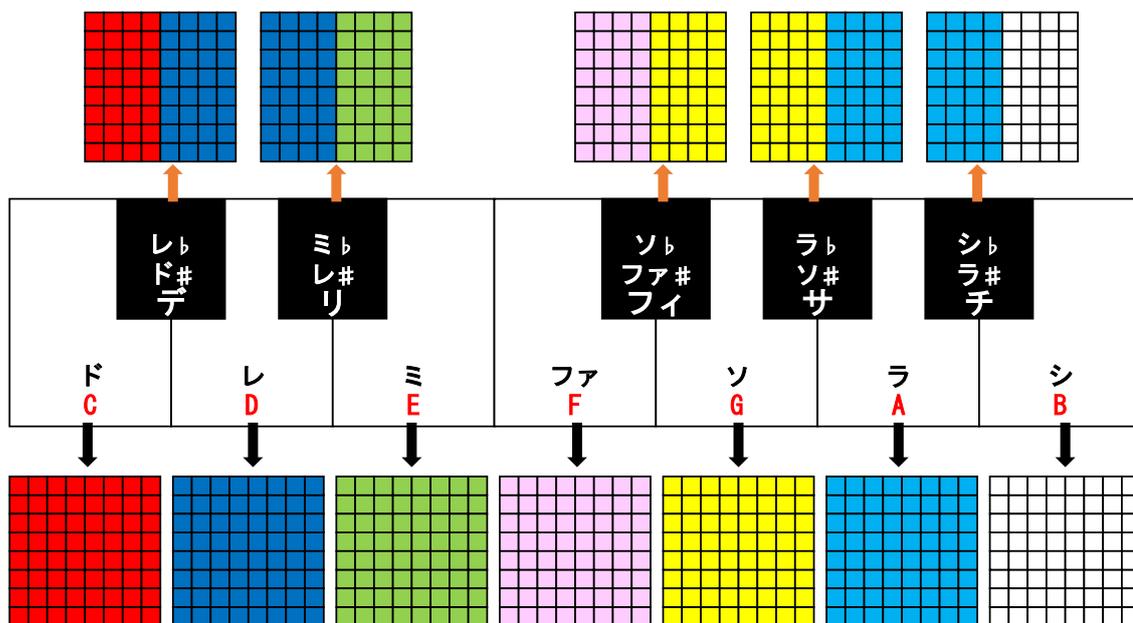


図 1-6 音階ごとの表示色(特定 MIDI チャンネル表示モード)

- 図 1-7 に示すとおり、表示対象チャンネルの NoteOn(打鍵)メッセージを受信するごとに、音階ごとに割り当てた色を表示し、一定時間ごとに表示範囲を徐々に小さくしていきます。
- 表示開始の際、NoteOn(打鍵)メッセージに含まれる Velocity 値を参照して、大きさを決定します。
図 1-7 に示すとおり、“ファ#” NoteOn メッセージの Velocity 値は最大値のため、最大サイズから表示開始しており、“ラ#” NoteOn メッセージでは Velocity 値は小さいため、一回り小さい範囲から表示開始しています。
- 輝度調整は行わず、明るさは一定です。

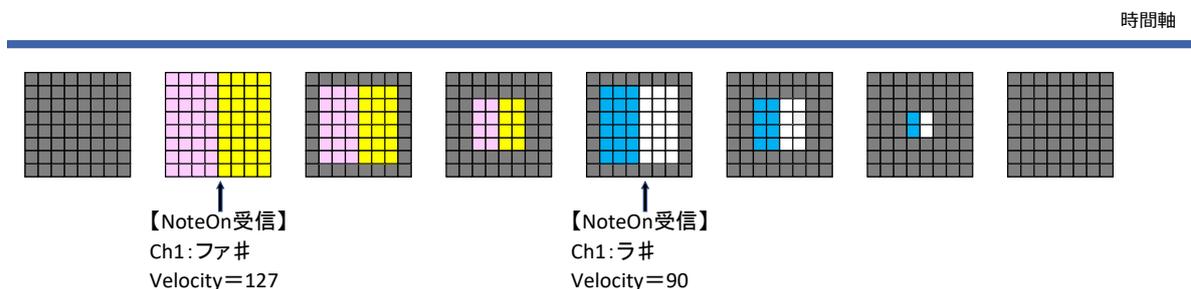


図 1-7 表示例(特定 MIDI チャンネル表示モード)

(3) メロディ表示中(全 MIDI チャンネル同時表示モード)のマトリクス LED 表示

- 8x8 あるマトリクス LED を 2x2 の 16 ブロックに小分けして、16 個の MIDI チャンネルを割り当てます。
- 受信した MIDI メッセージのうち、NoteOn(打鍵)メッセージに含まれる”音階”・”チャンネル番号”情報をもとにマトリクス LED の該当ブロックを点灯します。”チャンネル番号”にもとづくブロックは図 1-8 のとおりです。

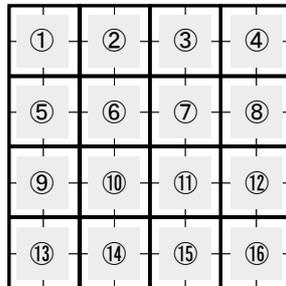


図 1-8 MIDI チャンネル別表示位置割り当て

- 各チャンネルの音階による色の割り当ては特定 MIDI チャンネル表示モードと同じとし、図 1-9 のとおり、表示枠が縮小されたものとしています。

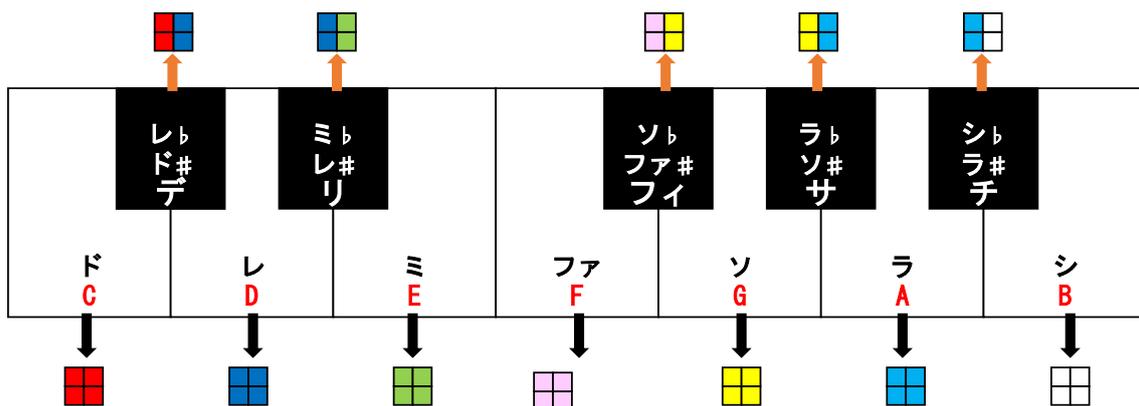


図 1-9 音階ごとの表示色(全 MIDI チャンネル同時表示モード)

- 図 1-10 に示すとおり、NoteOn(打鍵)メッセージを受信するごとに、MIDI チャンネルごとに割り当てたマトリクスのブロックに対して、音階ごとに割り当てた色を明るく表示し、徐々に暗くしていきます。

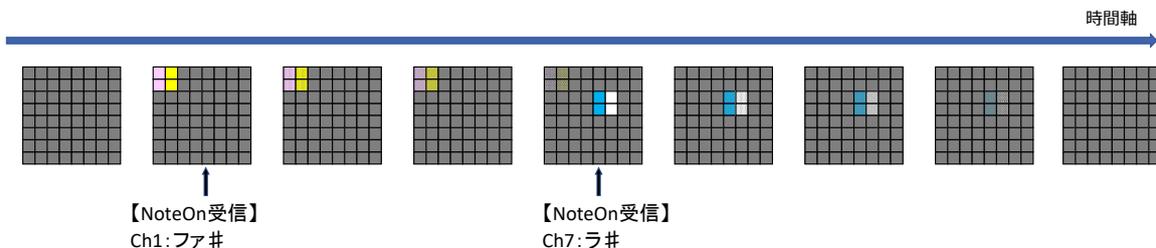


図 1-10 表示例(全 MIDI チャンネル同時表示モード)

2. ハードウェア説明

2.1 ハードウェア構成

表 2-1 に本サンプルプログラムで使用するハードウェアを示します。

表 2-1 ハードウェア一覧

ハードウェア	用途
使用ボード	ルネサスエレクトロニクス製 RL78/G16 Fast Prototyping Board (RTK5RLG160C00000BJ)
使用マイコン	RL78/G16 (R5F121BCAFP)
動作周波数	高速オンチップ・オシレータ・クロック (fIH): 16MHz
動作電圧	5.0V
MIDI シールドボード	SparkFun 製 MIDI Shield
MIDItoUSB ケーブル	Roland 製 UM-ONE
MIDItoMIDI(オス-オス)ケーブル	サンワサプライ製 KB-MID01-18K
MIDI 音源モジュール	ローランド製 SOUND Canvas SC-88 Pro
マトリクス LED モジュール	52pi 製 EP-0075 RPI-RGB-LED-Matrix

図 2-1 と図 2-2 に本アプリケーションノートで使用する構成を示します。

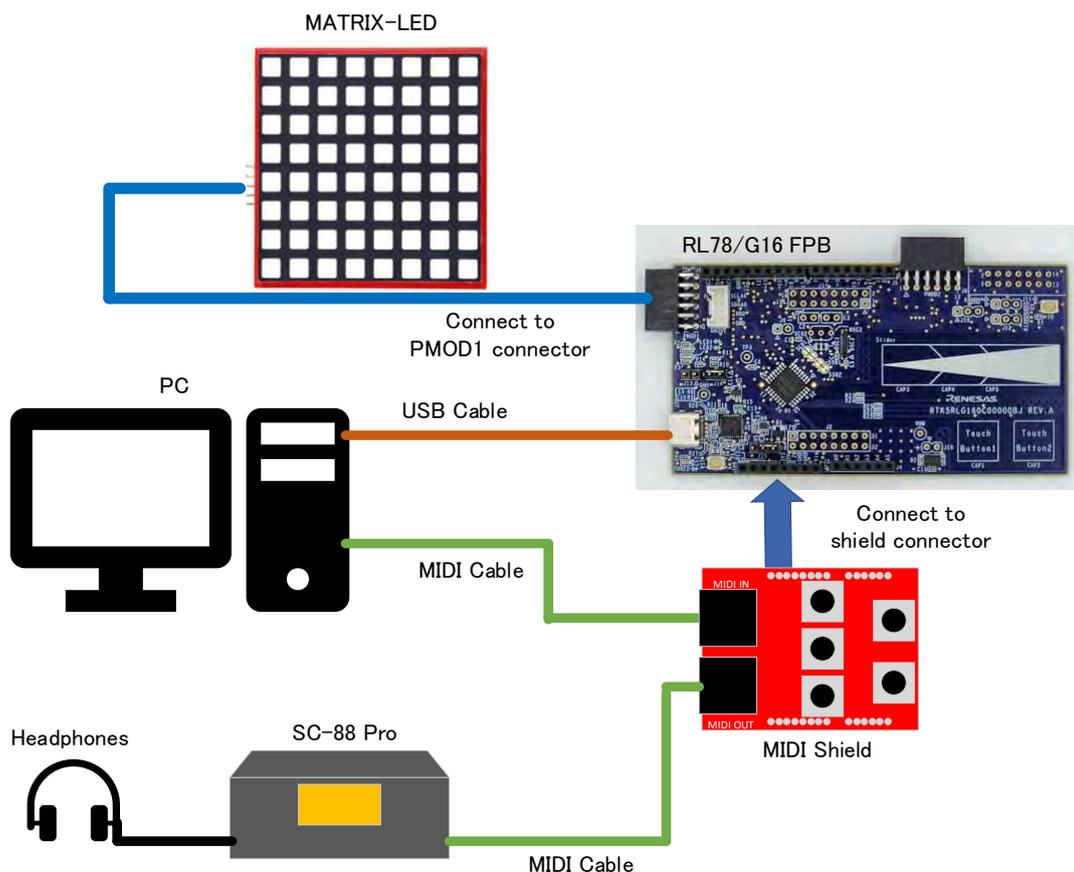


図 2-1 ハードウェア構成

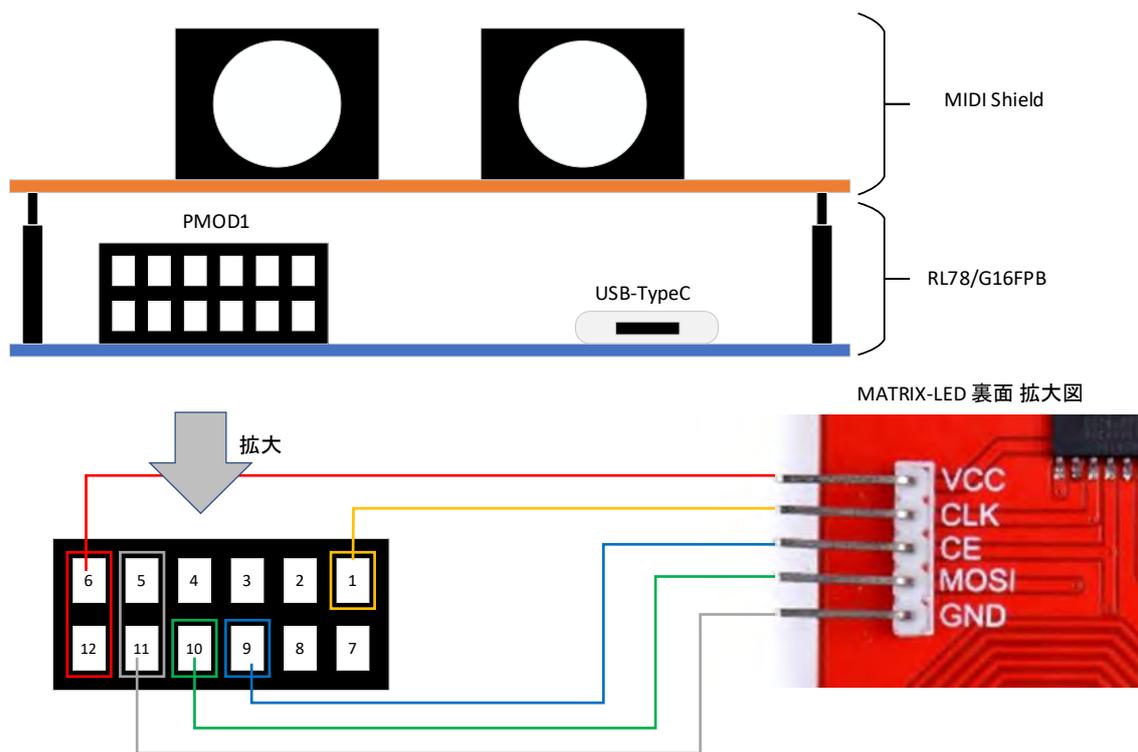


図 2-2 PMOD1 と MATRIX-LED 間の配線

2.2 端子結線概略図

図 2-3 と図 2-4 に MIDI Shield と MATRIX-LED の RL78/G16 FPB との端子結線概略図を示します。

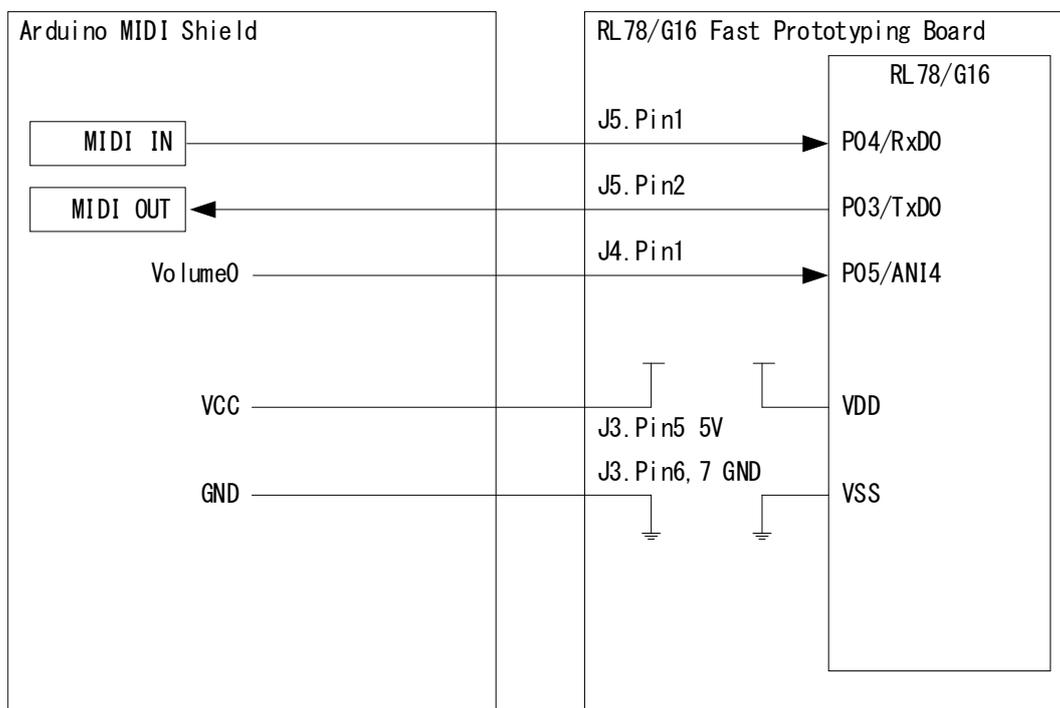


図 2-3 RL78/G16 FPB と MIDI Shield の端子結線概略図

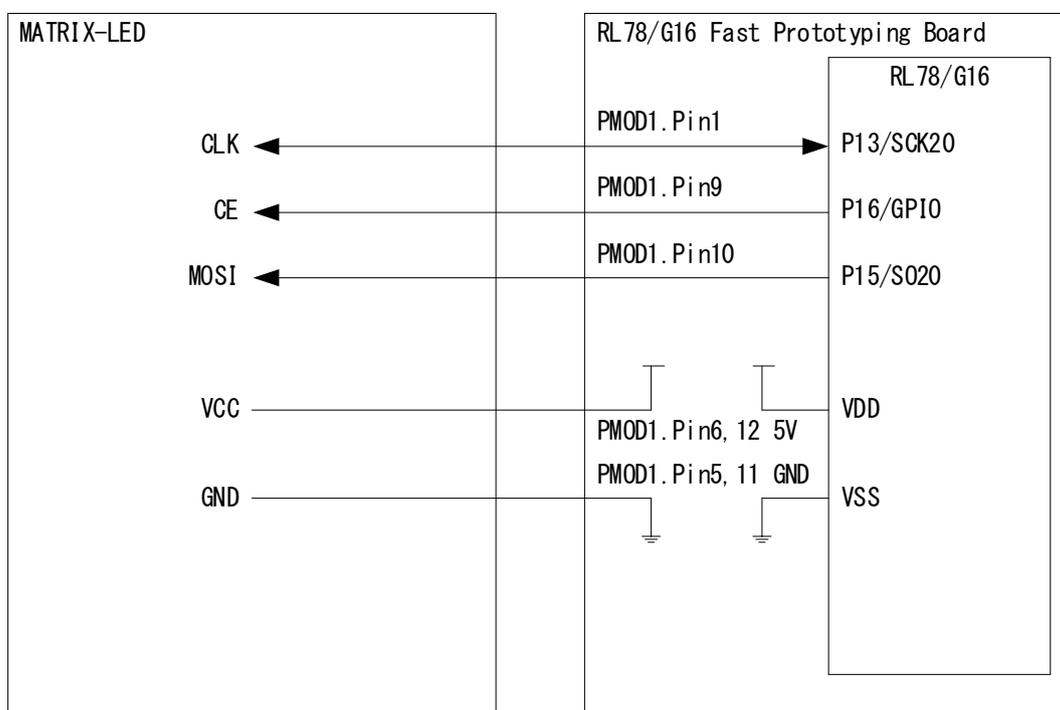


図 2-4 RL78/G16 FPB と MATRIX-LED の端子結線概略図

2.3 使用端子一覧

表 2-2 に使用端子と機能を示します。

表 2-2 使用端子と機能

端子名	入出力	内容
P03/TxD0	出力	MIDI メッセージ送信
P04/RxD0	入力	MIDI メッセージ受信
P05/ANI4	入力	MIDI チャンネル選択
P13/SCK20	出力	MATRIX-LED SPI クロック
P16/GPIO	出力	MATRIX-LED SPI チップセレクト
P15/SO20	出力	MATRIX-LED SPI MOSI

注意 本アプリケーションノートは、使用端子のみを端子処理しています。実際に回路を作成される場合は、「RL78/G16 ユーザーズマニュアル ハードウェア編 (R01UH0980)」の「2.3 未使用端子の処理」を参照し、本アプリケーションで使用していない端子の端子処理を適切に行い、電気的特性を満たすように設計してください。

3. ソフトウェア説明

3.1 ソフトウェア環境

表 3-1 に本サンプルプログラムで使用するソフトウェアを示します。

表 3-1 ソフトウェア一覧

ソフトウェア	内容
統合開発環境	ルネサスエレクトロニクス製 e ² studio 2024-04
C コンパイラ	ルネサスエレクトロニクス製 C Compiler Package for RL78 Family [CC-RL] V1.13.00
スマート・コンフィグレータ (SC)	Smart Configurator for RL78 V1.10.0
ボードサポートパッケージ (BSP)	ルネサス エレクトロニクス製 V1.62
SIS モジュール	MIDI インタフェースモジュール

注意 MIDI インタフェースモジュールは最新版をご使用ください。
再度 MIDI インタフェースモジュールを追加することで最新版をご使用いただけます。
追加方法は以下資料をご参考ください。

e2studio で追加する場合 : RL78 スマート・コンフィグレータ ユーザーガイド : e2studio 編 (R20AN0579)
IAR で追加する場合 : RL78 スマート・コンフィグレータ ユーザーガイド : IAR 編 (R20AN0581)

3.2 周辺機能の設定内容

図 3-1 に 1 ms周期のインターバルタイマの設定を示します。

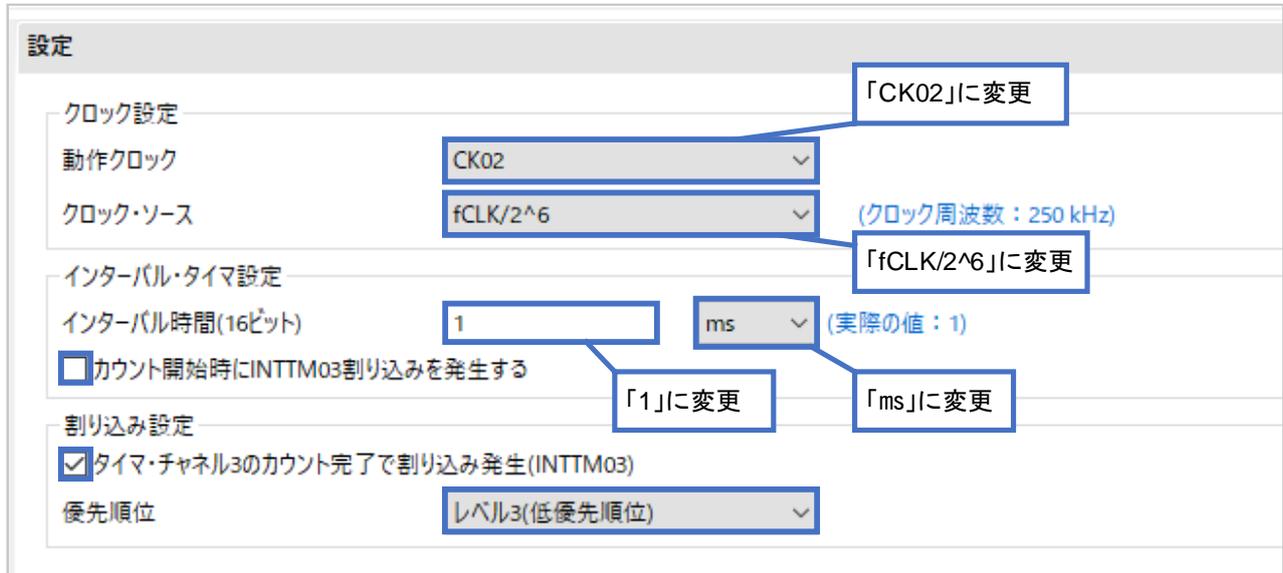


図 3-1 1 ms周期のインターバルタイマの設定

図 3-2 に 4 ms 周期のインターバルタイマの設定を示します。

図 3-2 4 ms 周期のインターバルタイマの設定

図 3-3 にアナログ入力チャンネルは 4 で、変換結果が 10bit データの A/D コンバータ設定を示します。

図 3-3 アナログ入力の設定

図 3-4 にデータ転送方向は MSB ファーストで、通信速度が 4Mbps の SPI 通信設定を示します。

設定

転送クロック設定

転送クロックモード: 内部クロック(マスタ) ▼

動作クロック: CK10 ▼

クロック・ソース: fCLK ▼ (クロック周波数: 16000 kHz)

転送モード設定

シングル転送モード 連続転送モード

データ・ビット長設定

8ビット 7ビット

データ転送方向設定

LSB MSB

「MSB」に変更

データ送受信タイミング設定
(下図はデータ転送方向がMSBの場合)

タイプ1

タイプ2

タイプ3

タイプ4

転送レート設定

ポーレート: 4000000 ▼ (bps) (実際の値: 4000000)

「4000000」に変更

割り込み設定

通信完了割り込み優先順位(INTCSI20): レベル3(低優先順位) ▼

コールバック機能設定

送信完了 オーバラン・エラー

図 3-4 SPI の通信設定

図 3-5 に MIDI 通信規格に準じた UART 送信の通信設定を示します。

設定

送信 受信

UART0クロック設定

動作クロック CK00

クロック・ソース fCLK/2⁵ (クロック周波数: 500 kHz)

転送モード設定

シングル転送モード 連続転送モード

データ・ビット長設定

7ビット 8ビット 9ビット

データ転送方向設定

LSB MSB

パリティ設定

パリティ・ビットなし 0パリティ 奇数パリティ 偶数パリティ

ストップ・ビット長設定

1ビット 2ビット

送信データ・レベル設定

非反転(通常) 反転

転送レート設定

転送レート設定 31250 (bps) (誤差: 0%)

割り込み設定

送信完了割り込み設定(INTST0) レベル3(低優先順位)

コールバック機能設定

送信完了

「fCLK/2⁵」に変更

「31250」に変更

図 3-5 MIDI 送信の通信設定

図 3-6 に MIDI 通信規格に準じた UART 受信の通信設定を示します。

設定

送信 受信

UART0クロック設定

動作クロック CK00

クロック・ソース fCLK/2⁵ (クロック周波数: 500 kHz)

データ・ビット長設定

7ビット 8ビット 9ビット

データ転送方向設定

LSB MSB

パリティ設定

パリティ・ビットなし 0パリティ 奇数パリティ 偶数パリティ

ストップ・ビット長設定

1ビット固定です

受信データ・レベル設定

非反転(通常) 反転

転送レート設定

転送レート設定 31250 (bps)

(誤差: 0%, 許容最小: -4%, 許容最大: 4%)

割り込み設定

受信完了割り込み設定(INTSR0) レベル2

エラー割り込み設定(INTSRE0) レベル3(低優先順位)

コールバック機能設定

受信完了 受信エラー

チェックを外す

図 3-6 MIDI 受信の通信設定

3.3 オプション・バイトの設定一覧

表 3-2にオプション・バイト設定を示します。

表 3-2 オプション・バイト設定

アドレス	設定値	内容
000C0H	11101111B	ウォッチドッグ・タイマ動作停止 (リセット解除後、カウント停止)
000C1H	11110111B	SPOR 検出電圧： 立ち上がり時 TYP. 2.90V (2.76 V ~ 3.02 V) 立ち下がり時 TYP. 2.84V (2.70 V ~ 2.96 V)
000C2H	11111001B	高速オンチップ・オシレータ・クロック : 16MHz
000C3H	10000101B	オンチップ・デバッグ許可

3.4 マクロ一覧

表 3-3 にサンプルプログラムで使用するマクロを示します。

表 3-3 サンプルプログラムで使用するマクロ

マクロ名	設定値	内容
DEMO_ALLCH_LIGHT_DIM	3	全チャンネル表示モード時の LED 調光段数
DEMO_MIDI_BRIGHTNESS_DIVISION	120	LED 調光時間を算出するために用いる分解能値
DEMO_MIDI_PITCH_NUM	12	1 オクターブの音階数
DEMO_MIDI_DISPLAY_ALLCH	16	全チャンネル表示モード時で表示する MIDI チャンネル数
DEMO_MIDI_CH_MAX	16	最大 MIDI チャンネル番号
DEMO_MIDI_CH_MIN	1	最小 MIDI チャンネル番号
DEMO_MIDI_DISPLAY_CH_MAX	17	チャンネル数 (Ch1~Ch16、全チャンネル)
DEMO_ALLCH_LIGHT_KEEP_THD	40	全チャンネル表示モード時の LED 調光時の明るさを維持する時間の閾値
DEMO_ADC_DATA_DIVISION	1024	ボリュームスイッチ入力分解能
DEMO_ADC_BUFF_SIZE	4	ボリュームスイッチデータバッファ数
DEMO_ADC_INPUT_DIFFERENCE	8	ボリュームスイッチ入力閾値
DEMO_DISPLAY_MODE_MELODY	0	表示情報種別：音階表示
DEMO_DISPLAY_MODE_CH_SET	1	表示情報種別：チャンネル表示
DEMO_DISPLAY_CH_SET_TIME	1000	チャンネル表示期間[ms]
DEMO_MATRIX_CATHODE_COLOR	3	LED 色要素数：Red、Blue、Green
DEMO_MATRIX_DIGIT	8	LED 行数
DEMO_SOUND_VOLUME_PATTERN	5	音量段数：5
DEMO_SOUND_VOLUME_PAT_LARGE	4	音量種別：大
DEMO_SOUND_VOLUME_PAT_MEDIUM	3	音量種別：中
DEMO_SOUND_VOLUME_PAT_SMALL	2	音量種別：小
DEMO_SOUND_VOLUME_PAT_MICRO	1	音量種別：極小
DEMO_SOUND_VOLUME_PAT_NONE	0	音量種別：なし
DEMO_SYSTEM_TIMER_START_FUNC	-	R_Config_TAU0_3_Start のエイリアス
DEMO_MATRIX_LED_PWM_START_FUNC	-	R_Config_TAU0_7_DEMO_MATRIX_LED_PWM_Start のエイリアス
DEMO_MATRIX_LED_SPI_CSPIN	-	CSI20 チップセレクト (P1_bit.no6) のエイリアス
DEMO_MATRIX_LED_SPI_START_FUNC	-	R_Config_CSI20_DEMO_MATRIX_LED_Start のエイリアス
DEMO_MATRIX_LED_SPI_SEND_FUNC	-	R_Config_CSI20_DEMO_MATRIX_LED_Send のエイリアス
DEMO_PWM_PERIOD_REG	-	タイマデータレジスタ TDR07 のエイリアス
DEMO_PWM_COUNT_REG	-	タイマカウントレジスタ TCR07 のエイリアス
DEMO_ANALOG_VOLUME_INPUT_START_FUNC	-	AD 変換スタート準備 R_Config_ADC_DEMO_VOLUME_Set_Operation()と R_Config_ADC_DEMO_VOLUME_Start()を連続でコールするマクロ

3.5 定数一覧

表 3-4 定数に定数を示します。

表 3-4 定数

型	定数名	内容	使用関数
const uint8_t	g_color_table_single_ch	MIDI チャンネル別表示モードの RGB 別の表示色設定値	demo_display_main
const uint8_t	g_sound_volume_digit_table	音量設定値テーブル	demo_display_main
const uint8_t	g_sound_volume_msk	特定チャンネル表示モード時の音量別 LED マスクテーブル	demo_display_main
const uint8_t	g_disp_volume_start	ベロシティ→音量変換テーブル	demo_convert_velocity_to_graph_pattern
const uint8_t	g_anti_blur	滲み防止用 LED データ	demo_display_main
const uint8_t	g_color_table_all_ch	全チャンネル表示モード時の音階別 LED データテーブル	demo_display_main
const uint8_t	g_display_ch_graph	チャンネル変更時のチャンネル表示の LED データテーブル	demo_display_main

3.6 変数一覧

表 3-5 と表 3-6 にグローバル変数を示します。

表 3-5 グローバル変数 (1/2)

型	変数名	内容	使用関数
uint8_t	g_demo_adc_finish	ボリュームスイッチの AD 変換完了を示すフラグ	main、 r_Config_ADC_DEMO_VOLUME_interrupt、 demo_volume_input
uint16_t	g_demo_adc_data	ボリュームスイッチの AD 値	main、 r_Config_ADC_DEMO_VOLUME_interrupt、 demo_volume_input
uint16_t	g_demo_adc_buff	ボリュームスイッチの AD 値バッファ	main、 demo_volume_monitor_main、 demo_volume_input
uint16_t	g_demo_adc_average	ボリュームスイッチの AD 値の平均値	main、 demo_volume_monitor_main
uint16_t	g_demo_adc_average_bak	ボリュームスイッチの AD 値の平均値の前回値	main、 demo_volume_monitor_main
uint8_t	g_demo_adc_step	ボリュームスイッチの AD 変換処理状態：0=変換停止中、1=変換中	main、 demo_volume_monitor_main、 demo_volume_input
uint8_t	g_demo_adc_input_index	ボリュームスイッチの AD 値バッファのインデックス	demo_volume_input
ch_blink_t	g_demo_matrix_ch_info	音階別の表示情報	main、 demo_display_main demo_convert_velocity_to_graph_pattern
uint16_t	g_demo_pwm_period	PWM タイマ周期	Main
uint16_t	g_demo_pwm_period_sub	Duty 比減少量	Main
uint16_t	g_demo_pwm_period_base	表示の明るさの段数	main
uint16_t	g_demo_pwm_count	PWM タイマカウンタ値	main、 demo_display_main
uint16_t	g_demo_pwm_count_bak	PWM タイマカウンタ値前回値	main
uint8_t	g_demo_spi_sending_flag	LED データ送信中フラグ：0=送信完了、1=送信中	main、 r_Config_CSI20_DEMO_MATRIX_LED_callback_sendend、 demo_matrix_led_data_send、 demo_matrix_led_send_busy_check

表 3-6 グローバル変数 (2/2)

型	変数名	内容	使用関数
uint32_t	g_demo_display_ch_start_time	チャンネル変更時のチャンネル表示開始時刻	main、 demo_display_main
uint8_t	g_demo_display_mode	LED 表示モード： DEMO_DISPLAY_MODE_MELODY=音階表示、 DEMO_DISPLAY_MODE_CHANNELSET=チャンネル表示	main、 demo_display_main
uint8_t	g_demo_display_ch	表示対象チャンネル： 0~15=Ch1~Ch16、16=全チャンネル	main、 demo_display_main
uint32_t	g_demo_timer	サンプルプログラムのシステムタイマ値[ms]	demo_time_now、 demo_timer_cycle
uint8_t	g_demo_read_ch	NoteOn(打鍵)メッセージからの MIDI チャンネル： 0~15=Ch1~Ch16	Main
uint8_t	g_demo_read_pitch	NoteOn(打鍵)メッセージからの音階（オクターブ情報は削除済み）	Main
uint8_t	g_demo_matrix_led_send_buff	LED 表示データ送信バッファ	demo_display_main
uint16_t	g_digit_now	更新対象 LED インデックス	demo_display_main

3.7 関数一覧

表 3-7 に関数を示します。

表 3-7 関数

関数名	概要
demo_volume_monitor_main()	モニタチャンネル更新処理
demo_volume_input()	ボリュームスイッチ状態取得処理
demo_display_main()	LED データ更新処理
demo_time_now()	プログラム起動からの経過時間を取得
demo_timer_cycle()	TAU0_3 周期割り込みから呼び出され、プログラム起動からの経過時間を更新
demo_convert_velocity_to_graph_pattern()	ベロシティから LED 表示パターンサイズへの変換処理
demo_matrix_led_data_send()	LED データ送信処理
demo_matrix_led_send_busy_check()	LED データ送信完了監視処理
R_Config_TAU0_3_Start()	TAU0_3 タイマスタート処理
r_Config_TAU0_3_interrupt()	TAU0_3 周期割り込みのコールバック処理
R_Config_TAU0_7_DEMO_MATRIX_LED_PWM_Start()	TAU0_7 タイマスタート処理
R_Config_ADC_DEMO_VOLUME_Start()	AD 変換完了割り込みフラグクリア、AD 変換完了割り込み許可、AD 変換動作許可を設定
R_Config_ADC_DEMO_VOLUME_Set_OperationOn()	AD 電圧コンパレータの動作許可
R_Config_ADC_DEMO_VOLUME_Get_Result_10bit()	AD 変換結果取得処理
r_Config_ADC_DEMO_VOLUME_interrupt()	INTAD AD 変換完了時のコールバック処理
R_Config_CSI20_DEMO_MATRIX_LED_Start()	CSI20 スタート処理
R_Config_CSI20_DEMO_MATRIX_LED_Send()	CSI20 データ送信処理
r_Config_CSI20_DEMO_MATRIX_LED_callback_sendend()	CSI20 送信完了時のコールバック処理
r_Config_CSI20_DEMO_MATRIX_LED_interrupt()	CSI20 転送完了割り込み処理
R_Config_UART0_Send()	UART0 送信処理
R_Config_UART0_Receive()	UART0 受信処理
r_Config_UART0_callback_sendend()	UART0 送信完了処理
r_Config_UART0_callback_receiveend()	UART0 受信完了処理
r_Config_UART0_interrupt_send()	UART0 送信割り込み処理
r_Config_UART0_interrupt_receive()	UART0 受信割り込み処理
r_Config_UART0_interrupt_error()	UART0 受信の通信エラーの割り込み処理

3.8 関数仕様

サンプルプログラムの関数仕様を示します。

demo_volume_monitor_main()

概要	MIDI チャンネル選択処理
ヘッダ	-
宣言	uint8_t demo_volume_monitor_main(void);
説明	ボリュームスイッチの変化からモニタする MIDI チャンネルを更新します。
引数	なし
リターン値	0: モニタする MIDI チャンネル変更なし 1: モニタする MIDI チャンネル変更あり

demo_volume_input()

概要	ボリュームスイッチの状態の取得処理
ヘッダ	-
宣言	uint8_t demo_volume_input(void);
説明	ボリュームスイッチの状態の取得処理です。
引数	なし
リターン値	0: ボリュームスイッチ入力取得中 1: ボリュームスイッチ入力取得完了

demo_display_main()

概要	マトリクス LED の表示更新処理
ヘッダ	-
宣言	void demo_display_main(void);
説明	マトリクス LED の表示内容を更新します。
引数	なし
リターン値	なし

demo_time_now()

概要	現在時刻の取得
ヘッダ	-
宣言	uint32_t demo_time_now(void);
説明	サンプルプログラム起動時からの経過時間を返します。
引数	なし
リターン値	uint32_t: サンプルプログラム起動時からの経過時間[ms]

demo_timer_cycle()

概要	時刻更新
ヘッダ	-
宣言	void demo_timer_cycle(void);
説明	定周期割り込み処理の中から呼び出されサンプルプログラム起動時からの経過時間を更新します。
引数	なし
リターン値	なし

demo_convert_velocity_to_graph_pattern()

概要	ベロシティから LED 表示パターンサイズへの変換
ヘッダ	-
宣言	void demo_convert_velocity_to_graph_pattern(uint8_t ch);
説明	時々変化する MIDI チャンネル別に保持しているベロシティ値を参照して、特定 MIDI チャンネル表示モードのマトリクス LED の表示サイズを決定します。
引数	uint8_t ch: MIDI チャンネル
リターン値	なし

demo_matrix_led_data_send()

概要	LED データ送信
ヘッダ	-
宣言	void demo_matrix_led_data_send(uint8_t * data, uint16_t len);
説明	マトリクス LED ヘデータを送信します。
引数	uint8_t * data: 送信データアドレス uint16_t len: 送信データサイズ
リターン値	なし

demo_matrix_led_send_busy_check()

概要	LED データ送信完了監視
ヘッダ	-
宣言	uint8_t demo_matrix_led_send_busy_check(void);
説明	通信処理中フラグ (g_demo_spi_sending_flag) を参照して送信状態を返します。
引数	なし
リターン値	0: 送信完了 1: 送信中

r_Config_TAU0_3_interrupt()

概要	インターバルタイマ割り込み処理
ヘッダ	r_cg_macrodriver.h、r_cg_userdefine.h、Config_TAU0_3.h、r_midi_rl78_if.h
宣言	static void __near r_Config_TAU0_3_interrupt(void); TAU0_3 のカウント完了割り込み関数です。
説明	MIDI の 1ms 周期通知関数を呼びます。 システムタイマカウント関数を呼びます。
引数	なし
リターン値	なし

r_Config_ADC_DEMO_VOLUME_interrupt()

概要	AD 変換完了割り込み処理
ヘッダ	r_cg_macrodriver.h、r_cg_userdefine.h、Config_ADC_DEMO_VOLUME.h
宣言	static void __near r_Config_ADC_DEMO_VOLUME_interrupt(void); A/D 変換終了割り込み関数です。
説明	AD 変換結果を読み取り、バッファ (g_demo_adc_data) に保存します AD 変換完了フラグ (g_demo_adc_finish) をセットします
引数	なし
リターン値	なし

r_Config_CSI20_DEMO_MATRIX_LED_callback_sendend()

概要	CSI20 送信完了時のコールバック処理
ヘッダ	r_cg_macrodriver.h、r_cg_userdefine.h、 Config_CSI20_DEMO_MATRIX_LED.h、midi_matrixled_demo.h
宣言	static void r_Config_CSI20_DEMO_MATRIX_LED_callback_sendend(void); CSI20 の送信完了時に呼ばれるコールバック関数です。
説明	SPI の CS 端子を High にします。 その後、通信処理中フラグ (g_demo_spi_sending_flag) をリセットします。
引数	なし
リターン値	なし

r_Config_UART0_callback_sendend()

概要	UART0 送信完了コールバック処理
ヘッダ	r_cg_macrodriver.h、r_cg_userdefine.h、Config_UART0.h、r_midi_rl78_if.h
宣言	static void r_Config_UART0_callback_sendend(void); UART0 送信完了時に呼ばれるコールバック処理です。
説明	(R_MIDI_NotifyEvent) を呼び出し、MIDI インタフェース SIS (Software Integration System) モジュールに送信完了を通知します。
引数	なし
リターン値	なし

r_Config_UART0_callback_receiveend()

概要	UART0 送信完了コールバック処理
ヘッダ	r_cg_macrodriver.h、r_cg_userdefine.h、Config_UART0.h、r_midi_rl78_if.h
宣言	static void r_Config_UART0_callback_receiveend(void); UART0 受信完了時に呼ばれるコールバック処理です。
説明	(R_MIDI_NotifyEvent) を呼び出し、MIDI インタフェース SIS (Software Integration System) モジュールに受信完了を通知します。
引数	なし
リターン値	なし

3.9 フローチャート

3.9.1 メイン処理

図 3-7 と 図 3-8 にメイン処理のフローチャートを示します。

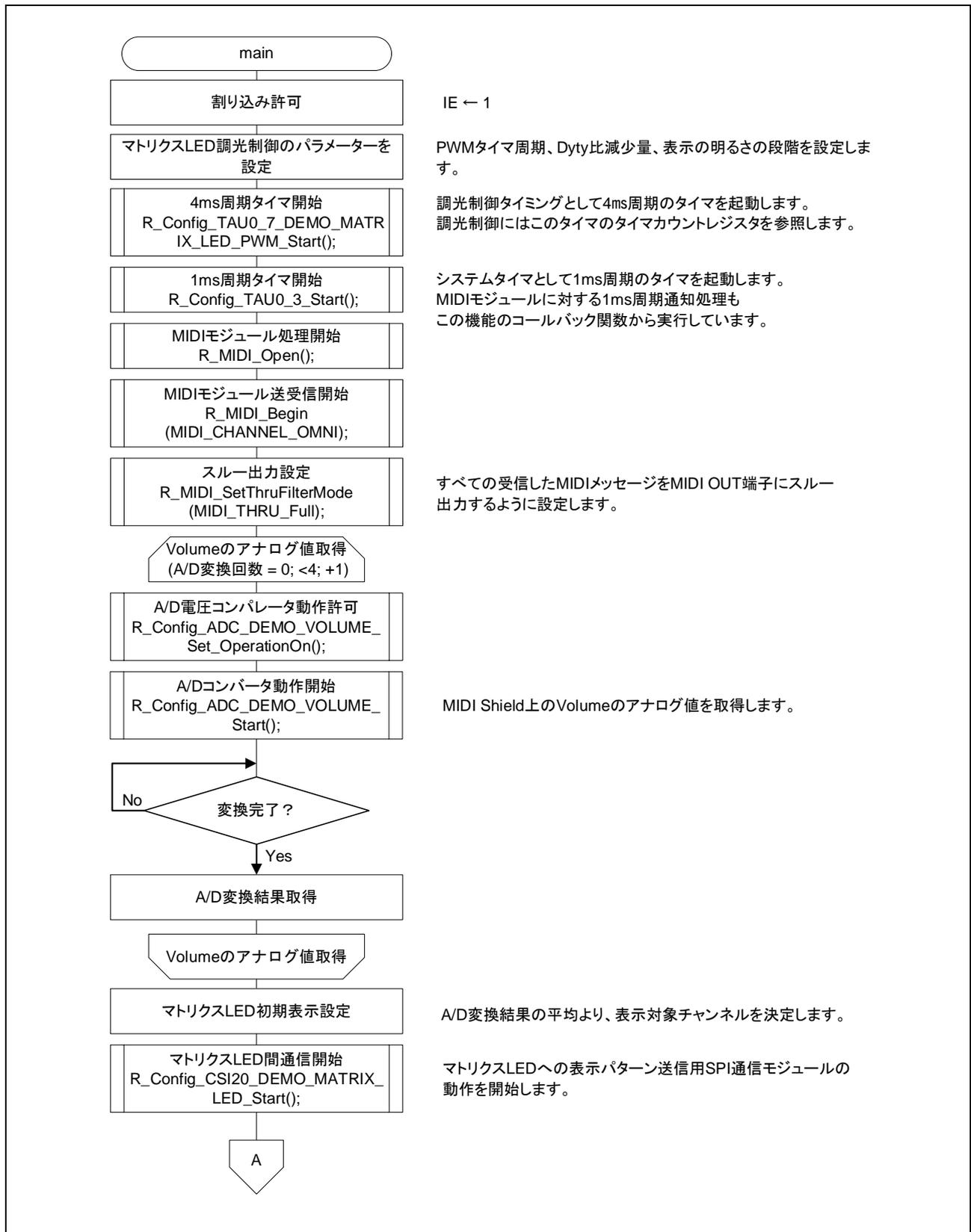


図 3-7 メイン処理(1/2)

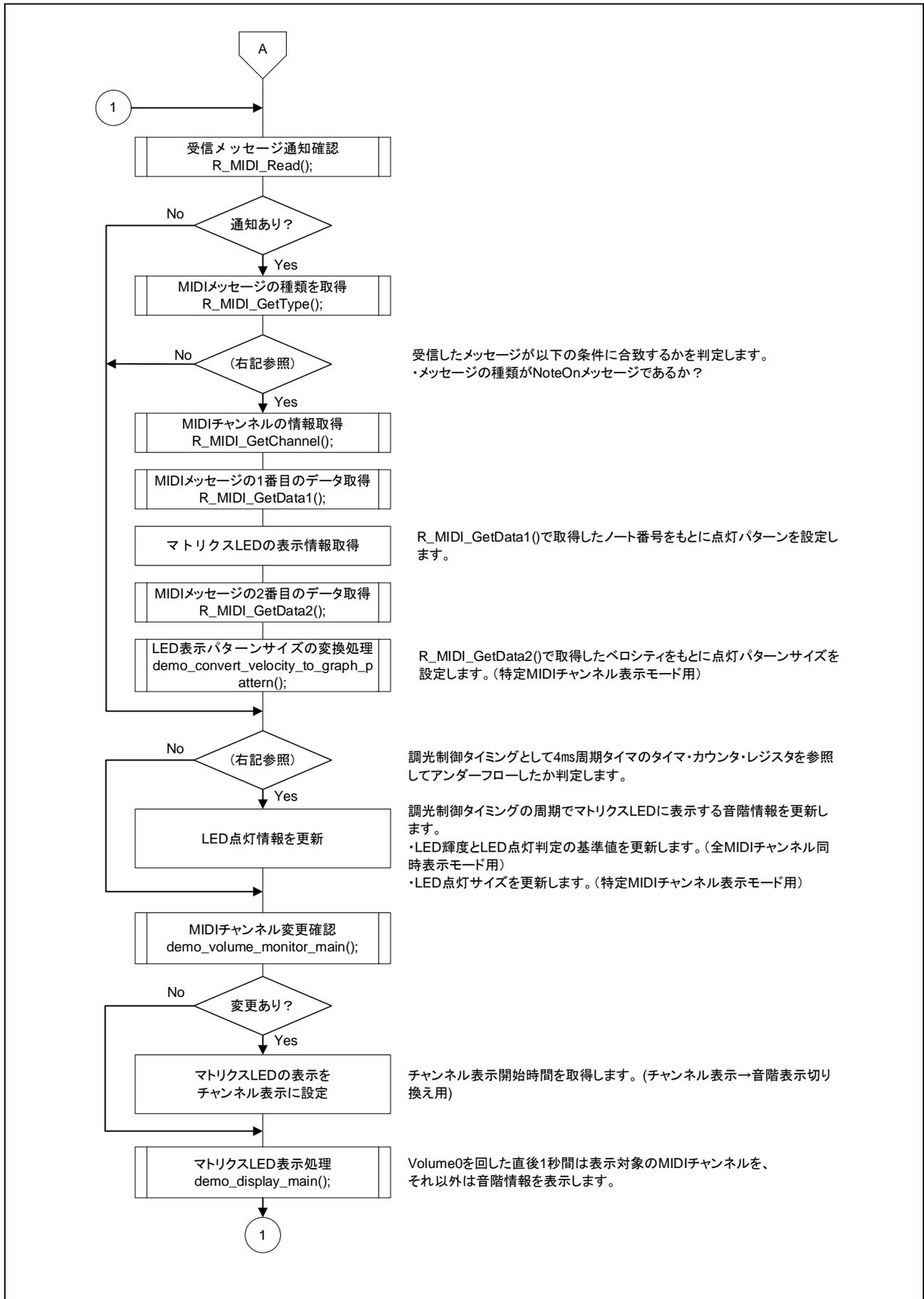


図 3-8 メイン処理(2/2)

3.9.2 ボリュームスイッチから表示対象の MIDI チャンネル決定処理

図 3-9 にボリュームスイッチから表示対象の MIDI チャンネル決定処理のフローチャートを示します。

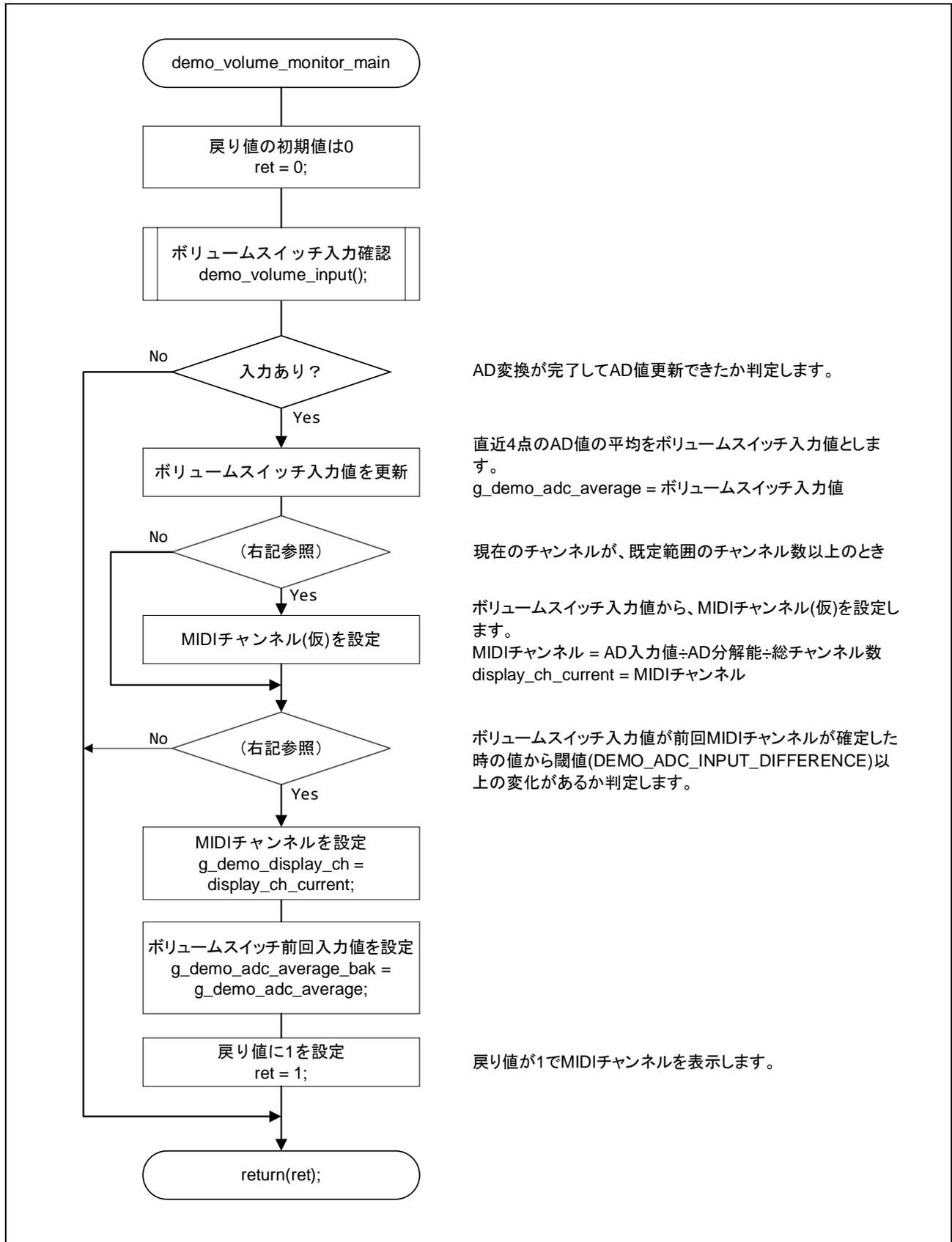


図 3-9 ボリュームスイッチから表示対象の MIDI チャンネル決定処理

3.9.3 ボリュームスイッチ入力取得処理

図 3-10 にボリュームスイッチ入力取得処理のフローチャートを示します。

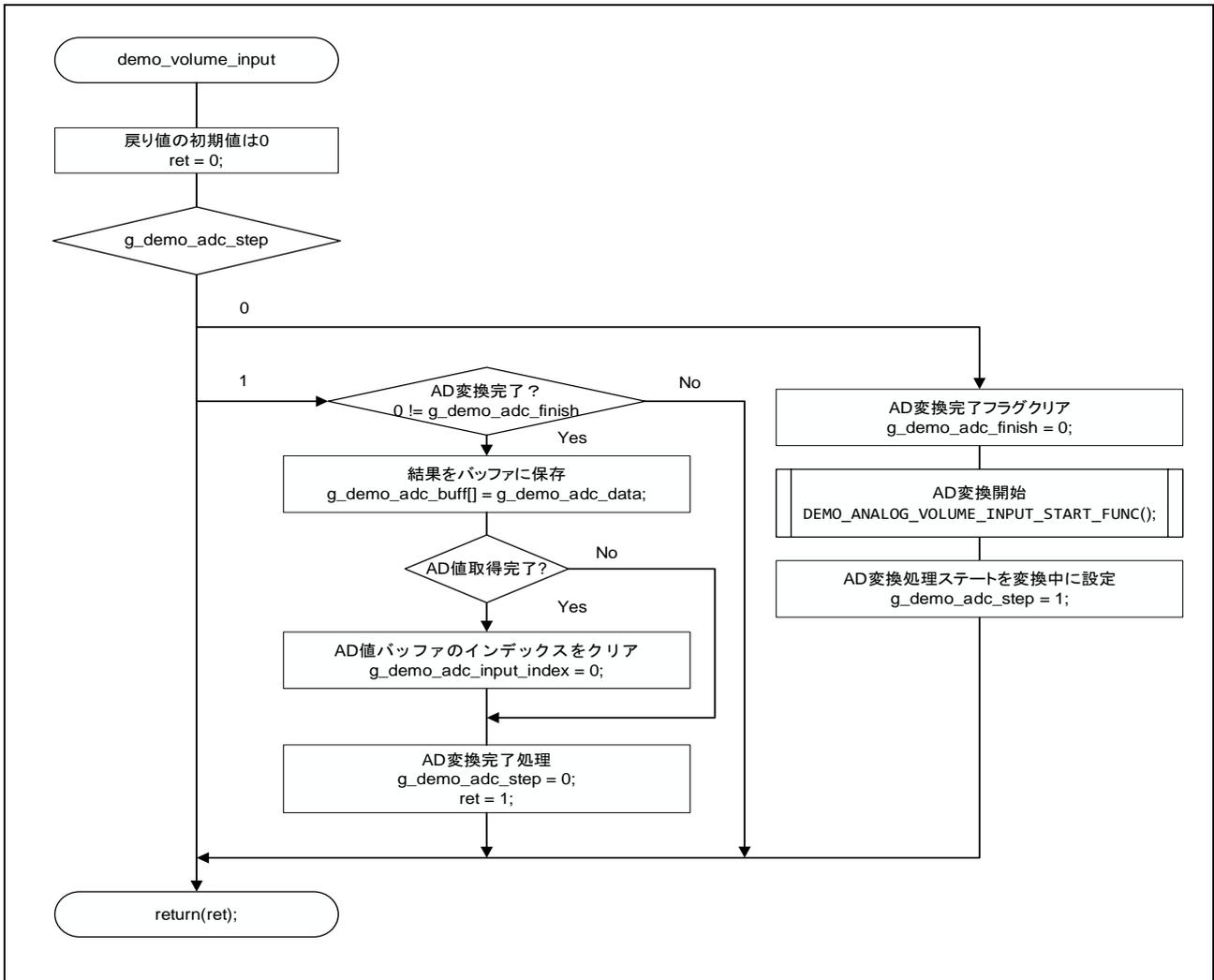


図 3-10 ボリュームスイッチ入力取得処理

3.9.4 マトリクス LED 表示処理

図 3-11 と図 3-12 にマトリクス LED 表示処理のフローチャートを示します。

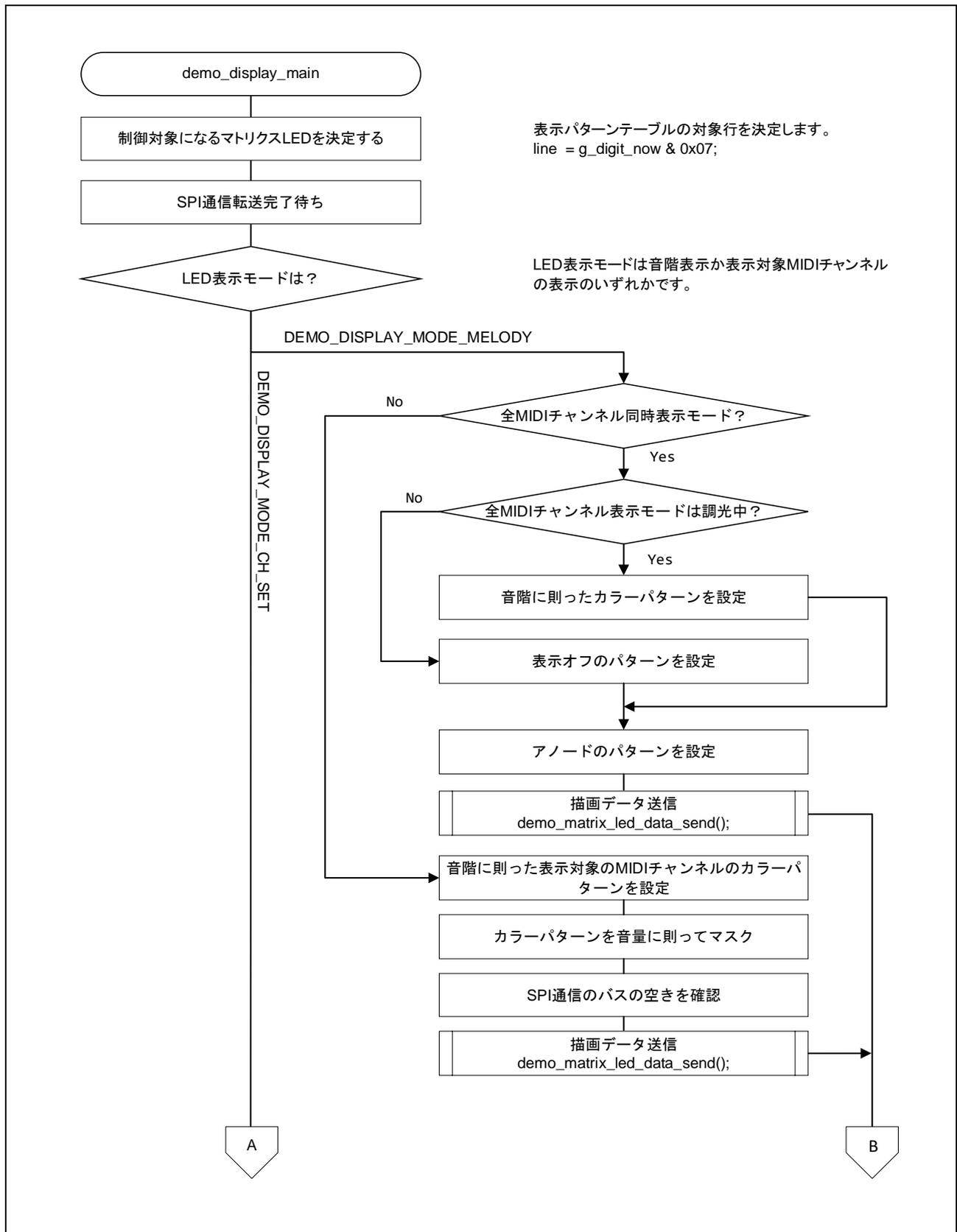


図 3-11 マトリクス LED 表示処理 (1/2)

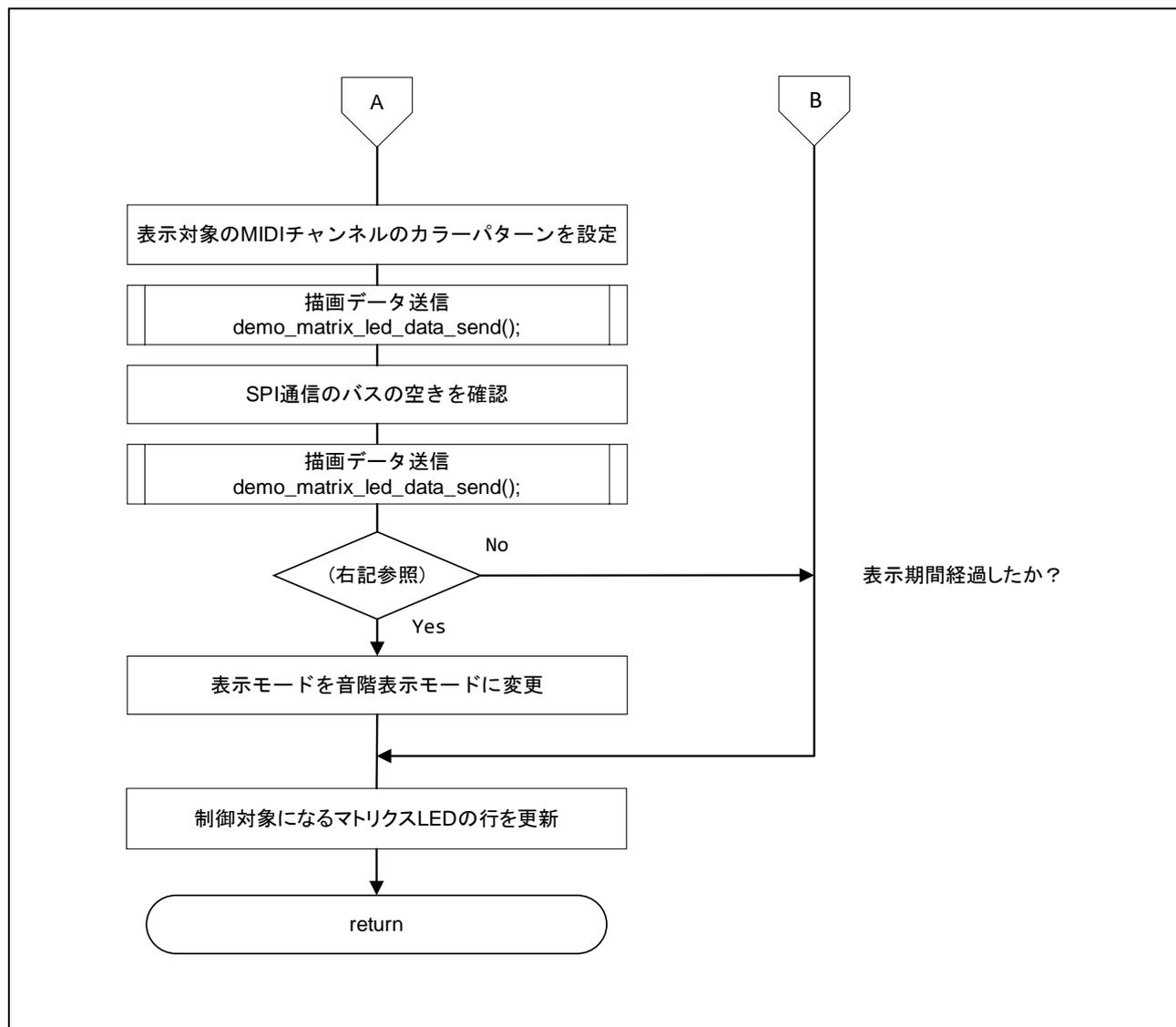


図 3-12 マトリクス LED 表示処理 (2/2)

3.9.5 システムタイマ取得処理

図 3-13 にシステムタイマ取得処理のフローチャートを示します。

※この関数はグローバル変数を返すだけの処理ですが複数箇所から参照され、同じ変数を参照していることを明示するため関数化しています。

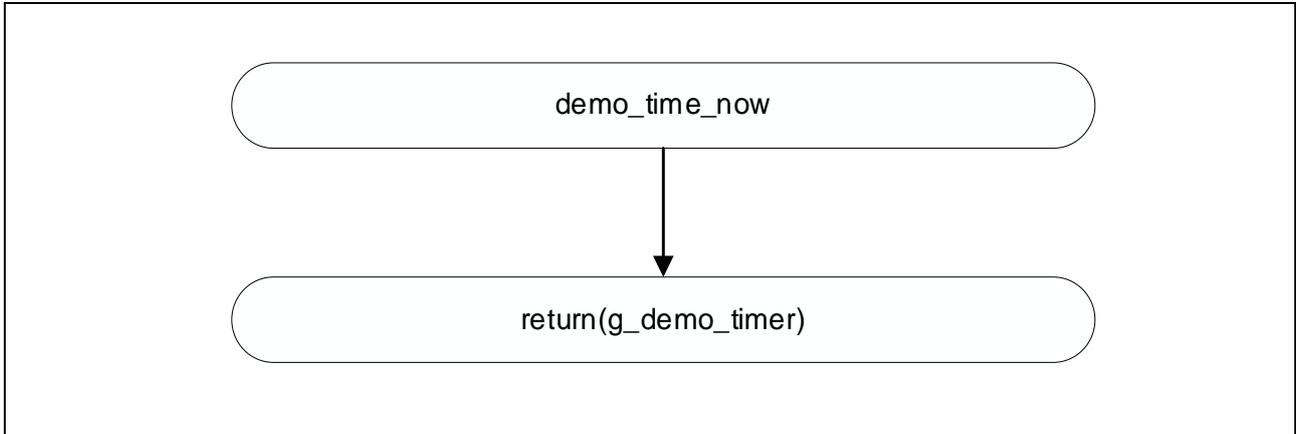


図 3-13 システムタイマ取得処理

3.9.6 システムタイマカウント処理

図 3-14 にシステムタイマカウント処理のフローチャートを示します。

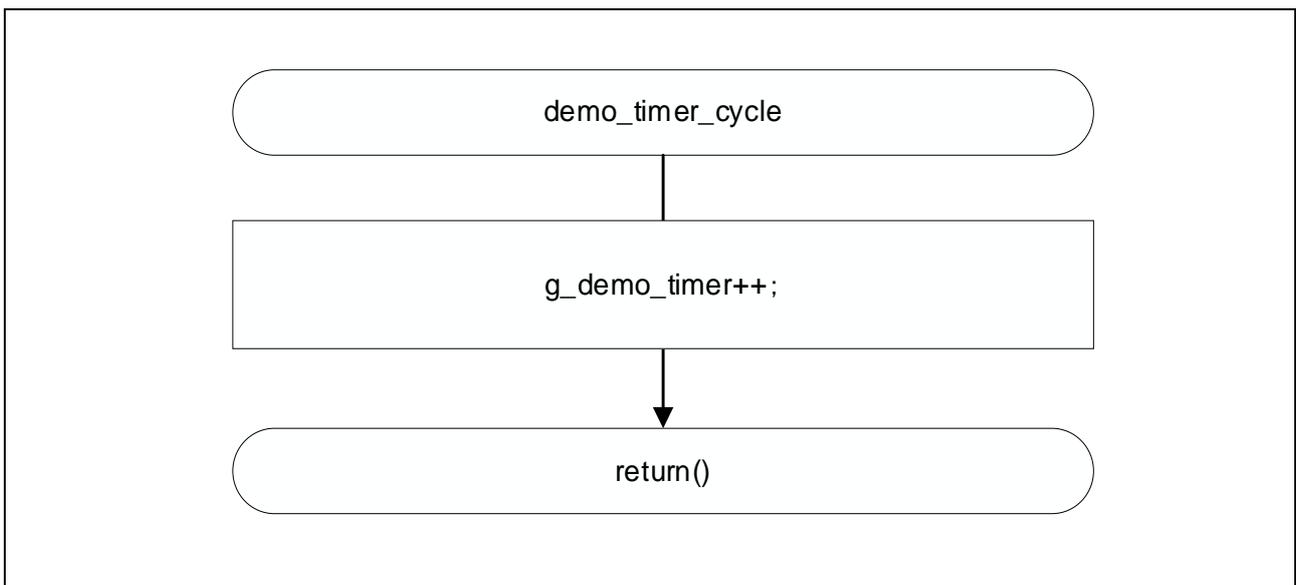


図 3-14 システムタイマカウント処理

3.9.7 マトリクス LED データ表示サイズ決定処理

図 3-15 にマトリクス LED データ表示サイズ決定処理のフローチャートを示します。

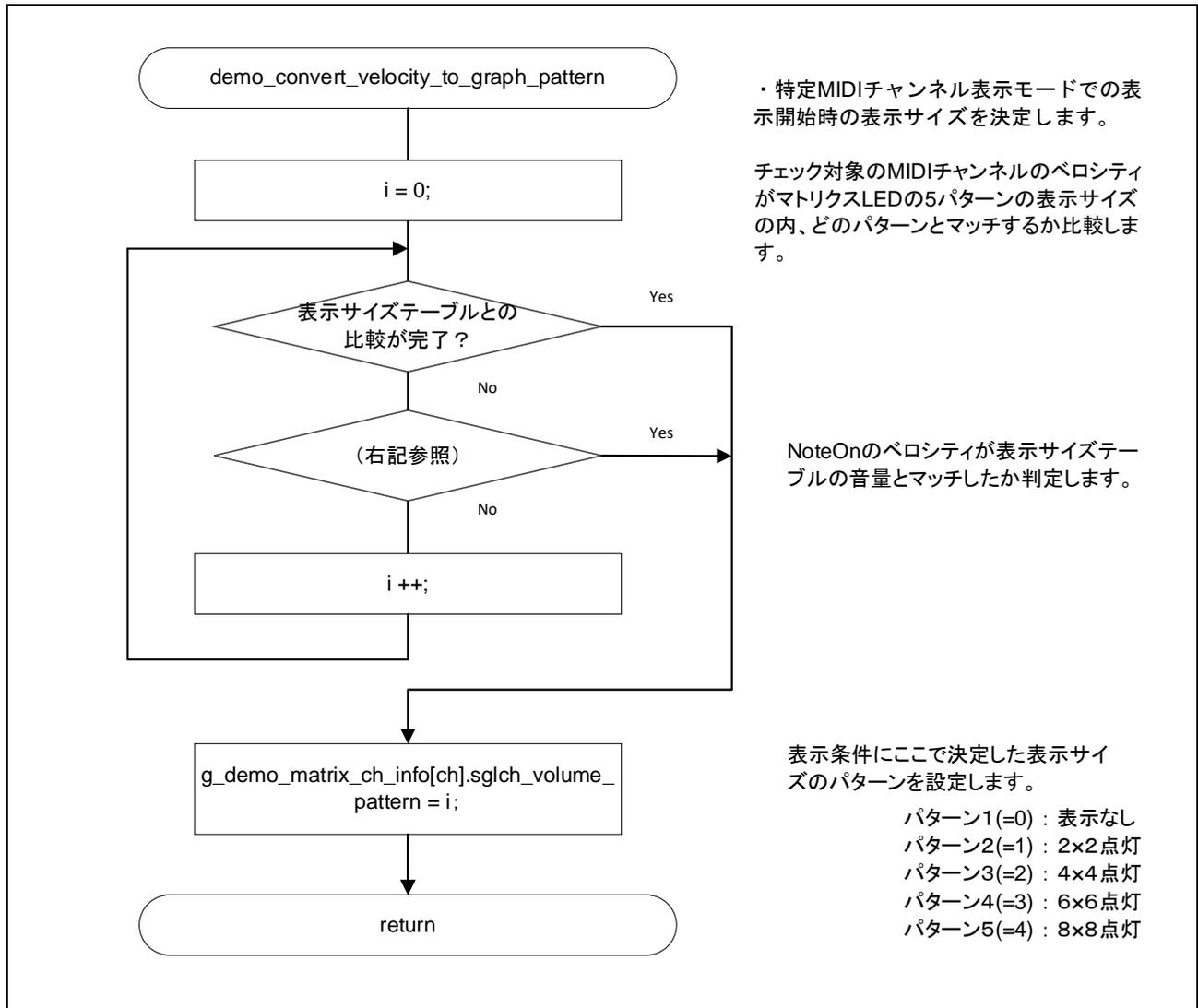


図 3-15 マトリクス LED データ表示サイズ決定処理

3.9.8 マトリクス LED データ送信処理

図 3-16 にマトリクス LED データ送信処理のフローチャートを示します。

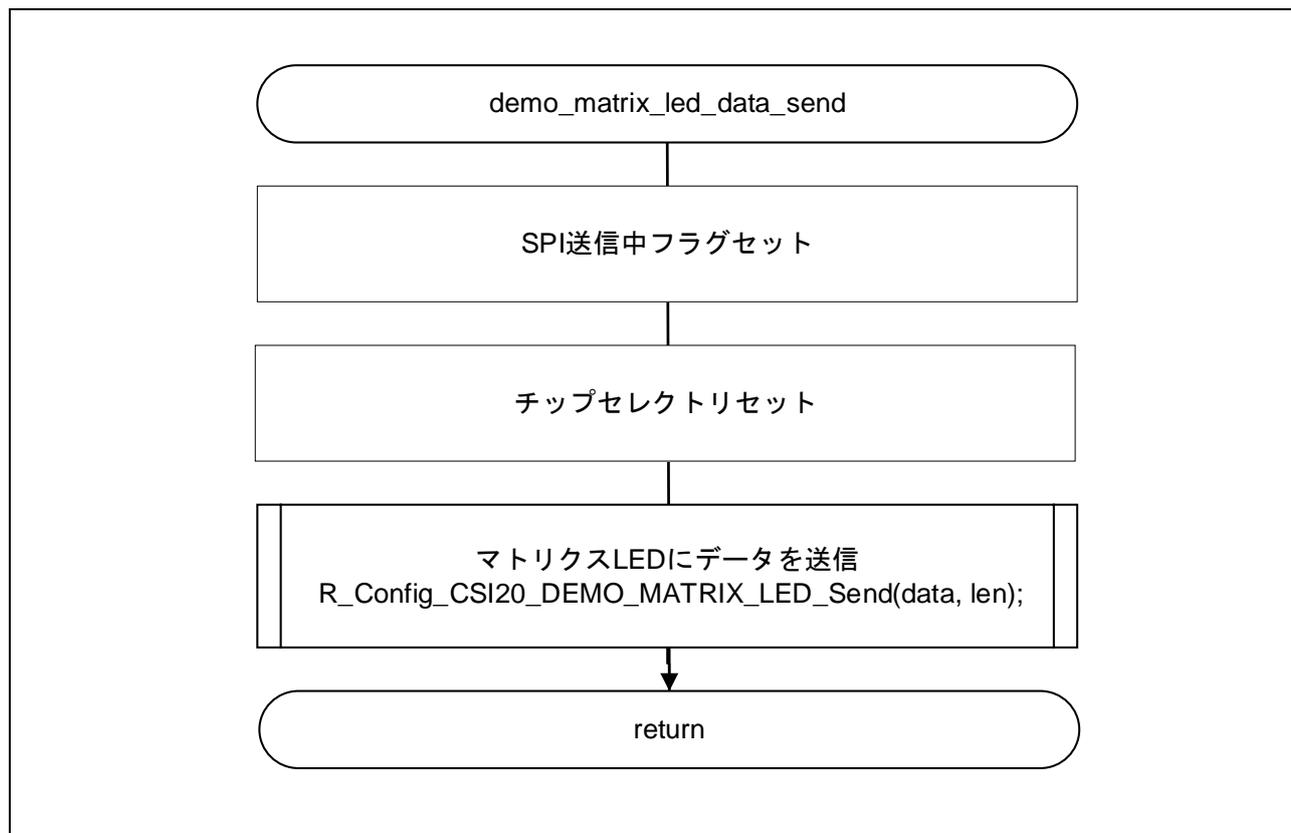


図 3-16 マトリクス LED データ送信処理

3.9.9 マトリクス LED データ送信完了チェック処理

図 3-17 にマトリクス LED データ送信完了チェック処理のフローチャートを示します。

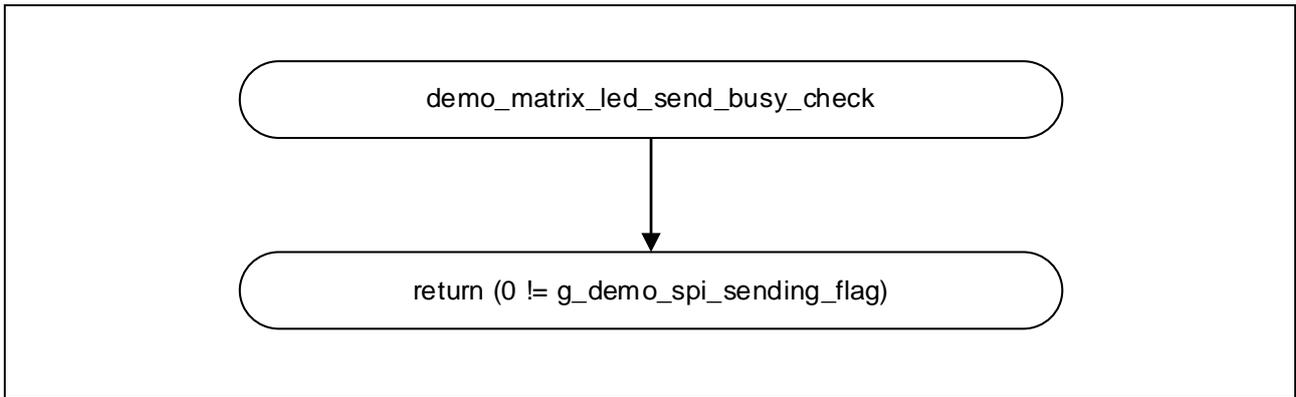


図 3-17 マトリクス LED データ送信完了チェック処理

3.9.10 TAU0_3 割り込み処理

図 3-18 に TAU0_3 割り込み処理のフローチャートを示します。

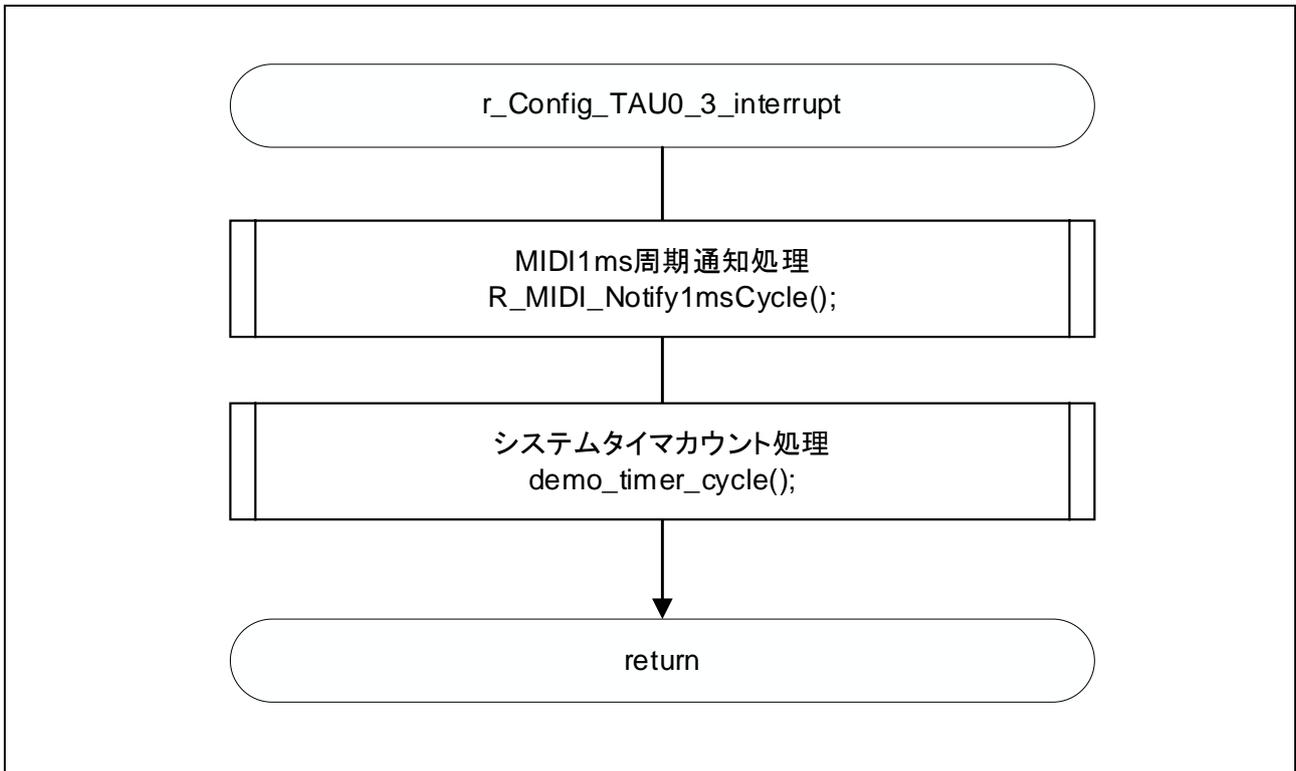


図 3-18 TAU0_3 割り込み処理

3.9.11 AD 変換完了割り込み処理

図 3-19 に AD 変換開始変換完了割り込み処理のフローチャートを示します。

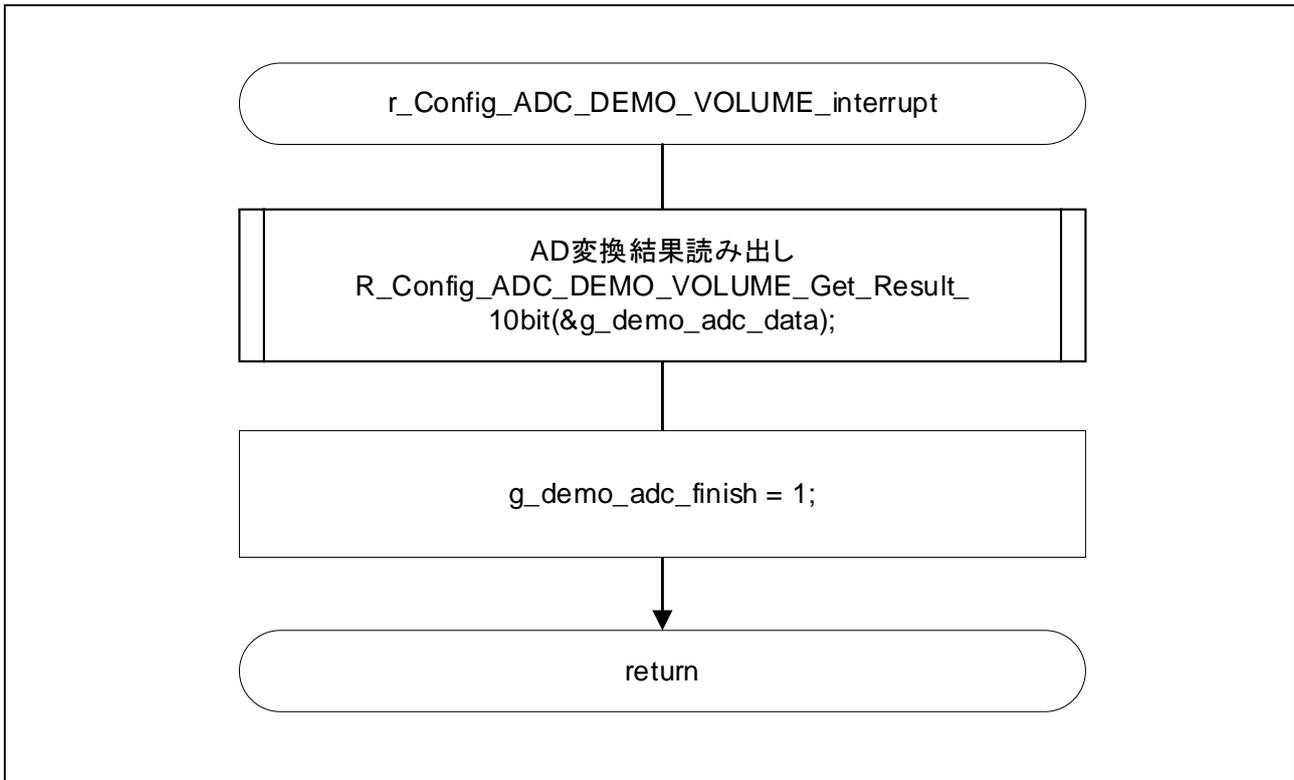


図 3-19 AD 変換完了割り込み処理

3.9.12 マトリクス LED データ CSI 送信完了処理

図 3-20 にマトリクス LED データ CSI 送信完了処理のフローチャートを示します。

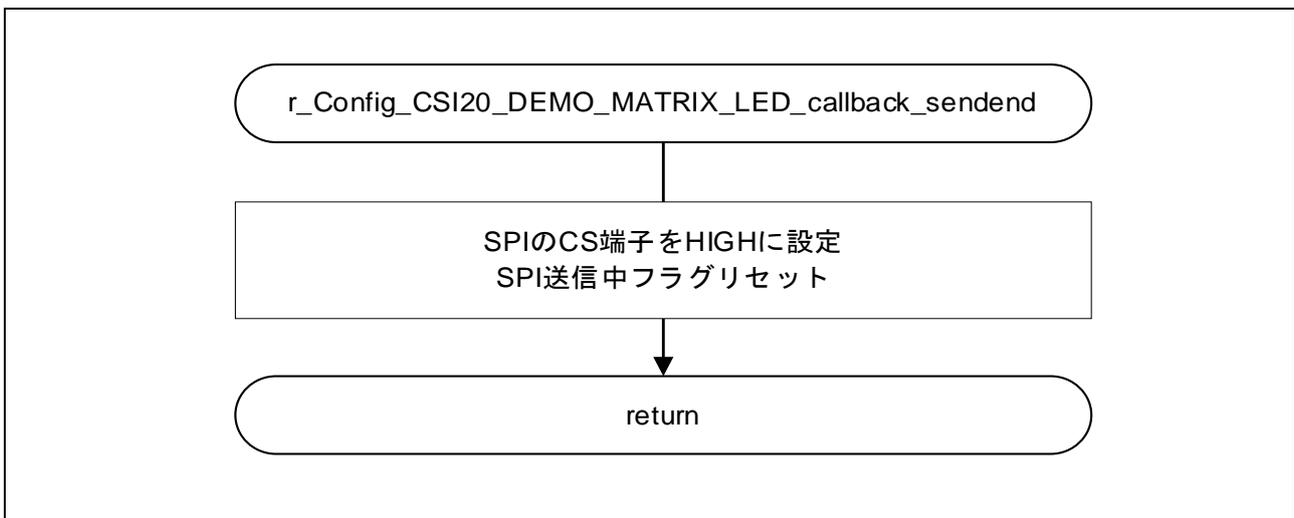


図 3-20 マトリクス LED データ CSI 送信完了処理

3.9.13 UART0 送信完了処理

図 3-21 に UART0 送信完了処理のフローチャートを示します。

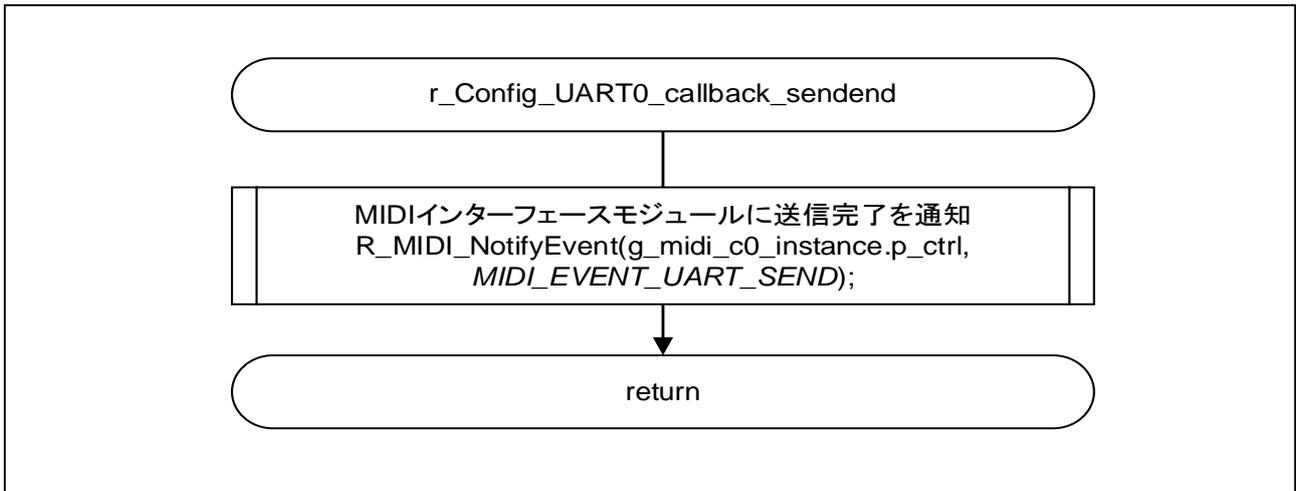


図 3-21 UART0 送信完了処理

3.9.14 UART0 受信完了処理

図 3-22 に UART0 受信完了処理のフローチャートを示します。

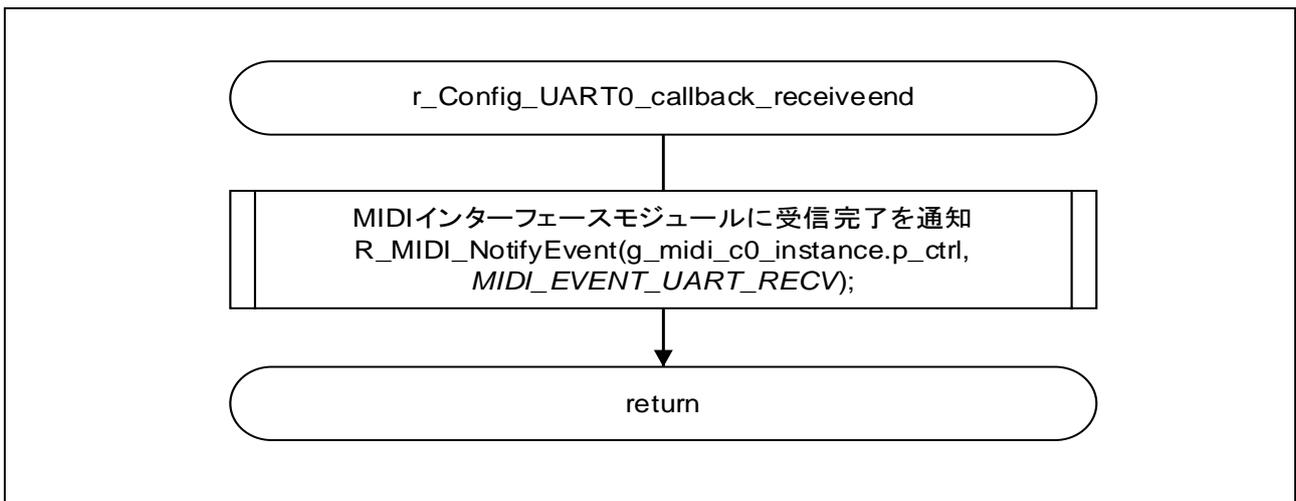


図 3-22 UART0 受信完了処理

4. DAW の設定方法

DAW として Domino(<https://takabosoft.com/domino>)を用いた例で説明します。

ダウンロードしたファイルは解凍すると図 4-1 のとおり展開されます。

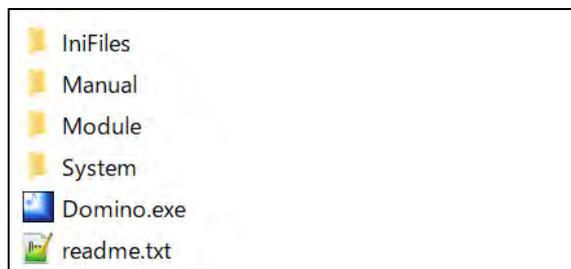


図 4-1 Domino 展開イメージ

展開したファイルの中の Domino.exe 実行すると図 4-2 のとおりアプリケーションが立ち上がります。

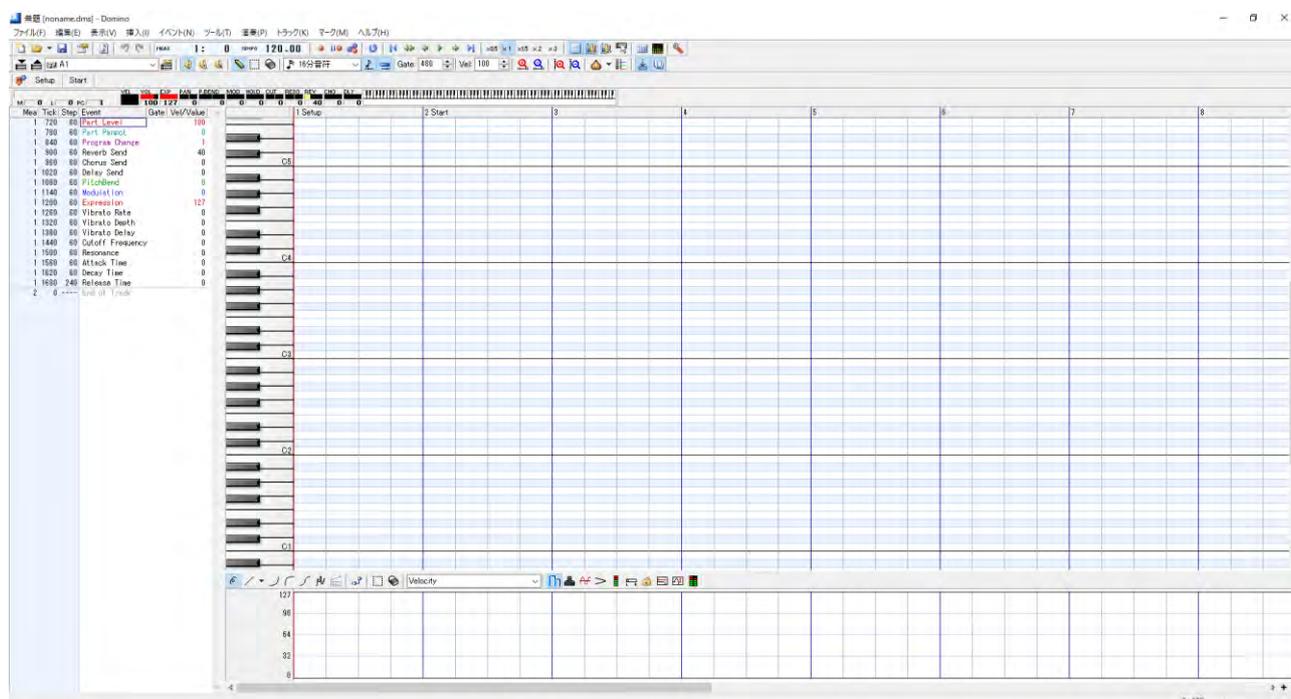


図 4-2 Domino メイン画面

PC と MIDI IN 端子の接続は表 2-1 ハードウェア一覧のとおり、Roland 製 UM-ONE を使用しているので“ファイル-環境設定”または“F12”を押して図 4-3 環境設定ダイアログで設定します。また、音源モジュールとして SC-88Pro を設定します。

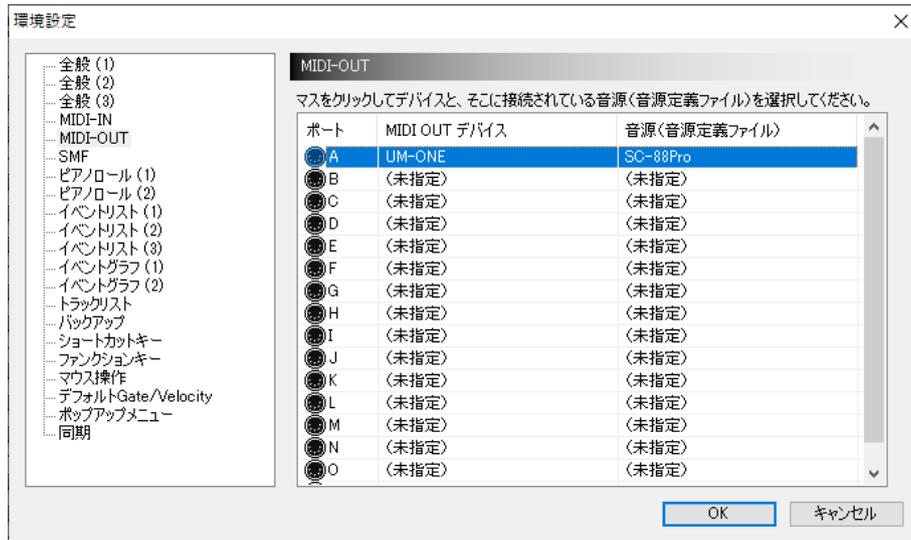


図 4-3 Domino 環境設定

演奏する MIDI ファイルを“ファイル-開くメニュー”または“Ctrl+O”を押して図 4-4 “ファイルを開く”ダイアログから開きます。

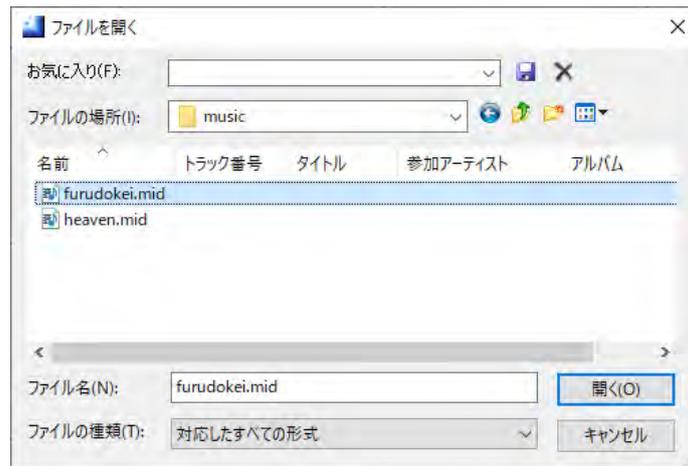


図 4-4 Domino ファイルを開くダイアログ

図 4-5 は MIDI ファイルが読み込まれ演奏準備完了している状態です。

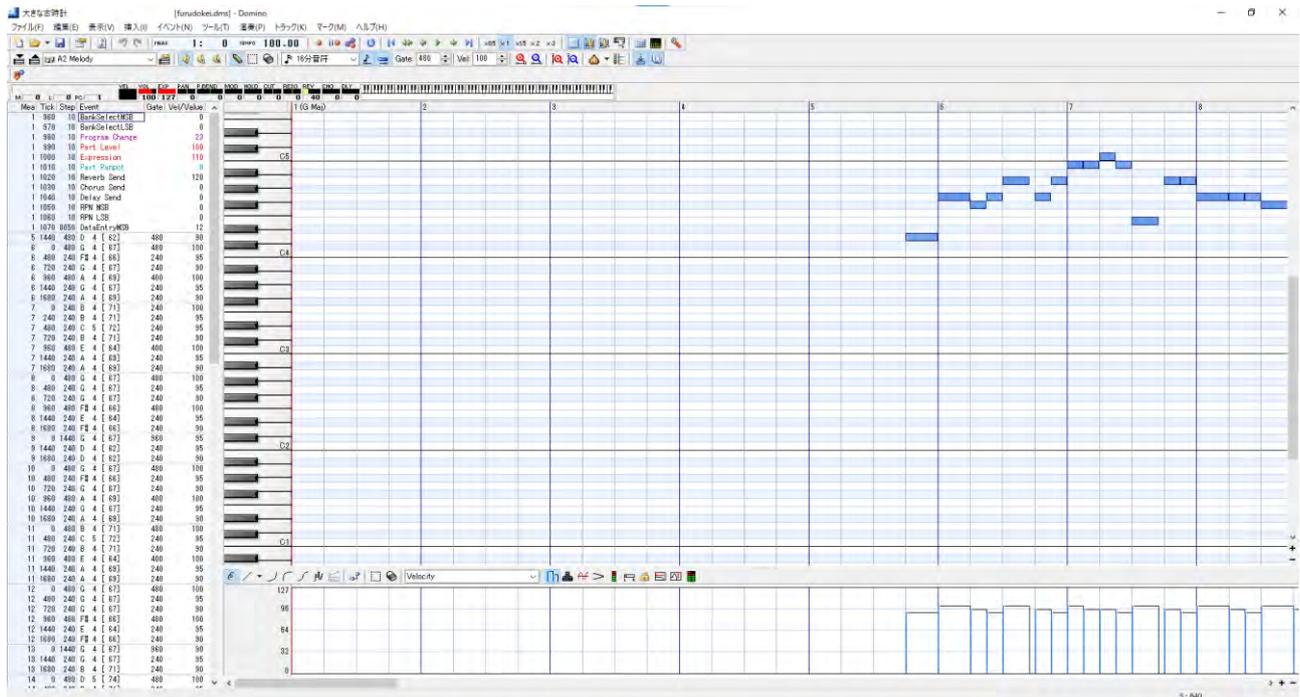


図 4-5 Domino メイン画面 (MIDI ファイル開いた状態)

図 4-6 に示す “演奏-演奏開始/停止” ボタン、または “スペースキー” を押すと演奏を開始します。



図 4-6 Domino 演奏

5. サンプルプログラム

サンプルプログラムは、ルネサスエレクトロニクスホームページから入手してください。

6. 参考ドキュメント

RL78 ファミリ MIDI インタフェースモジュール Software Integration System (R01AN7265J)

RL78 ファミリ SIS を用いた MIDI 演奏制御サンプルソフトウェア (R01AN7491J)

RL78/G16 ユーザーズマニュアルハードウェア編 (R01UH0980J)

RL78 ファミリユーザーズマニュアルソフトウェア編 (R01US0015J)

RL78/G16 Fast Prototyping Board ユーザーズマニュアル (R12UM0048)

(最新版をルネサスエレクトロニクスホームページから入手してください。)

テクニカルアップデート

(最新の情報をルネサスエレクトロニクスホームページから入手してください。)

MIDI シールド(SparkFun MIDI Shield)

<https://www.sparkfun.com/products/12898>

マトリクス LED

<https://wiki.52pi.com/index.php?title=EP-0075>

すべての商標および登録商標は、それぞれの所有者に帰属します。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2024.11.29	—	初版発行

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れしないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後、切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
 2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
 5. 当社製品を、全部または一部を問わず、改造、変更、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、変更、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
 6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
 7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
 8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
 9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
 11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとなります。
 13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。