



B-SA4TM-TS15 - Ethernet 内蔵型 CPU 接続用  
サンプルラダー説明書

---

**東芝テック株式会社**

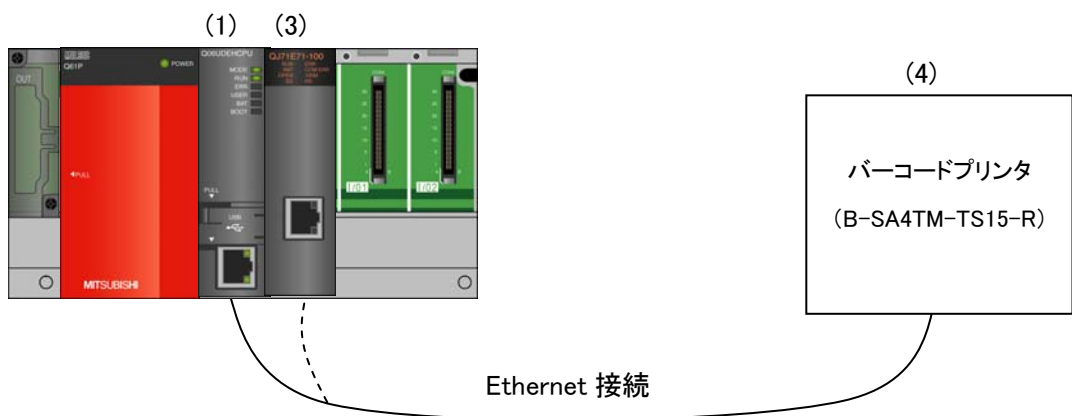
- 1. 概要..... 3
  - 1.1 サンプルラダー概要 ..... 3
  - 1.2 システム構成例 ..... 3
  - 1.3 Ethernet内蔵形QnUCPUユニット(Q06UDEHCPU)の設定 ..... 4
    - 1.3.1 Ethernet内蔵形QnUCPUユニット(Q06UDEHCPU)の動作設定 ..... 4
    - 1.3.2 Ethernet内蔵形QnUCPUユニット(Q06UDEHCPU)のオープン設定 ..... 5
  - 1.4 EthernetユニットQJ71E71-100 の設定 ..... 6
    - 1.4.1 EthernetユニットQJ71E71-100 の動作設定 ..... 6
    - 1.4.2 EthernetユニットQJ71E71-100 のイニシャル設定..... 7
    - 1.4.3 EthernetユニットQJ71E71-100 のオープン設定 ..... 8
- 2. サンプルラダー詳細 ..... 9
  - 2.1 プログラム全体 ..... 9
  - 2.2 プログラム詳細 ..... 13
    - 2.2.1 実行時初期処理..... 13
    - 2.2.2 実行待機状態..... 14
    - 2.2.3 実行開始処理..... 15
    - 2.2.4 初期化処理 ..... 16
    - 2.2.5 回線オープン処理..... 18
    - 2.2.6 データ送信処理 ..... 20
    - 2.2.7 タイムアウト判定処理..... 22
    - 2.2.8 終了処理 ..... 23
- 3. サンプルラベル内容 ..... 24

1. 概要

1.1 サンプルラダー概要

本サンプルラダーは、Ethernet 内蔵 CPU を利用して、東芝テック製バーコードプリンタ(B-SA4TM-TS15-R)で QR コードを印字するサンプルラダーです。

1.2 システム構成例



No.	機器名	説明
1	Q シリーズ シーケンサ	ベースユニット、電源ユニット、Q シリーズ Ethernet 内蔵形 QnUCPU ユニット (シリアル No.の上 5 桁が“11012”以降の CPU ユニット)を使用します。
		モデル適用 CPU 形名
		MELSEC-Q シリーズユニバーサルモデル
		※QCPU(A モード)使用不可
2	GX Developer	Version8.78G 以降を使用します。
3	QJ71E71-100	Q シリーズ シーケンサ Ethernet ユニット
4	バーコードプリンタ	東芝テック製バーコードプリンタ(B-SA4TM-TS15-R)

### 1.3 Ethernet内蔵形QnUCPUユニット(Q06UDEHCPU)の設定

#### 1.3.1 Ethernet内蔵形QnUCPUユニット(Q06UDEHCPU)の動作設定

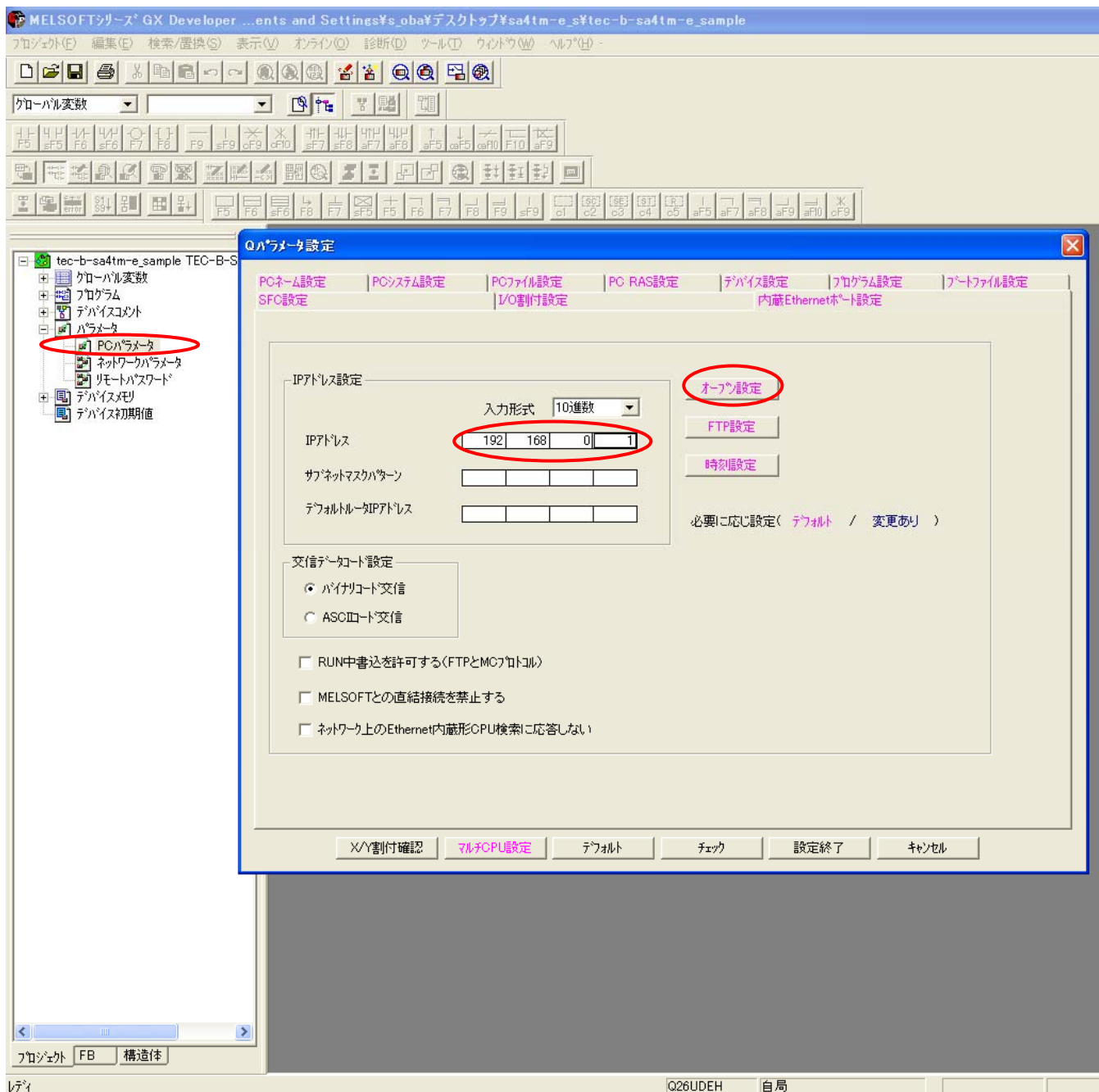
本プログラムで使用する Ethernet 内蔵形 QnUCPU ユニット(Q06UDEHCPU)の設定を説明します。

GX Developer を用いて、以下の項目を設定します。

(1)IP アドレス

Ethernet ユニットの IP アドレスを「192.168.0.1」と設定します。

下図は GX Developer です。



1. 3. 2 Ethernet内蔵形QnUCPUユニット(Q06UDEHCPU)のオープン設定

(1)ソケット通信設定

コネクション No.はシステムに合わせて選択してください。  
このコネクション No.はラダーに入力するコネクション No.となります。

- (a)プロトコル

「TCP」を設定します。
- (b)オープン方式

「ソケット通信」を指定します。
- (c)TCP 接続方式

「Active」を指定します。
- (d)自局ポート番号(16 進数)

パートナー様接続機器の設定に合わせて、16 進数で入力します。  
「0BB8」を入力します。
- (e)交信相手 IP アドレス

パートナー様接続機器設定に合わせて、10 進数で入力します。  
「192.168.0.2」を入力します。
- (f)交信相手ポート番号(16 進数)

パートナー様接続機器の設定に合わせて、16 進数で入力します。  
「0BB8」を入力します。

GX Developer を用いて、2 つのコネクションオープン設定をします。

内蔵Ethernetポート オープン設定

	プロトコル	オープン方式	TCP接続方式	自局 ポート番号	交信相手 IPアドレス	交信相手 ポート番号
1	TCP	ソケット通信	Active	0BB8	192.168.0.2	0BB8
2	TCP	MELSOFT接続				
3	TCP	MELSOFT接続				
4	TCP	MELSOFT接続				
5	TCP	MELSOFT接続				
6	TCP	MELSOFT接続				
7	TCP	MELSOFT接続				
8	TCP	MELSOFT接続				
9	TCP	MELSOFT接続				
10	TCP	MELSOFT接続				
11	TCP	MELSOFT接続				
12	TCP	MELSOFT接続				
13	TCP	MELSOFT接続				
14	TCP	MELSOFT接続				
15	TCP	MELSOFT接続				
16	TCP	MELSOFT接続				

自局ポート番号、交信相手ポート番号：ポート番号を16進数で入力してください。

設定終了

キャンセル

## 1. 4 EthernetユニットQJ71E71-100 の設定

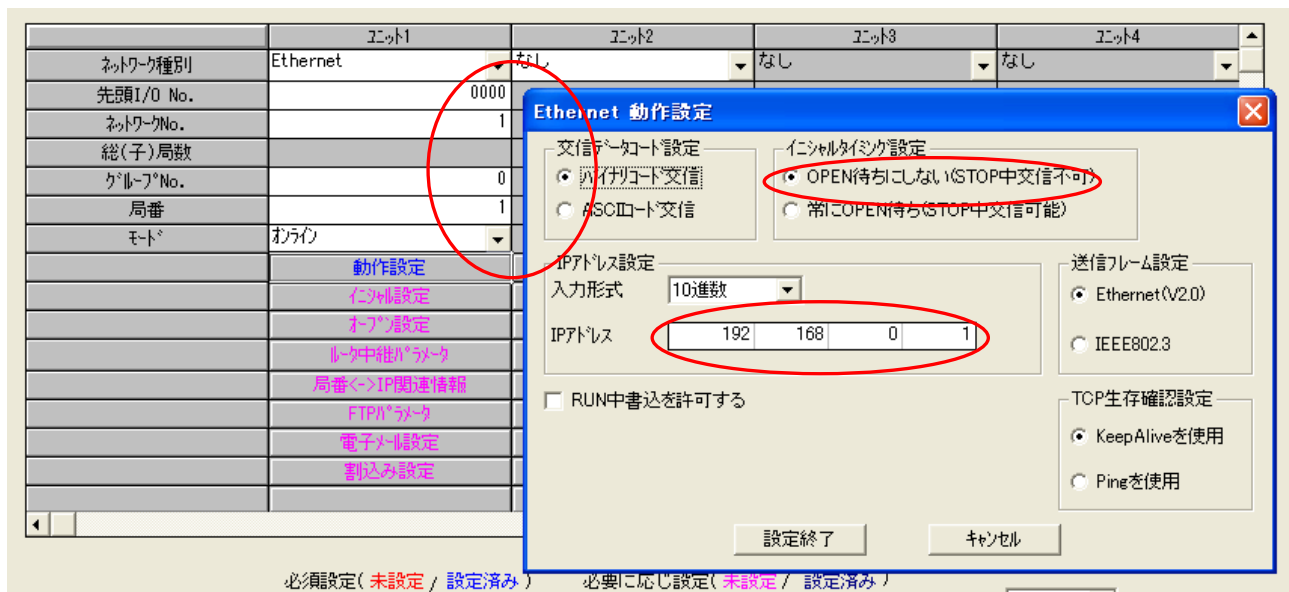
### 1. 4. 1 EthernetユニットQJ71E71-100 の動作設定

本プログラムで使用する QJ71E71-100 の設定を説明します。

GX Developer を用いて、以下の項目を設定します。

- |                  |   |
|------------------|---|
| (1) 先頭 I/O No.   | 装着ユニット No.に合わせて入力します。<br>スロット 0 に装着した場合、「0000」を設定します。         |
| (2) ネットワーク No.   | ネットワーク構成に合わせて入力します。<br>「1」を設定します。                             |
| (3) 局番           | ネットワーク構成に合わせて入力します。<br>「1」を設定します。                             |
| (4) 自局 IP アドレス   | Ethernet ユニットの IP アドレスを設定します。<br>本プログラムでは「192.168.0.1」を設定します。 |
| (5) イニシャルタイミング設定 | 「OPEN 待ちにしない(STOP 中交信不可)」を選択します。                              |

下図は GX Developer です。



1. 4. 2 EthernetユニットQJ71E71-100 のイニシャル設定

QJ71E71-100 の生存確認の設定値は、以下の項目を設定することで変更が可能です。

- 相手先 生存確認開始間隔タイマ
- 相手先 生存確認間隔タイマ
- 相手先 生存確認再送回数

タイマ設定

設定値をブラウザで設定すると、ユニットはデフォルト値で動作します。

	設定値	デフォルト値	単位
TCP ULPタイマ		60	×500ms
TCP セロウインドウタイマ		20	×500ms
TCP 再送タイマ		20	×500ms
TCP 終了タイマ		40	×500ms
IP組立てタイマ		10	×500ms
レスポンス監視タイマ		60	×500ms
相手先 生存確認開始間隔タイマ		1200	×500ms
相手先 生存確認間隔タイマ		20	×500ms
相手先 生存確認再送回数		3	回

DNS設定

入力形式 10進数

DNSサーバ1のIPアドレス				
DNSサーバ2のIPアドレス				
DNSサーバ3のIPアドレス				
DNSサーバ4のIPアドレス				

設定終了

キャンセル

1. 4. 3 EthernetユニットQJ71E71-100 のオープン設定

(1)TCP 通信設定

コネクション No.はシステムに合わせて選択して下さい。  
このコネクション No.はラダーに入力するコネクション No.となります。

- (a)プロトコル

「TCP」と設定します。
- (b)オープン方式

「Active」と設定します。
- (c)固定バッファ送信手順

「手順無し」と設定します。
- (d)ペアリングオープン

「ペアにする」と設定します。
- (e)生存確認

「確認する」と設定します。
- (f)自局ポート番号(16 進数)

バーコードプリンタの設定に合わせて 16 進数で入力します。  
「0BB8」を入力します。
- (g)送信相手 IP アドレス

バーコードプリンタの設定に合わせて 10 進数で入力します。  
「192.168.0.2」を入力します。
- (h)送信相手ポート番号(16 進数)

バーコードプリンタの設定に合わせて 16 進数で入力します。  
「0BB8」を入力します。

GX Developer を用いて、2 つのコネクションオープン設定をします。

	プロトコル	オープン方式	固定バッファ	固定バッファ 送信手順	ペアリング オープン	生存確認	自局 ポート番号	送信相手 IPアドレス	送信相手 ポート番号
1	TCP	Active	受信	手順無し	ペアにする	確認する	0BB8	192.168. 0. 2	0BB8
2	TCP	Active	送信	手順無し	ペアにする	確認する	0BB8	192.168. 0. 2	0BB8
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									

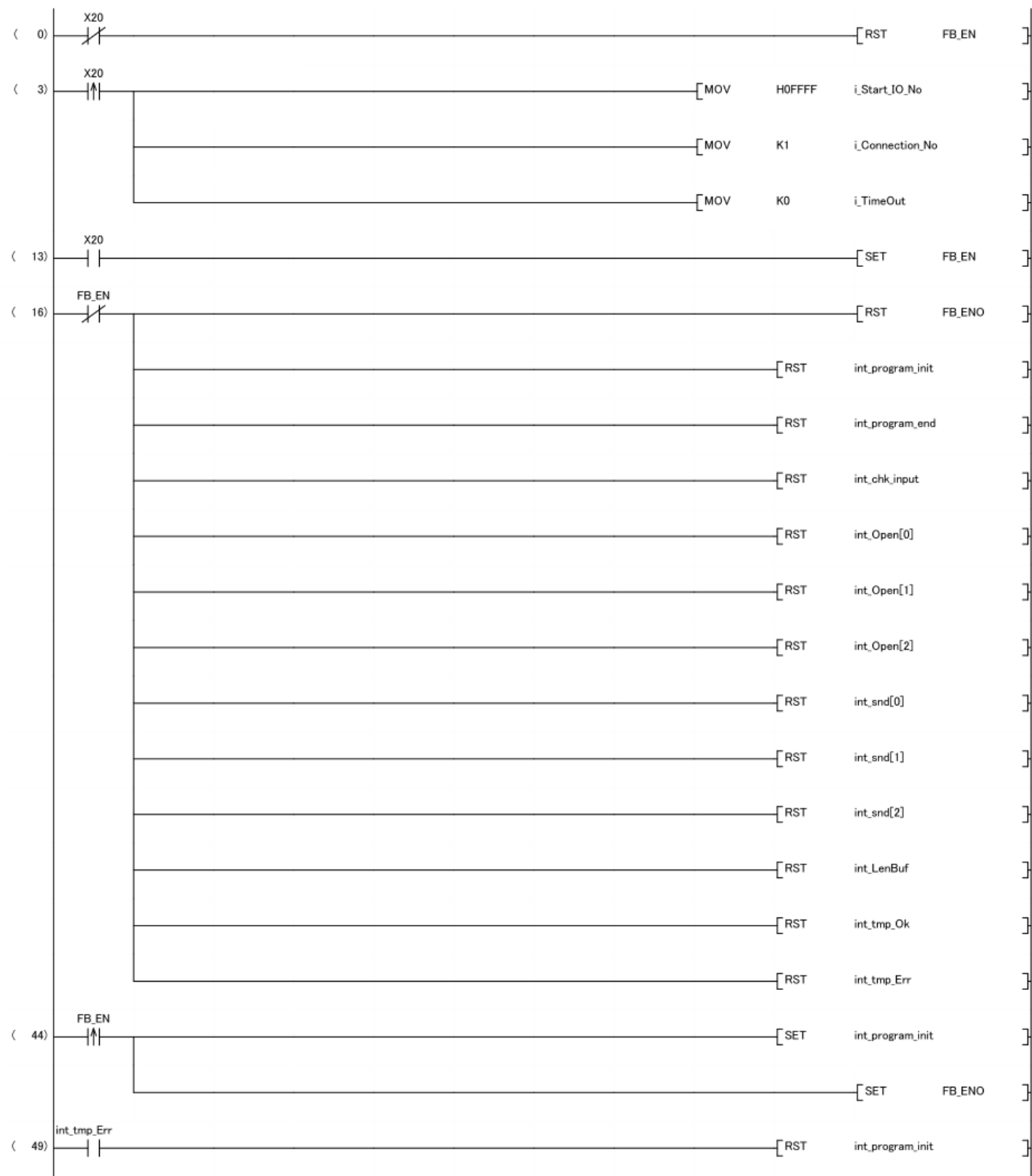
設定終了

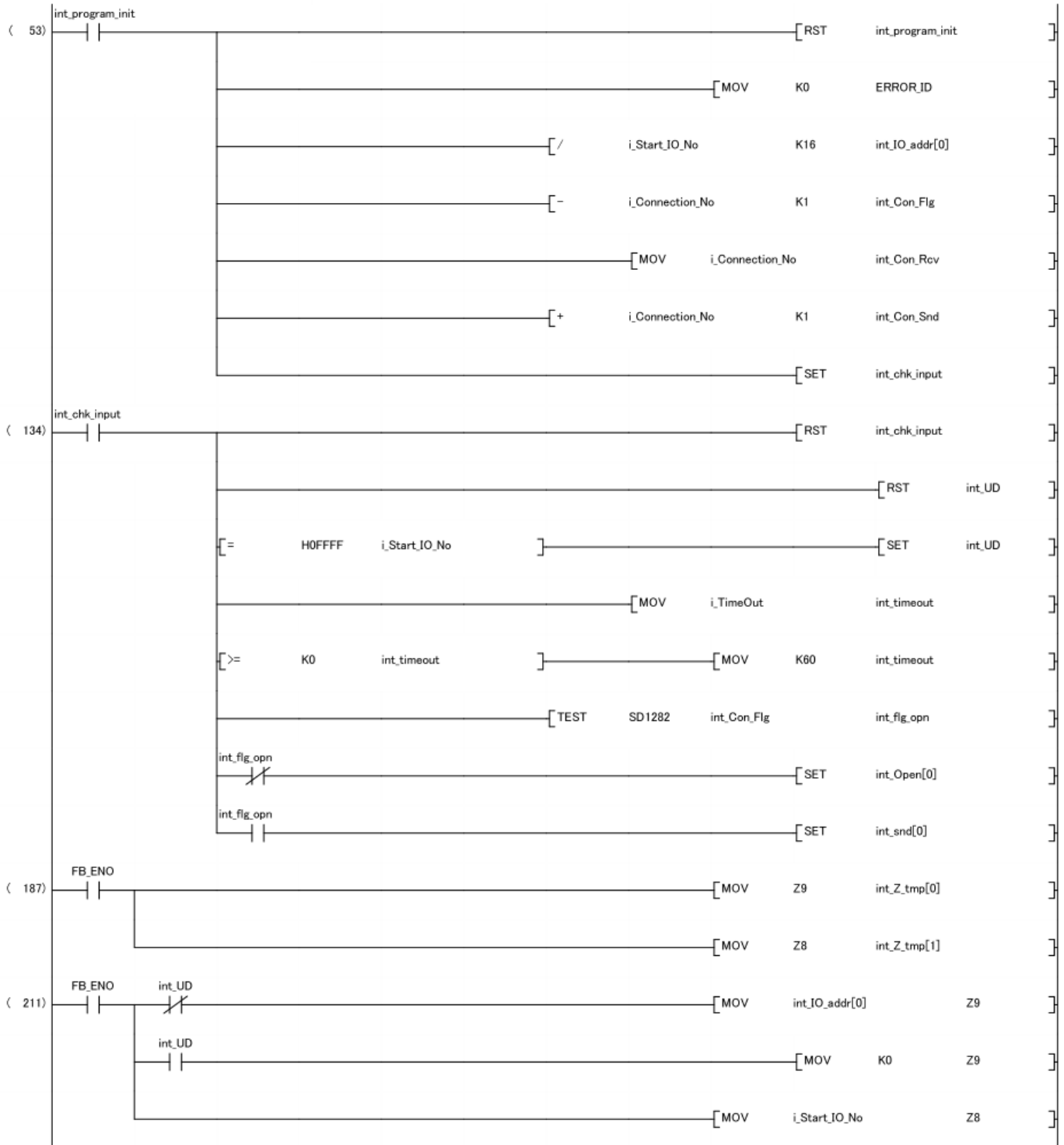
キャンセル

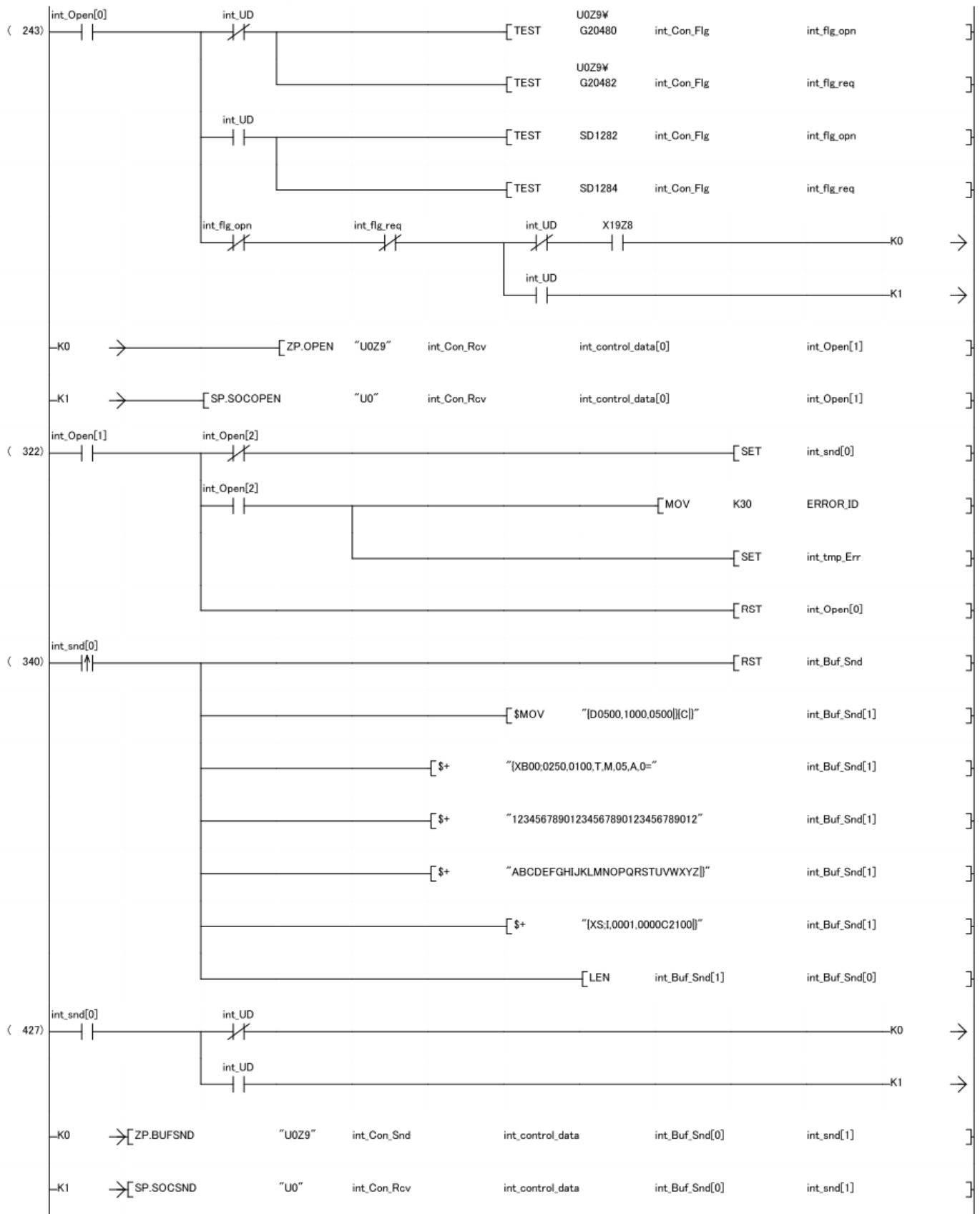


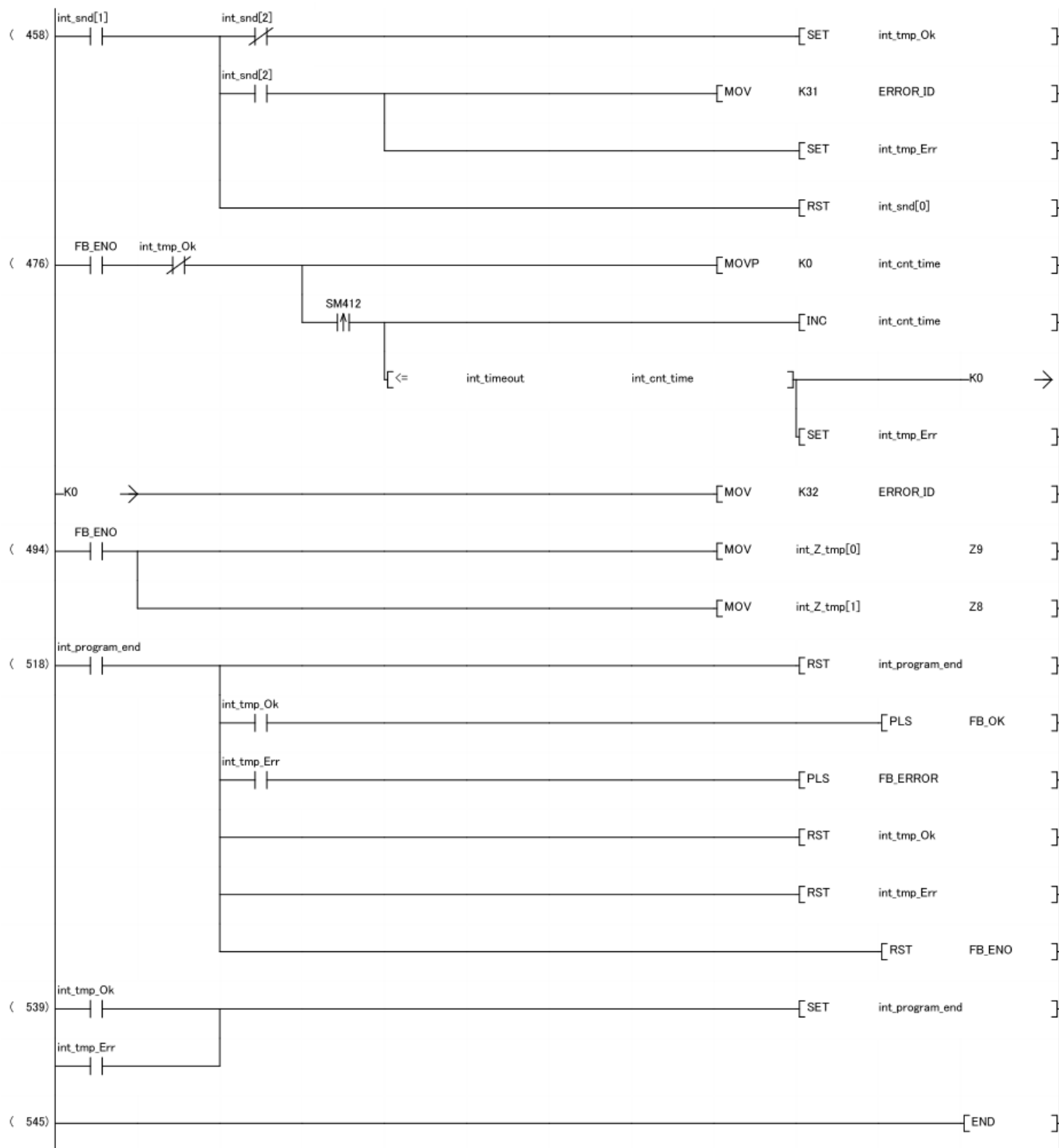
2. サンプルラダー詳細

2.1 プログラム全体









2. 2 プログラム詳細

2. 2. 1 実行時初期処理



X20:OFF 時

処理	説明
RST FB_EN	実行トリガを OFF します。

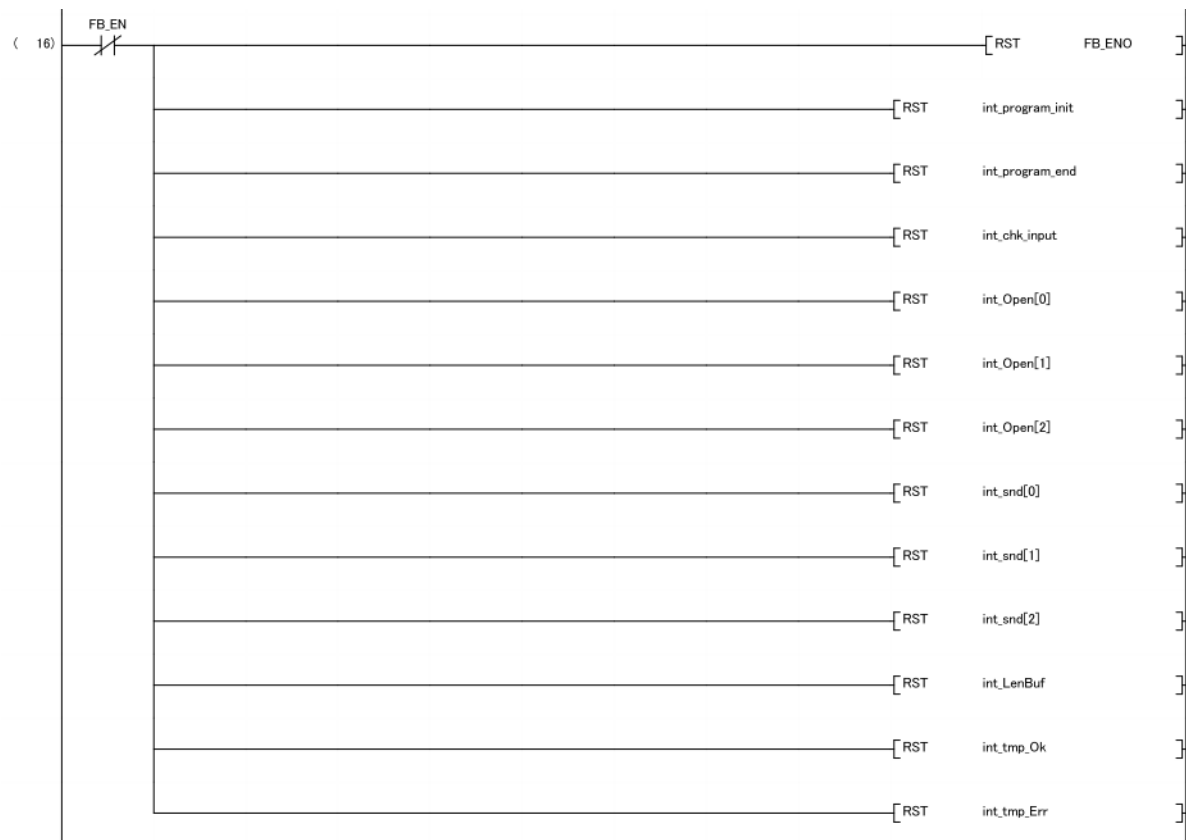
X20:ON 時(立ち上がりパルス)

処理	説明
MOV H0FFFF i_StartIO_No	対象の Ethernet ユニットが設定されている先頭 I/O アドレスとして 0FFFF をセットします。
MOV K1 i_Connection_No	コネクションを確立するコネクション No として 1 をセットします。
MOV K0 i_TimeOut	処理開始から終了までのタイムアウトの時間として 0 をセットします。(本件の場合 0 をセットすると自動的に 60 秒がセットされます)

X20:ON 時

処理	説明
SET FB_EN	実行トリガを ON します。

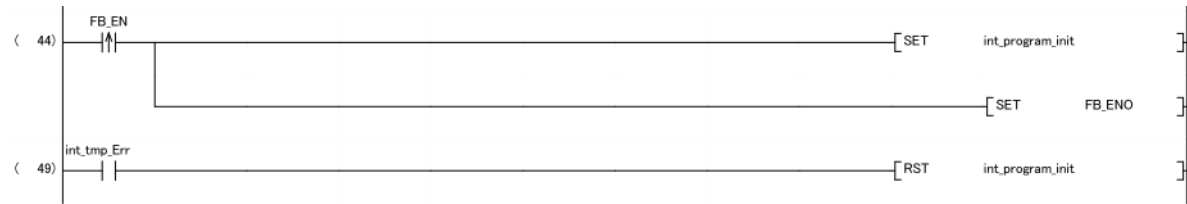
2. 2. 2 実行待機状態



FB\_EN:OFF 時

処理	説明
RST FB_ENO	実行トリガを OFF します。
RST int_program_init	int_program_init を OFF します。
RST int_program_end	int_program_end を OFF します。
RST int_chk_input	int_chk_input を OFF します。
RST int_Open[0]/[1]/[2]	int_Open[0]～[2] を OFF します。
RST int_snd[0]/[1]/[2]	int_snd[0]～[2] を OFF します。
RST int_LenBuf	int_LenBuf を OFF します。
RST int_tmp_Ok	int_tmp_Ok を OFF します。
RST int_tmp_Err	int_tmp_Err を OFF します。

2. 2. 3 実行開始処理



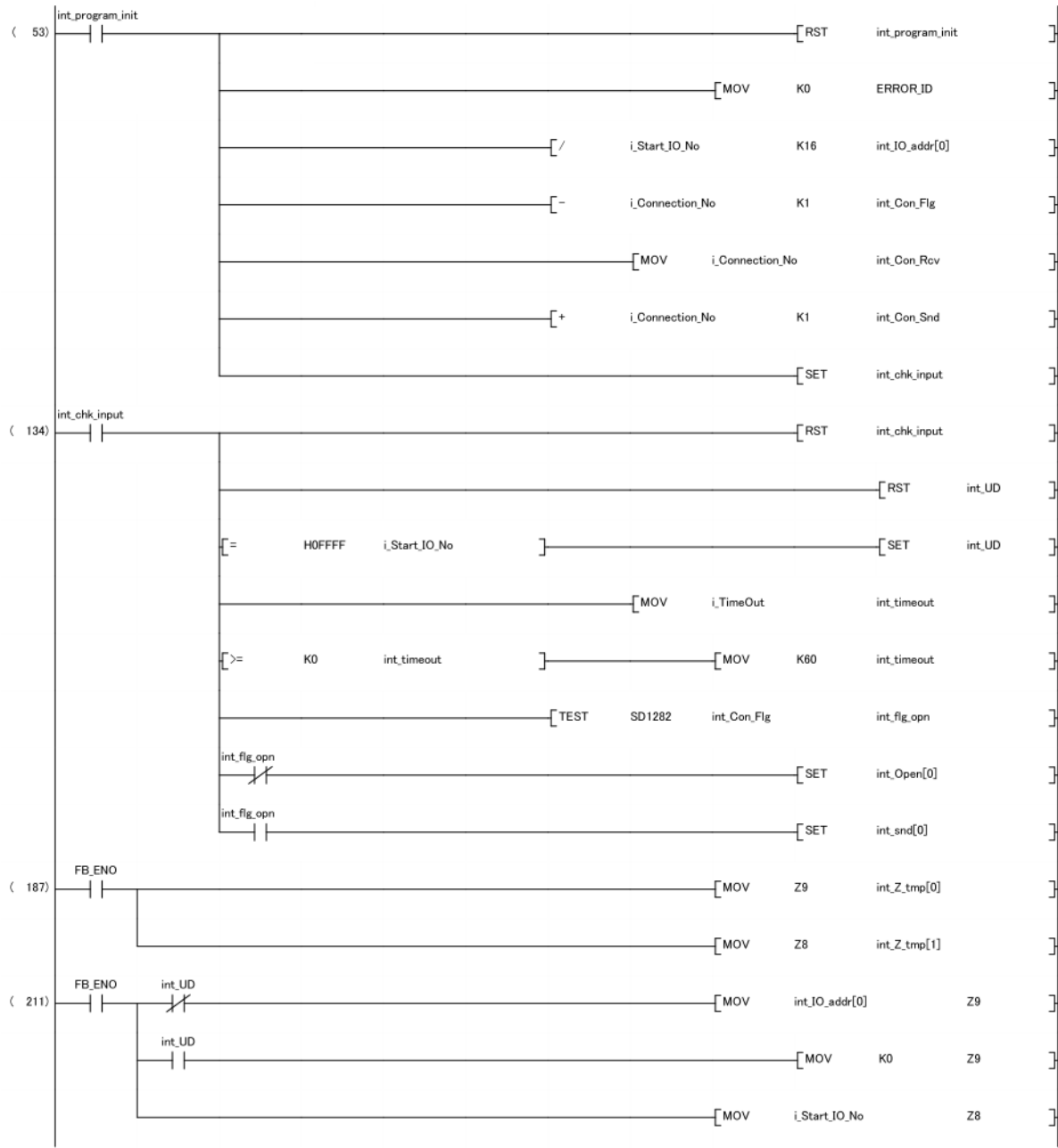
FB\_EN:ON 時(立ち上がりパルス)

処理	説明
SET int_program_init	int_program_init を ON します。
SET FB_ENO	FB_ENO を ON します。

int\_tmp\_Err:ON 時

処理	説明
RST int_program_init	int_program_init を OFF します。

2. 2. 4 初期化处理



int\_program\_init:ON 時

処理	説明
RST int_program_init	int_program_init を OFF します。
MOV K0 ERROR_ID	ERROR_ID に 0 をセットします。
/ i_Start_IO_No K16 int_IO_addr[0]	I_Start_IO_No を 16 で割り、得られた値を int_IO_addr[0]にセットします。
- i_Connection_No K1 int_Con_Flg	i_Connection_No より 1 を減じ、得られた値を int_ConFlg にセットします。
MOV i_Connection_No int_Con_Rcv	i_Connection_No の値を int_ConRcv にセットします。
+ i_Connection_No K1 int_Con_Snd	i_Connection_No に 1 を加え、得られた値を int_Con_Snd にセットします。



処理	説明
SET int_chk_input	int_chk_input を ON します。

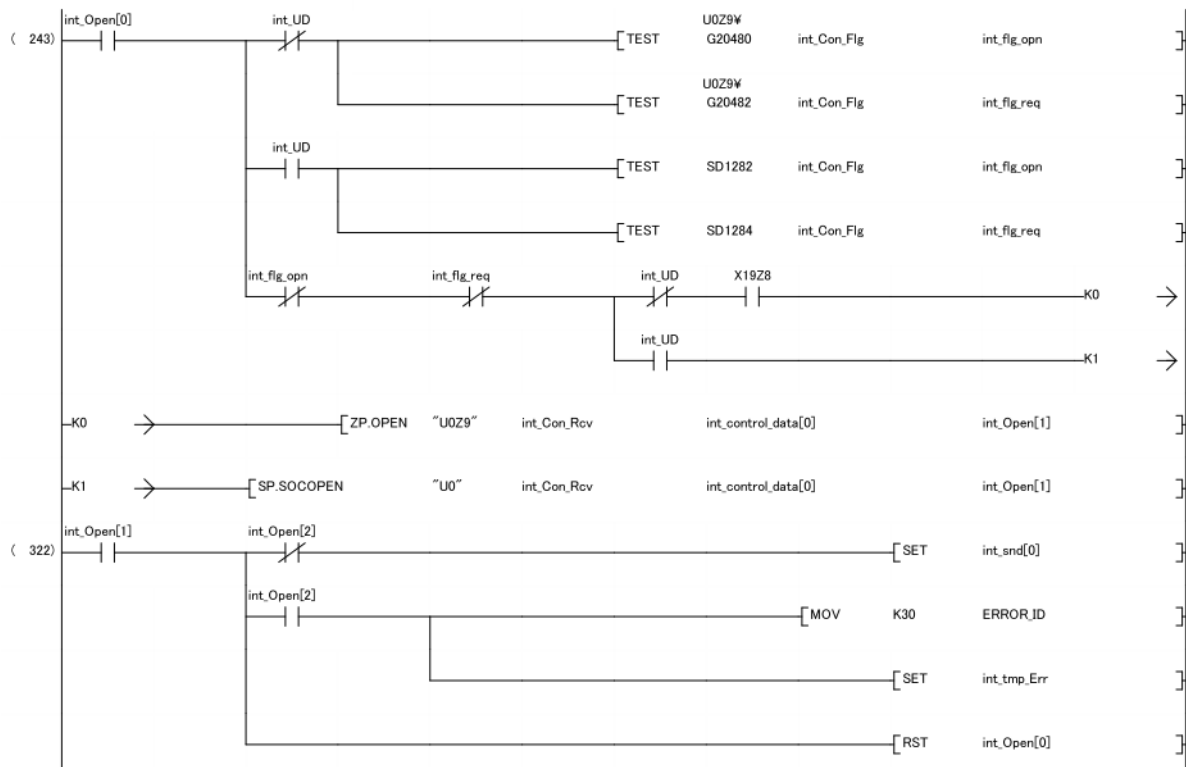
int\_chk\_input: ON 時

処理	説明
RST int_chk_input	int_chk_input を OFF します。
RST int_UD	int_UD を OFF します。
SET int_UD	int_UD を ON します。 (i_Start_IO_No が H0FFFF の場合のみ)
MOV i_TimeOut int_timeout	I_TimeOut の値を int_timeout にセットします。
MOV K60 int_timeout	int_timeout に 60 をセットします。 (セット前の int_timeout に入っている値が 0 以下の場合のみ)
TEST SD1282 int_Con_Flg int_flg_opn	SD1282 内の int_Con_Flg の位置にあるビットデータを int_flg_opn にセットします。
SET int_Open[0]	int_Open[0]を ON します。 (int_flg_opn が OFF の場合のみ)
SET int_snd[0]	int_snd[0]を ON します。 (int_flg_opn が ON の場合のみ)

FB\_ENO:ON 時

処理	説明
MOV Z9 int_Z_tmp[0]	int_Z_tmp[0]に Z9 の値をセットします。
MOV Z8 int_Z_tmp[1]	int_Z_tmp[1]に Z8 の値をセットします。
MOV int_IO_addr[0] Z9	Z9 に int_IO_addr[0]の値をセットします。 (int_UD が OFF の場合のみ)
MOV K0 Z9	Z9 に 0 をセットします。 (int_UD が OFF の場合のみ)
MOV int_IO_addr[1] Z8	Z8 に int_IO_addr[1]の値をセットします。

2. 2. 5 回線オープン処理



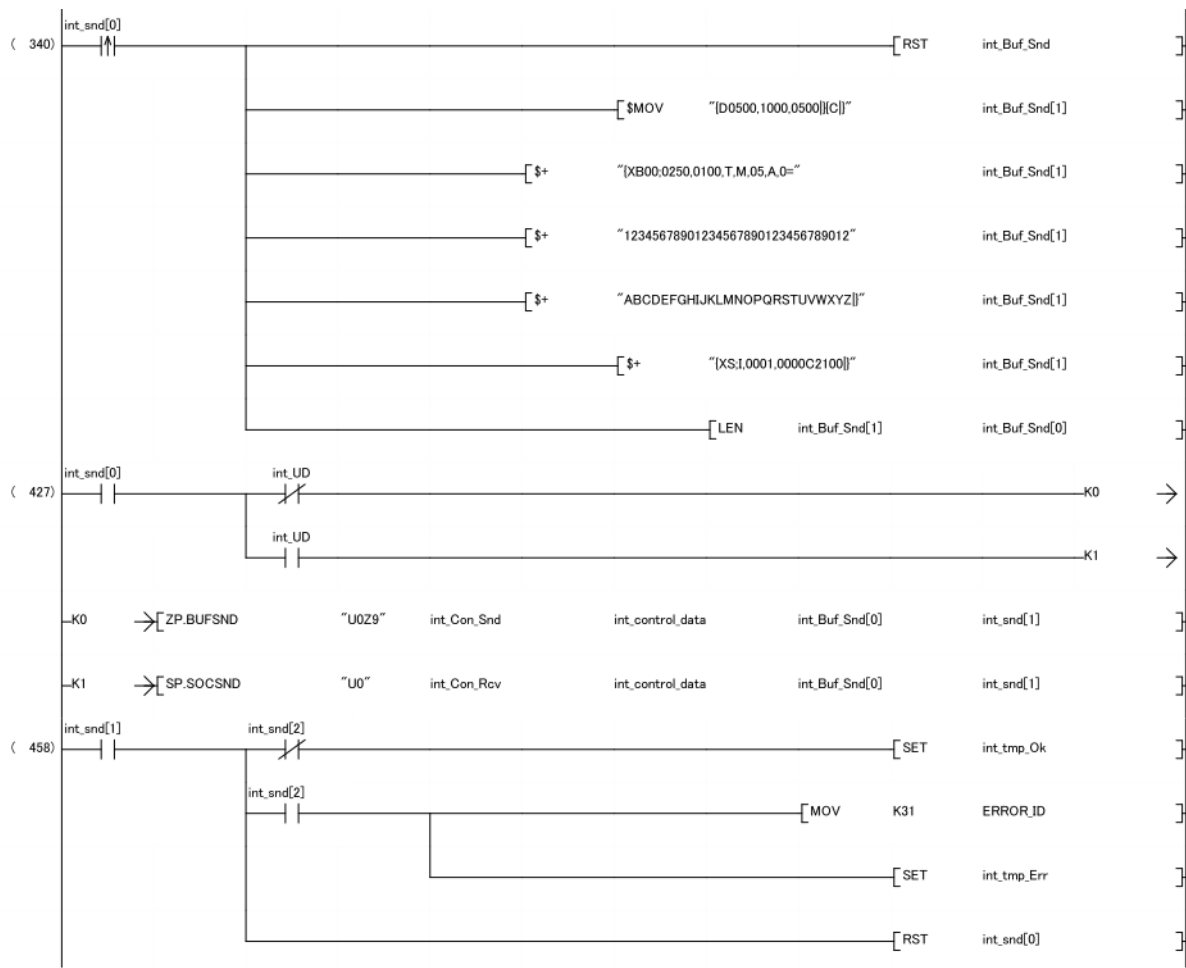
int\_Open[0]: ON 時

処理	説明
TEST U0Z9¥G20480 int_Con_Flg int_flg_opn	U0Z9¥G20480 内の int_Con_Flg の位置にあるビットデータを int_flg_opn にセットします。 (int_UD が OFF の場合のみ)
TEST U0Z9¥G20482 int_Con_Flg int_flg_req	U0Z9¥G20482 内の int_Con_Flg の位置にあるビットデータを int_flg_req にセットします。 (int_UD が OFF の場合のみ)
TEST SD1284 int_Con_Flg int_flg_opn	SD1284 内の int_Con_Flg の位置にあるビットデータを int_flg_opn にセットします。 (int_UD が ON の場合のみ)
TEST SD1284 int_Con_Flg int_flg_req	SD1284 内の int_Con_Flg の位置にあるビットデータを int_flg_req にセットします。 (int_UD が ON の場合のみ)
ZP.OPEN “U0Z9” int_Con_Rcv int_control_data[0] int_Open[1]	前述までに指定された内容で回線をオープンします。 (int_flg_opn、int_flg_req、int_UD が OFF され、X19Z8 が ON されている場合のみ)
SP.SOCOPEN “U0” int_Con_Rcv int_control_data[0] int_Open[1]	前述までに指定された内容で回線をオープンします。 (int_flg_opn、int_flg_req が OFF され、int_UD が ON されている場合のみ)

int\_Open[1]: ON 時

処理	説明
SET int_snd[0]	int_snd[0]を ON します。 (int_Open[2]が OFF されている場合のみ)
MOV K30 ERROR_ID	ERROR_ID に 30 をセットします。 (int_Open[2]が ON されている場合のみ)
SET int_tmp_Err	int_tmp_Err を ON します。 (int_Open[2]が ON されている場合のみ)
RST int_Open[0]	int_Open[0]を OFF します。

2. 2. 6 データ送信処理



int\_snd[0]: ON 時(立ち上がりパルス)

処理	説明
RST int_Buf_Snd	int_Buf_Snd を OFF します。
\$MOV "{D0500,1000,0500}[C]}" int_Buf_Snd[1]	int_Buf_Snd[1]より"{D0500,1000,0500}[C]}"をセットします。 (ラベルフォーマット設定、バッファクリア処理)
\$+ "{XB00;0250,0100,T,M,05,A,0=" int_Buf_Snd[1]	int_Buf_Snd[1]より続く文字列に "{XB00;0250,0100,T,M,05,A,0="を連結します。 (QRコードフォーマット設定です)
\$+ "12345678901234567890123456789012" int_Buf_Snd[1]	int_Buf_Snd[1]より続く文字列に "12345678901234567890123456789012"を連結します。 (QRコード印字文字列設定です)
\$+ "ABCDEFGHIJKLMNOPQRSTUVWXYZ" int_Buf_Snd[1]	int_Buf_Snd[1]より続く文字列に "ABCDEFGHIJKLMNOPQRSTUVWXYZ"を連結します。 (QRコード印字文字列設定です)
\$+ "{XS;I,0001,0000C2100}" int_Buf_Snd[1]	int_Buf_Snd[1]より続く文字列に "{XS;I,0001,0000C2100}"を連結します。 (ラベル発行コマンドです)

処理	説明
LEN int_Buf_Snd[1] int_Buf_Snd[0]	int_Buf_Snd[0]に int_Buf_Snd[1]以降にセットされた文字列の長さをセットします。

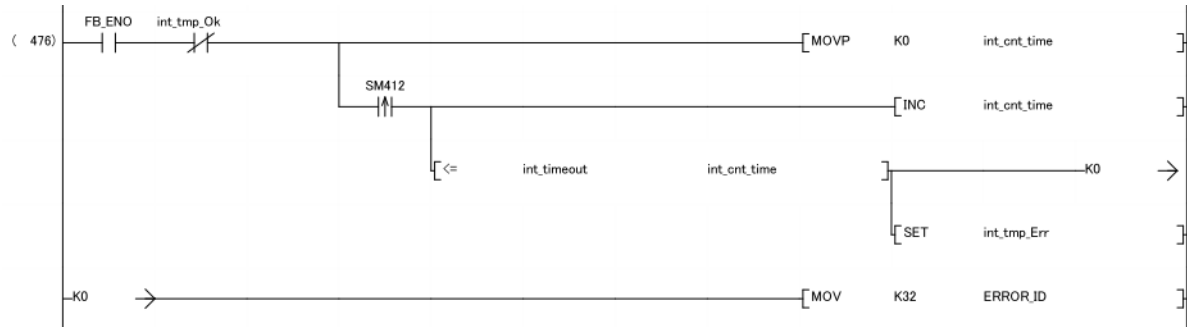
int\_snd[0]: ON 時

処理	説明
ZP.BUFSND “U0Z9” int_Con_Snd int_control_data int_Buf_Snd[0] int_snd[1]	上記処理でセットされた文字列を、オープンされている回線で送信します。 (int_UD が OFF の場合のみ)
SP.SOCSND “U0” int_Con_Rcv int_control_data int_Buf_Snd[0] int_snd[1]	上記処理でセットされた文字列を、オープンされている回線で送信します。 (int_UD が ON の場合のみ)

int\_snd[1]: ON 時

処理	説明
SET int_tmp_OK	int_tmp_Ok を ON します。 (int_snd[2]が OFF の場合のみ)
MOV K31 ERROR_ID	ERROR_ID に31をセットします。 (int_snd[2]が ON の場合のみ)
SET int_tmp_Err	int_tmp_Err を ON します。 (int_snd[2]が ON の場合のみ)
RST int_snd[0]	int_snd[0]を OFF します。

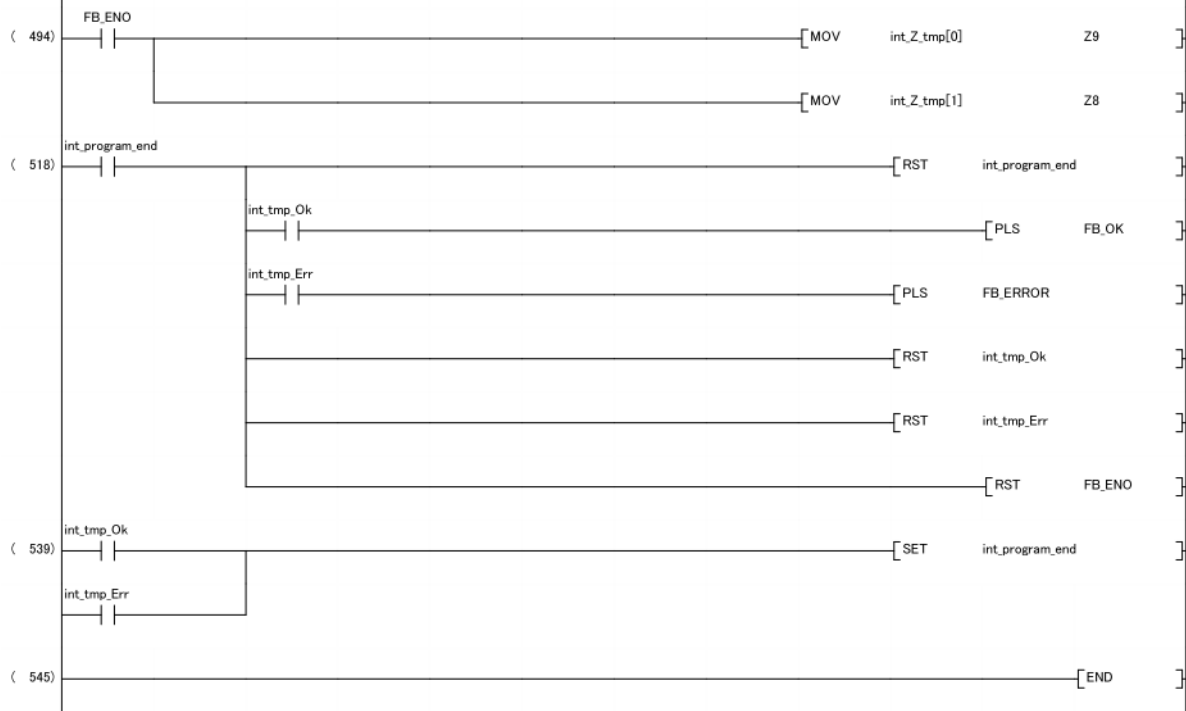
2. 2. 7 タイムアウト判定処理



FB\_ENO:ON 時かつ int\_tmp\_Ok:OFF 時

処理	説明
MOVP K0 int_cnt_time	int_cnt_time に 0 をセットします。 (処理実行1回につき1回、初回遷移時のみ)
INC int_cnt_time	int_cnt_time を1ずつ加算します。
MOV K32 ERROR_ID	ERROR_ID に 32 をセットします。 (int_cnt_time が int_timeout 以上になった場合のみ)
SET int_tmp_Err	int_tmp_Err を ON します。 (int_cnt_time が int_timeout 以上になった場合のみ)

2. 2. 8 終了処理



FB\_ENO:ON 時

処理	説明
MOV int_Z_tmp[0] Z9	Z9 に int_Z_tmp[0]の値をセットします。
MOV int_Z_tmp[1] Z8	Z8 に int_Z_tmp[1]の値をセットします。

Int\_program\_end: ON 時

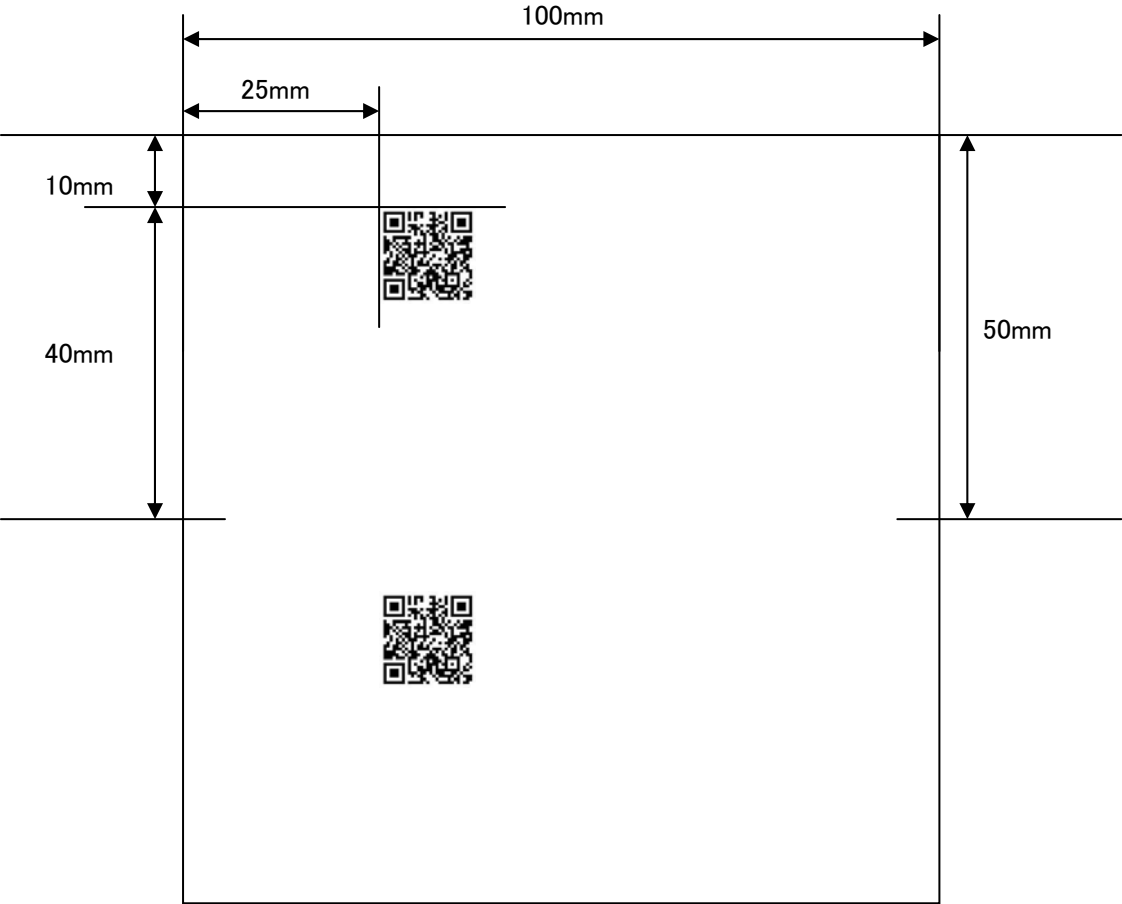
処理	説明
RST int_program_end	int_program_end を OFF します。
PLS FB_OK	FB_OK を1スキャンのみ ON します。 (int_tmp_Ok が ON 時のみ)
PLS FB_ERROR	FB_ERROR を 1 スキャンのみ ON します。 (int_tmp_Err が ON 時のみ)
RST int_tmp_Ok	int_tmp_Ok を OFF します
RST int_tmp_Err	int_tmp_Err を OFF します
RST FB_ENO	FB_ENO を OFF します

int\_tmp\_Ok: ON 時又は int\_tmp\_Err:ON 時

処理	説明
SET int_program_end	int_program_end を ON します。

3. サンプルラベル内容

下図はイメージです。



タグのサイズ等

項目	位置、長さ等
タグピッチ	50mm
有効印字幅	100mm
有効印字長	50mm

QRコードについて

項目	位置、長さ等
QRコードの基点 X 座標	25mm
QRコードの基点 Y 座標	10mm
誤り訂正レベル	標準
1 セル幅	5 ドット
モード指定	自動モード
QRコードの回転方向	0°
内容	12345678901234567890123456789012ABCDEF GHIJKLMNOPQ RSTUVWXYZ