

78K0/LF3 マンチェスタ・コード・ジェネレータ (Cソース版)

本サンプル・プログラムでは、マンチェスタ・コード・ジェネレータを使用したプログラムを提供します。無線送信など、マンチェスタ・コードを用いた送信をするアプリケーションにおいて、ソフト処理の負荷を低減可能です。転送レートは最大 250kbps まで、内蔵ボーレートジェネレータを用いて任意に設定可能です。また、送信データ長は 1bit ~ 8bit まで任意に設定可能です。送信バッファはダブルバッファとなっており、送信開始割り込みの処理で、次の送信データを設定することにより、連続で任意の bit 長の送信が可能です。

対象製品 : uPD78F049x

【目次】

1	概要	2
2	回路図	3
2.1	回路図	3
2.2	使用する周辺ハードウェア	3
3	ソフトウェアの構成	4
3.1	ファイルの構成	4
3.2	マイコン周辺機能リソース	5
3.3	動作概要	6
3.4	フローチャート	6
3.5	関数説明	7
3.5.1	関数構成図	7
3.5.2	関数一覧 (Lx3_MCG.asm ファイル内)	8
3.5.3	関数詳細	8
3.6	ソフトウェアの説明	9
4	リファレンス	9
5	Programming Code	10
6	改版履歴	11

《使用許諾条件》

本サンプル・プログラムおよび本資料は予告なく変更する場合があります。本サンプル・プログラムおよび本資料に記載された情報の使用に際して当社は当社もしくは第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。上記使用に起因する第三者所有の権利にかかわる問題が発生した場合、当社はその責を負うものではありません。本サンプル・プログラムおよび本資料に記載された情報は、半導体製品の動作例、応用例を説明するためのものです。したがってこれらの情報をお客様の機器に使用される場合には、お客様の責任において機器設計をしてください。これらの使用に起因するお客様もしくは第三者の損害に対して、当社は一切その責を負いません。

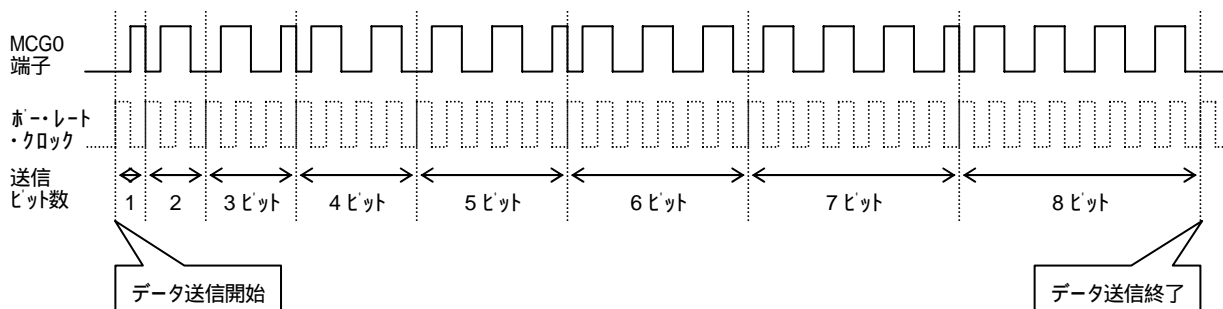
上記内容に合意の上で、ご使用ください。

1 概要

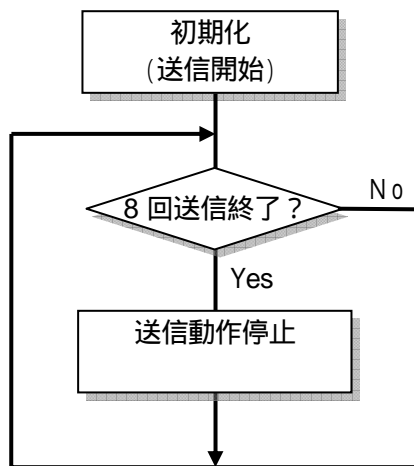
本サンプル・プログラムは、マンチェスタ・コード・ジェネレータを使用し、固定データ 0x55 を送信開始割り込みの発生毎に、送信ビット長を1ビット、2ビット・・・3ビット、8ビットの順に変更して送信し、8ビットのデータ送信が終了するとデータの送信を停止します。

リセット直後に一回だけ送信します。

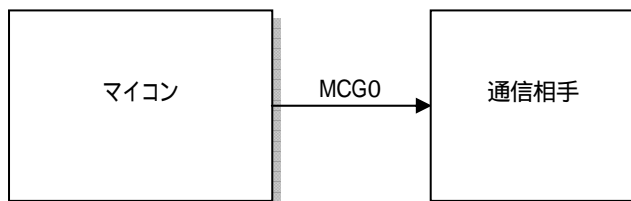
《送信タイミング》



《フローチャート概要》



《ブロック図/回路図イメージ》

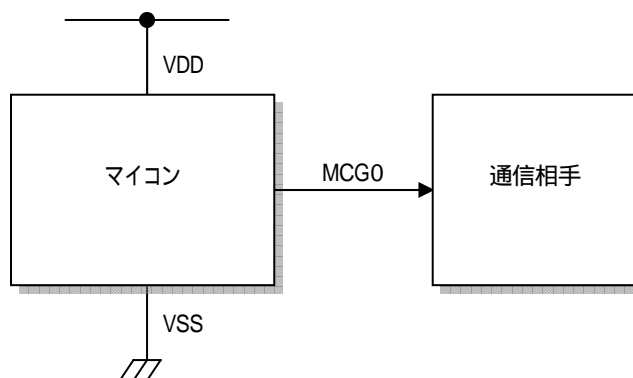


2 回路図

本サンプル・プログラムを動作させるための、回路図の例および使用する周辺ハードウェアを示します。

2.1 回路図

《回路図》



上図はリセット処置等のすべての必要な処置を示しているとは限りません。

2.2 使用する周辺ハードウェア

マンチェスタ・コードを受信できる通信相手。

3 ソフトウェアの構成

3.1 ファイルの構成

本ソフトウェアは、以下のファイルで構成されています。

ファイル名	機能	種別
Lx3_Mem.inc	オプション・バイトの設定値の定義	インクルード・ファイル (ユーザ向け定数定義)
Lx3_Mem.asm	オプション・バイトの設定	ソース・ファイル
Lx3_Res.h	メモリ・サイズ(IXS、IMS)の定義	ヘッダ・ファイル (ユーザ向け定数定義)
Lx3_Opt.h	オプション設定 (クロック、ビットレートの設定を行います。)	ヘッダ・ファイル (ユーザ向け定数定義)
Lx3_Sfr.h	ハードウェア初期設定値(SFR)の定義	ヘッダ・ファイル (ユーザ向け定数定義)
Lx3_Init.c	ハードウェア初期化処理	ソース・ファイル (ユーザ向けインタフェース処理)
Lx3_MCG.c	マンチェスタ・コード・ジェネレータ処理	ソース・ファイル (ユーザ向けインタフェース処理)

上記、7つのファイルはすべてコンパイルの対象としてください。インクルード・ファイル、ヘッダ・ファイルはソースと同じディレクトリ、またはサーチパスの通ったディレクトリに置いてください。

コンパイラオプション(-D)設定にて、品番の選択(記述)を行ってください。

【品番の選択】

品番	ROM	RAM
uPD78F0495	60KB	2KB
uPD78F0494	48KB	2KB
uPD78F0493	32KB	1KB
uPD78F0492	24KB	1KB
uPD78F0491	16KB	768B

上記、品番のシンボル定義により、Lx3_Res.h ファイルの各定義が品番別に定義されます。

その他、Lx3_Mem.inc ファイルにて、オプション・バイトの内容を選択定義、スタック・サイズの設定を行ってください。また、Lx3_Sfr.h ファイルにて各種 SFR レジスタの初期値の設定、Lx3_Opt.h ファイルにて使用するクロックの設定およびマンチェスタ・コード・ジェネレータのビットレート選択を行ってください。

3.2 マイコン周辺機能リソース

本ソフトウェアでは、マイコンの以下のリソースを使用します。

マイコン初期化部

リソース	内容		備考
RAM	スタック領域の確保	120 byte	標準スタートアップ・ルーチンにより確保されます。
ROM	標準スタートアップ・ルーチン	156 byte	
	オプション・バイト	5 byte	セグメント定義擬似命令により確保しています。
	マイコン初期化 / メイン処理	400 byte	ROM サイズはシリーズ、初期化ルーチンの呼び出し等により異なります。
割り込み	リセット入力 (RESET)		
その他	未使用		

マンチェスタ・コード・ジェネレータ処理部分

リソース	内容		備考
RAM	未使用		
ROM	サンプル・プログラム	62 byte	
関連ハードウェア	MCG		
割り込み	送信開始割り込み (INTMCG)		
その他	未使用		

3.3 動作概要

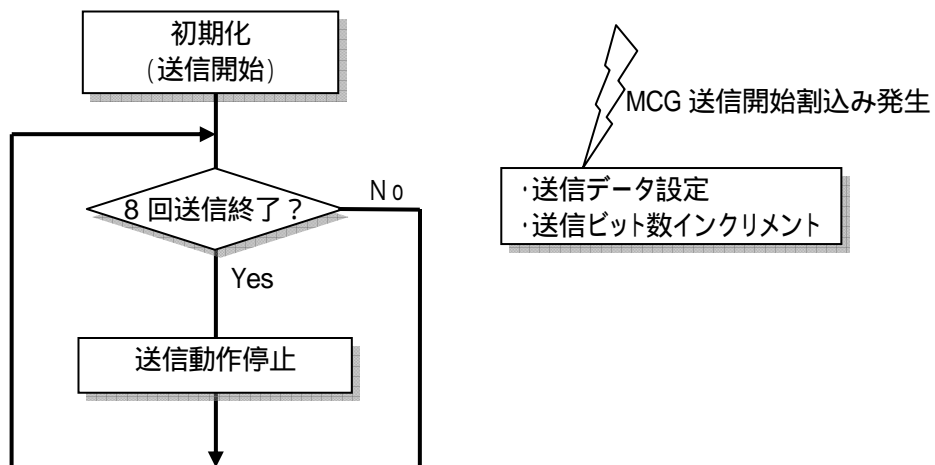
本サンプル・プログラムは、マンチェスタ・コード・ジェネレータを使用し、固定データ 0x55 を送信開始割り込みの発生毎に1ビット、2ビット…3ビット、8ビットの順に送信し、8ビットのデータ送信が終了するとデータの送信停止します。

リセット直後に一回だけ送信します。

3.4 フローチャート

以下に、本ソフトウェアのフローチャートを示します。

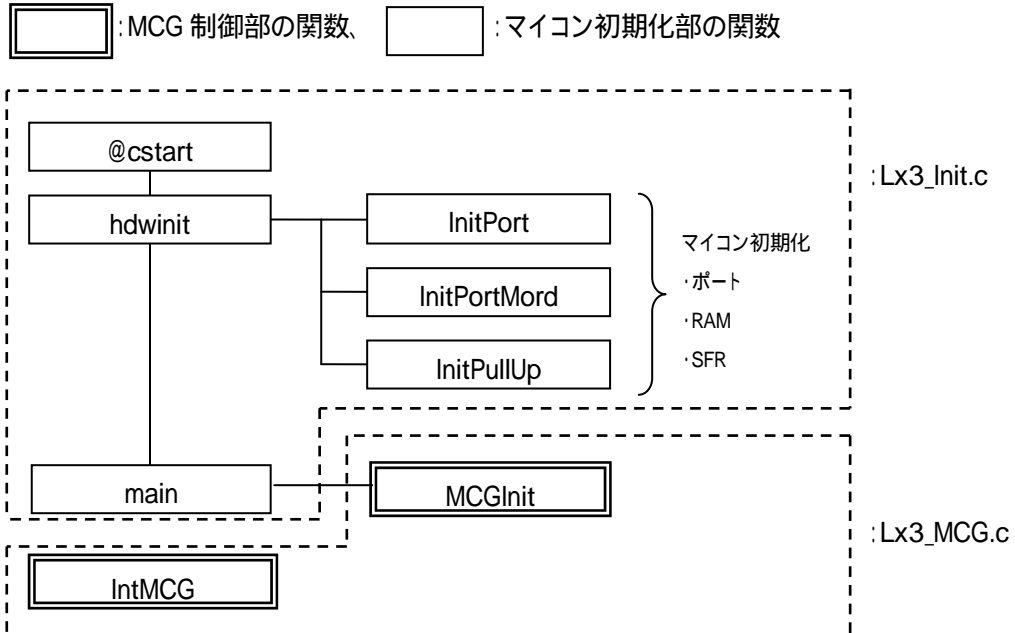
《フローチャート概要》



3.5 関数説明

3.5.1 関数構成図

関数との関係を示します。



3.5.2 関数一覧(Lx3_MCG.asm ファイル内)

インタフェース関数の一覧を以下に示します。

関数名	処理概要	引数	戻り値
void MCGInit(void)	MCG 初期化処理	なし	なし
void MCGMain(void)	MCG メイン処理	なし	なし
_interrupt void IntMCG(void)	MCG 送信開始割込み処理 (INTMCG 使用)	なし	なし

3.5.3 関数詳細

void MCGInit(void)

関数名	void MCGInit(void)
処理	MCG 初期化処理
引数	-
戻り値	-
内容	マンチェスタ・コード・ジェネレータに必要な設定を行い、送信動作を開始する。
備考	リセット後に呼び出します。

void MCGMain(void)

関数名	void MCGMain(void)
処理	MCG メイン処理
引数	-
戻り値	-
内容	送信動作終了時に MCG 動作を停止させます。
備考	

_interrupt void IntMCG(void)

関数名	_interrupt void IntMCG(void)
処理	MCG 送信開始割込み処理 (INTMCG 使用)
引数	-
戻り値	-
内容	割込み毎に、送信の終了判定を行う。 送信ビット数が8ビットであれば送信動作を停止し、送信ビット数が8ビット未満であれば送信ビット数を1つ増やし、データを送信する。
備考	

3.6 ソフトウェアの説明

以下に、本ソフトウェアの説明をします。

システム設定内容は以下の通りです。

- ・メインシステムクロック : 高速内蔵発振(8MHz(TYP.))

マンチェスタ・コード・ジェネレータの設定はMCOBIT、MCOCTL0、MCOCTL1、MCOCTL2 レジスタで行います。設定内容は以下の通りです。

- ・送信ビット数: 1 ~ 8 ビット
- ・先頭ビット: LSB
- ・ボーレート: 9600bps (Lx3_Opt.h ファイルにて変更可能)

4 リファレンス

- ・78K0/LF3 ユーザーズ・マニュアル
- ・78K/0 シリーズ 命令編
- ・RA78K0 言語編
- ・CC78K0 言語編
- ・PM+ ユーザーズ・マニュアル
- ・SM+ システム・シミュレータ 操作編

5 Programming Code

弊社 WEB サイトからダウンロードして下さい。以下2種類のサンプル・プログラムが利用可能です。

- ・プロジェクト付
- ・ソースのみ

“プロジェクト付き”サンプル・プログラムとは、統合開発環境 PM+用のプロジェクト(ワークスペース)が添付されています。PM+からオープンすることにより、新たにワークスペースを作成することなく、アセンブル/コンパイルが可能です。

ただし、インストールされた環境によっては、正常にアセンブル/コンパイルができない可能性がありますので、その場合は、一旦、PM+ワークスペースを削除して新規に作成してください。

6 改版履歴

	発行日	ページ	改版内容
Rev1.0	2007/03/30	-	初版