

弊社マイコン KL5C16030 / KL5C16005 におきまして、TRA 命令を使用する場合に、以下に示す動作上の不具合が発生することが判明いたしました。ご使用頂くにあたり、充分にご注意頂けるようお願いいたします。

原因

NMI の直後に TRA 命令を実行したとき、および RST 命令の直後に TRA 命令を実行したとき、内部処理で使われているレジスタと外部端子(アドレスバス)に現れるレジスタとの間に不整合が発生する。内部処理で使われているレジスタが不具合となる。

影響およびその範囲

RST 命令 + TRA 命令、または / および、NMI + TRA 命令を使用したプログラムで影響が及ぶ危険性があります。これらの命令を使用していない場合は問題がありません。

この不整合の影響が現れる、すなわち、内部処理で使われているレジスタが参照されるのは、

- (a) 割り込み受け付け (INT、NMI とも)
このとき、スタックに積まれる戻り番地が誤った値になる
- (b) CALL、CALL3、RST を実行したとき
このとき、スタックに積まれる戻り番地が誤った値になる
- (c) JR、JR cc、DJNZ を実行したとき
このとき、飛び先番地が誤った値になる

の 3 つのケースです。ちなみに、JP、JP3、JP cc、JP3 cc は影響を受けません。

尚、上記 (b)(c) の命令が影響を受けるのは、RST 命令 + TRA 命令 などにより不整合が生じた状態で実行された場合のみであり、普通に (b)(c) の命令が実行される場合は何も問題はありません。

この不整合は、JP、JP3、JP cc (成立)、JP3 cc (成立) によりジャンプすると解消します。

回避策

下記の対策を全て講じることで回避することができます。これらの内容につきましては、次ページ以降のサンプルプログラムもご参照ください。プログラム記述内に下記番号を記載しました。

RST 命令の開始番地 (000038H など) に TRA 命令を置く場合、その直後に JP 命令または JP3 命令を配置することによりジャンプさせ、不整合を速やかに解消する。
NMI の開始番地である 000066H に TRA 命令を置く場合、その直後に JP 命令または JP3 命令を配置することによりジャンプさせ、不整合を速やかに解消する。
RST 命令の開始番地 (000038H など) に置かれた TRA 命令の直後に割り込みを受け付けた場合、誤った戻り番地 (000039H など、正しくは 00003AH など) がスタックに積まれるため、これを訂正する処理ルーチンを、割り込みハンドラの先頭に置く。

別の方法として、RST 命令 + TRA 命令を使わない (CALL3 命令で代用) でも回避できます。

補足

RST 命令を DI 状態で実行することは、厳密には回避策になりません。DI 状態でも NMI は受け付けるため、NMI からの戻り番地が誤った値になる恐れがあるからです。

その他ご不明な点等がございましたら弊社までご連絡ください。

川崎製鉄株式会社 LSI 事業部 営業部

URL: <http://www.kawatetsu-lsi.co.jp/assp/micro/index.htm>

E-mail: mcu@lsidiv.kawasaki-steel.co.jp

Bug Report

KL5C16030/05

```

000000 : ; ===== TRA-after-RST bug patch routine =====
000000 : ; ver.1.0 2000-09-28 mcu@lsidiv.kawasaki-steel.co.jp
000000 :
000000 : 000020          KP63A equ 20h          ; 汎用タイマ
000000 : 000034          KP69 equ 34h          ; 割り込みコントローラ
000000 : 000038          KP67 equ 38h          ; パラレルポート
000000 :
000000 :                org 000000h
000000 :
000000 : 31 FF FF          ld sp, 0ffffh      ;
000003 : 3E 80            ld a, 80h           ;
000005 : ED F4            ld zp, a           ; スタックポインタ ZSP の初期化 ( 80FFFFH ~ )
000007 : ED 4E            im 3                ;
000009 : C3 00 20         jp SETUP           ; さまざまな初期化ルーチンへ
00000C :
000020 :                org 000020h          ; RST 20H は互換ボックス内からしか
000020 : C3 00 60          RST20: jp M_RST20  ; 使わない前提で TRA は置かない場合の例
000023 :
000030 :                org 000030h          ; RST 30H はグローバルに使うので
000030 : ED 54            RST30: tra          ; TRA を置く場合の例
000032 : C3 00 70          jp M_RST30         ; RST 30H の本体は 互換ボックス内にある
000035 :                ; ；ので JP で飛ぶ
000035 :
000038 :                org 000038h          ; RST 38H もグローバルに使うので
000038 : ED 54            RST38: tra          ; TRA を置く場合の例
00003A : ED C3 05 60 01   jp3 M_RST38        ; RST 38H の本体は 互換ボックス外にある
00003F :                ; ；ので JP3 で飛ぶ
00003F :
00003F :                ; *****
00003F :                ; NMI 処理 ( 本体は 互換ボックス外 に配置 )
00003F :                ; *****
000066 :                org 000066h
000066 : ED 54            NMI: tra            ; PPC の内容を 3 バイト分スタックに退避
000068 : C3 6B 00          jp NMI1           ; ただちに JP 命令 を実行 ( JR 命令 は不可 )
00006B : CD 2E 01          NMI1: call TRA_PATCH ; パッチ当ルーチンを実行
00006E : ED C3 00 00 02   jp3 M_NMI         ; NMI 処理メインルーチンへ
000073 :
000073 :                ; *****
000073 :                ; ； 割り込みアドレステーブル
000073 :                ; *****
000073 :                ; ； 割り込み処理ルーチンを 互換ボックス外 に置く場合
000073 :                ; ； o 必ず 互換ボックス内 から処理を始める
000073 :                ; ； 割り込みアドレステーブル には 互換ボックス内の
000073 :                ; ； 実行開始番地しか登録できない
000073 :                ; ； o JP3 命令 により 互換ボックス外 へ飛ぶ
000073 :                ; *****
000100 :                org 000100h
000100 :
000100 : 80 16            dw INTERR          ; IR0 不使用
000102 : 80 16            dw INTERR          ; IR1 不使用
000104 : 80 16            dw INTERR          ; IR2 不使用
000106 : 80 16            dw INTERR          ; IR3 不使用
000108 : 80 16            dw INTERR          ; IR4 不使用
00010A : 80 16            dw INTERR          ; IR5 不使用
00010C : 80 16            dw INTERR          ; IR6 不使用

```

Bug Report

KL5C16030/05

```

00010E: 28 01          dw  P_INT7      ; IR7 ( 外部割込み )
000110: 80 16          dw  INTERR      ; IR8  不使用
000112: 80 16          dw  INTERR      ; IR9  不使用
000114: 80 16          dw  INTERR      ; IR10  不使用
000116: 80 16          dw  INTERR      ; IR11  不使用
000118: 80 16          dw  INTERR      ; IR12  不使用
00011A: 80 16          dw  INTERR      ; IR13  不使用
00011C: 20 01          dw  P_INT14     ; IR14 ( タイマ割込み )
00011E: 80 16          dw  INTERR      ; IR15  不使用
000120:
000120:          ; %%%%%%%%%%%
000120:          ; %   パッチ当てを経由してジャンプする   %
000120:          ; %%%%%%%%%%%
000120:
000120: CD 2E 01      P_INT14: call  TRA_PATCH ; パッチ当ルーチンを実行
000123: ED C3 30 60 01      jp3    INT14      ; IR14 ( タイマ割り込み ) 処理メインルーチンへ
000128:
000128: CD 2E 01      P_INT7:  call  TRA_PATCH ; パッチ当ルーチンを実行
00012B: C3 12 80      jp      INT7      ; IR7 ( 外部割込み ) 処理メインルーチンへ
00012E:
00012E:
00012E:
00012E:          ; 【ここから RST+TRA のバグ回避用パッチ】
00012E:          ;
00012E:          ;   NMI ハンドラ および 割込みハンドラ の先頭で実行する
00012E:          ;   これは 00xxxxh に置くことを前提としたコード
00012E:          ;
00012E:          ;   TRA を置いている RST xx の直後か否かを調べ、戻り番地を修復する。
00012E:          ;   TRA を置いていない RST xx は調べない。
00012E:          ;
00012E: F5          TRA_PATCH: push  af          ; 作業用レジスタを確保
00012F: E5          push  hl
000130:
000130: ED 2A 06      ld      ahl, (sp+6)      ; 割込みの戻り番地を取り出す
000133: B4          or      h          ; A=H=0 ?
000134: 20 09      jr      nz, TRA_P1      ; AHL=0000xx 以外なら何もしない
000136:
000136: 7D          ld      a, l
000137: FE 31      cp      31h          ; RST 30H の直後 ?
000139: 28 07      jr      z, TRA_H1
00013B: FE 39      cp      39h          ; RST 38H の直後 ?
00013D: 28 03      jr      z, TRA_H1
00013F:
00013F: E1          TRA_P1: pop  hl          ; レジスタを復元
000140: F1          pop  af
000141: C9          ret
000142:
000142: 2C          TRA_H1: inc  l          ; 戻り番地を修復
000143: ED A6 06      ld      (sp+6), hl      ; 割込みの戻り番地 ( 下位 ) を書き戻す
000146: 18 F7      jr      TRA_P1
000148:
000148:          ; 【ここまで RST+TRA のバグ回避用パッチ】
000148:
000148:
000148:
000148:

```

Bug Report

KL5C16030/05

```

000148 : ; *****
000148 : ; 不使用チャネルの割込み処理ルーチン (暴走を回避)
000148 : ; 割込みから戻らないことが前提なので TRA_PATCH は省略可
000148 : ; *****
001680 : org 001680h
001680 : 3E 3F INTERR: ld a, 00111111b ; ビット操作コマンド
001682 : D3 39 out (KP67+1), a ; P37 を high にセット
001684 : 18 FE SELF: jr SELF ; ここで無限ループ = 処理の停止
001686 :
002000 : org 002000h
002000 : 00 SETUP: nop ; *****
002001 : 00 nop ; さまざまな初期化ルーチンを記述する
002002 : 00 nop ; 適宜 メインルーチンを記述する
002003 : 00 nop ; *****
002004 :
006000 : org 006000h ; *****
006000 : ; RST 20H ... 互換ボックス内からの使用に限定
006000 : ; *****
006000 : 00 M_RST20: nop ; 適宜 RST 20H 本体ルーチンを記述する
006001 : C9 ret ; RST 20H には TRA を置いてないため RET で戻る
006002 :
007000 : org 007000h ; *****
007000 : ; RST 30H ... グローバルな使用を想定
007000 : ; *****
007000 : 00 M_RST30: nop ; 適宜 RST 30H 本体ルーチンを記述する
007001 : ED 5C ret3 ; RST 30H には TRA を置いてあるため RET3 で戻る
007003 :
008012 : org 008012h ; *****
008012 : ; 割込みルーチンを 互換ボックス内 に配置した例
008012 : ; IR7 処理ルーチン ( 外部割込み )
008012 : ; *****
008012 : 00 INT7: nop ; 実際には必要な処理を記述する
008013 : D3 1D out (1dh), a ; 3 バイト RETI 命令相当の処理
008015 : FB ei ;
008016 : ED 5C ret3 ;
008018 :
016005 : org 016005h ; *****
016005 : ; RST 38H ... グローバルな使用を想定
016005 : ; *****
016005 : 00 M_RST38: nop ; 適宜 RST 38H 本体ルーチンを記述する
016006 : ED 5C ret3 ; RST 38H には TRA を置いてあるため RET3 で戻る
016008 :
016030 : org 016030h ; *****
016030 : ; 割込みルーチンを 互換ボックス外 に配置した例
016030 : ; IR14 処理ルーチン ( タイマ割込み )
016030 : ; *****
016030 : 00 INT14: nop ; 実際には必要な処理を記述する
016031 : D3 1D out (1dh), a ; 3 バイト RETI 命令相当の処理
016033 : FB ei ;
016034 : ED 5C ret3 ;
016036 :
020000 : org 020000h ; *****
020000 : ; NMI ルーチンを 互換ボックス外 に配置した例
020000 : ; *****
020000 : 00 M_NMI: nop ; 実際には必要な処理を記述する
020001 : ED 55 retn3 ;
020003 :

```