

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日  
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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。お客様の機器の設計において、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には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、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に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
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  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障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おりますが、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、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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。お客様がかかる法令を遵守しないこと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。
11. 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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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とその総株主の議決権の過半数を直接または間接に保有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。

## 漢字表示デモンストレーション用V850ES/JG3ボード

---

### 対象デバイス

V850ES/JG3マイクロコントローラ

V850ES/JJ3マイクロコントローラ

V850ES/JF3-Lマイクロコントローラ

V850ES/JG3-Lマイクロコントローラ

(メモ)

# 目次要約

第1章 概 説 ...	10
第2章 スタンド・アローン操作 ...	13
第3章 ホスト・コマンド操作 ...	16
第4章 ボード仕様 ...	55
第5章 ボード設計情報 ...	62
第6章 プログラム構成とビルド方法 ...	69
付録A 改版履歴 ...	78

### 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デバイスの入力が入力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。入力レベルが固定な場合はもちろん、 $V_{IL}$  (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等が入らないようご使用ください。

### 未使用入力の処理

CMOSデバイスの未使用端子の入力レベルは固定してください。

未使用端子入力については、CMOSデバイスの入力に何も接続しない状態で動作させるのではなく、プルアップかプルダウンによって入力レベルを固定してください。また、未使用の入出力端子が出力となる可能性（タイミングは規定しません）を考慮すると、個別に抵抗を介して  $V_{DD}$  または GND に接続することが有効です。

資料中に「未使用端子の処理」について記載のある製品については、その内容を守ってください。

### 静電気対策

MOSデバイス取り扱いの際は静電気防止を心がけてください。

MOSデバイス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、または導電性の緩衝材、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、MOSデバイス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。

### 初期化以前の状態

電源投入時、MOSデバイスの初期状態は不定です。

電源投入時の端子の出力状態や入出力設定、レジスタ内容などは保証しておりません。ただし、リセット動作やモード設定で定義している項目については、これらの動作ののちに保証の対象となります。

リセット機能を持つデバイスの電源投入後は、まずリセット動作を実行してください。

### 電源投入切断順序

内部動作および外部インタフェースで異なる電源を使用するデバイスの場合、原則として内部電源を投入した後に外部電源を投入してください。切断の際には、原則として外部電源を切断した後に内部電源を切断してください。逆の電源投入切断順により、内部素子に過電圧が印加され、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「電源投入切断シーケンス」についての記載のある製品については、その内容を守ってください。

### 電源OFF時における入力信号

当該デバイスの電源がOFF状態の時に、入力信号や入出力プルアップ電源を入れないでください。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OFF時における入力信号」についての記載のある製品については、その内容を守ってください。

- 本資料に記載されている内容は2009年9月現在のものです。今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
- 当社は、本資料に記載された当社製品の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、一切その責を負いません。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。
- 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。お客様の機器の設計において、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には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責を負いません。
- 当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。また、当社製品は耐放射線設計については行っておりません。当社製品をお客様の機器にご使用の際には、当社製品の不具合の結果として、生命、身体および財産に対する損害や社会的損害を生じさせないように、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を行ってください。
- 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。

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

特別水準：輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器

特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等

当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

(注)

- (1) 本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいう。
- (2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

# はじめに

**対象者** このマニュアルは、V850ESマイクロコントローラの応用システムを設計、開発するユーザを対象とします。

**目的** 漢字表示デモンストレーション用V850ES/JG3ボードの仕様および漢字表示デモンストレーション・プログラムの操作方法についてユーザに理解していただくことを目的とします。

**構成** このマニュアルは、大きく分けて次の内容で構成しています。

- ・概 説
- ・操作方法
- ・ボード仕様
- ・ボード設計情報
- ・プログラム構成

**読み方** このマニュアルの読者には、電気、論理回路およびマイクロコンピュータに関する一般知識を必要とします。

本ボードに組み合わせる漢字表示デモンストレーション用ベース・ボードの取り扱い方法を理解しようとするとき

ベース・ボードの**ユーザーズ・マニュアル**を参照してください。

V850ES/JG3の詳細を理解しようとするとき

V850ES/JG3の**ユーザーズ・マニュアル**を参照してください。

**凡 例** データ表記の重み：左が上位桁，右が下位桁

アクティブ・ローの表記： $\overline{\text{xxx}}$ （端子，信号名称に上線）

メモリ・マップのアドレス：上部 - 上位，下部 - 下位

注：本文中に付けた注の説明

注意：気を付けて読んでいただきたい内容

備考：本文の補足説明

数の表記：2進数 ... xxxxまたはxxxxB

10進数 ... xxxx

16進数 ... xxxxH

2のべき数を示す接頭語（アドレス空間，メモリ容量）：

K（キロ）...  $2^{10} = 1024$

M（メガ）...  $2^{20} = 1024^2$

G（ギガ）...  $2^{30} = 1024^3$



**関連資料** 関連資料は暫定版の場合がありますが、この資料では「暫定」の表示をしておりません。あらかじめご了承ください。

**漢字表示デモンストレーション・ボードの関連資料**

資料名	資料番号	
	和文	英文
漢字表示デモンストレーション用ベース・ボード ユーザーズ・マニュアル	U19207J	未定
漢字表示デモンストレーション用78K0/KF2ボード ユーザーズ・マニュアル	U19208J	未定
漢字表示デモンストレーション用78K0R/KG3ボード ユーザーズ・マニュアル	U19209J	未定
漢字表示デモンストレーション用V850ES/JG3ボード ユーザーズ・マニュアル	このマニュアル	未定
78K0/Kx2サンプル・プログラム（簡易OS編）アプリケーション・ノート	U19214J	未定
78K0R/Kx3サンプル・プログラム（簡易OS編）アプリケーション・ノート	U19215J	未定
V850ES/Jx3サンプル・プログラム（簡易OS編）アプリケーション・ノート	U19216J	未定
フォント・ユーティリティ ユーザーズ・マニュアル	U19527J	未定
78K0/Kx2 サンプル・プログラム（フォント選択編）アプリケーション・ノート	U19528J	未定
78K0R/Kx3 サンプル・プログラム（フォント選択編）アプリケーション・ノート	U19529J	未定
V850ES/Jx3 サンプル・プログラム（フォント選択編）アプリケーション・ノート	U19530J	未定
漢字表示デモンストレーション用拡張ボード ユーザーズ・マニュアル	U19526J	未定
78K0/Kx2 サンプル・プログラム（ドットLCD制御編）アプリケーション・ノート	U19531J	未定
78K0R/Kx3 サンプル・プログラム（ドットLCD制御編）アプリケーション・ノート	U19532J	未定
V850ES/Jx3 サンプル・プログラム（ドットLCD制御編）アプリケーション・ノート	U19533J	未定
78K0/Kx2 サンプル・プログラム（タッチスクリーン編）アプリケーション・ノート	U19720J	未定
78K0R/Kx3 サンプル・プログラム（タッチスクリーン編）アプリケーション・ノート	U19721J	未定
V850ES/Jx3 サンプル・プログラム（タッチスクリーン編）アプリケーション・ノート	U19722J	未定

**V850ESマイクロコントローラ・デバイスの関連資料**

資料名	資料番号	
	和文	英文
V850ES/JG2 ユーザーズ・マニュアル	U17715J	U17715E
V850ES/JJ2 ユーザーズ・マニュアル	U17714J	U17714E
V850ES/JG3 ユーザーズ・マニュアル	U18708J	U18708E
V850ES/JJ3 ユーザーズ・マニュアル	U18376J	U18376E
V850ES/JF3-L ユーザーズ・マニュアル	U18952J	U18952E
V850ES/JG3-L ユーザーズ・マニュアル	U18953J	U18953E
V850ES ユーザーズ・マニュアル アーキテクチャ編	U15943J	U15943E

**注意** 上記関連資料は予告なしに内容を変更することがあります。設計などには、必ず最新の資料をご使用ください。

# 目 次

## 第1章 概 説 ... 10

- 1.1 特 徴 ... 10
- 1.2 セットアップ方法 ... 11
- 1.3 システム構成 ... 12

## 第2章 スタンド・アローン操作 ... 13

- 2.1 デモンストレーション概要 ... 13
- 2.2 メニュー画面構成 ... 13
- 2.3 キー操作 ... 14
- 2.4 メニュー項目 ... 14
  - 2.4.1 トップ・メニュー (Top Menu) ... 14
  - 2.4.2 各種内部設定メニュー (System Settings) ... 15
  - 2.4.3 総合デモ・メニュー (Demonstration) ... 15
  - 2.4.4 メロディの演奏メニュー (Melody Select) ... 15
  - 2.4.5 LEDの点滅動作メニュー (LED Blink) ... 15
  - 2.4.6 A/D変換値の表示メニュー (ADC Display) ... 15
  - 2.4.7 スタンバイ・メニュー (Standby) ... 15

## 第3章 ホスト・コマンド操作 ... 16

- 3.1 機能ユニット構成 ... 16
- 3.2 メッセージ・フォーマット ... 17
- 3.3 リード/ライト・コマンド ... 19
- 3.4 ユニット別詳細 ... 20
  - 3.4.1 全体制御 (MainユニットP) ... 20
  - 3.4.2 キー・LEDスキャン (BTimerユニットS) ... 22
  - 3.4.3 ホスト・コマンド受信 (COM1\_rxユニットU) ... 23
  - 3.4.4 ホスト・マシンへのメッセージ送信 (COM1\_txユニットu) ... 24
  - 3.4.5 LCD制御 (LCDユニットL) ... 25
  - 3.4.6 文字表示 (PrintユニットJ) ... 26
  - 3.4.7 キー・コード生成 (Key\_codeユニットG) ... 28
  - 3.4.8 メロディ出力 (MelodyユニットM) ... 29
  - 3.4.9 LEDブリンク (LED\_blinkユニットB) ... 33
  - 3.4.10 日付・時刻表示, アラーム (CalendarユニットC) ... 35
  - 3.4.11 内蔵A/D変換 (ADC\_inユニットA) ... 38
  - 3.4.12 内蔵D/A変換 (DAC\_outユニットD) ... 41
  - 3.4.13 シリアル接続D/A変換 (CSI\_DACユニットF) ... 42
  - 3.4.14 シリアル接続EEPROM (CSI\_EEPユニットE) ... 43
  - 3.4.15 拡張IOデバイス (I2C\_IOユニットI) ... 45
  - 3.4.16 赤外リモコン受信 (IR\_rxユニットR) ... 46
  - 3.4.17 赤外リモコン送信 (IR\_txユニットT) ... 47
  - 3.4.18 グラフィック・パターン描画 (GraphユニットN) ... 48
  - 3.4.19 メニュー制御 (MenuユニットQ) ... 49

## 第4章 ボード仕様 ... 55

- 4.1 構成図 ... 55
- 4.2 諸元 ... 55
- 4.3 コネクタ信号配置 ... 56
- 4.4 MCU端子割り当て ... 58
- 4.5 内蔵周辺機能の割り当て ... 61

## 第5章 ボード設計情報 ... 62

- 5.1 回路情報 ... 62
  - 5.1.1 回路ファイル構成 ... 62
  - 5.1.2 回路図(縮小版) ... 62
- 5.2 基板情報 ... 63
  - 5.2.1 基板ファイル構成 ... 63
  - 5.2.2 基板仕様例 ... 63
  - 5.2.3 基板図面(縮小版) ... 63
- 5.3 実装情報 ... 67
  - 5.3.1 実装ファイル構成 ... 67
  - 5.3.2 実装図およびメタル・マスク図(縮小版) ... 67

## 第6章 プログラム構成とビルド方法 ... 69

- 6.1 ビルド済プログラム・ファイル ... 69
- 6.2 ファイル構成 ... 69
- 6.3 プログラム実行の依存関係 ... 71
- 6.4 ビルド方法 ... 71
- 6.5 起動時の注意 ... 73
- 6.6 プログラム改造方法 ... 74
  - 6.6.1 端子変更 ... 74
  - 6.6.2 発振周波数変更 ... 76
  - 6.6.3 内蔵周辺機能の割り当て ... 76

## 付録A 改版履歴 ... 78

# 第1章 概 説

この章では、本ボードの特徴、セットアップ方法およびデモンストレーション用のシステム構成について説明します。

## 1.1 特 徴

本ボードは、漢字表示デモンストレーション用ベース・ボードに取り付けることにより、漢字表示を始めとする各種デモンストレーションを行うことができます。

### (1) ハードウェア

- ・ MCU : V850ES/JG3 (  $\mu$  PD70F3740GC-UEU-AX, Flash 512Kバイト, RAM 40Kバイト )
- ・ 動作電圧 : 3.0 ~ 3.6 V ( A/D ・ D/Aコンバータを使用しない場合は2.85 V ~ 3.6 V )
- ・ メイン・クロック発振 : 4MHz ( セラミック発振子 )
- ・ サブ・クロック発振 : 32.768kHz ( 水晶発振子 )
- ・ ベース・ボード接続機能 : ベース・ボードの次の機能を使用可能です。  
128x64ドットLCD, 12桁7セグメントLED, 48キー・マトリクス, IOポート11本, EEPROM, シリアル接続8ビットD/Aコンバータ, ポテンショメータ4個, I<sup>2</sup>C拡張デバイス, UART 2ch, ブザー, 赤外LED, 赤外受光モジュール
- ・ ツール接続 : QB-MINI2 ( オンチップ・デバッグ・エミュレータ ) を接続可能。

### (2) デモンストレーション・ソフトウェア

リビジョン3.05以降では、次のデモが可能です。

- ・ メイン・クロック 16MHz ( 4MHz x 4通倍 ) / 32MHz ( 4MHz x 8通倍 ) 切り替え設定。
- ・ 漢字テキストのLCD表示およびLCDバックライトのPWM調光。  
JIS X0213 第3水準の14ドット・フォント表示。  
NEC製FontAvenueのサンプル・フォント ( 12, 16, 20, 24, 32ドット ) による例文表示。
- ・ 時刻やアナログ入力値の自動表示 ( LCDまたはLED ) 。
- ・ ポート接続LEDの自動点滅。
- ・ キー・マトリクスからASCII文字入力 ( LCDまたはホスト・マシンへ表示 ) 。
- ・ メロディ・スクリプトから自動的にメロディを生成してブザー出力。
- ・ キー・クリック音のブザー出力。
- ・ 赤外リモコン送信, 受信。

またSM06B2拡張ボードの追加により次のデモが可能です。

- ・ 各種サイズのドットLCDモジュールの接続と表示。
- ・ タッチパネル接続での座標検出。
- ・ ADPCMによる音声再生。

## 1.2 セットアップ方法

本ボードをベース・ボードに取り付けた後、ジャンパ設定やターミナル・ソフトウェアの設定を行います。

### (1) 電源選択端子のVDD設定

3.3 Vにジャンパを設定します。5 Vに設定した場合は電源が供給されず、動作しません。

### (2) ターミナル・ソフトウェアの通信条件

ホスト・コマンドによる操作を行う場合は設定が必要です。

ベース・ボードのユーザズ・マニュアルに記載してある代表例と同じ設定をします。

ビット/秒：57600

データ・ビット：8

パリティ：なし

ストップ・ビット：1

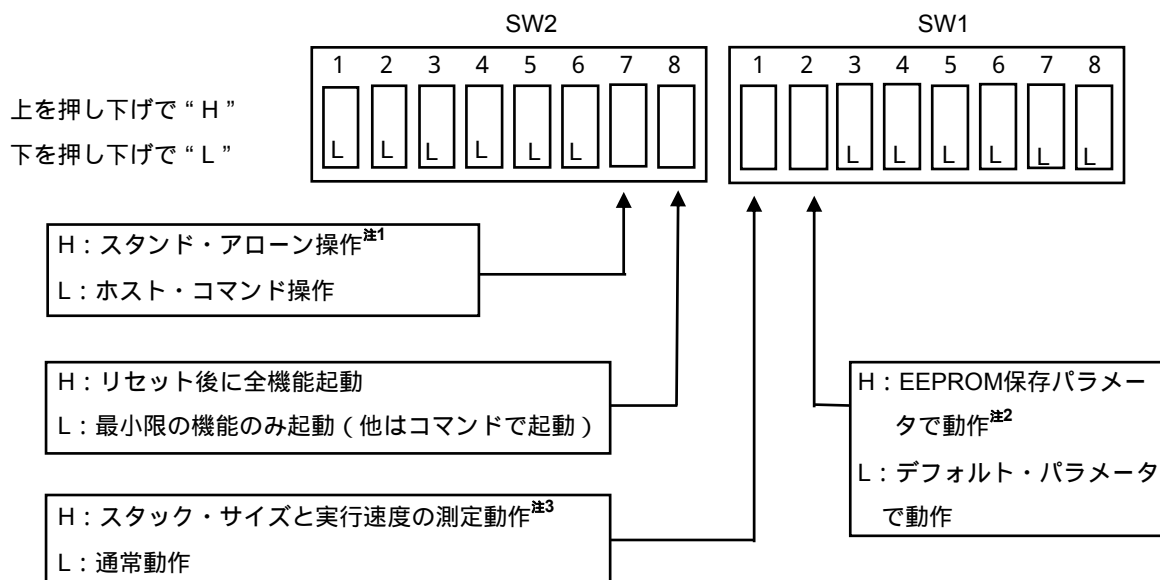
フロー制御：ハードウェア

その他必須設定：ローカル・エコーあり，送信時行末にCR付加，Shift-JIS漢字コード。

その他推奨設定：キーボードによるコピー & ペースト有効。

### (3) 操作モード設定

本ボードはスタンド・アローンでの操作とホスト・コマンドによる操作が可能です。スタンド・アロンの操作を行うかどうかは，ポート・レベル設定スイッチの内容によって決まります。



注1. スタンド・アローン設定でもホスト・コマンドによる操作は可能ですが，スタンド・アローン操作（メニュー操作）とホスト・コマンド操作の内容に競合があると，表示が乱れたり期待どおりに動作しなくなる可能性があります。

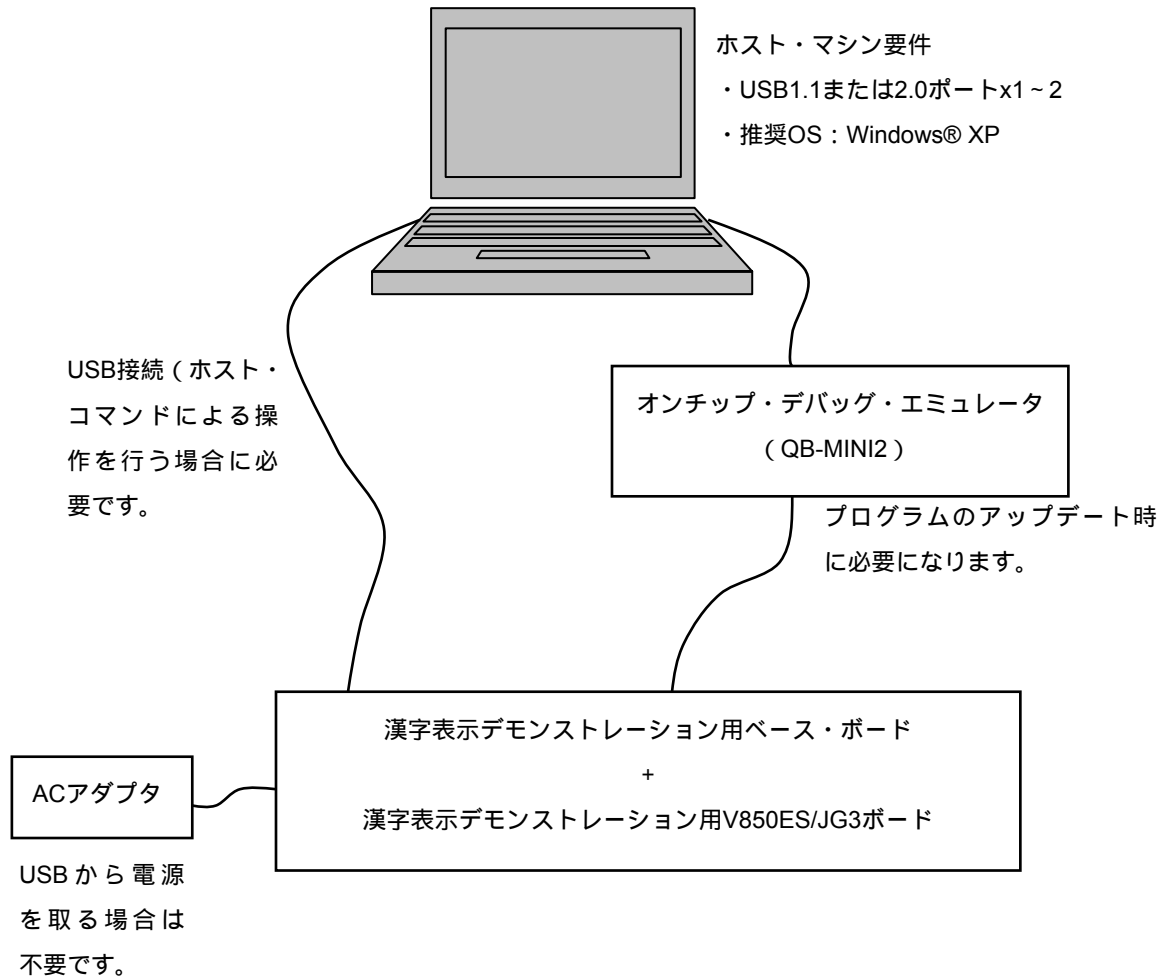
注2. パラメータが保存されていない場合はデフォルト・パラメータで動作します。

注3. 詳細は簡易OSのアプリケーション・ノート（U19216（第2版以降））を参照してください。

### 1.3 システム構成

スタンド・アローンのデモンストレーションは本ボード、ベース・ボード、ACアダプタだけで実行できます。ホスト・コマンドによりデモンストレーションを制御する場合は、ホスト・マシンが必要です。またデモンストレーション・プログラムをアップデートする場合にはオンチップ・デバッグ・エミュレータ（QB-MINI2）も必要になります。全体の接続構成を次の図に示します。

図1-1 システム構成



## 第2章 スタンド・アローン操作

この章では、スタンド・アローン・モードで起動した場合の操作方法について説明します。

### 2.1 デモンストレーション概要

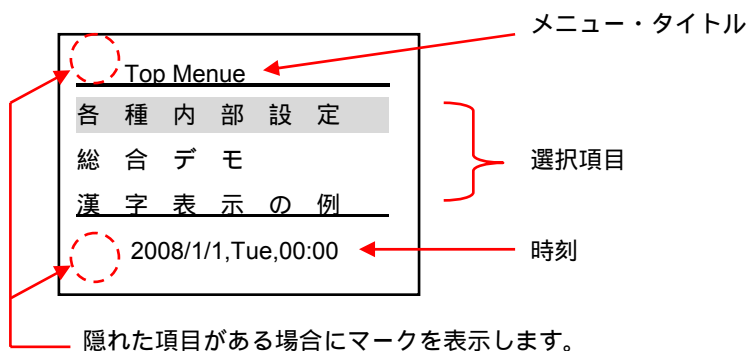
スタンド・アローン・モードで起動するとLCDにメニューが表示されます。ベース・ボード上のキー操作でメニュー項目を選択することにより、以下のデモンストレーションが可能です<sup>注1</sup>。

- ・ 漢字テキスト（例文）のスクロール表示
- ・ メロディ演奏
- ・ LED自動点滅
- ・ A/D変換値の自動表示
- ・ キーによるテキスト入力
- ・ リモコン受信コードの表示
- ・ スタンバイ動作

注1. デモンストレーション・プログラムのリビジョンによって内容が変わる場合があります。変更点についてはプログラムと同梱のリリース・ノート（SM05KV2j.TXT）を参照してください。

### 2.2 メニュー画面構成

メニュー画面の一番上がメニュー・タイトル、一番下が時刻、真中がメニュー項目のスクロール・エリアとなっています。



## 2.3 キー操作

キー操作によりメニューの選択・実行・取消しを行うことができます。各キーの機能は次の通りです。

- : カーソルを上に移動します。メニュー・カーソルが一番上の状態でも、LCD画面の左上に  が表示されている場合は上に隠れた項目があり、本キーでスクロールすることができます。
- : カーソルを下に移動します。メニュー・カーソルが一番下の状態でも、LCD画面の左下に  が表示されている場合は下に隠れた項目があり、本キーでスクロールすることができます。
- ENT : カーソルで選択されているデモンストレーション項目を実行します。項目によっては下位階層のメニューが表示されます。
- BS : デモンストレーション実行中や下位階層メニュー表示中に押すと、1つ前のメニューに戻ります。ただしテキスト入力時にはバックスペースとなるので、メニューに戻るにはENTキーを押します。

テキスト入力時は、以下のキーも使用できます。

- : カーソルを右に移動します。
- : カーソルを左に移動します。
- SYB : 記号入力モードに切り替わります。
- TAB : タブ
- ALP : 英数字モードに切り替わります。英数字モードの時に押すと大文字 / 小文字が切り替わります。
- SP : スペース
- その他 : 英数字または記号

## 2.4 メニュー項目

以下、メニュー項目とデモンストレーション内容を説明します。

### 2.4.1 トップ・メニュー (Top Menu)

- ・各種内部設定      クリック音の有無、バックライトの輝度、などの設定メニューを表示します。
- ・総合デモ          デモンストレーション選択メニューを表示します。
- ・漢字表示の例      漢字テキストをスクロール表示します。
- ・メロディの演奏      メロディ選択メニューを表示します。
- ・LEDの点滅動作      点滅パターン選択メニューを表示します。
- ・A/D変換値の表示      ポテンショメータなどのアナログ値をLCDまたはLEDに表示するためのメニューを表示します。
- ・キー入力による表示      キー入力した内容をLCDに表示します。同時にリモコン送信も行います。
- ・リモコン受信表示      受信したリモコン・コードをLCDに表示します。
- ・スタンバイ          2種類のスタンバイ復帰方法を選べます。
- ・メニュー制御終了      画面をクリアしてメニュー制御を終了します。再度メニューを動作させるにはリセット・ボタンを押すか、DIP SW2-7をいったん“L”にした後“H”に戻します。



### 2.4.2 各種内部設定メニュー（System Settings）

- ・ クリック音の有無      クリック音有り / 無しを選択できます。どちらかにカーソルを合わせてENTキーを押すと確定します。
- ・ バックライト輝度      キーで輝度を選択できます。選択した輝度でENTキーを押すと確定します。
- ・ ライト自動消灯      指定時間キーを押さないときにバックライトを消灯します。      キーで時間を選択後、ENTキーを押すと確定します。
- ・ ライト自動点灯      キーを押したときにバックライトが点灯するかどうかを選択できます。
- ・ 時計表示形式      画面下の時計表示の形式を24時制 / 12時制のどちらかにすることができます。
- ・ 初期化      上記の各設定をデフォルトの設定に戻します。

### 2.4.3 総合デモ・メニュー（Demonstration）

- ・ デモ1      メロディ出力、LEDブリンク、漢字テキストのスクロール表示、A/Dで読み取ったポテンシオメータの値のLED表示、を同時に実行します。

### 2.4.4 メロディの演奏メニュー（Melody Select）

- ・ 演奏停止（stop）      メロディ演奏を止めます。
- ・ 曲名      サンプルとして実装されている曲名が表示されています。カーソルを合わせてENTキーを押すと演奏が始まります。

### 2.4.5 LEDの点滅動作メニュー（LED Blink）

- ・ 全消灯      DIP SW1の上下の右側4つのLEDが全て緑表示になります。
- ・ 4つ同時on/off      4つのLEDが同時に同じ点滅をします。
- ・ 流れる感じ      4つのLEDが順番に消灯します。
- ・ バラバラ      4つのLEDが異なる点滅をします。

### 2.4.6 A/D変換値の表示メニュー（ADC Display）

- ・ 変換停止      A/D変換を停止し、LED表示を消します。
- ・ LCD表示（3行）      ポテンシオメータRV1～RV3の値をLCDに表示します。
- ・ LED表示（SW3）      DIP SW3に連動した簡易D/Aコンバータの出力値をLEDに表示します。
- ・ LED表示（vol1）      ポテンシオメータRV1の値をLEDに表示します。
- ・ LED表示（vol2）      ポテンシオメータRV2の値をLEDに表示します。
- ・ LED表示（vol3）      ポテンシオメータRV3の値をLEDに表示します。
- ・ LED表示（vol4）      ポテンシオメータRV4の値をLEDに表示します。

### 2.4.7 スタンバイ・メニュー（Standby）

- ・ キーによる起動      BSキーを2秒以上押すとスタンバイを解除します。ホスト・コマンドでアラーム設定されている場合は、その時刻になると自動的にスタンバイ解除します。
- ・ 目覚まし機能      10～70秒後に自動的にスタンバイを解除します（1分後の時刻にアラーム設定を行ってスタンバイに入ります）。

## 第3章 ホスト・コマンド操作

この章では、ホスト・コマンドによる操作方法について説明します。

### 3.1 機能ユニット構成

デモンストレーション・プログラムは、以下の機能ユニットから構成しています。ホスト・コマンドは、機能ユニットごとに定めてあります。

表3-1 デモンストレーション・プログラムのユニット構成

ユニット記号	ユニット名	ユニット機能
P	Main	システム全体制御，省電力制御。
S	BTimer	キー・マトリクスおよび数字LEDのスキャン制御。LED表示パターン生成。
U	COM1_rx	ホスト・マシンからのコマンド受信。
u	COM1_tx	ホスト・マシンへのメッセージ送信。
L	LCD	ビットマップ表示メモリの内容をLCDへ転送。バックライトの輝度調整。
J	Print	文字列からフォント・データを検索し，ビットマップ表示メモリに展開。
G	Key_code	キー検出情報からASCIIコードを生成。クリック音，バックライト点灯の要求制御。
M	Melody	メロディ演奏，クリック音生成。
B	LED_Blink	ポート接続LEDの自動点滅制御。
C	Calendar	日付・時刻の表示，指定時刻でのアラーム動作。
A	ADC_in	MCU内蔵A/Dコンバータで読み取った値の表示，保存，ホスト・マシンへの報告。
D	DAC_out	MCU内蔵D/Aコンバータから指定値を出力。
F	CSI_DAC	3線シリアル接続のD/Aコンバータから指定値を出力。
E	CSI_EEP	3線シリアル接続のEEPROMアクセス（データやパラメータの保存，読み出し）。
I	I2C_IO	I <sup>2</sup> C拡張デバイスに接続されたLEDの点灯やDIP SWの読み取り制御。
R	IR_rx	赤外線リモコン受信。
T	IR_tx	赤外線リモコン送信。
N	Graph	グラフィック・パターンをビットマップ表示メモリに展開。
Q	Menu	テキストのスクロール表示制御，メニューの自動階層制御。
H	Self_test	スタンド・アローン・デモンストレーションの制御。
V	COM2_rx	ローカル・デバイスからのコマンド受信。
v	COM2_tx	ローカル・デバイスへのメッセージ送信。
X	User	ユーザ・プログラムの雛型。

注 リビジョンによっては内容が変わる可能性があります。ユニットV, vはR3.01で未実装です。

ユニットH～Xの説明は省略します。

## 3.2 メッセージ・フォーマット

### (1) コマンド・メッセージの基本フォーマット

ホスト・マシンから本ボードへ送るコマンド・メッセージの基本フォーマットは次のとおりです。

(コメント文字列) \$AA コマンド・パラメータ ¥r

#### コメント文字列

\$以外の文字は無視します。

#### コマンド開始記号

この文字からコマンドとして解釈します。

#### ユニット記号

どのユニットに対するコマンドかを指定します(表3-1のユニット記号を指定)。

#### コマンド種別

どのような動作を行うかを指定します。各ユニット共通のリード・コマンド(R),ライト・コマンド(W)と、個々のユニットごとに定めてあるコマンドがあります。

#### コマンド・パラメータ

個々のコマンドによって定まるパラメータを記述します。

- ・そのまま文字を記述した場合(アポストロフィを除く)は、ASCIIコードまたはシフトJISコードのデータ列と解釈します。
- ・' (アポストロフィ) を記述すると、以降16進数表記と解釈します。文字としては、0~9およびA~F(大文字限定)を使用します。2文字で1バイトと解釈しますが、1文字+空白でも1バイトと認識します(この場合1文字は下4ビットと解釈)。16進数2文字ごとに挿入する空白は無視します。再度' (アポストロフィ) を記述すると文字によるデータ列と解釈します。

#### 終端記号

¥r (キャリッジ・リターン, 0DH) を検出するとコマンド内容が確定したと解釈します。

確定前であれば、バックスペース(08H)による修正が可能です。バックスペース以外の編集コードは無効です。

注 後述のコマンド説明では終端記号(¥r)の記載を省略します。

### (2) 応答メッセージの基本フォーマット

本ボードからホスト・マシンへ返す応答メッセージの基本フォーマットは次のとおりです。

#AA\$ 応答データ ¥r

#### 応答開始記号

メッセージはこの文字から始まります。

#### ユニット記号

どのユニットからの応答かを表します(表3-1のユニット記号を参照)。

#### 応答種別

どのような動作結果について応答するかを表します。コマンドに対する応答の場合は、コマンドで指定したコマンド種別をそのまま返します。

**ステータス記号**

コマンドを正常に受け付けたかどうかの区分を表します。詳細は表3 - 2を参照してください。

**応答データ**

内容は応答種別とステータスによって定まります。

**終端記号**

¥r (キャリッジ・リターン, 0DH) で応答メッセージの終了を表します。

注 後述の応答メッセージ説明では終端記号 (¥r) の記載を省略します。

表3 - 2 ステータス一覧

ステータス記号	意味
a	パラメータで指定したアドレスが許容範囲を超えました。 リード・コマンドに対する応答の場合、リード開始アドレスとリード・バイト数の合計が許容範囲を超えたと考えられます。 ライト・コマンドに対する応答の場合、ライト開始アドレスと書き込みデータ数の合計が許容範囲を超えたと考えられます。
c	無効コマンドです (未定義のコマンド種別です)。
e	コマンドが長すぎます。
m	コマンドで要求された情報をテキスト形式で返信します。
o	コマンドを正常に受け付けました。要求情報がある場合は16進数表記で返信します。
p	コマンド・パラメータで指定した値が不正か、個数が足りません。
r	リビジョンが異なるため要求された動作を行いません。
s	対象ユニットが停止しています (リード/ライト・コマンド以外のコマンドはユニット停止中は受け付けられません)。元々対象ユニットを起動していないか、もしくはエラーにより強制停止させられたと考えられます。MainユニットPの\$PCコマンドにより対象ユニットの状態を確認できます。また\$PTコマンドで対象ユニットを起動することが可能ですが、停止原因が解消されていない場合は、また停止する可能性があります。
t	対象ユニットは以前に要求したコマンドを処理中のため、新規コマンドを受け付けられません。100ms以上の時間を要するコマンドの場合、対象ユニットが各種不具合や資源競合によりコマンド受付状態になれない場合が考えられます。自然回復しない場合、MainユニットPの\$PEコマンドにより対象ユニットをいったん停止し、\$PTコマンドにより起動することで回復できる場合があります。。
u	対象ユニットがありません。ユニット記号の誤記か、ビルド時に組み込み忘れたと考えられます。
v	資源不足でコマンドを実行できません。
!	イベントが発生したときに自動的に送られるメッセージです。コマンドに対する応答ではありません。

### 3.3 リード/ライト・コマンド

機能ユニットによっては、動作指示用のパラメータ領域と操作対象データを格納するためのデータ領域を持っています。この領域の内容を読み出したり、値を設定するためにリード・コマンド、ライト・コマンドを使用します。

コマンドのフォーマットは各ユニット共通ですが、領域のサイズや内容はユニットごとに異なります。

#### (1) リード・コマンド

$\$ \{ \text{ユニット記号} \} R \{ \text{リード開始アドレス} \} \{ \text{リード・バイト数} \}$

##### ・リード開始アドレス

パラメータ領域の場合は、0～7FHの範囲で指定します。

データ領域の場合は、実際のアドレスに8000Hを加算した8000H～EFFFHの範囲で指定します。このパラメータはビッグ・エンディアン（“8000”の場合80が上位バイト）です。

##### ・リード・バイト数

1～FCHの範囲で指定します。リード開始アドレス＋リード・バイト数が領域サイズを超えないように指定する必要があります。

本コマンドが正常に受け付けられると、以下の応答メッセージが返ってきます。

$\# \{ \text{ユニット記号} \} Ro \{ \text{読み出しデータ列 (16進数表記)} \}$

#### (2) ライト・コマンド

$\$ \{ \text{ユニット記号} \} W \{ \text{ライト開始アドレス} \} \{ \text{書き込みデータ列} \}$

##### ・ライト開始アドレス

パラメータ領域の場合は、0～7FHの範囲で指定します。

データ領域の場合は、実際のアドレスに8000Hを加算した8000H～EFFFHの範囲で指定します。このパラメータはビッグ・エンディアンで指定します。

##### ・書き込みデータ列

文字表記あるいは16進数表記で必要バイト数を指定します。ここで指定したバイト数＋ライト開始アドレスが領域サイズを超えないように指定する必要があります。

本コマンドが正常に受け付けられると、以下の応答メッセージが返ってきます。

$\# \{ \text{ユニット記号} \} Wo$

## 3.4 ユニット別詳細

### 3.4.1 全体制御 (MainユニットP)

本ユニットは、他のユニットの起動/停止を制御したり、全ユニットを停止してスタンバイ状態にする制御を行います。

特に起動時は、DIP SW設定 (1.2項参照) に応じて各ユニットを自動的に起動したり、各ユニットのパラメータ領域にデフォルト値またはEEPROM保存値を設定したりします。

#### (1) パラメータ領域

アドレス	内容
0~15	プログラム名とリビジョンを格納する領域です。
16~19	リセット後あるいはスタンバイ解除後にベース・ボードのDIP SW2-8が 'H' の場合、このパラメータを参照してユニットの起動を行います。 ソース・プログラムUDEF_a.hの「タスク名定義」で指定した順に、パラメータ・アドレス0のLSB側から対応付けされており、対応ビットが '1' であれば起動します。 ただし、Main, BTimer, ホスト通信ユニット (次のパラメータで指定するユニット) は、指定にかかわらず必ず起動します。デフォルト値はall "1" です。
20	最初に起動したいホスト通信ユニット (受信側) のユニット記号を設定します。起動しない場合はFFHを設定しますが、その場合はホスト・コマンドを受け付けられません。デフォルト値は "U" です。
21	起動したことをホストへ通知する場合にホスト通信ユニット (送信側) のユニット記号を設定します。デフォルト値は "u" です。通知しない場合はFFHを設定します。 通知設定してある場合は起動時に次のメッセージをホスト・マシンへ送信します。 #PU! {MCUのリセット・コントロール・フラグ・レジスタの値}

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### リビジョン情報要求

\$PV

応答データとして以下のようにプログラム名とリビジョンを文字列で返します。

#PVm {プログラム名文字列} {リビジョン文字列}

##### スタンバイ要求

\$PSave

これによりスタンバイ状態になります。解除するにはベース・ボードのBSキーを2秒以上押します。なお、Calendarユニットでアラーム設定されている場合は、その時刻になると自動的に起動します。

##### ユニット状態情報要求

\$PC {ユニット記号}

次の形式で1バイトのユニット状態情報を返します。

#PCo' { ユニット状態コード }

ユニット状態コードの意味は次のとおりです。

00：停止状態です（起動されていません）。

8x：起動されていますが、準備中でコマンド受付はできません。この状態は通常一瞬しかありませんが、必要資源が確保できない場合などはこの状態が継続する場合があります。

9x：起動されていますが、ホスト・コマンドあるいはイベントを待っている状態です。

Bx：10ms間隔で定期的に動作している状態です。

Dx：常時動作している状態です。

5xまたは4x：なんらかの原因で処理時間が10msを超え、OSにより強制停止された状態です。

#### ユニット起動要求

\$PT { ユニット記号 }

対象ユニットを起動します。起動済ユニットに対してこのコマンドを使うと資源待ちで停止する可能性があります。

#### ユニット停止要求

\$PE { ユニット記号 }

対象ユニットに停止要求を出します。

#### ユニット待機要求

\$PH { ユニット記号 }

対象ユニットを強制的に待機状態にします。

#### ユニット待機解除要求

\$PU { ユニット記号 }

対象ユニットを強制的に実行可能状態にします。

### (4) 操作例

#### リビジョン取得の例

\$PV

このコマンドにより以下のメッセージが返信されます（リビジョンは例です）。

#PVmSM05KV2 R02.00'00

#### スタンバイの例

\$PSave

LCD画面、数値LED表示が消え、スタンバイになります。

### 3.4.2 キー・LEDスキャン (BTimerユニットS)

本ユニットは、キー・マトリクスおよび数値LEDのダイナミック・スキャンを制御します。キー、LEDのスキャンを個別にオン/オフしたり、ASCIIコードで指定した数字をLEDに表示できます。

#### (1) パラメータ領域

アドレス	内容
0	キー・スキャン開始桁位置を1(左端)~CH(右端)の範囲で設定します。デフォルト値は1です。
1	キー・スキャン終了桁位置を上記同様に設定します。デフォルト値はCHです。
2	数値LEDスキャン開始桁位置を上記同様に設定します。デフォルト値は1です。
3	数値LEDスキャン終了桁位置を上記同様に設定します。デフォルト値はCHです。
4	未定義
5	bit7:1を設定するとキー・スキャンを実行します。デフォルト値は1です。 bit6:1を設定すると数値LEDのスキャンを実行します。デフォルト値は1です。

#### (2) データ領域

アドレス	内容
0~BH	数値LEDの表示パターン・メモリです。LED1桁が1バイトに対応し、bit0~6がセグメントa~g、bit7がdpに対応します。アドレス0が数値LED左端の桁に対応します。

#### (3) 個別コマンド

##### ASCIIコード文字列のLED表示

\$SP { 開始桁位置 } { 文字列 }

##### ・開始桁位置

0(左端)~BH(右端)の範囲で指定します。

##### ・文字列

‘0’~‘9’あるいは英字などを指定します。ただし数字以外は必ずしも判読できる表示とは限りません。また小数点を表示する場合は、ASCIIコードに80Hを加算します。

#### (4) 操作例

##### 直接表示パターンを書き込む例

0123456789EF.表示の例 \$SW'8000 3F 06 5B 4F 66 6D 7D 07 7F 6F 79 F1

全消灯の例 \$SW'8000 00 00 00 00 00 00 00 00 00 00 00

##### ASCIIコードによる表示例

F.E9876543210表示の例 \$SP'00C6'E9876543210'

中8桁にABCDEFGHを表示する例 \$SP'2'ABCDEFGH

全消灯の例(注 後ろに12個の空白があります) \$SP'0'



### 3.4.3 ホスト・コマンド受信 (COM1\_rxユニットU)

ホスト・マシンから受け取ったコマンドを各ユニットへ伝達したり、各ユニットのパラメータ領域やデータ領域のリード/ライト制御を行います。

#### (1) パラメータ領域

アドレス	内容
0	bit7-6 : フロー制御方法を設定します。00 : 無し, 11 : ハードウェア制御 (デフォルト値) bit4-3 : パリティ方式を設定します。00 : 無し (デフォルト値), 10 : 偶数, 11 : 奇数 bit2 : ストップ・ビットの長さを設定します。0 : 1bit (デフォルト値), 1 : 2bit
1	bit3-0 : 通信速度を設定します。 2 : 2400, 3 : 4800, 4 : 9600, 5 : 14400, 6 : 19200, 7 : 28800, 8 : 38400, 9 : 57600 (デフォルト値), A : 76800, B : 115200, C : 230400 (注 他のユニットの動作状況によって実際に動作する上限が異なります)

#### (2) データ領域

ありません。

#### (3) 個別コマンド

ありません。

#### (4) 操作例

通信速度を変えるには、パラメータ変更後にいったんスタンバイにし、スタンバイ解除による起動を行う必要があります。

速度を115200bpsに変更する設定 \$UW'1 B

スタンバイ \$PSave

スタンバイからの解除 (BSキーの長押し) を行うと速度が変わります。

なお、変更後に以下のコマンドでパラメータを保存し、DIP SW1-2を 'H' に設定すれば、リセットや電源再投入した後でも変更した値で通信が可能となります。

パラメータの保存 \$EB'0

### 3.4.4 ホスト・マシンへのメッセージ送信 (COM1\_txユニットu)

各ユニットから受け取った応答データを編集し、応答メッセージとしてホスト・マシンへ送信します。

#### (1) パラメータ領域

アドレス	内容
0	未定義
1	<p>ホスト・マシンへの応答フォーマットを設定します。</p> <p>0: 標準フォーマット (デフォルト値)</p> <p>1: 応答メッセージ最後の<math>\%r</math>の後ろに<math>\%n(0AH)</math>を付加します。</p> <p>2: 16進数表記のデータ並びに、1バイトごとにスペースを挿入します。</p> <p>3: 上記1と2の機能。</p>

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### ループ・バック試験

$\$u$  { コマンド種別 } { ステータス } { データ並び }

##### ・コマンド種別

R, W以外のコード (20H ~ FEH) を指定します。

##### ・ステータス

m以外を指定した場合は、応答メッセージのデータ並びが全て16進数表記になります。

mを指定した場合は、表示可能なコードは文字として返信します。

##### ・データ並び

任意のデータ並びを指定します。

#### (4) 操作例

$\$uCm$ Command Loopback Test

応答は、 $\#uCm$ Command Loopback Test

$\$uCo$ Command Loopback Test

応答は、 $\#uCo$ '436F6D6D616E64204C6F6F706261636B2054657374

### 3.4.5 LCD制御 (LCDユニットL)

ビットマップ表示メモリの内容をLCDへ転送して表示します。またバックライトの輝度調整や、指定時間経過によるバックライト自動消灯を行います。

#### (1) パラメータ領域

アドレス	内容
0~1	キー操作がなくなってからバックライトが自動消灯するまでの時間を10ms単位で設定します。FF FFを設定すると自動消灯しません。デフォルト値はFF FFです。 このパラメータはリトル・エンディアンで指定します。例えば15秒 (1500 05DCH) は、DC 05と指定します。
2~3	バックライト輝度を設定します。0が消灯、34 0E (0E34H=3636) が最大輝度です。 ユニット起動時に0の場合はバックライト機能がオフとなり、パラメータを変更してもバックライトは点灯しません。デフォルト値は最大輝度です。

#### (2) データ領域

アドレス	内容
0~7	下記の表示メモリ領域を書き換えた場合に1を書き込みます。1ブロック・ライン (8ドット・ライン) 単位で1バイトに対応しています。
8H~407H	テキスト用ビットマップ表示メモリです。文字フォントを展開します。LCDサイズ128x64ドット分のメモリを確保しています。1バイトが縦8ドットx横1ドット分に対応します。LSBが画面上側に対応します。
408H~807H	グラフィック用ビットマップ表示メモリです。テキスト面にOR演算して表示します。

**補足** R2.xxではカーソル用ビットマップ表示メモリも割り当てていましたが、R3.01で削除しました。

#### (3) 個別コマンド

ありません。

#### (4) 操作例

##### バックライト制御の例

- ・ 60秒で自動消灯、やや暗めの輝度の設定例 \$LW'0 70 17 00 08  
キー押下によりバックライトを再点灯するはKey\_codeユニットでパラメータ設定します。
- ・ 自動消灯せず、輝度最大の例 \$LW'0 FF FF 34 0E  
消灯状態で設定しても自動的に点灯しませんので、1回だけキー操作が必要になります。

##### ビットマップ表示メモリの直接操作による表示例

```
$LW'8008 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA
```

```
$LW'8000 1
```

```
$LW'8088 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA 55 AA
```

```
$LW'8001 1
```

以上によりLCD画面左上に市松模様を表示します。

### 3.4.6 文字表示 (PrintユニットJ)

文字列をビットマップ文字フォントに変換してビットマップ表示メモリへ書き込みます。

フォントに関する説明については「V850ES/Jx3サンプル・プログラム(フォント選択編)アプリケーション・ノート(U19530)」を参照してください。

#### (1) パラメータ領域

アドレス	内容
0~3	描画許可領域1。表示文字列中に制御コード11Hを指定した場合、以後の描画はここで指定した範囲内だけになります。開始位置(水平ドット位置, 垂直ドット位置), 終了位置(水平ドット位置, 垂直ドット位置)の順に設定します。ただし, 垂直開始ドット位置は8の倍数, 垂直終了ドット位置は8の倍数 - 1を指定します。
4~7	描画許可領域2。表示文字列中に制御コード12Hを指定した場合、以後の描画はここで指定した範囲内だけになります。設定内容は描画許可領域1に準じます。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 文字表示コマンド

`$JF { 水平位置 } { 垂直位置 } { 属性 } { 表示文字列 }`

##### ・水平位置

表示を開始する水平ドット位置を設定します。有効範囲は0からLCD\_lib\_c.hで定義している水平サイズ(nLCD\_xs) - 1までです。

##### ・垂直位置

表示を開始する垂直ドット位置を設定します。有効値は8の倍数です。なお文字が占有する垂直ドット数も、フォント・サイズに応じて8の倍数のドット数となります。例えば、9~16ドット・フォントを表示する場合は垂直16ドットを占有します。

##### ・属性

フォント, ピッチ, 字間を指定します。

bit7-5: フォント・サイズ

0: 5x7ドット(半角のみ), 1: 7x14ドット(半角のみ),  
2: 14x14ドット, 3: 16x16ドット, 4: 24x24ドット, 5: 20x20ドット,  
6: 12x12ドット, 7: 32x32ドット

2~7は、全角/半角混在で表示文字列を指定可能ですが、半角と判定すると1(7x14ドット)のフォントを使用します。

bit4 : 0で固定ピッチ, 1でプロポーショナル(可変ピッチ)になります。

bit3-0: 文字の後ろに付加する空白ドット数を指定します。ただし、全角/半角混在文字列における半角文字の後ろは、指定値の半分(端数切り上げ)の空白になります。

##### ・表示文字列

全角文字はシフトJISコードを使って指定します。改行は'A', '(アポストロフィ)は'27'と記述します。また、指定水平位置まで空白詰めを行いたい場合は,'09 xx'(xxは水平ドット位置)と記述します。属性を変更したい場合は,'0C'+属性, 位置を変更したい場合は,'0C'+水平位置+垂直位置を記述します。また'11'および'12'を記述することにより、描画範囲を制限できます。描画範囲をLCD全域に戻す場合は'10'を記述します。

このコマンドを実行すると、次の描画位置を次の形式で返信します。

`#JFo { 水平位置 } { 垂直位置 }`

## (4) 操作例

**全角半角混在，プロポーショナルの表示例**

全角，半角，アポストロフィ，改行の混在した例を次に示します。

```
$JF'0 0 51'Printユニット ('27'J'27')は,'A' 日本語テキストを表示'A'する demonstration'A'プログラムです。'A
```

**全角，16ドット固定ピッチの表示例**

以降の表示例では，平家物語の冒頭部分を引用しています。固定ピッチ表示の場合はプロポーショナル表示との差異を示すために，同じ行数で一部文字を省いて表示しています。1行1画面です。

```
$JF'0 0 42'祇園精舎の鐘の声'A'諸行無常の響有り'A'沙羅双樹の花の色'A'盛者必衰の理表す'A
```

```
$JF'0 0 42'驕者も久しからず'A'唯春の夜の夢の如'A'猛者も遂に亡びぬ'A'偏に風前の塵に同'A
```

**全角，プロポーショナルの表示例**

```
$JF'0 0 53'祇園精舎の鐘の声'A 0C52'諸行無常の響き有り'A 0C53'沙羅双樹の花の色'A 0C52'盛者必衰の理を表す'A
```

```
$JF'0 0 51'驕れる人も久しからず'A 0C52'唯春の夜の夢の如し'A 0C51'猛き者も遂には亡びぬ'A'偏に風の前の塵に同じ'A
```

**全角，24ドット・フォントの表示例**

```
$JF'0 0 81'電源ボタン'A 0C92'押して下さい'A 0C42 A
```

**1/4角，6ドット固定ピッチとプロポーショナル字間1ドットの交互表示例**

```
$JF'0 00 01'ABCDEFGHIJKLMNQRSTU'A
```

```
$JF'0 08 11'ABCDEFGHIJKLMNQRSTU'A
```

```
$JF'0 10 01'VWXYZabcdefghijklmnop'A
```

```
$JF'0 18 11'VWXYZabcdefghijklmnop'A
```

```
$JF'0 20 01'qrstuvwxyz0123456789.'A
```

```
$JF'0 28 11'qrstuvwxyz0123456789.'A
```

```
$JF'0 30 01'!"#$%&'27'()=~¥-^|@`{[<>'A
```

```
$JF'0 38 11'!"#$%&'27'()=~¥-^|@`{[<>'アカタ'A
```

**半角限定，8ドット固定ピッチの表示例**

```
$JF'0 00 22'-アイエオカキクコサスセ'A
```

```
$JF'0 10 22'ぢぢつナニネノハヒフホ'A
```

```
$JF'0 20 22E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEEF0A
```

```
$JF'0 30 22F0F1F2F3F4F5F6F7F8F9F0AFBFCFDFF0A
```

英語用半角フォントを組み込んでいるため半角カタカナの代わりにラテン文字が表示されます。

逆さ疑問符の左右逆は参照元データによるものです。

### 3.4.7 キー・コード生成 (Key\_codeユニットG)

キー・スキャン結果に対してチャタリング除去処理を行い、ASCIIコードに変換して出力します。また、ホスト・コマンドやリモコン受信コードによってキーを押した時と等価なコードを出すことが可能です。

#### (1) パラメータ領域

アドレス	内容
0	チャタリング除去処理用一致期間を10 ms単位で設定します。デフォルト値は2です。
1	動作モードを設定します。デフォルト値は、FAHです。 bit7: 1ならキー・オフ時にオフ・コード (FFH) を生成します。赤外線モコン送信を使用する場合は必ず1にします。 bit6: 1ならキー押下時にクリック音を出すようにMelodyユニットへ要求します。 bit5: 1ならキー押下時にバックライトを点灯するようにLCDユニットへ要求します。 bit4: 1ならスタンド・アローンでのキー操作が有効になります。 bit3: 1ならキー押下時に赤外線モコン送信をIR_txユニットへ要求します。 bit2: 未定義。 bit1: 1ならキー押下時に指定ユニットへ押されたキーを自動的に報告します。 bit0: 1なら赤外線モコン受信をキー入力として扱います。
2	キーの自動報告を行う場合に、報告先のユニットの記号を設定します。デフォルト値は“u”です。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### ホスト・マシンからのキー操作エミュレーション

\$GE { ASCIIコード }

- ・ASCIIコード: 1バイトのコード (1~FFH) を文字表記または16進数表記で指定します。FFHは、キー・オフを意味します。赤外線モコン送信のようにキー・オフがくるまで動作を続けるユニットがある場合は、必ずキー・オフもコマンドで送ってください。

#### (4) 操作例

キー・オフ・コード生成せず、クリック音無し、バックライト点灯無しの設定例

\$GW'1 2

キー・オフ・コード生成あり、クリック音あり、バックライト点灯ありの設定例

\$GW'1 FA

##### キー操作エミュレーションの例

Gキーの例 \$GEG

キー・オフの例 \$GE'FF

注 赤外線モコン送信機能が有効な場合は、必ず最後にキー・オフを送ってください。

### 3.4.8 メロディ出力 (MelodyユニットM)

データ領域に格納したメロディ・スクリプトに従ってメロディ演奏を行います。また、キー押下時にクリック音を出します。メロディ演奏中は、キー押下により一時的に演奏を中断してクリック音を出します。

#### (1) パラメータ領域

アドレス	内容
0	演奏終了時の自動報告先のユニット記号を設定します。デフォルト値は“u”です。
1	演奏中にキー・クリック音を出さず場合の前後の無音時間を設定します。 bit7-4：事前の無音時間を10ms単位で指定します。デフォルト値は7です。 bit3-0：事後の無音時間を10ms単位で指定します。デフォルト値はFHです。
2	クリック音のノート番号を0~7FHの範囲で設定します。デフォルト値は6DHです。
3	クリック音の長さを10ms単位で指定します(1~E6Hの範囲内)。デフォルト値はFHです。

#### (2) データ領域

アドレス	内容
0~3E7H	メロディ・スクリプト格納領域です。スクリプトの書式については(5)を参照してください。なお、スクリプトは、他のユニットのデータ領域にあっても演奏可能です。

#### (3) 個別コマンド

##### 演奏開始

\$MP {ユニット番号} {先頭アドレス}

- ・ユニット番号

メロディ・スクリプトを格納したユニットの記号を指定します。

- ・先頭アドレス

データ領域内でのスクリプトの先頭アドレスを指定します。このパラメータはビッグ・エンディアンのため、アドレスを上位バイト、下位バイトの順に指定します。

スクリプト演奏を終了すると次のメッセージが返信されます。

#MF! {ステータス}

- ・ステータス

正常終了時は6FH (“o”)、不正スクリプト検出時は70H (“p”)になります。

##### 演奏強制停止

\$ME

**(4) 操作例****「浜辺の歌（作曲：成田為三（1945年没））」のスク립ト格納の例**

次の例はリピート回数1に設定してあります。

```
$MW'8140 0A200444048C0410026804F00149E749E84E1250EA52E850124EEA50484BE74E
$MW'8160 E84D124BEA49E84EE752E850124EEA506C50E7FE2449E749E84E1250EA52E850
$MW'8180 124EEA50484BE74EE84D124BEA494852E749E850E74E6C4EE7FE2455E755E850
$MW'81A0 E7554851E755EB52E857E757E853E74EE850E7556C55E7FE2449E749E84E1250
$MW'81C0 EA524850124EEA50E84BE74EE84D124BEA49E852E749E850E74E6C4EE7FE48F4
$MW'81E0 01 96 00 FF FF
```

**演奏開始の例**

浜辺の歌 \$MPM'0140

**演奏強制停止の例**

の演奏開始例の後にコマンド入力 \$ME

**(5) スクリプトのフォーマット**

スクリプトは、マクロ定義とスクリプト本体から構成します。スクリプト本体は、制御コード（1バイト）+パラメータ（0~2バイト）を1組とし、これを連続して並べたデータ列となります。

**スクリプトの先頭バイト（1バイト）**

マクロのバイト数（=スクリプト本体の先頭までのバイト数）を記述します。

**マクロ定義**

1マクロ当り発音時間1バイト+無音時間1バイトの2バイトで構成します。時間はそれぞれ10ms単位で1（10ms）~E6H（2.3s）の範囲で指定します。

**スクリプト本体**

制御コードとパラメータの組み合わせで記述します。

## ・音階制御コード（0~7FH）

音階を示すノート番号です。対応する周波数をブザーへ出力します。ノート番号と周波数の対応は表3-3を参照してください。

出力時間は続くパラメータ1バイトで指定します。続くパラメータが1~E6Hであれば、10ms単位の時間と見なします。続くパラメータがE7H以上であればマクロ定義の時間に従います。E7Hがマクロ定義の先頭の組に対応します。

## ・終了制御コード（FFH）

スクリプトを終了します。続くパラメータはありません。



・無音制御コード (FEH)

出力を無音にします。続くパラメータ1バイトで10 ms単位の無音時間を指定します。

・リピート・カウンタ制御コード (F0H~F3H)

リピート・カウンタの回数設定を行います。F0H~F3H の4つに対応して独立に設定可能です。続く1バイトで反復数 (1~255回) を指定します。

・リピート制御コード (F4H~F7H)

リピート制御を行います。F4HがF0Hで設定したカウンタに対応します。

続くパラメータ3バイトとカウンタの値で動作を決定します。

条件1: パラメータ1バイト目が0で、リピート・カウンタが0の場合

パラメータ2~3バイト目 (リトル・エンディアン数値) を進み移動バイト数として、実行スクリプト位置に加算します。

条件2: パラメータ1バイト目が1で、リピート・カウンタが0以外の場合

パラメータ2~3バイト目 (リトル・エンディアン数値) を戻り移動バイト数として、実行スクリプト位置から減算します。

条件3: その他の場合

リピート・カウンタを-1して次に進みます。

なお移動バイト数は3バイト・パラメータの直後を0 (基点) として計算します。

表3-3 ノートと周波数の対応

階名	ノート	周波数	ノート	周波数	ノート	周波数	ノート	周波数	ノート	周波数
ド	00	8.2	18	32.7	30	130.8	48	523	60	2093
ド#,レb	01	8.7	19	34.6	31	138.6	49	554	61	2217
レ	02	9.2	1A	36.7	32	146.8	4A	587	62	2349
レ#,ミb	03	9.7	1B	38.9	33	155.6	4B	622	63	2489
ミ	04	10.3	1C	41.2	34	164.8	4C	659	64	2637
ファ	05	10.9	1D	43.7	35	174.6	4D	698	65	2794
ファ#,ソb	06	11.6	1E	46.2	36	185.0	4E	740	66	2960
ソ	07	12.2	1F	49.0	37	196.0	4F	784	67	3136
ソ#,ラb	08	13.0	20	51.9	38	207.7	50	831	68	3322
ラ	09	13.8	21	55.0	39	220.0	51	880	69	3520
ラ#,シb	0A	14.6	22	58.3	3A	233.1	52	932	6A	3729
シ	0B	15.4	23	61.7	3B	246.9	53	988	6B	3951
ド	0C	16.4	24	65.4	3C	261.6	54	1047	6C	4186
ド#,レb	0D	17.3	25	69.3	3D	277.2	55	1109	6D	4435
レ	0E	18.4	26	73.4	3E	293.7	56	1175	6E	4699
レ#,ミb	0F	19.4	27	77.8	3F	311.1	57	1245	6F	4978
ミ	10	20.6	28	82.4	40	330	58	1319	70	5274
ファ	11	21.8	29	87.3	41	349	59	1397	71	5588
ファ#,ソb	12	23.1	2A	92.5	42	370	5A	1480	72	5920
ソ	13	24.5	2B	98.0	43	392	5B	1568	73	6272
ソ#,ラb	14	26.0	2C	103.8	44	415	5C	1661	74	6645
ラ	15	27.5	2D	110.0	45	440	5D	1760	75	7040
ラ#,シb	16	29.1	2E	116.5	46	466	5E	1865	76	7459
シ	17	30.9	2F	123.5	47	494	5F	1976	77	7902
ド									78	8372
ド#,レb									79	8870
レ									7A	9397
レ#,ミb									7B	9956
ミ									7C	10548
ファ									7D	11175
ファ#,ソb									7E	11840
ソ									7F	12544

注 ノートは16進数表記です。周波数[Hz]は、計算値です。実際に再生可能な周波数は、ハードウェアの特性に依存します。

### 3.4.9 LEDブリンク (LED\_blinkユニットB)

PIO0～3接続 (スタティック接続) のLEDの点灯, 消灯, 自動点滅を制御します。以下の説明では, このLEDをLED0～3と表現します。

#### (1) パラメータ領域

アドレス	内容
0～3	LED0～3の点滅状況ステータスです。\$BCコマンドで02H～FDHを指定した場合に, 1回点滅するごとにダウン・カウントします。設定は\$BCコマンドで行います。\$BWで直接書き換えた場合の動作は保証されません。
4～7	LED0～3の点滅のオン時間を10ms単位で各1バイトで設定します。
8～BH	LED0～3の点滅のオフ時間を10ms単位で各1バイトで設定します。
CH	LED0～3の点滅完了を指定ユニットへ自動報告するかどうかを設定します。 bit0～3: LED0～3に対応します。“1”なら自動報告します。デフォルト値は“1”です。
DH	自動報告先のユニット記号を設定します。デフォルト値は“u”です。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 点滅コマンド

\$BC {動作コード}

・動作コード

LED0～3のそれぞれの動作を1LED当り下記コードで指定します。

00: 現在の状態を変更しません。

01: 消灯します (赤LEDが消灯, 緑LEDが点灯)。

FFH: 点灯します (赤LEDが点灯, 緑LEDが消灯)。

FEH: 次のコマンドが来るまで点滅を繰り返します。

02H～FDH: (指定値-1) 回の点滅後, 消灯します。

#### (4) 操作例

##### 全点灯の例

\$BC'FF FF FF FF

##### 全消灯の例

\$BC'1 1 1 1

##### 点滅の例

\$BW'4 0A 14 46 28 0A 14 1E 3C 0F 75

\$BC'15 15 15 15

上記例は, 各LEDが20回点滅します。設定時間は次のとおりです。

LED0 : オン100ms , オフ100ms

LED1 : オン200ms , オフ200ms

LED2 : オン700ms , オフ300ms

LED3 : オン400ms , オフ600ms

点滅完了により下記メッセージが返信されます。

#BF!01      LED0の点滅完了 ( 4秒後 )

#BF!02      LED1の点滅完了 ( 8秒後 )

#BF!0C      LED2と3の点滅完了 ( 20秒後 )

### 3.4.10 日付・時刻表示，アラーム（CalendarユニットC）

日付や時刻をLEDあるいはLCDに表示します。予約時刻に自動的にLEDを点滅したりメロディを演奏するアラーム機能もあります。

#### (1) パラメータ領域

アドレス	内容
0	LCDへ時刻表示を行う場合の位置を設定します。 <ul style="list-style-type: none"> <li>・1/4角表示の場合 日付・時刻表示を開始する水平ドット位置を指定します。垂直位置はアドレス3で設定します。</li> <li>・全角半角表示の場合 bit7-4：時刻表示を開始する垂直位置（8ドット単位）を設定します。 bit3-0：時刻表示を開始する水平位置（8ドット単位）を設定します。</li> </ul>
1	LCDへ日付表示を行う場合の位置を設定します。 <ul style="list-style-type: none"> <li>・1/4角表示の場合 日付・時刻表示を終了する水平ドット位置を設定します。</li> <li>・全角半角表示の場合 bit7-4：日付表示を開始する垂直位置（8ドット単位）を設定します。 bit3-0に日付表示を開始する水平位置（8ドット単位）を設定します。</li> </ul>
2	LCD表示の書式を設定します。下記ビットが"1"の項目を表示します。ただし表示可能範囲を超えた場合は打ち切ります。 bit7：年上2桁，bit6：年下2桁，bit5：月，bit4：日，bit3：曜日，bit2：時，bit1：分，bit0：秒
3	0を設定するとLCD表示を停止します（表示済の内容はクリアしません）。 0以外は，LCD表示の文字フォント設定になります。 bit7-1：1～31を設定すると1/4角フォントで1～31行目に表示します。 32なら半角表示，64なら全角3行表示，65なら全角2行表示，66なら全角3行で年月日を漢字表示します。 bit0：1を設定すると12時制でam/pmを表示します。0なら24時制で表示をします。
4	LED表示の場合の位置を設定します。LED位置は左端が0，右端がBHになります。 bit7-4：表示開始桁を設定します（0～BHの範囲）。 bit3-0：表示終了桁を設定します（0～BHの範囲）。
5	LED表示の書式を設定します。0なら表示を停止します（表示済の内容はクリアしません）。 下記ビットで"1"の項目があれば表示します。ただし表示可能範囲を超えた場合は打ち切ります。 bit7：年上2桁，bit6：年下2桁，bit5：月，bit4：日，bit2：時，bit1：分，bit0：秒 また時制を下記ビットで設定します。 bit3：1を設定すると12時制でam/pmを表示します。0なら24時制で表示をします。 am表示は時刻の左下の縦棒点灯，pm表示は時刻の左上の縦棒点灯です。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 日付と時刻の設定

\$CD {年（西暦下2桁）月2桁 日2桁 曜日2桁 時2桁 分2桁 秒2桁の10進数文字列}

注 曜日は00～06が日～土に対応しますが，日付から計算した値を設定するため，ここで指定した値は無視されます。時は表示が12時制でも24時制で設定します。

**日付と時刻の読み取り**

\$CI

\$CDのコマンド・パラメータと同じ形式で日付と時刻を返信します。

**アラーム設定**

\$CS {区分} {曜日} {時} {分} {LED0} {LED1} {モード} {メロディ} {報告先}

## ・区分

0を指定してください。

## ・曜日

bit0～6が日～土に対応します。アラームを出す曜日（複数可）のbitを1にします。

## ・時, 分 (各1バイト)

時は24時制で設定します。時, 分ともBCDコードで指定します

## ・LED0, LED1

点滅動作コードを指定します。設定値は3. 4. 9 LEDリンクを参照してください。

## ・モード

アラーム時刻になった時の動作を設定します。

bit7: 1ならLED3を消灯（赤LED消灯）します。

bit6: 1ならLED3を点灯（赤LED点灯）します（bit7同時設定時は一瞬消灯後に再点灯）。

bit5: 1ならLED1に指定動作コードを設定します。

bit4: 1ならLED0に指定動作コードを設定します。

bit3: 1ならLCDバックライトの点灯要求を行います。

bit2: 1なら1回アラームを検出するとアラーム設定を自動消去します。

bit1: 1なら指定報告先へアラーム発生を通知します。

bit0: 1ならメロディ・スクリプトを実行します。

## ・メロディ

スクリプトが格納されているユニット番号, データ領域のスクリプト開始アドレス（上位, 下位の順）を指定します。

## ・報告先

アラーム時の報告先ユニット記号を指定します。

アラーム設定は20件まで登録できます。すでに曜日・時・分が同じ設定があった場合は, 内容を置き換えます。ただしアラーム・モードが0なら登録を抹消します。

登録, 置換え, 抹消が行われた場合は, 登録リスト上での位置（0～13H）を返信します。

**アラーム設定内容確認**

\$CL {登録リスト上での位置（0～13H）}

アラーム設定の内容を返信します。

**(4) 操作例****日時の設定例**

2008年1月16日水曜日18時37分00秒の例 \$CD08011603183700

2008年02月28日木曜日23時59分56秒の例 \$CD08022800235956

#### LEDへの日付，時刻の表示例

年（4桁）月日時分 \$CW'4 0B F7  
 年（2桁）月日時分秒 \$CW'4 0B 77  
 月日ampm時分秒 \$CW'4 2B 3F  
 LED表示停止 \$CW'5 0

#### LCDへの日付，時刻の表示例

1/4角，年（2桁）月日曜日時分秒ampm \$CW'0 00 7F 7F 03  
 半角，年（4桁）月日曜日時分秒ampm（2行） \$CW'0 42 20 FF 41  
 全角，年月日，時分秒（2行） \$CW'0 40 20 77 82  
 全角，年（4桁）月日曜日ampm時分 \$CW'0 40 00 FE 85  
 LCD表示停止 \$CW'3 0

#### アラームの設定と実行の例

まずアラーム・メロディを設定します。

```
$MW'8140 0A200444048C0410026804F00149E749E84E1250EA52E850124EEA50484BE74E
$MW'8160 E84D124BEA49E84EE752E850124EEA506C50E7FE2449E749E84E1250EA52E850
$MW'8180 124EEA50484BE74EE84D124BEA494852E749E850E74E6C4EE7FE2455E755E850
$MW'81A0 E7554851E755EB52E857E757E853E74EE850E7556C55E7FE2449E749E84E1250
$MW'81C0 EA524850124EEA50E84BE74EE84D124BEA49E852E749E850E74E6C4EE7FE48F4
$MW'81E0 01 96 00 FF FF
```

次にアラームLEDの点滅周期を設定します。

```
$BW'4 10 20
$BW'8 10 20
```

確認用の時計表示 \$CW'0 40 20 77 82

時計を設定（2008年1月16日水曜日18:59:00） \$CD"08011603185900

アラーム時刻，動作内容を設定します。

```
水曜19時1分 $CS'0 08 19 01 20 20 7B 'M' 01 40 75
水曜19時2分 $CS'0 08 19 02 20 20 BB 'M' 01 40 75
```

時計を進めます（待ち時間を短縮のため）。

```
2008年01月16日(水)19:00:50 $CD"08011603190050
```

以上の設定で，7時1分にアラームが実行されます。

次に7時2分までにスタンバイ・コマンドを入力します。 \$PSave

7時2分になるとスタンバイが解除され，アラームが実行されます。ただし，スタンド・アローン・モードで実行した場合は，LED点滅，メロディ演奏の実行がキャンセルされます。

#### アラーム設定の削除例

```
$CS'0 08 19 01 0 0 0 0 0 0
$CS'0 08 19 02 0 0 0 0 0 0
```

これにより で設定したアラーム設定が無効になります。

## 3.4.11 内蔵A/D変換 (ADC\_inユニットA)

各種アナログ入力の値を読み取り、結果をLED、LCDに表示したり、ホスト・マシンへ報告します。また読み取り結果を自動的にEEPROMに格納する機能もあります。

## (1) パラメータ領域

アドレス	内容
0	<p>定期的にアナログ値を読み取る場合は、開始A/Dチャンネル番号を設定します。定期的な変換をしない場合は、0を設定します。</p> <p>A/Dチャンネル番号とベース・ボードの信号の関係は次のとおりです。</p> <p>80H：AN00 DIP SW3の設定内容を簡易D/Aコンバータで変換した信号。</p> <p>81H：AN01 3線シリアル接続のD/Aコンバータのチャンネル1出力信号。</p> <p>82H：AN02 3線シリアル接続のD/Aコンバータのチャンネル2出力信号。</p> <p>83H：AN03 3線シリアル接続のD/Aコンバータのチャンネル3出力信号。</p> <p>84H：AN04 ポテンショメータRV1の出力信号。</p> <p>85H：AN05 ポテンショメータRV2の出力信号。</p> <p>86H：AN06 ポテンショメータRV3の出力信号。</p> <p>87H：AN07 ポテンショメータRV4の出力信号。</p> <p>88H：AUin テスト端子の拡張アナログ入力信号。</p> <p>89H：MCU内蔵D/Aコンバータからのループバック信号。</p> <p>注 動作後にチャンネルや周期を変更する場合は、いったん0を設定してから変更してください。</p>
1	定期的にアナログ値を読み取る場合の終了A/Dチャンネル番号を設定します。
2	<p>定期的に読み取った値（採取データ）をどのように処理するかを設定します。</p> <p>bit6：1なら採取データをEEPROMに格納する処理が完了した時に指定報告先へ報告します。</p> <p>bit4：1なら採取データを指定報告先へ通知します。</p> <p>bit2：1なら採取データをLCDに表示します。</p> <p>表示内容は、採取時刻（時分秒）、チャンネル番号、採取データ（10進数表示）です。</p> <p>bit1：1なら採取データをLEDに表示します。</p> <p>表示内容は、採取時刻（時分秒）、チャンネル番号、採取データ（10進数表示）です。</p> <p>bit0：1なら採取データをEEPROMに格納します。</p> <p>採取データ1件につき8バイト（月日時分秒、チャンネル番号、データ2バイト（ビッグ・エンディアン））を格納します。</p>
3	未定義です。
4~7	変換周期を10ms単位の4バイトの数値で設定します。リトル・エンディアンです。
8~BH	1変換周期内に複数チャンネルを変換する場合、チャンネル間の変換間隔を10ms単位の4バイトの数値で設定します。リトル・エンディアンです。
CH	<p>LCD表示の場合の表示位置を設定します。</p> <p>bit7-4：表示の終了行（垂直ドット位置）を8ドット単位で指定します。</p> <p>bit3-0：表示の開始行（垂直ドット位置）を8ドット単位で指定します。</p>
DH	<p>LCD表示フォントを設定します。</p> <p>bit7：0なら半角フォント，1なら1/4角フォントです。</p> <p>bit4-0：表示の開始桁（水平ドット位置）を8ドット単位で指定します。</p>
EH	採取データを自動報告する場合の報告先ユニット記号を指定します。
FH	EEPROM格納完了報告の場合の報告先ユニット記号を指定します。
10H~11H	EEPROM格納時の開始アドレス（リトル・エンディアン）を指定します。
12H~13H	EEPROM格納時の格納件数（リトル・エンディアン）を指定します。



**(2) データ領域**

ありません。

**(3) 個別コマンド****アナログ値の読み取り**

\$AD { A/Dチャンネル番号 (80H ~ 89H) }

・ A/Dチャンネル番号

チャンネル番号と信号の対応はパラメータ・アドレス0の説明を参照してください。

本コマンド実行後に次のメッセージが返信されます。

#ADo { A/Dチャンネル番号 } { 採取データ2バイト (ビッグ・エンディアン) }

**(4) 操作例****周期変換を停止する例**

\$AW'0 0 0 0

変換は停止しますが、表示済の内容をクリアしません。

**1チャンネルの読み取り値をLEDに表示する例**

以下の例は、指定チャンネルを0.5秒ごとに読み取ってLEDに表示します。

DIP SW3                   \$AW'0 80 80 02 0 32 0 0 0 32 0 0 0

シリアルD/A1             \$AW'0 81 81 02 0 32 0 0 0 32 0 0 0

シリアルD/A2             \$AW'0 82 82 02 0 32 0 0 0 32 0 0 0

シリアルD/A3             \$AW'0 83 83 02 0 32 0 0 0 32 0 0 0

ポテンショメータ1     \$AW'0 84 84 02 0 32 0 0 0 32 0 0 0

ポテンショメータ2     \$AW'0 85 85 02 0 32 0 0 0 32 0 0 0

ポテンショメータ3     \$AW'0 86 86 02 0 32 0 0 0 32 0 0 0

ポテンショメータ4     \$AW'0 87 87 02 0 32 0 0 0 32 0 0 0

外部アナログ (AUin)   \$AW'0 88 88 02 0 32 0 0 0 32 0 0 0

内蔵D/Aループバック   \$AW'0 89 89 02 0 32 0 0 0 32 0 0 0

注 内蔵D/Aループバックの読み取り値はD/A設定値の4倍です。

**連続チャンネルをLEDに表示する例**

以下の例は、0.5秒ごとにチャンネルを切り替えながら読み取り、LED表示を行います。

シリアルD/A1~3     \$AW'0 81 83 02 0 96 0 0 0 32 0 0 0

ポテンショメータ1~4   \$AW'0 84 87 02 0 C8 0 0 0 32 0 0 0

**ホスト・マシンへ採取データを自動報告する例**

以下の例は、1秒ごとにポテンショメータ1~4を切り替えながら読み取ってホスト・マシンへ報告します。

\$AW'E 75

\$AW'0 84 87 12 0 A0 1 0 0 64 0 0 0

**LCDに表示する例**

次の例は、全チャンネルを1秒ごとに切り替えながら、1/4角でLCDへ表示します。

```
$AW'0 80 89 04 0 E8 3 0 0 64 0 0 0 70 80
```

次の例は、ポテンシオメータ1~4を0.5秒ごとに切り替えながら、半角でLCDへ表示します。

```
$AW'0 84 87 04 0 C8 0 0 0 32 0 0 0 70 00
```

**EEPROMへ自動記録する例**

以下の例は、ポテンシオメータ1~4を0.5秒ごとに切り替えながらデータ採取し、合計32件を記録します。

```
$AW'F 75 00 08 20 00
```

```
$AW'0 84 87 41 0 C8 0 0 0 32 0 0 0
```

以上のコマンド入力から約16秒後に#AF!が表示されたら下記で内容確認します。

```
$ELE'00 00 00 08 80 00
```

```
#ELoが返信されたらリード $ER'800 80
```

```
$ELE'00 00 80 08 80 00
```

```
#ELoが返信されたらリード $ER'800 80
```

**ホスト・マシンからオン・デマンドでアナログ値を読み取る例**

```
DIP SW3 $AD'80
```

```
ポテンシオメータ1 $AD'84
```

```
ポテンシオメータ2 $AD'85
```

```
ポテンシオメータ3 $AD'86
```

```
ポテンシオメータ4 $AD'87
```

以上のコマンドを入力すると以下のメッセージが返信されます。

```
#ADo80xxxx
```

```
#ADo84xxxx
```

```
#ADo85xxxx
```

```
#ADo86xxxx
```

```
#ADo87xxxx
```

注 xxxxは採取データ（ビッグ・エンディアン表記）です。

### 3.4.12 内蔵D/A変換 (DAC\_outユニットD)

内蔵D/Aコンバータから設定値に応じた電圧を出力します。

出力電圧は、 $VDD \times \text{設定値} / 256$ になります。

#### (1) パラメータ領域

アドレス	内容
0	ユニット起動時にD/Aチャンネル80H (AUoutへ接続) に設定する初期値
1	ユニット起動時にD/Aチャンネル81H (A/Dチャンネル89Hへ接続) に設定する初期値

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 出力値設定コマンド

`$DD {チャンネル番号 (80H~81H)} {設定値 (0~FFH)}`

このコマンドは指定チャンネルのD/Aコンバータに設定値を書き込みます。

#### (4) 操作例

##### D/Aチャンネル81Hの出力値をLEDに表示する例

LED表示設定 `$AW'0 89 89 02 0 32 0 0 0 32 0 0 0`

以下、D/Aに設定した値に応じて表示が変わります。表示値は設定値の4倍になります。

`$DD'81 80`

`$DD'81 FF`

`$DD'81 30`

`$DD'81 0`

表示停止 `$AW'0 0 0 0`

### 3.4.13 シリアル接続D/A変換 (CSI\_DACユニットF)

ベース・ボード搭載の3線シリアル接続D/Aコンバータから設定値に応じた電圧を出力します。

#### (1) パラメータ領域

ありません。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 出力値設定コマンド

\$FD {チャンネル番号 (1~3)} {設定値 (0~FFH)}

このコマンドは3線シリアル接続D/Aコンバータの指定チャンネルに設定値を書き込みます。

#### (4) 操作例

##### D/Aチャンネル1~3の出力値をLCDに表示する例

LCD表示設定 \$AW'0 81 83 04 0 14 0 0 0 0A 0 0 0 61 00

以下, D/Aに設定した値に応じて表示が変わります。表示値は設定値の4倍になります。

\$FD'1 80

\$FD'2 40

\$FD'3 20

\$FD'1 1

\$FD'2 2

\$FD'3 3

表示停止 \$AW'0 0 0 0

### 3.4.14 シリアル接続EEPROM (CSI\_EEPROMユニットE)

指定ユニットの指定データ範囲の内容をEEPROMに格納したり、EEPROMから読み出すことができます。また、全ユニットの全パラメータを一括でEEPROMに保存したり、復旧することができます。

#### (1) パラメータ領域

アドレス	内容
0~1	接続されているEEPROMの容量を256バイト単位の数値（リトル・エンディアン）で設定します。デフォルト値は128（32Kバイト）です。
2	EEPROMのページ・サイズを設定します。デフォルト値は64です。

#### (2) データ領域

アドレス	内容
0~FFH	パラメータ領域の保存・復旧用のワーク・メモリとして使います。保存・復旧作業時以外は自由に使えます。

#### (3) 個別コマンド

##### 全ユニットの全パラメータを保存

\$EB {保存先エリア番号 (0~3)}

保存先は4つ選べます。EEPROMの先頭から1エリア当り256バイト、合計1024バイトを割り当てています。

なお、ベース・ボードのSW1\_2の設定による起動時のパラメータ読み込みは、エリア番号0固定です。

##### 全ユニットの全パラメータを復旧

\$EP {保存先エリア番号 (0~3)}

##### 指定データ領域の保存

\$ES {ユニット} {データ・アドレス} {EEPROMアドレス} {バイト数}

###### ・ユニット

保存対象データ領域があるユニットの記号を指定します。

###### ・データ・アドレス

保存対象データの開始アドレスを2バイト（リトル・エンディアン）で指定します。

###### ・EEPROMアドレス

保存先のEEPROMアドレスを2バイト（リトル・エンディアン）で指定します。

###### ・バイト数

保存するデータのバイト数を2バイト（リトル・エンディアン）で指定します。

**指定データ領域へのデータ復旧**

\$EL {ユニット} {データ・アドレス} {EEPROMアドレス} {バイト数}

- ・ ユニット

復旧先データ領域があるユニットの記号を指定します。

- ・ データ・アドレス

データ領域の復旧先の開始アドレスを2バイト（リトル・エンディアン）で指定します。

- ・ EEPROMアドレス

保存データがあるEEPROMアドレスを2バイト（リトル・エンディアン）で指定します。

- ・ バイト数

復旧するデータのバイト数を2バイト（リトル・エンディアン）で指定します。

**(4) 操作例****全パラメータの保存 / 復旧の例**

エリア0への保存例 \$EB'0

エリア0からの復旧例 \$EP'0

**データの保存 / 復旧の例**

まず例として保存するデータをMelodyユニットのデータ領域に書き込みます。

```
$MW'8200 0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF
```

このデータをEEPROMの7400Hから保存します \$ESM'00 02 00 74 20 00

完了メッセージ#ESoを待ちます。

次にいったんメロディ・ユニットのデータ領域に別な値を書きます。

```
$MW'8200 FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
```

内容確認のためリードします。 \$MR'8200 20

次に保存データを復旧します。

\$ELM'00 02 00 74 20 00

完了メッセージ#ELoを待ちます。

内容確認のためリードします。 \$MR'8200 20

最初に書き込んだ0123...が表示されます。

### 3.4.15 拡張IOデバイス (I2C\_IOユニット)

拡張デバイスに接続された黄色LEDの点灯・消灯や、スイッチの読み取りを行います。

#### (1) パラメータ領域

アドレス	内容
0	拡張デバイスのアドレスを設定します。デフォルト値は4EHです。
1	bit7 : 1なら高速モードで動作します。デフォルト値は0です。 bit1 : 1ならDIP SW3の変化を指定報告先へ自動的に報告します。デフォルト値は1です。
2	自動報告先のユニット記号を設定します。デフォルト値は“u”です。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 拡張IOデバイスへのデータ出力

\$ID {レジスタ番号} {出力データ}

- ・レジスタ番号

ベース・ボードの黄色LED接続レジスタは、3です。

- ・出力データ

ベース・ボードの黄色LEDを点灯する場合は対応ビットを“0”にします。

##### 拡張IOデバイスからのデータ入力

\$II {レジスタ番号}

- ・レジスタ番号

DIP SW3接続レジスタは、0です。

#### (4) 操作例

##### 黄色LEDの点灯制御の例

全点灯の例 \$ID'3 00

部分点灯の例 \$ID'3 55

全消灯の例 \$ID'3 FF

##### DIP SW3の内容読み取りの例

\$II'0

次のメッセージが返信されます。

#IIo { DIP SW3の内容 }

##### DIP SW3の変化を自動報告する / しない設定の例

ホスト・マシンへ自動報告しない場合 \$IW'1 00

ホスト・マシンへ自動報告する場合 \$IW'1 02 75

### 3.4.16 赤外リモコン受信 (IR\_rxユニットR)

NECフォーマット (16ビット・カスタム・コード, コマンド単発送信モード) のコードを受信します。

#### (1) パラメータ領域

アドレス	内容
0~1	NECフォーマットのカスタム・コードを設定します。カスタム・コードの割り当てを受けてない場合は、デフォルト値 (00 FF) のままで使用してください。
2	bit1 : 1なら受信コードを指定されたユニットへ自動報告します。デフォルト値は1です。 bit0 : 1なら受信コードをキー押下信号として扱うようにキー・コード生成ユニットへ要求します。デフォルト値は0です。
3	自動報告先のユニット記号を指定します。デフォルト値は“u”です。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

ありません。

#### (4) 操作例

##### 受信データをホスト・マシンへ通知する例

```
$RW'2 02 75
```

この状態で、例えばデータ4EHを受信した場合、ホスト・マシンに下記メッセージが送信されます。

```
#RI!'4E
```



### 3.4.17 赤外リモコン送信 (IR\_txユニットT)

NECフォーマット (16ビット・カスタム・コード, コマンド単発送信モード) のコードを送信します。

#### (1) パラメータ領域

アドレス	内容
0~1	NECフォーマットのカスタム・コードを設定します。カスタム・コードの割り当てを受けてない場合は、デフォルト値 (00 FF) のままで使用してください。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### コード送信

\$TD {コード}

・コード

送信可能コードは, 1~FEHの範囲です。

送信可能コードを送信すると, 以降自動的にリピート・コードを送信し続けるため, 送信停止コード (FFH) 指定したコマンドを必ず入力する必要があります。

#### (4) 操作例

##### ホスト・コマンドによる送信例

4EHを送信 \$TD'4E

キー・オフ送信 \$TD'FF

注 必ずキー・オフを送ってください。送らないとリピート・コードが出力され続けます。

##### ベース・ボードのキーに対応したコードを送信する例

キー・コード生成ユニットGのパラメータ・アドレス1のビット3および7を“1”に設定すればキー押下時に対応したコード (ASCIIコード) が送信されます。キーを離すとリピート・コード送信が止まります。

### 3.4.18 グラフィック・パターン描画 (GraphユニットN)

LCDビットマップ表示メモリにグラフィック・パターンを展開します。

#### (1) パラメータ領域

ありません。

#### (2) データ領域

ありません。

#### (3) 個別コマンド

##### 図形パターン描画

\$NP {水平開始位置} {垂直開始位置} {水平終了位置} {垂直終了位置} {属性} {図形パターン}

- ・ 水平開始位置 / 水平終了位置

ドット単位で指定します。

- ・ 垂直開始位置

8ドット刻みで指定します。

- ・ 垂直終了位置

8ドット刻み-1で指定します。

- ・ 属性:

bit7-4 : ビットマップ表示メモリへ図形パターンを書き込む場合の演算方法を指定します。

0 : 置換, 1 : AND, 2 : OR, 3 : XOR

bit3-0 : ビットマップ面 (0~3) を指定します。ただし本版では下記のみ指定可能です。

0 : テキスト面, 1 : グラフィック面

- ・ 図形パターン

縦8ドットx横1ドットのパターンを1バイトで指定します。本版では、LSBが上側になります。

#### (4) 操作例

##### 画面中央付近を反転表示する例

反転したことが分かりやすいように、まず例として文字を表示します。

\$JF'0 0 42'グラフィック描画'A'ユニットNで画面'A'中央を反転表示す'A'の例を示します。'A'

上記表示が出てから次のコマンドを入力します。以下の例では、文字の反転を行います。

テキスト面XOR \$NP'10 08 6F 30 30 FF

テキスト面XOR \$NP'10 08 6F 30 30 FF

### 3.4.19 メニュー制御 (MenuユニットQ)

メニュー制御スクリプトに基づいて自動的にメニューの表示・階層移動などの制御を行います。また、メニューではなく、単純なテキスト・スクロール表示の制御も可能です。

#### (1) パラメータ領域

アドレス	内容
0~1	メニュー制御スクリプトまたはスクロール用テキストのバイト・サイズをリトル・エンディアンで設定します。アクセス保護用です。
2	0を設定するとスタンド・アロンのキー操作を受付ません。 1を設定するとベース・ボードの キ - によるスクロール操作が可能になります。 2を設定すると キ - , BSキー , ENTキーによるメニュー操作が可能になります。 デフォルト値は0です。
3	bit1 : 1を設定するとメニュー制御スクリプトの指示に基づく値を指定報告先へ報告します。デフォルト値は1です。
4	報告先のユニット記号を指定します。デフォルト値は "u" です。

#### (2) データ領域

アドレス	内容
0~17FFH	スクロール・テキストやメニュー・スクリプトを格納する領域です。他のユニットのデータ領域にあっても制御できます。

#### (3) 個別コマンド

##### スクロール属性設定

\$QA{ユニット}{アドレス}{表示行位置}{レコード・サイズ}{開始行}{終了行}{開始桁}{終了桁}

- ・ユニット

スクロール表示用テキストが格納されているユニット記号を指定します。

- ・アドレス (2バイト)

スクロール表示用テキストが格納されている先頭アドレスをリトル・エンディアンで指定します。

- ・表示行位置 (2バイト)

実際に表示を開始するテキスト上の行番号 (0から始まる番号) をリトル・エンディアンで指定します。

- ・レコード・サイズ

1行の情報を何バイトで構成するか指定します。なお、行情報の1バイト目には、文字フォントやピッチを指定します (内容は3.6.4 (3) の\$JFコマンドの属性と同じです)。行情報の2バイト目から文字列を格納します。文字列の最後は¥n¥0(0AH 00H)にする必要があります。

- ・開始行 / 終了行

スクロール表示領域の垂直ドット範囲を指定します。開始行は8ドット刻みの値を指定し、8ドット刻み-1の値を指定します。

- ・開始桁 / 終了桁  
スクロール表示領域の水平ドット範囲を指定します。

#### メニュー属性設定

\$QM {ユニット} {アドレス} {レコード・サイズ} {開始行} {終了行} {開始桁} {終了桁} {実行スクリプト位置} {表示開始項目番号} {カーソル行} {ヘッダ属性} {ヘッダ高さ}

- ・ユニット  
メニュー制御スクリプトが格納されているユニット記号を指定します。
- ・アドレス (2バイト)  
メニュー制御スクリプトが格納されている先頭アドレスをリトル・エンディアンで指定します。
- ・レコード・サイズ  
1行の情報を何バイトで構成するか指定します。
- ・開始行 / 終了行  
メニュー表示領域の垂直ドット範囲を指定します。開始行は8ドット刻みの値を指定し、8ドット刻み-1の値を指定します。
- ・開始桁 / 終了桁  
メニュー表示領域の水平ドット範囲を指定します。
- ・実行スクリプト位置 (2バイト)  
メニュー制御スクリプト (書式の詳細は (4) 参照) の実行レコード番号をリトル・エンディアンで指定します。タイトル記述のレコード番号を指定しないと制御不能になります。
- ・表示開始項目番号  
メニュー選択項目が1画面に表示しきれない場合、どこから表示するか項目番号 (0から始まる連番) を指定します。
- ・カーソル行  
カーソルの位置を指定します。全角 / 半角の場合は0~2, 1/4角の場合は0~5の範囲で選びます。
- ・ヘッダ属性 / ヘッダ高さ  
ヘッダ (メニューのタイトル行) の文字フォントやピッチを指定します (内容は3.6.4 (3) の \$JFコマンドの属性と同じです)。またヘッダ領域の垂直ドット・サイズを指定します。

**(4) メニュー制御スクリプトの書式**

メニュー制御スクリプトは、メニューの1階層の情報をタイトル記述レコード（1つ）と複数の選択項目記述レコードから構成します。

レコード番号	← レコード・サイズで指定 →
0	トップ階層メニューのタイトル記述
1	トップ階層メニューの選択項目記述1（項目番号0）
2	トップ階層メニューの選択項目記述2（項目番号1）
:	:
n	トップ階層メニューの選択項目記述n（項目番号n）
n+1	サブ・メニュー1のタイトル記述
n+2	サブ・メニュー1の選択項目記述1（項目番号0）
n+3	サブ・メニュー1の選択項目記述2（項目番号1）
:	:

**タイトル記述レコードの構成**

## ・戻りレコード番号（1～2バイト目）

BSキーが押された場合に戻るべき上位階層のタイトル記述レコード番号をリトル・エンディアンで指定します。トップ階層のタイトル記述の場合は一般的には0（自分自身へ戻る）を指定します。

## ・選択項目数（3バイト目）

表示される選択項目数を指定します。格納する選択項目記述レコードはこれより多くても構いません。逆に少ない場合は表示異常などの現象が出ます。

## ・改行および戻り項目番号（4バイト目）

bit7-6：項目の1行の高さを次のように定めてカーソル制御します。

10：8ドット，00：16ドット，11：24ドット，01：32ドット

bit5-0：上位階層へ戻ったときのカーソル表示位置を項目番号で指定します。

## ・メニュー・タイトル（5バイト目～）

ASCIIコードで文字列を記述します。文字列の最後に¥n¥0（0AH 00H）を格納します。

ここで指定した文字列は、メニュー表示領域の上位行にQMコマンド・パラメータのヘッダ属性で指定したフォント、ピッチで表示されます。

**選択項目記述レコードの構成**

## ・実行ポインタ（1～2バイト目）

ENTキーが押された場合に実行するポインタ値をリトル・エンディアンで指定します。

実行ポインタをどのように使うかは4バイト目の実行タイプの内容に従います。

## ・選択項目の表示属性（3バイト目）

選択項目の文字列を表示する場合のフォントやピッチなどを指定します。内容は3.6.4 (3) の\$JFコマンドの属性と同じです

## ・実行タイプ（4バイト目）

ENTキーが押された場合にどのような処理を行うか指定します。

0の場合：下位メニューへ制御を移します。実行ポインタが下位メニューのタイトル記述のレコード番号を指すようにします。

1の場合：実行ポインタの内容をgMenu.select\_indexに格納します。他のユニットに対して動作を指示するために使用できます。例えば他のユニットがこの変数を参照し、自分が処理すべき番号帯であれば処理を行ってクリアするようにします。

2の場合：実行ポインタの内容を他のユニットへ通知します。パラメータ領域アドレス3で報告指定がある場合に、パラメータ領域アドレス4のユニットに対して実行ポインタの内容を通知します。通知先ユニットがホストへの応答ユニットであれば、ホストへ通知されることとなります。

・ 選択項目名（5バイト目～）

ASCIIコードまたはシフトJISコードで文字列を記述します。文字列の最後に¥n¥0(0AH 00H)を格納します。ここで指定した文字列は、3バイト目で指定した属性に応じたフォントやピッチで表示されます。

## (5) 操作例

### スクロール表示例

まず、データ領域にスクロール用のテキストを格納します。以下の例は、平家物語の冒頭部分を引用しています。1/4角表示の行と混在したときの動作を示すために、テスト用の英字行を挿入しています。

```
$QW'8000 11'A famous story in Japan'A 0
$QW'8020 42'祇園精舎の鐘の声'A 0
$QW'8040 52'諸行無常の響き有り'A 0
$QW'8060 42'沙羅双樹の花の色'A 0
$QW'8080 51'盛者必衰の理を表す'A 0
$QW'80A0 01'8dot-line characters'A 0
$QW'80C0 51'驕れる人も久しからず'A 0
$QW'80E0 52'唯春の夜の夢の如し'A 0
$QW'8100 51'猛き者も遂には亡びぬ'A 0
$QW'8120 51'偏に風の前の塵に同じ'A 0
$QW'8140 01'ABCDEFGHIJKLMNOPQRSTUVWXYZ'A 0
$QW'8160 11'ABCDEFGHIJKLMNOPQRSTUVWXYZ'A 0
$QW'8180 01'VWXYZabcdefghijklmnop'A 0
$QW'81A0 11'VWXYZabcdefghijklmnop'A 0
$QW'81C0 01'qrstuvwxyz0123456789.'A 0
$QW'81E0 11'qrstuvwxyz0123456789.'A 0
$QW'8200 01'!"#$%&'27'()-~¥-^@`[<>'A 0
$QW'8220 11'!"#$%&'27'()-~¥-^@`[<>アカク'A 0
$QW'8240 42'この行は非表示にします'A 0
```

次にローカル・キーを有効にしてスクロール属性を設定します。

```
ローカル・キー・オン $QW'0 4002 1
```

```
属性設定 $QAQ'0000 0000 20 08 37 04 77
```

キーによりスクロール表示します。

### メニュー表示例

まずデータ領域にメニュー・スクリプトを書き込みます。

```
$QW'8800 0000 07 00'Top Menu'A 0
$QW'8820 0A00 42 00'家族の健康管理'A 0
$QW'8840 0E00 42 00'家計簿、住居・車'A 0
$QW'8860 1E00 42 00'平家物語序文'A 0
$QW'8880 2000 42 00'演奏メロディ選曲'A 0
$QW'88A0 2100 51 00'ユーティリティ起動'A 0
$QW'88C0 2200 51 00'システム設定の変更'A 0
$QW'88E0 6900 42 02'電源オフ'A 0

$QW'8940 0000 02 00'Health Control'A 0
$QW'8960 5100 42 02'体重・体脂肪記録'A 0
$QW'8980 5200 42 02'飲酒量・値記録'A 0

$QW'89C0 0000 03 01'House Hold'A 0
$QW'89E0 1200 42 00'家計管理'A 0
$QW'8A00 1700 42 00'住居管理'A 0
$QW'8A20 1900 42 00'車管理'A 0

$QW'8A40 0E00 04 00'Budget'A 0
$QW'8A60 6100 42 02'日常の収支管理'A 0
$QW'8A80 6200 42 02'預貯金・有価証券'A 0
$QW'8AA0 6300 42 02'住宅ローン残高'A 0
$QW'8AC0 6400 42 02'カード払い'A 0

$QW'8AE0 0E00 01 01'House'A 0
$QW'8B00 6A00 42 02'掃除当番表'A 0

$QW'8B20 0E00 02 02'Car'A 0
$QW'8B40 7100 42 02'車検関係'A 0
$QW'8B60 7200 42 02'車両保険関係'A 0

$QW'8BC0 0000 01 02'Heike Story'A 0
$QW'8BE0 0300 42 02'祇園精舎の鐘の声'A 0

$QW'8C00 0000 00 03'Melody'A 0
$QW'8C20 0000 00 04'Utility'A 0
$QW'8C40 0000 05 85'System Setting'A 0
$QW'8C60 0800 11 02'Volume Setting'A 0
$QW'8C80 0900 11 02'Click Sound Setting'A 0
$QW'8CA0 0100 11 02'Melody Setting'A 0
$QW'8CC0 0200 11 02'LCD Brightness Setting'A 0
$QW'8CE0 0300 11 02'LCD auto off time'A 0
```

次にメニュー・キー制御を有効にします。

ホスト・マシンへメニュー・セレクト報告 \$QW'3 2 75

メニュー・キー・オン \$QW'0 0005 2

次の例はトップ階層からメニュー制御を開始します。

```
$QMQ'0008 0000 20 00 3F 02 7C 0 0 11 8
```

次の例は途中の階層からメニュー制御を開始します。

```
$QMQ'0008 1200 20 00 3F 02 7C 1 2 11 8
```

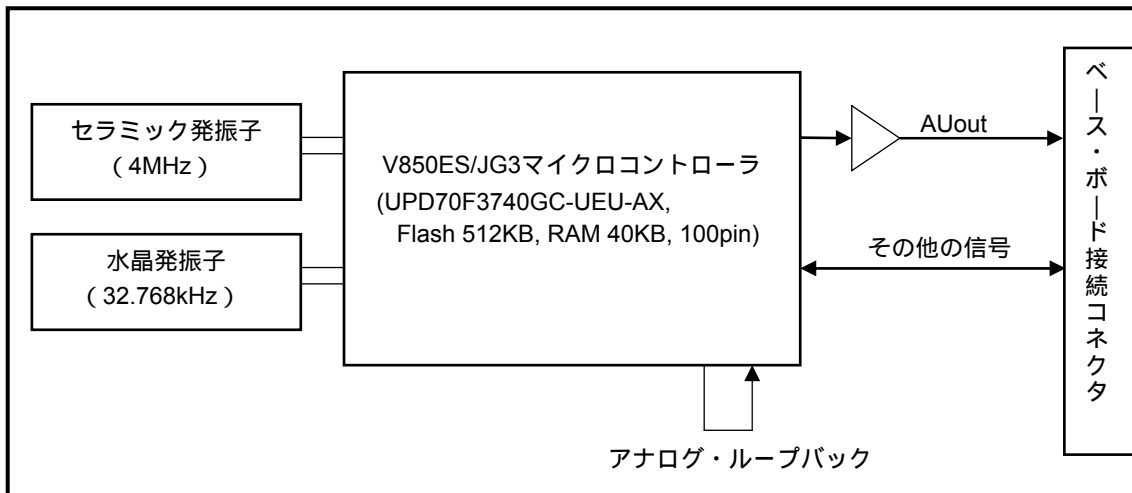
メニュー制御を終了します。 \$QM'0



## 第4章 ボード仕様

この章では、ボードの構成、諸元、コネクタの信号配置について説明します。

### 4.1 構成図



### 4.2 諸元

項目	内容
電源電圧	・ 3.0 V ~ 3.6 V (内蔵A/Dコンバータまたは内蔵D/Aコンバータ使用時) ・ 2.85 V ~ 3.6 V (内蔵A/Dコンバータおよび内蔵D/Aコンバータを使用しない時)
消費電流	・ 64mA (max.) (ポート駆動電流含まず) ・ 平均20mA以下 (SM05KV2デモンストレーション・プログラム動作時)
インタフェース	・ ベース・ボード接続コネクタ (140ピン・プラグ) AUout端子ドライブ能力: $\pm 10\text{mA}$ 可能 (1/2VDD出力時)。 その他端子のドライブ能力: UPD70F3740のドライブ能力に依存。
寸法	基板サイズ 69 x 55

### 4.3 コネクタ信号配置

コネクタ形状：ヒロセFX8C-140P-SV2（プラグ）

信号配置：次の表に示します。

表4 - 1 コネクタ信号配置一覧(1/2) (Bottom view)

MCU接続ポート	信号名	No.	No.	信号名	MCU接続ポート
RESET	QB_RESO	2	1	QB_SI	P30/TXDA0/SOB4
無接続	QB_CLKI	4	3	QB_SO	P31/RXDA0/INTP7/SIB4
P914/A14/INTP5/TIP51/TOP51	IRout	6	5	QB_SCK	無接続
P33/TIP01/TOP01	IRin	8	7	QB_HS	無接続
P03/INTP0/ADTRG	I2C_INT	10	9	QB_CLK	無接続
グラウンド	GND	12	11	GND	グラウンド
無接続	TP20	14	13	QB_FMD1	PDL5/AD5/FLMD1
無接続	TP19	16	15	QB_DATA	無接続
無接続	TP18	18	17	QB_FMD0	FLMD0
無接続	TP17	20	19	QB_RESI	10k 経由でRESETに接続
無接続	TP16	22	21	LCD_BL	P96/A6/TIP21/TOP21
無接続	TP15	24	23	LCD_PS	ブルアップ
無接続	TP14	26	25	LCD_RES	PCM2/HLDAK
無接続	TP13	28	27	LCD_CS	PCM3/HLDRQ
グラウンド	GND	30	29	GND	グラウンド
無接続	TP12	32	31	D7/SDA	PDL7/AD7
無接続	TP11	34	33	D6/SCK	PDL6/AD6
無接続	TP10	36	35	D5	PDL5/AD5/FLMD1
無接続	TP09	38	37	D4	PDL4/AD4
無接続	TP08	40	39	D3	PDL3/AD3
無接続	TP07	42	41	D2	PDL2/AD2
無接続	TP06	44	43	D1	PDL1/AD1
無接続	TP05	46	45	D0	PDL0/AD0
グラウンド	GND	48	47	GND	グラウンド
無接続	TP04	50	49	WR	PCT0/WR0
無接続	TP03	52	51	RD	PCT4/RD
無接続	TP02	54	53	LCD_ON	PCM0/WAIT
無接続	TP01	56	55	A00	P913/A13/INTP4
PDH0/A16	CSI_CS0	58	57	CSI_SI	P97/A7/SIB1/TIP20/TOP20
PDH1/A17	CSI_CS1	60	59	CSI_SO	P98/A8/SOB1
P35/TIP11/TOP11	BUZ	62	61	CSI_SCK	P99/A9/SCKB1
P40/SIB0/SDA01	SDA	64	63	SCL	P41/SOB0/SCL01
グラウンド	GND	66	65	GND	グラウンド
無接続	VCC	68	67	VCC	無接続
デバイス電源	VDD	70	69	VDD	デバイス電源

表4 - 1 コネクタ信号配置一覧(2/2) (Bottom view)

内容	信号名	No.	No.	信号名	内容
無接続	VBU	72	71	VBU	無接続
P71/ANI1	AUin	74	73	AUout	P11/ANO1 (バッファ経由)
グラウンド	AGND	76	75	AGND	グラウンド
P76/ANI6	AN04	78	77	AN00	P72/ANI2
P77/ANI7	AN05	80	79	AN01	P73/ANI3
P78/ANI8	AN06	82	81	AN02	P74/ANI4
P79/ANI9	AN07	84	83	AN03	P75/ANI5
グラウンド	GND	86	85	GND	グラウンド
P50/TIQ01/KR0/TOQ01/RTP00	$\overline{\text{KR0}}$	88	87	PIO0	P54/SOB2/KR4/RTP04
P51/TIQ02/KR1/TOQ02/RTP01	$\overline{\text{KR1}}$	90	89	PIO1	P55/SCKB2/KR5/RTP05
P52/TIQ03/KR2/TOQ03/RTP02	$\overline{\text{KR2}}$	92	91	PIO2	P42/SCKB0
P53/SIB2/KR3/TIQ00/TOQ00/RTP03	$\overline{\text{KR3}}$	94	93	PIO3	P915/A15/INTP6/TIP50/TOP50
PDL8/AD8	SEG0	96	95	PIO4	PCM1/CLKOUT
PDL9/AD9	SEG1	98	97	PIO5	P06/INTP3
PDL10/AD10	SEG2	100	99	PIO6	P710/ANI10
PDL11/AD11	SEG3	102	101	PIO7	P711/ANI11
グラウンド	GND	104	103	GND	グラウンド
PDL12/AD12	SEG4	106	105	PIO8	PDH3/A19
PDL13/AD13	SEG5	108	107	PIO9	PDH4/A20
PDL14/AD14	SEG6	110	109	PIO10	PDH5/A21
PDL15/AD15	SEG7	112	111	PIO11	無接続
PCT6/ASTB	$\overline{\text{DIG0}}$	114	113	PIO12	無接続
PCT1/WR1	$\overline{\text{DIG1}}$	116	115	PIO13	無接続
PDH2/A18	$\overline{\text{DIG2}}$	118	117	PIO14	無接続
P912/A12/SCKB3	$\overline{\text{DIG3}}$	120	119	PIO15	無接続
グラウンド	GND	122	121	GND	グラウンド
P911/A11/SOB3	$\overline{\text{DIG4}}$	124	123	TxD1	P90/A0/KR6/TXDA1/SDA02
P910/A10/SIB3	$\overline{\text{DIG5}}$	126	125	RxD1	P91/A1/KR7/RXDA1/SCL02
P95/A5/TIP30/TOP30	$\overline{\text{DIG6}}$	128	127	$\overline{\text{CTS1}}$	P04/INTP1
P94/A4/TIP31/TOP31	$\overline{\text{DIG7}}$	130	129	$\overline{\text{RTS1}}$	P32/ASCKA0/SCKB4/TIP00/TOP00
P93/A3/TIP40/TOP40	$\overline{\text{DIG8}}$	132	131	TxD2	P38/TXDA2/SDA00
P92/A2/TIP41/TOP41	$\overline{\text{DIG9}}$	134	133	RxD2	P39/RXDA2/SCL00
P37	$\overline{\text{DIG10}}$	136	135	$\overline{\text{CTS2}}$	P05/INTP2/DRST
P36	$\overline{\text{DIG11}}$	138	137	$\overline{\text{RTS2}}$	P34/TIP10/TOP10
グラウンド	GND	140	139	GND	グラウンド

## 4.4 MCU端子割り当て

本ボードに搭載のV850ES/JG3の端子割り当て（漢字表示デモンストレーション用）を次の表に示します。

表4 - 2 端子割り当て一覧(1/3)

端子名	ピン 番号	io	割当て内容
P02/NMI	17	o	PROG：予約（セルフ書き込み用）
P03/INTP0/ADTRG	18	i	I2C_INT：I2C割り込み入力
P04/INTP1	19	i	CTS1：UART1送信許可入力
P05/INTP2/DRST	20	i	CTS2：UART2送信許可入力
P06/INTP3	21	i	PIO5：ウエイクアップ入力
P10/ANO0	3	o	P70/ANI0ヘルプバック
P11/ANO1	4	o	AUout：拡張アナログ出力
P30/TXDA0/SOB4	25	o	QB_SI：ツール接続
P31/RXDA0/INTP7/SIB4	26	i	QB_SO：ツール接続
P32/ASCKA0/SCKB4/TIP00/TOP00	27	o	RTS1：UART1データ要求出力
P33/TIP01/TOP01	28	i	IRin：リモコン入力
P34/TIP10/TOP10	29	o	RTS2：UART2データ要求出力
P35/TIP11/TOP11	30	o	BUZ：ブザー出力
P36	31	o	DIG11：デジット信号
P37	32	o	DIG10：デジット信号
P38/TXDA2/SDA00	35	o	TxD2：UART2送信信号
P39/RXDA2/SCL00	36	i	RxD2：UART2受信信号
P40/SIB0/SDA01	22	io	SCL：I2Cデータ
P41/SOB0/SCL01	23	io	SDA：I2Cクロック
P42/SCKB0	24	o	PIO2：LEDブリンク出力
P50/TIQ01/KR0/TOQ01/RTP00	37	i	KR0：キー入力
P51/TIQ02/KR1/TOQ02/RTP01	38	i	KR1：キー入力
P52/TIQ03/KR2/TOQ03/RTP02	39	i	KR2：キー入力
P53/SIB2/KR3/TIQ00/TOQ00/RTP03	40	i	KR3：キー入力
P54/SOB2/KR4/RTP04	41	o	PIO0：LEDブリンク出力
P55/SCKB2/KR5/RTP05	42	o	PIO1：LEDブリンク出力
P70/ANI0	100	i	P10/ANO0からループバック
P71/ANI1	99	i	AUin：拡張アナログ入力
P72/ANI2	98	i	AN00：アナログ入力
P73/ANI3	97	i	AN01：アナログ入力
P74/ANI4	96	i	AN02：アナログ入力
P75/ANI5	95	i	AN03：アナログ入力
P76/ANI6	94	i	AN04：アナログ入力

表4 - 2 端子割り当て一覧(2/3)

端子名	ピン 番号	io	割当て内容
P77/ANI7	93	i	AN05 : アナログ入力
P78/ANI8	92	i	AN06 : アナログ入力
P79/ANI9	91	i	AN07 : アナログ入力
P710/ANI10	90	i	PIO6 : デモンストレーション・モード選択入力MS0
P711/ANI11	89	i	PIO7 : デモンストレーション・モード選択入力MS1
P90/A0/KR6/TXDA1/SDA02	43	o	TxD1 : UART1送信信号
P91/A1/KR7/RXDA1/SCL02	44	i	RxD1 : UART1受信信号
P92/A2/TIP41/TOP41	45	o	$\overline{\text{DIG9}}$ : デジット信号
P93/A3/TIP40/TOP40	46	o	$\overline{\text{DIG8}}$ : デジット信号
P94/A4/TIP31/TOP31	47	o	$\overline{\text{DIG7}}$ : デジット信号
P95/A5/TIP30/TOP30	48	o	$\overline{\text{DIG6}}$ : デジット信号
P96/A6/TIP21/TOP21	49	o	LCD_BL : LCDバックライト駆動信号
P97/A7/SIB1/TIP20/TOP20	50	i	CSI_SI : 3線シリアル入力
P98/A8/SOB1	51	o	CSI_SO : 3線シリアル出力
P99/A9/SCKB1	52	o	CSI_SCK : 3線シリアル・クロック
P910/A10/SIB3	53	o	$\overline{\text{DIG5}}$ : デジット信号
P911/A11/SOB3	54	o	$\overline{\text{DIG4}}$ : デジット信号
P912/A12/SCKB3	55	o	$\overline{\text{DIG3}}$ : デジット信号
P913/A13/INTP4	56	o	A00 : LCDアドレス
P914/A14/INTP5/TIP51/TOP51	57	o	IRout : リモコン出力
P915/A15/INTP6/TIP50/TOP50	58	o	PIO3 : LEDブリンク出力
PCM0/WAIT	61	o	LCD_ON : LCD電源ON信号
PCM1/CLKOUT	62	o	PIO4 : レディ出力
PCM2/HLDAK	63	o	$\overline{\text{LCD\_RES}}$ : LCDリセット
PCM3/HLDRQ	64	o	$\overline{\text{LCD\_CS}}$ : LCDチップ・セレクト
PCT0/WR0	65	o	$\overline{\text{WR}}$ : LCDデータ・ライト信号
PCT1/WR1	66	o	$\overline{\text{DIG1}}$ : デジット信号
PCT4/RD	67	o	$\overline{\text{RD}}$ : LCDデータ・リード信号
PCT6/ASTB	68	o	$\overline{\text{DIG0}}$ : デジット信号
PDH0/A16	87	o	$\overline{\text{CSI\_CS0}}$ : シリアルEEPROMチップ・セレクト
PDH1/A17	88	o	$\overline{\text{CSI\_CS1}}$ : シリアルD/Aコンバータ・ストローブ
PDH2/A18	59	o	$\overline{\text{DIG2}}$ : デジット信号

表4 - 2 端子割り当て一覧(3/3)

端子名	ピン 番号	io	割当て内容
PDH3/A19	60	i	PIO8 : デモンストレーション・モード選択入力MS2
PDH4/A20	6	i	PIO9 : デモンストレーション・モード選択入力MS3
PDH5/A21	7	i	PIO10 : デモンストレーション・モード選択入力MS4
PDL0/AD0	71	io	D0 : LCDデータ・バス
PDL1/AD1	72	io	D1 : LCDデータ・バス
PDL2/AD2	73	io	D2 : LCDデータ・バス
PDL3/AD3	74	io	D3 : LCDデータ・バス
PDL4/AD4	75	io	D4 : LCDデータ・バス
PDL5/AD5/FLMD1	76	io	D5 : LCDデータ・バス ( ツールにも接続 )
PDL6/AD6	77	io	D6 : LCDデータ・バス
PDL7/AD7	78	io	D7 : LCDデータ・バス
PDL8/AD8	79	o	SEG0 : セグメント駆動信号
PDL9/AD9	80	o	SEG1 : セグメント駆動信号
PDL10/AD10	81	o	SEG2 : セグメント駆動信号
PDL11/AD11	82	o	SEG3 : セグメント駆動信号
PDL12/AD12	83	o	SEG4 : セグメント駆動信号
PDL13/AD13	84	o	SEG5 : セグメント駆動信号
PDL14/AD14	85	o	SEG6 : セグメント駆動信号
PDL15/AD15	86	o	SEG7 : セグメント駆動信号
REGC	10	-	コンデンサ接続
RESET	14	-	QB_RESO : ツール接続
VDD	9	-	電源
VSS	11	-	グラウンド
X1	12	-	4MHz発振子
X2	13	-	4MHz発振子
XT1	15	-	32.768kHz発振子
XT2	16	-	32.768kHz発振子
AVREF0	1	-	アナログ電源
AVREF1	5	-	アナログ電源
AVSS	2	-	アナログ・グラウンド
EVDD	34	-	電源
EVDD	70	-	電源
EVSS	33	-	グラウンド
EVSS	69	-	グラウンド
FLMD0	8	-	QB_FMD0 : ツール接続

## 4.5 内蔵周辺機能の割り当て

本ボード搭載のV850ES/JG3の汎用内蔵周辺機能は、漢字表示デモンストレーション・プログラムでは次のように割り当てています。

### (1) タイマ

各タイマの用途は次のとおりです。

- 16ビット・タイマTMP0：赤外リモコン受信用パルス間隔測定
- 16ビット・タイマTMP1：ブザー（メロディ）用PWM波形生成
- 16ビット・タイマTMP2：LCDバックライトの輝度調整用PWM
- 16ビット・タイマTMP3：基本周期生成用インターバル・タイマ
- 16ビット・タイマTMP4：赤外リモコン送信用インターバル・タイマ
- 16ビット・タイマTMP5：赤外リモコン送信用PWM（33%デューティの38kHzキャリア生成）
- 16ビット・タイマTMQ0：使用していません。
- 16ビット・インターバル・タイマTMM0：使用していません。

### (2) シリアル・インタフェース

各インタフェースの用途は次のとおりです。

- UARTA0 / CSIB4：開発ツールUART接続用
- UARTA1 / I<sup>2</sup>C02：ベース・ボードのUART1用（ホスト・マシン接続用）
- UARTA2 / I<sup>2</sup>C00：ベース・ボードのUART2用
- CSIB0 / I<sup>2</sup>C01：ベース・ボードのI<sup>2</sup>C用
- CSIB1：ベース・ボードの3線シリアルI/O用
- CSIB2：使用していません。
- CSIB3：使用していません。

## 第5章 ボード設計情報

この章ではボードの回路情報，基板情報，実装情報として公開しているファイルの概要について説明します。実際のファイルは下記URLのボード設計情報からダウンロードしてください。

<http://www.necel.com/micro/ja/designsupports/board/index.html>

### 5.1 回路情報

#### 5.1.1 回路ファイル構成

ダウンロード・ファイル (SM05K2\_SCH.zip) には，以下のファイルが含まれています。

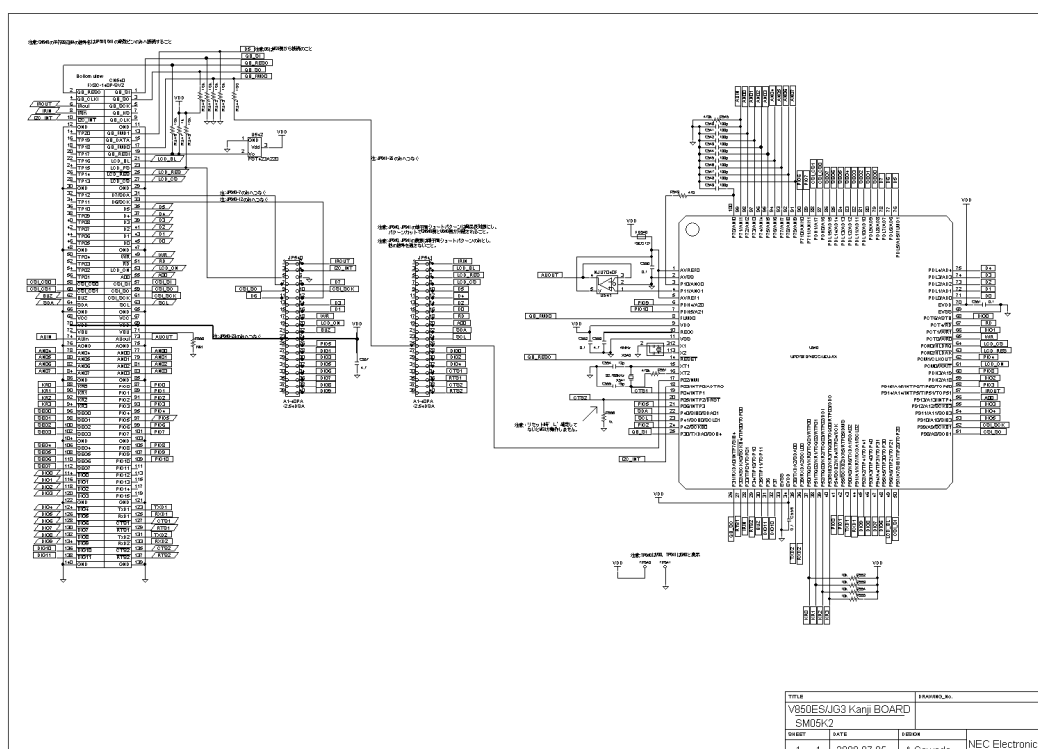
ファイル名	内容 (フォーマット)
SM05K2.pdf	回路図 (PDF)
SM05K2.csv	部品表 (csv)
SM05K2.net	ネット・リスト (CADVANCE形式テキスト) <sup>注1</sup>
SM05K2.dcd	回路CADデータ (D2 CAD) <sup>注2</sup>

注1. 他の形式が必要な場合はD2 CADをインストールして生成してください。

注2. D2 CADは下記からダウンロード可能です。

<http://www.yansoft.com/d2cad/>

#### 5.1.2 回路図 (縮小版)





## 5.2 基板情報

### 5.2.1 基板ファイル構成

ダウンロード・ファイル (SM05K2\_PWB.zip) には、以下のファイルが含まれています。

ファイル名	内容 (フォーマット)
SM05K2_dim.pdf	寸法図 (PDF)
SM05K2_layer.pdf	基板図 (PDF)
SM05K2_gbr.zip	ガーバー・データ <sup>注1</sup>
SM05K2.pcpa	ビューワ・データ (CADVANCE) <sup>注2</sup>

注1. 各層のファイル名称は、同梱のSM05K2.lstに記載されています。

注2. “ Eye-PCB (ビューワ版) ” でパターンを見たり、寸法を測ることができます。このソフトウェアは下記URLから探してダウンロードできます。

<http://www.ydc.co.jp/>

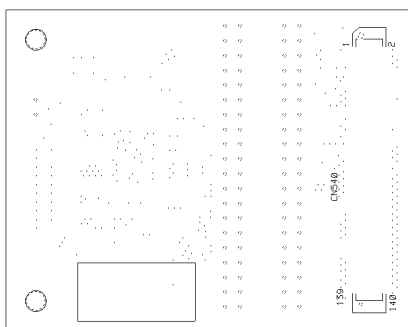
### 5.2.2 基板仕様例

- ・材質 : FR-4
- ・構成 : 4層スルー・ホール
- ・板厚 : 1.6 mm
- ・レジスト : 両面
- ・シルク : 両面
- ・仕上げ : 水溶性耐熱フラックス

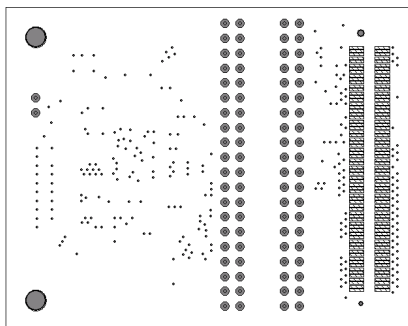
### 5.2.3 基板図面 (縮小版)

以下の順に掲載します。

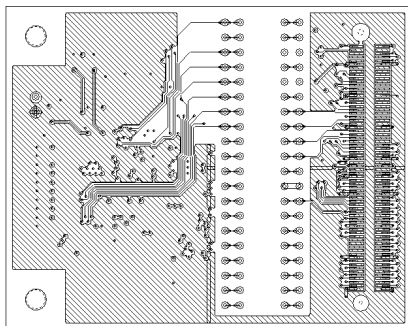
- ・部品面シルク
- ・部品面レジスト
- ・部品面パターン
- ・L2パターン
- ・L3パターン
- ・半田面パターン
- ・半田面レジスト
- ・半田面シルク



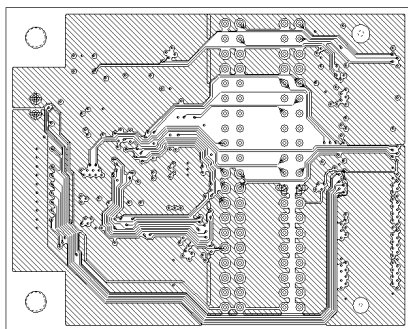
部品面シルク



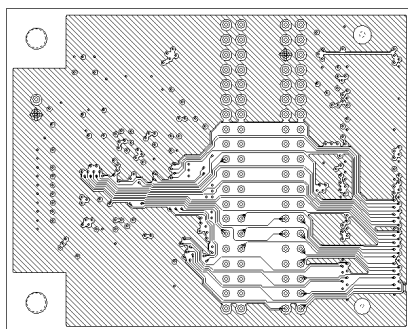
部品面レジスト



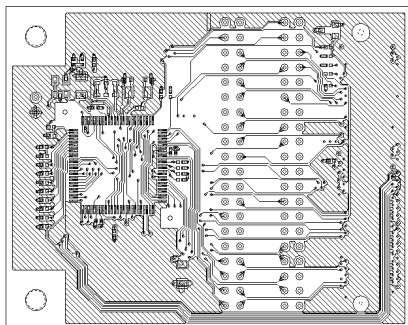
部品面パターン



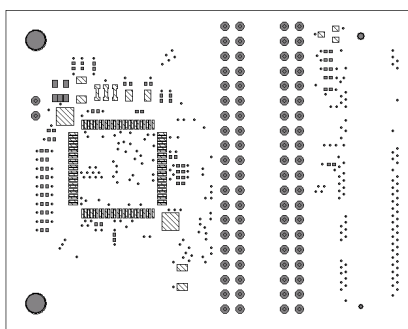
L2 パターン



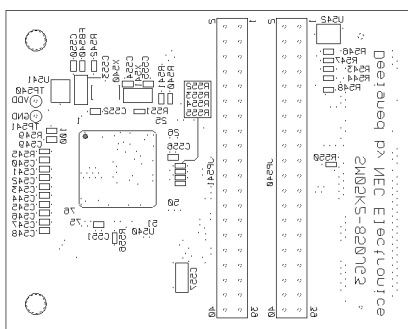
L3 パターン



バーマの面田半



半田面ミスマス



半田面ミスマス

## 5.3 実装情報

### 5.3.1 実装ファイル構成

ダウンロード・ファイル (SM05K2\_ASY.zip) には、以下のファイルが含まれています。

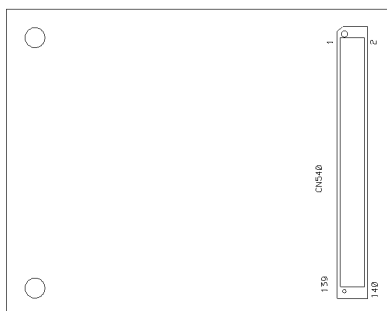
ファイル名	内容 (フォーマット)
SM05K2_mm.pdf	メタル・マスク図 (PDF) <sup>注1</sup>
SM05K2_mm.zip	メタル・マスクのガーバー・データ <sup>注2</sup>
SM05K2_mnt.pdf	実装図 (PDF)
SM05K2_mnt.lst	部品座標リスト (テキスト)

注1. 外形図が必要な場合は基板外形図を参照してください。

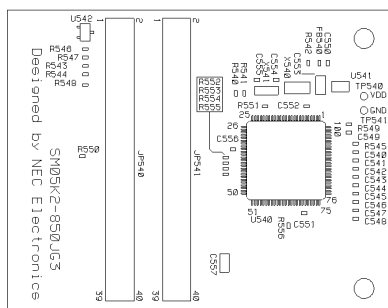
注2. 各層のファイル名称は、同梱のSM05K2.lstに記載されています。

ビューワ・データは基板のビューワ・データ・ファイルに含まれています。

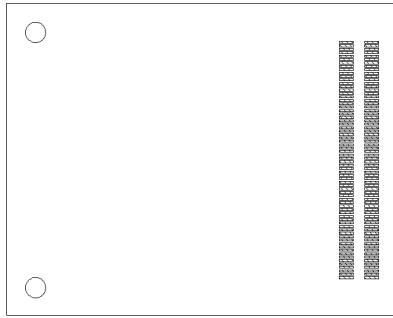
### 5.3.2 実装図およびメタル・マスク図 (縮小版)



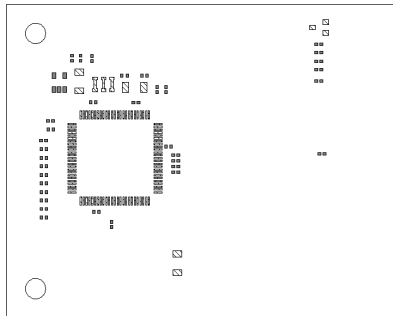
部品面シルク



半田面シルク



部品面メタル



半田箇所

## 第6章 プログラム構成とビルド方法

本章ではデモンストレーション・プログラムの構成とビルド方法について説明します。各プログラムはサンプル・ソースとしては必ずしも適していないため、内容の説明はありません。一部の機能については、サンプル・プログラムのアプリケーション・ノートに解説があります（本書の“はじめに”に記載の関連資料参照）。

### 6.1 ビルド済プログラム・ファイル

デモンストレーション・プログラムを動作させるだけであれば、ダウンロード・ファイルのTASK\_KVディレクトリにある次のビルド済ファイルをプログラマで書き込んで使用します。

SM05KV2.hex

### 6.2 ファイル構成

ダウンロード・ファイルには、ソース・プログラムをはじめ各種ファイルがいくつかのディレクトリに分けて格納してあります。構成の概要は次のとおりです。

SM05KV2\_R3xx.zip（3xxはリリース番号です）

— SM05KV2.TXT	リリース・ノートです。本書の説明と異なる部分についての記載があります。
— SM05KV2.prw	ワークスペース・ファイルです（有償ツールではこれを開きます）。
— SM05KV2.prj	プロジェクト・ファイルです（有償ツール用）。
— SM05KV2_F.prw	ワークスペース・ファイルです（フリー・ツールではこれを開きます）。
— SM05KV2_F.prj	プロジェクト・ファイルです（フリー・ツール用）。
— <b>FONT_LIB</b>	フォント・データおよびアクセス・ライブラリが格納されているディレクトリです。
— <b>KERNEL</b>	簡易OSや初期化プログラムなどが格納されているディレクトリです。
— <b>TASK_K</b>	基本的なデモ用タスクなどのファイルが格納されているディレクトリです。
— <b>TASK_KV</b>	SM05KV2固有のファイルが格納されているディレクトリです。

R3.05までの主要な変更履歴について表6-1に示します。R3.06以降の変更点はリリース・ノートを参照してください。

表6 - 1 主要な変更履歴

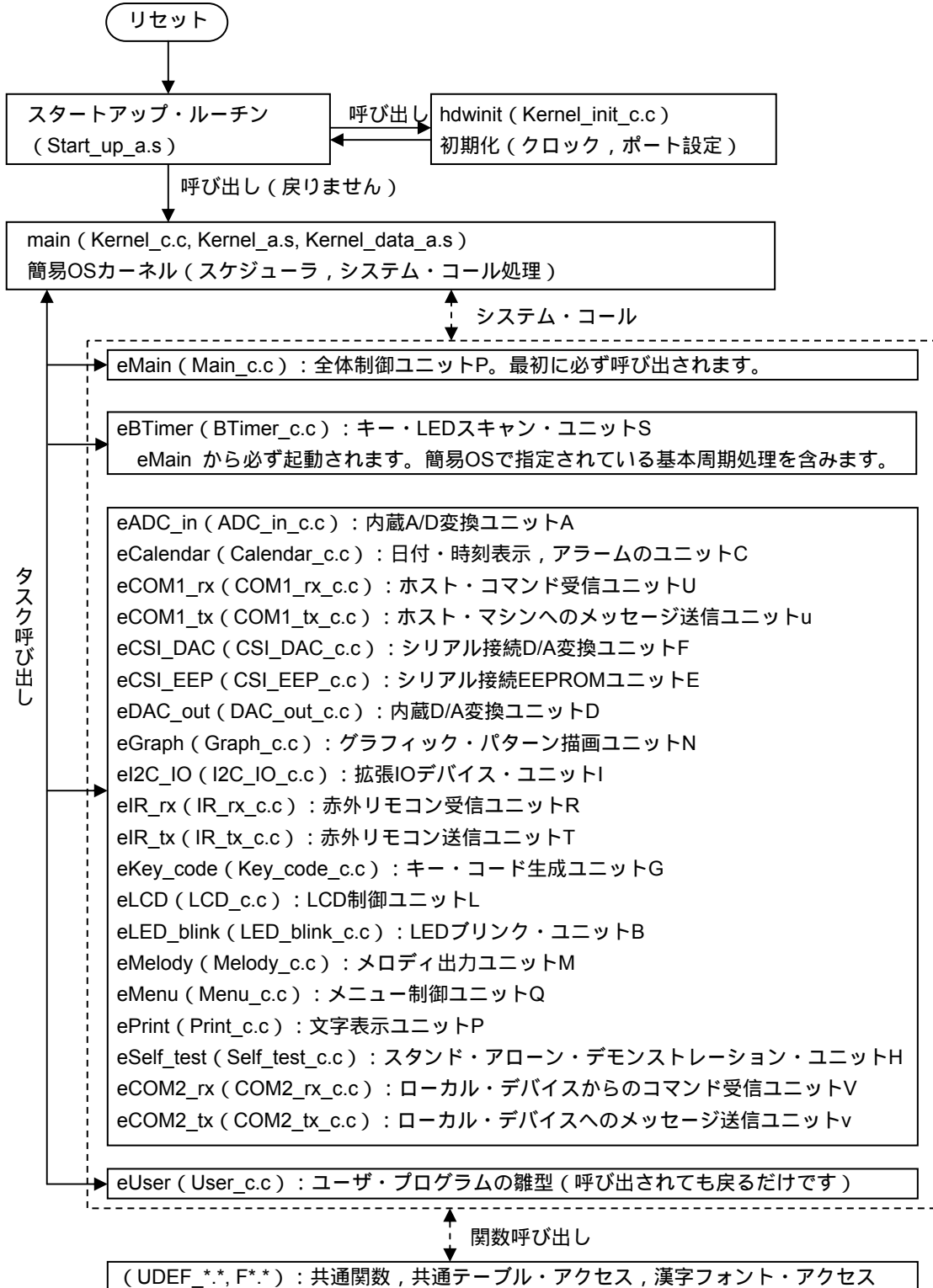
変更項目	R2.11	R3.01	R3.05
5x7ドット・フォント (FA207LVH)	C言語配列データとして提供。	フォント・ユーティリティ対応。	
7x14ドット・フォント (FH114LVH)	C言語配列データとして提供。	フォント・ユーティリティ対応。	
14x14ドット・フォント (FL314LVH)	フォント・データをライブラリとして提供。	フォント・ユーティリティ対応。	第3水準文字セットに加えて、フリー・ツールで直接ビルド可能な278文字セットのファイルを添付。
その他の漢字フォント (FL024LVH, NL8xxLVH)	無し	24ドット・フリー・フォントのサンプルを組み込み。	12, 16, 20, 24, 32ドットのFontAvenueサンプル・フォントを組み込み。
KERNEL (簡易OS) 関係	コード空間128KB限定	コード空間2MB対応を追加。 スタック・サイズと実行時間の測定機能を追加。	CPU動作クロック16MHz / 32MHz切り替えを追加。
LCD制御 (LCD_XXX)	画面サイズより広い表示メモリが必要。	画面サイズ分の表示メモリのみ必要。	各種LCDサイズとコントローラに対応したLCDライブラリを使用。
文字表示 (Print_c)	-	フォント・ユーティリティ対応。 複数漢字フォント対応。 描画領域制限機能追加。	12, 16, 20, 24, 32ドットのFontAvenueサンプル・フォントの表示に対応。
メロディ出力 (Melody_c)	-	constエリアのスキ립ト・データを演奏可能なように変更。	
スタンド・アローン操作 (Self_test)	-	各種改良。	漢字表示例にFontAvenueサンプル・フォントを使用した例文を追加。
プロジェクト・ファイル	無し	有り	(ディレクトリ構成の変更に対応)
ROM配置	フリー・ツール使用時の14ドット・フォント配置は、1E000Hから。	フリー・ツール使用時の14ドット・フォント配置は、20000Hから。	



### 6.3 プログラム実行の依存関係

リセット以降、どのように全体のプログラムが動作するかを次に示します。

図6 - 1 プログラム実行の依存関係



## 6.4 ビルド方法

使用するツールとしては、SP850（有償ソフトウェア・パッケージ）を推奨します。フリー・ダウンロード版のCA850では漢字フォントの組み込み作業が別途必要になります。以下、有償版について説明します。フリー・ツールで漢字フォントを組み込む場合は、「フォント・ユーティリティ ユーザーズ・マニュアル（U19527）」を参照してください。

ツールの推奨リビジョンは、CA850：3.20以上です。

### （1）添付ワークスペース・ファイルによるビルド

添付ワークスペース・ファイルSM05KV2.prw（フリー・ツールの場合はSM05KV2\_F.prw）を開くと、簡単にビルドができます。ただし、次の手順でツール選択が必要です。うまく動作しない場合は、（2）以降の手順によりビルドします。

リビジョン選択を要求されたら、いったんOKを押した後、キャンセルします。

“プロジェクト プロジェクト設定 ツールバージョン設定 詳細設定”によりツール選択します。

### （2）ソース・ファイルの指定

#### 必須指定ファイル

FONT\_LIB : FA207LVH.s, FH114LVH.s, FL314LVH\_8836.s（フリー・ツール時は  
FL314LVH\_8836\_F.sまたはFL314LVH\_278.sを選択）,  
KERNEL : Kernel\_a.s, Kernel\_c.c, Kernel\_data\_a.s, Kernel\_init\_c.c,  
UDEF\_data\_a.s, UDEF\_func\_c.c, Tools\_c.c（測定機能使用時）  
TASK\_K : BTimer\_c.c, COM1\_rx\_c.c, COM1\_tx\_c.c, Main\_c.c

#### 選択可能ファイル

FONT\_LIB : NL012LVH.s, NL016LVH.s, NL020LVH.s, NL024LVH.s,  
NL032LVH.s（Self\_test\_c.c選択時は以上5つを全て指定して下さい）,  
FL024LVH\_28.s（Self\_test\_c.cを選択しない時は指定して下さい）  
TASK\_K : ADC\_in\_c.c, Calendar\_c.c, DAC\_out\_c.c, Graph\_c.c, I2C\_IO\_c.c, IR\_rx\_c.c,  
IR\_tx\_c.c, Key\_code\_c.c, LED\_blink\_c.c, Melody\_c.c, Menu\_c.c, Print\_c.c,  
User\_c.c, CSI\_DAC\_c.c, CSI\_EEP\_c.c, LCD\_c.c, LCD\_lib\_c.c  
TASK\_KV : Self\_test\_c.c

以上のソースは必要に応じて選択可能ですが、選択しない場合は以下のファイルの編集が必要です。

- ・UDEF\_a.h  
タスク名定義において、組み込まないタスク定義をコメント・アウトするか削除します。
- ・UDEF\_func\_c.h  
組み込みレベル定義において、組み込まないタスクの定義を0にするか行を削除します。

### (3) プロジェクト関連ファイルの指定

スタートアップ・ファイル ... Start\_up\_a.s  
リンク・ディレティブ・ファイルの指定 ... SM05KV2.dir  
ライブラリの指定 ... FDJ850ES\_26.a (フォント・アクセス・ライブラリ)

### (4) コンパイラ・オプション

- ・プリプロセッサ 定義マクロ：通常はKOS\_CODE\_TYPE=0を設定します。コード空間拡張や測定機能を使用する場合は、アプリケーション・ノート簡易OS編 (U19216JJ2) の表1 - 2を参照して設定します。  
なお、CPUクロックはデフォルトでは16MHzです。32MHzにする場合は、APP\_CODE\_TYPE=3を追加で指定します。
- ・プリプロセッサ インクルード・ファイル・パス：\*\_c.hが存在するディレクトリを追加します。
- ・出力コード sconstセクションのデータ位置：“配置する”にチェックします。
- ・出力コード レジスタモード：“26レジスタ”を選択します。22レジスタあるいは32レジスタを選択した場合はビルド時にワーニングがでますが、動作は可能です。
- ・アセンブラ：“「アセンブラオプション」を使用”にチェックをつけます。

### (5) アセンブラ・オプション

- ・インクルード・ファイル・パス：\*\_a.hが存在するディレクトリを追加します。
- ・シンボル定義：コンパイラ・オプションの定義マクロと同じ指定をします。

### (6) リンカ・オプション

- ・必要に応じて出力ファイル名などを設定します。

## 6.5 起動時の注意

- ・デバッガを使用する場合は、V3.41以上を使用してください。
- ・デバッガで起動する場合、環境設定でMain OSC 4MHz，Multiply rate 4に設定してください。ただし、コンパイラ/アセンブラ・オプションでAPP\_CODE\_TYPE=3を指定した場合は、Multiply rate 8に設定してください。

## 6.6 プログラム改造方法

端子変更，発振周波数変更，内蔵周辺機能の割り当て変更の方法について以下説明します。不要タスクの切り離し方法は，「6.4(2) 選択可能ファイル」を参照してください。文字フォントの実装数や配置の変更方法は「フォント・ユーティリティ ユーザーズ・マニュアル (U19527)」を参照してください。

### 6.6.1 端子変更

プログラムで使用している端子名は，UDEF\_port\_c.hの中で定義した値を使用しています。これらのファイルで定義しているポート・レジスタ名を変更すれば，プログラム・コード部を変更することなく端子変更ができます。SM05KVおよびSM06KWの定義例を表6-2に示します。

なお，ポート・モード・レジスタは接頭辞zmを付けて定義しています。また，端子を削除したい場合，使用していない割り込みフラグをダミー・ポートとして割り当てると，ソース・コードを修正しなくてもコンパイル・エラーを防ぐことができます。ただし，無駄なオブジェクトが生成されるため最終的には削除が必要です。

表6-2 ポート定義一覧 (1/2)

定義名 (意味)	SM05KV2	SM06KW2
zD (LCDデータ・バス)	PDLL	
zWR (LCDデータ・ライト信号)	PCT.0	
zRD (LCDデータ・リード信号)	PCT.4	
zLCD_DIR (LCD入出力切り替え)	-	P4.2
zA00 (LCDアドレス)	P9H.5	
zLCD_RES (LCDリセット)	PCM.2	
zLCD_CS (LCDチップ・セレクト)	PCM.3	
zLCD_CS2 (LCDチップ・セレクト2)	-	P5.5
zLCD_ON (LCD電源ON信号)	PCM.0	
zLCD_BL (LCDバックライト駆動信号)	P9L.6	
zRxD1 (UART1受信信号)	P9L.1	
zTxD1 (UART1送信信号)	P9L.0	
zRTS1 (UART1データ要求出力)	P3L.2	P9H.7
zCTS1 (UART1送信許可入力)	P0.4	PDH.5
zRxD2 (UART2受信信号)	P3H.1	
zTxD2 (UART2送信信号)	P3H.0	
zRTS2 (UART2データ要求出力)	P3L.4	
zCTS2 (UART2送信許可入力)	P0.5	
zCSI_CS0 (シリアルEEPROMチップ・セレクト)	PDH.0	
zCSI_CS1 (シリアルD/Aコンバータ・ストローブ)	PDH.1	
zCSI_CS2 (拡張チップ・セレクト)	-	P5.4
zCSI_SO (3線シリアル出力)	P9H.0	
zCSI_SI (3線シリアル入力)	P9L.7	
zCSI_SCK (3線シリアル・クロック)	P9H.1	

表6 - 2 ポート定義一覧 (2/2)

定義名 (意味)	SM05KV2	SM06KW2
zKR (キー入力)	P5	
zSCL (I <sup>2</sup> Cクロック)	P4.1	
zSDA (I <sup>2</sup> Cデータ)	P4.0	
zI2C_INT (I <sup>2</sup> C割り込み入力)	P0.3	
zSEG (セグメント駆動信号)	PDLH	
zDIG11 (デジット信号11)	P3L.6	
zDIG10 (デジット信号10)	P3L.7	
zDIG9 (デジット信号9)	P9L.2	
zDIG8 (デジット信号8)	P9L.3	
zDIG7 (デジット信号7)	P9L.4	
zDIG6 (デジット信号6)	P9L.5	
zDIG5 (デジット信号5)	P9H.2	
zDIG4 (デジット信号4)	P9H.3	
zDIG3 (デジット信号3)	P9H.4	
zDIG2 (デジット信号2)	PDH.2	
zDIG1 (デジット信号1)	PCT.1	
zDIG0 (デジット信号0)	PCT.6	
zBUZ (ブザー出力)	P3L.5	
zIRin (リモコン入力)	P3L.3	
zIRout (リモコン出力)	P9H.6	
zPIO0 (LEDブリンク出力0)	P5.4	-
zPIO1 (LEDブリンク出力1)	P5.5	-
zPIO2 (LEDブリンク出力2)	P4.2	-
zPIO3 (LEDブリンク出力3)	P9H.7	-
zMS0 (モード選択入力0)	P7H.2	固定値 " 0 "
zMS1 (モード選択入力1)	P7H.3	
zMS2 (モード選択入力2)	PDH.3	固定値 " 1 "
zMS3 (モード選択入力3)	PDH.4	
zREADY (レディ出力)	PCM.1	DUMMY
zPEN_INQ (ペンダウン検出)	-	PDH.3
zCSI_SPI (拡張シリアル入力)	-	P7H.2
zCSI_SPO (拡張シリアル出力)	-	P0.4
zCSI_SPC (拡張シリアル・クロック)	-	P3L.2
zPCL_850 (コーデック用クロック出力)	-	PCM.1
zFRAME (コーデック用フレーム信号入力)	-	P0.6

### 6.6.2 発振周波数変更

X1クロック発振周波数の変更により影響を受けるモジュールは表6 - 3のとおりです。

すでに16MHz / 32MHz切り替えのための記述が入っているので、その定義内容を変更するだけで済む場合もあります。通信ボー・レートや音階周波数のようにテーブルで記述されている場合は再計算が必要になることがあります。

### 6.6.3 内蔵周辺機能の割り当て

タイマやシリアル・インタフェースなどの内蔵周辺機能の割り当て状況は表6 - 3のとおりです。

SM05KVやSM06KWでは、内蔵周辺機能の名称を直接使用しているため、変更の場合はソース・コード修正が必要です。

表6 - 3 各ソース・プログラムの変更対象箇所 (1/2)

ソース・プログラム名	使用端子	X1周波数依存性	内蔵周辺機能
Kernel_init_c.c (初期化)	全端子	クロック基本設定	-
Kernel_a.s (カーネルのアセンブラ部)		基本周期10msを生成するためのBTimer_c.c内定義値 (測定機能オプションを有効にした場合のみ必要)	タイマ TMP3 (測定機能オプションを有効にした場合のみ必要)
ADC_in_c.c (内蔵A/D変換)	-	内蔵A/Dコンバータの変換時間に関する設定	内蔵A/Dコンバータ
ADPCM_c.c (ADPCM再生)	zmcPCL_850, zmPCL_850, zmcFRAME, zFRAME, zmCSI_SCK, zmCSI_SI, zCSI_SO, zmCSI_SO	クロック出力周波数の設定およびその周波数に対するオーディオ・コーデックの各種パラメータ・テーブル。 シリアル・インタフェース IIC01 の速度設定 (I2C_IO_c.cが組み込まれて無い場合のみ必要)。	クロック出力回路。 シリアル・インタフェース CSIB1, IIC01。
BTimer_c.c (キー・LEDスキャン)	zSEG, zmSEG, zDIGn, zmDIGn, zmKRn, zKR	基本周期10msを生成するためのカウント値。	タイマ TMP3
Calendar_c.c (日付・時刻表示)	zmPIO3, zPIO3	-	時計タイマ
COM1_rx_c.c (ホスト・コマンド受信)	zRTS1	通信ボー・レート生成値	シリアル・インタフェース UARTA1
COM1_tx_c.c (ホスト・マシンへの応答)	zCTS1	-(COM1_rx_c.cで設定)	シリアル・インタフェース UARTA1

表6-3 各ソース・プログラムの変更対象箇所(2/2)

ソース・プログラム名	使用端子	X1周波数依存性	内蔵周辺機能
CSI_DAC_c.c(シリアル接続D/A変換)	zCSI_CS1, zmCSI_CS1。 zCSI_SPC, zCSI_SPO (SM06B2使用時)。	シリアル・インタフェースCSIB1の速度設定。 ポート制御シリアル タイミング設定 (SM06B2使用時)。	シリアル・インタフェースCSIB1
CSI_EEP_c.c(シリアル接続EEPROM)	zCSI_CS0, zmCSI_CS0。 zCSI_SPC, zCSI_SPO, zCSI_SPI (SM06B2使用時)。	シリアル・インタフェースCSIB1の速度設定。 ポート制御シリアル タイミング設定 (SM06B2使用時)。	シリアル・インタフェースCSIB1
DACout_c.c(内蔵D/A出力)	-	-	内蔵D/Aコンバータ。
I2C_IO_c.c(拡張IOデバイス)	-	シリアル・インタフェースIIC01の速度設定	シリアル・インタフェースIIC01
IR_rx_c.c(赤外線リモコン受信)	-	パルス幅測定用クロックの設定。	タイマ TMP0
IR_tx_c.c(赤外線リモコン送信)	-	キャリア周波数およびビット基本周期の設定	タイマ TMP4(ビット基本周期), TMP5(キャリア生成)
LCD_c.c(LCD制御)	zLCD_ON	バックライトPWMの周期設定。	タイマ TMP2(バックライトPWM用)
LCD_lib_c.c(LCD制御ライブラリ)	zLCD_DIR, zmD, zD, zA00, zRD, zRD2, zWR, zLCD_RES	LCD制御タイミング	-
LED_blink_c.c(LEDブリンク)	zPIO0, zmPIO0, zPIO1, zmPIO1, zPIO2, zmPIO2, zPIO3, zmPIO3	-	-
Main_c.c(全体制御)	zMS0, zMS2, zMS3, zREADY。 zDIG11, zmDIG11, zKR0(スタンバイ解除用)。	スタンバイ解除用10msのカウント値。	タイマ TMP3 (スタンバイ時のみ使用)
Melody_c.c(メロディ出力)	-	音階周波数テーブル。	タイマ TMP1
Self_test_c.c(スタンド・アローン・デモ制御)	zMS3	LCD_c.cのバックライト定義依存(輝度テーブル)。	-
Tools_c.c(タスクの測定機能)	zMS1	-	-
TSC_c.c(タッチスクリーン制御)	zCSI_CS2, zCSI_SPC, zCSI_SPO, zCSI_SPI, zPEN_INQ	ポート制御シリアル タイミング設定。	-

## 付録A 改版履歴

### A.1 2版, 3版で改訂された主な箇所

箇所	内容
<b>はじめに</b>	
p.7	関連資料を変更
<b>第1章 概 説</b>	
p.10	1.1 (2) デモンストレーション・ソフトウェアの内容を変更
p.11	1.2 (3) 操作モード指定に測定動作オプションを追加
<b>第3章 ホスト・コマンド操作</b>	
p.25	3.4.5 (2) データ領域の構成とサイズを変更
p.26	3.4.6 文字表示のフォントに関する参照資料を変更
p.26	3.4.6 (1) パラメータ領域を追加
p.26	3.4.6 (3) 文字表示コマンドのパラメータ内容を変更
p.27	3.4.6 (4) 操作例を追加・変更
p.35	3.4.10 (1) パラメータ領域の一部パラメータ範囲を変更
p.35	3.4.10 (3) 日付と時刻の設定の曜日指定に関する注を追加
p.38	3.4.11 (1) パラメータ領域にLCD表示開始桁の指定を追加
p.48	3.4.18 (3) 図形パターン描画の一部パラメータを変更
p.48	3.4.18 (4) 操作例を変更
p.49	3.4.19 (3) スクロール属性設定にパラメータを追加
p.50	3.4.19 (3) メニュー属性設定にパラメータを追加
p.51~p.52	3.4.19 (4) メニュー制御スクリプトの書式の一部を変更
p.52~p.54	3.4.19 (5) 操作例の一部のコマンドにパラメータを追加
<b>第6章 プログラム構成とビルド方法</b>	
p.69	6.2 ファイル構成を変更
p.70	表6-1 変更ファイル一覧を追加(2版) 主要な変更履歴に変更(3版)
p.72~p.73	6.4 ビルド方法にワークスペース・ファイルを使用したビルド方法を追加。またオプション指定を変更
p.73	6.5 起動時の注意を変更
p.74~p.77	6.6 プログラム改造方法を追加
<b>付録A 改版履歴</b>	
p.78	付録A 改版履歴を追加



〔メモ〕

## 【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：(044)435-5111

## 【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.necel.co.jp/>

## 【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか、NECエレクトロニクスの販売特約店へお申し付けください。

---

## —— お問い合わせ先 ——

### 【営業関係、デバイスの技術関係お問い合わせ先】

半導体ホットライン

(電話：午前 9:00～12:00, 午後 1:00～5:00)

電 話 : (044)435-9494

E-mail : [info@necel.com](mailto:info@necel.com)

---

### 【マイコン開発ツールの技術関係お問い合わせ先】

開発ツールサポートセンター

E-mail : [toolsupport-micom@ml.necel.com](mailto:toolsupport-micom@ml.necel.com)

### 【漢字表示プログラム／ボードの技術関係お問い合わせ先】

E-mail : [kanji-demo@ml.necel.com](mailto:kanji-demo@ml.necel.com)