

# 32×32 RGB LEDマトリクスVer.2

型番：KP-RGB3232E

KP-RGB3232Eは横32ドット、縦32ドットのLEDマトリクスパネルです。  
各ドットは独立したRGBの三色LEDから構成されます。  
点灯パターンはデータ6線＋クロック線による同期式転送により制御を行います。

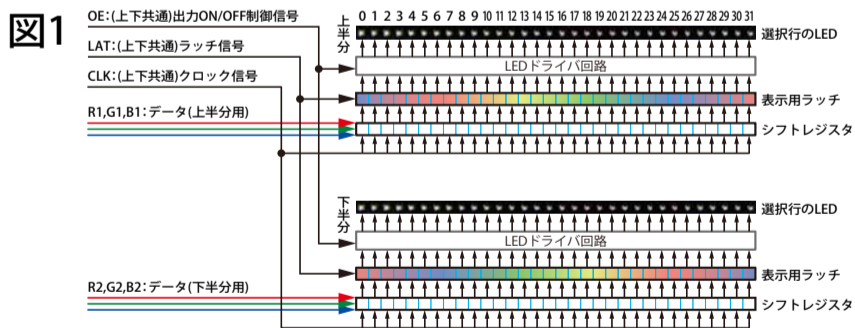
**注意** 旧製品「KP-3232D」とは互換性がありません

## ■概要

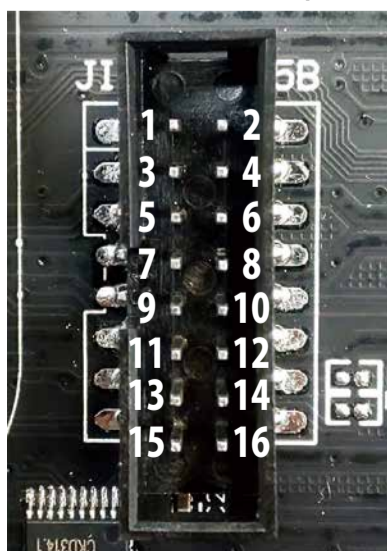
KP-RGB3232Eは横32ドット、縦32ドットのLEDマトリクスパネルです。  
各ドットは独立したRGBの三色LEDから構成されます。  
点灯パターンはデータ6線＋クロック線による同期式転送により制御を行います。

## ■動作

本パネルの論理的な構成は上半分と下半分に分かれています。  
横方向の32ドットを1行として、上半分16行、下半分16行で合計32行になります。  
一度に点灯するのは、上半分16行の中から選択された1行と、下半分16行の中から選択された1行のみです。  
上下それぞれの行に対して、横32ドット分のデータをセットした後、それを表示させる行を16行の中から選択します。行の指定は上下で共通となっているため、上下ともに同じ位置の行が選択されます。  
パネル全面を表示するには、16行の選択を順次切り替えながら点灯させるダイナミック点灯が要求されます。



## ■入力インターフェース (JIN-HUB75Bと記載)

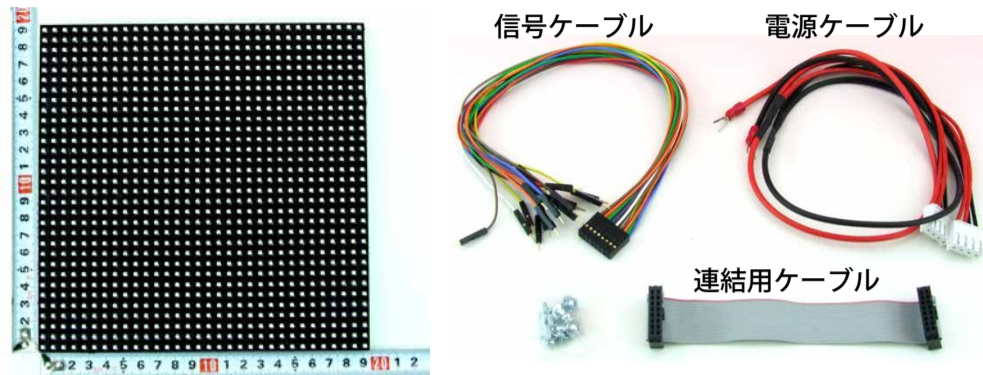


信号名	ピン番号	信号名
R1	1	上半分用のシリアル赤データ
G1	2	上半分用のシリアル緑データ
B1	3	上半分用のシリアル青データ
GND	4	
R2	5	下半分用のシリアル赤データ
G2	6	下半分用のシリアル緑データ
B2	7	下半分用のシリアル青データ
GND	8	
A	9	上下共通の行選択線A (最下位ビット)
B	10	上下共通の行選択線B
C	11	上下共通の行選択線C
D	12	上下共通の行選択線D (最下位ビット)
CLK	13	上下共通のシリアルデータ取込みクロック
LAT	14	上下共通のデータラッチ
OE	15	全ドットの点灯制御 L=ON、H=OFF、H→Lで取込み
GND	16	

制御用のマイコンとの接続は  
付属の信号ケーブルをご使用ください。

## ■仕様

- ・サイズ：約(W)192×(H)192×(t)13mm
- ・ドット数：横32ドット×縦32ドット×3色LED
- ・スキャンレート：1/16
- ・電源：5V 2.5A(3A以上推奨)
- ・付属品  
電源ケーブル、信号ケーブル (QI 8P×2列-16Pバラ線)、  
連結用ケーブル (両端BOX 8P×2列)、固定用ネジ8本



## ■電源

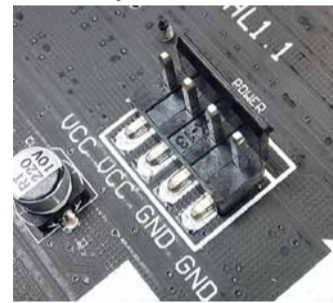
電源電圧：DC 5V  
消費電流：約 2.5A (全ドット点灯、実測値)

付属の電源ケーブルを使用して、POWERコネクタにDC5V電源を供給してください。  
赤色が[+]、黒色が[-]です。先端が2個ありますが、もう一方は制御装置などに供給する際に使用可能な予備のコネクタです。  
消費電流が大きい事に加えて変動も大きいため、スイッチング式電源の使用を推奨します。  
全点灯時が消費電流の最大値になりますが、個体差を考慮して1枚あたり3Aを目安に、余裕をもって電源を選定してください。

**本パネルには、過電圧保護や逆極性接続保護等の保護回路は一切ありません。  
電源の電圧間違い、極性接続の間違いには十分注意してください。**

## ■適合コネクタ：コネクタ番号未記入 (Powerと記載)

信号名	ピン番号	信号名
VCC	1	+5V
VCC	2	+5V
GND	3	GND
GND	4	GND



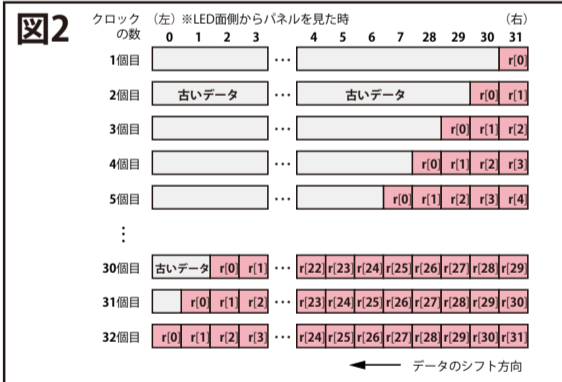
## ■表示の制御手順

(はじめに)  
以下の制御手順については、当社が独自に行った実験をもとに作成したものです。  
結果として発生する現象に関して、当社にて保証するものではない事をご了承の上、ご利用ください。

### (1) 発光データの入力

- ・1行32ドット分の発光データは、上半分用の各色(R1,G1,B1)ピンと、下半分用の各色(R2,G2,B2)ピンの合計6本のデータ線で入力します。
- ・上記に加えて、制御クロック(CLK)ピン
- ・ラッチ制御(LATピン、出力制御(OE)ピンの3本を使用します。これらは上半分と下半分で共通となります。
- ・行のデータの入りは、表示面から見て左端のドットから順に入力します。
- ・各ドットの各色はHレベルで点灯、Lレベルで消灯になります。
- ・入力した各点灯データ(R1,G1,B1,R2,G2,B2)はクロック信号(CLK)に同期して、パネル上のシフトレジスタに取り込まれます。
- ・以上の方法で発光させるパターンに合わせたデータを、横1行32ドット全体のシフトレジスタに保持させます。

以下はデータ転送のイメージです。例として上半分の赤色発光データを、左のドットから順に r[0] ~ r[31] とします。



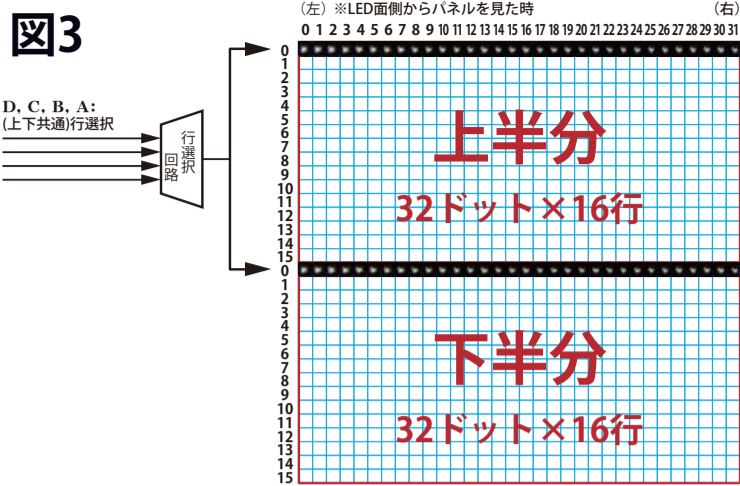
前回に送信した1行分のデータ(古いデータ)が「ところてん式」に順番に押し出され、新しい発光データに置き換わっていきます。32個のクロック信号(CLK)送信で、1行分の新しい発光データ r[0] ~ r[31] がシフトレジスタ内に揃うことになります。  
上記は上半分の赤色(R1)のみで説明しましたが、実際には上半分の3色+下半分の3色で合計6線を一度に指定しなければなりません。

### (2) 発光データの取り込み

- ・ラッチ制御ピンLATをHレベルにすると、シフトレジスタのデータが表示用ラッチに反映されます。
- ・LATをLレベルにしている場合、シフトレジスタと表示用ラッチは切り離されます。
- ・パネルには、上下半分それぞれ表示用ラッチ内のデータが、A,B,C,Dによって選択された行の位置に表示されます。

全てのデータが揃った時点で上下共通のLATピンをHにしてからLに戻すことで、シフトレジスタに溜まった上下・RGB各32個のデータがLED点灯用ドライバに送られて、実際のLEDを点灯させます。  
なお、LATをHにしない限り、データをシフトレジスタに送っている途中のデータがLED表示に影響することはありません。LATがHのままデータを送ると、データのシフト過程がそのまま表示に現れてしまいますので、色の滲みなどの現象として知覚される可能性があります。

### (3) 発光を制御する行の選択



- A,B,C,Dピンの各信号の組み合わせで対象となる行を選択します。
- 行選択方法はDCBAを4ビットのバイナリ(二進数)値とみなし、0~15の行番号で指定します。  
(Dが最上位ビット、その下にC、Bと続き、Aが最下位ビットです)
- パネル上下の各16行のうち、一番上の行から順に0,1,2番と進み、一番下が15番です。
- 行選択の4本のピンは、Dピンの変化のタイミングで取り込まれるようになっています。  
[行番号0~7]の場合は、DピンをHからLに変化させた時点で確定します。  
まずDピンをHにして、2μS程度時間をあけてからLを出力してください。  
[行番号8~15]の場合は、DピンをLからHに変化させた時点で確定します。  
まずDピンをLにして、2μS程度時間をあけてからHを出力してください。  
以上の操作を行った時点のA,B,Cの値が、選択した行番号となります。

### (4) 出力制御

- 出力制御OEピンをHレベルの間は、パネル全体の表示が消灯します。
- OEピンがHからLレベルになる時、データラッチに入っている1行分の点灯データが更にLEDドライバ回路へ伝送されます。
- OEピンがLレベルの間、上記A,B,C,Dピンで選択された行番号の位置にある各LEDがデータにあわせて点灯します。  
手順(3)で行選択を切り替える前にOEをHにして表示を一時的に消去し、A,B,C,Dピンで選択行を変更後、再度OEをLにすることでデータ内容が選択位置に表示されるようになります。

※旧仕様製品(KP-3232D)と異なり、当製品ではOEピンの制御が必須となります。

### (5) パネル全体へのダイナミック表示

このパネルは、全行のドットを同時点灯する回路構成ではなく、上下の各1行分がその瞬間に指定した通りに点灯するのみです。点灯するLEDは、行選択ピンA,B,C,Dで指定した上半分・下半分の各1行だけです。それ以外の各15行は消灯状態となります。

パネル全面に表示させるには、

- 0番の行を指定→0番用の行データを送る→しばらく待つ
- 1番の行を指定→1番用の行データを送る→しばらく待つ
- ...

- 15番の行を指定→15番用の行データを送る→しばらく待つ

といった手順を高速で繰り返す処理(いわゆる1/16デューティのダイナミック表示)を行わなければなりません。

このため、データを送る作業を止める事ができず、制御用のコントローラ(CPU)は相当な負荷となります。

通常は、秒あたり60回の全面描画ができればチラつきを感じることなく表示できますが、余裕を見て秒100回の更新として計算します。

切替行は16行で一巡するので、×16で1600行/秒のデータ更新が必要です。

さらに各行は32ドットあるため、×32で51200ドット/秒のデータを送り出す能力が必要となります。

## ■参考情報

### (1) PWMによる階調表現

上記で説明した手順は、単純にRGB各色を点灯または消灯の2値で表現する場合の方法です。実際には各ドットの点灯と消灯の時間比率をさらに細かく指定することで、各色の明るさを中間的に調整することができます。(PWM制御)

各色の明るさを何段階で表現するかに応じて、1画面の表示に要する処理量が增大していきます。

一般的にフルカラーと呼ばれる各色256段階の調整を行うには、コントローラ側のデータ送り出しを256倍高速に行う必要があります。これを概算すると13.1MHzの送り出しレートとなりますが、FPGAなどのハードロジックが要求される速度です。

一般的なマイクロコントローラでソフト制御する場合は、より分解能を低くした16階調程度が現実的です。

本説明書のサンプルコードでは各色10段階の例を示しています。

### (2) 出力インタフェース

本パネルには、シフトレジスタの最終段から押し出される各色データと制御信号が引き出された出力コネクタを装備しています。(JOUT-HUB75Bと記載)

この出力コネクタと別の本製品の入力コネクタ(JIN-HUB75Bと記載)を連結することで、コントローラから入力する信号線はそのまま、全体を横64ドット×32行のパネルとして扱えるようになります。連結には付属の「連結用ケーブル」を使用してください。

横ドットが増えることにより、1行あたりの送出データが増加しますので、描画処理はより忙しくなります。

※電源は必ずすべてのパネルに接続してください。単一の5V電源から給電する場合は、必ず接続枚数分の電流を供給できることをご確認ください。

※旧仕様製品(KP-3232D)を使用することはできません。

## ■ご注意

本パネルは、ダイナミック表示を前提とした制御が必要となります。

A,B,C,Dピンによる行選択を高速で切り替える処理が必須となりますが、この制御が停止すると、特定行のLEDのみに電流が流れ負荷が集中することでLEDの劣化が生じる可能性があります。

コントローラ(CPU)の制御ソフトの試作時など、動作が安定するまでは「可変電源」を使用して5Vより低い電圧で試験される事をお勧めします。

## ■サンプルコード

下記コードは、マイクロコントローラ「mbed Nucleo F401RE」向けのサンプルプログラムです。mbed用プログラムは一般的なC言語系(コンパイラはC++)で、オンライン開発環境

(<https://developer.mbed.org/>)で作成しています。

サンプルコードのリストは当社ホームページからダウンロードできます。

<http://prod.kyohritsu.com/KP-RGB3232E.html>

```
#include "mbed.h"

DigitalOut R1 (D2); // シリアルデータ 上半分用・赤
DigitalOut G1 (D3); // シリアルデータ 上半分用・緑
DigitalOut B1 (D4); // シリアルデータ 上半分用・青
DigitalOut R2 (D5); // シリアルデータ 下半分用・赤
DigitalOut G2 (D6); // シリアルデータ 下半分用・緑
DigitalOut B2 (D7); // シリアルデータ 下半分用・青
DigitalOut CLK (D8); // シリアルクロック
DigitalOut LAT (D10); // データラッチ
DigitalOut OE (D9, 1); // 出力制御 (Hレベルで初期化)
BusOut sel_ABC (A0, A1, A2); // 行選択A, B, C
DigitalOut sel_D (A3); // 行選択D

Ticker ticker; // 繰返しタイマー割込み

uint8_t F_BUF_R[32][32]; // 表示パターン格納エリア 赤用 0~10
uint8_t F_BUF_G[32][32]; // 表示パターン格納エリア 緑用 0~10
uint8_t F_BUF_B[32][32]; // 表示パターン格納エリア 青用 0~10

void timerIsr() { // 100uS 周期で繰返し実行
    static uint8_t line=0; // 行選択カウンタ 0~15
    static uint8_t PWM=0; // PWM カウンタ 0~10
    uint8_t x;

    // 1行分の各色データを送信
    for (x=0; x<32; x++) {
        R1=F_BUF_R[line][x]>PWM; // 各色データをピンに出力
        R2=F_BUF_R[line+16][x]>PWM; // 現在のPWMカウンタよりドットの明るさを
        G1=F_BUF_G[line][x]>PWM; // 明るい場合に1を、
        G2=F_BUF_G[line+16][x]>PWM; // 暗い場合に0を出力する
        B1=F_BUF_B[line][x]>PWM;
        B2=F_BUF_B[line+16][x]>PWM;
        CLK=1; // クロックを1つ送ってデータをシフト
        CLK=0;
    }
    OE=1; // 出力を一時的にOFFにする
    LAT=1; // シフトレジスタ内容をラッチへ取り込み
    LAT=0;
    sel_ABC=line&7; // まずA, B, Cピンをセット
    sel_D=line<8 ? 1 : 0; // Dピンに一旦逆の値を出力してから
    wait_us(2); // 少し待機した後
    sel_D=line<8 ? 0 : 1; // 本来の値を出力するとABCDが取り込まれる
    OE=0; // 出力をONにする
    line++; // 選択行を1つ次の行にする
    if (line==16) { // (16まで達したら0に戻し、
        line=0;
        PWM++; if (PWM>=10) PWM=0; // PWMカウンタを1増加 最大まで達したら0に戻す)
    }
}

int main(void) {
    uint8_t x, y;
    // LEDパネル描画処理を100uS間隔で繰返し割り込み開始
    ticker.attach_us(&timerIsr, 100);
    wait(0.2);
    // ↓以下、表示データを変化される処理
    while(1) { // ランダムなx, y点を、ランダムなRGB色で描画
        x=32.0*rand()/RAND_MAX;
        y=32.0*rand()/RAND_MAX;
        F_BUF_R[y][x]=10.0*rand()/RAND_MAX;
        F_BUF_B[y][x]=10.0*rand()/RAND_MAX;
        F_BUF_G[y][x]=10.0*rand()/RAND_MAX;
        wait_ms(2);
    }
}
```

## ■製品の補償について

- 本製品およびそれらを構成するパーツ類は、改良・性能向上のため予告なく外觀変更・仕様変更・非純正品品使用等があることをあらかじめご了承ください。
- 本製品は組立キットまたは半完成品です。製作作業中の安全確保のため説明書をよくお読みになり、正しい工具の使用・手順を守ってください。
- 本製品は最終製品ではなく、そこに組み込まれる構成部品のひとつです。そのため、本製品単独時の性能は説明書記載の仕様を保証しますが、他の構成部品との組み合わせ、組み立て方、相性によっては総合的、最終的な性能・品質がお客様の期待に添わない場合があることをあらかじめご了承ください。
- 本製品は機器への組込み他、工業製品としての使用を想定した設計は行っていません。また、本製品に起因する直接、間接の損害につきましては当社修理サポートの規定範囲を超えての補償には応じられません。
- 本製品はハード基板のみの供給・補償となります。プログラムなどソフト面のご質問はお答えできません。

Electronic Devices, Parts, Kits & Robots  
**KYOHITSU**  
共立電子産業株式会社 共立プロダクツ事業所  
〒556-0004 大阪市浪速区日本橋西2-5-1  
TEL:06-6644-4447 FAX:06-6644-4448

【“共立プロダクツ”ブランドとは】  
当ブランドの製品はユーザーニーズを捉えた製品をリーズナブルな価格でのご提供を目指しています。  
そのためユーザーサポートはメールに限定しておりますことをご理解、ご了承ください。  
✉Email:wonderkit@keic.jp  
Twitterやblogで応用例や製品紹介を更新中です。ぜひご覧になって下さい。

共立プロダクツ