

บริษัท ซิลาร์รี่ จำกัด 1108/41 ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ 10110
TEL. 0-2712-2850-2 FAX. 0-2381-1447 www.silaresearch.com

SILA

DP-74F

v1.0

4 Inchx1, 7-Segment Display (Serial)

DP-74F เป็นบอร์ด 7-SEGMENT ขนาดความสูงของตัวเลข 4" สีแดง 1 หลัก มองเห็นได้ชัดเจนในระยะ 10 เมตร สามารถต่ออนุกรมกันได้หลายๆ หลักโดยวิธีต่อพ่วงกันไปตามที่ต้องการ แต่ต้องคำนึงถึงกระแสไฟที่เลี้ยง Display ซึ่งแต่ละหลักต้องการกระแสประมาณ 120mA/หลัก การต่อใช้งานทำได้สะดวกโดยต่อผ่าน 3BPORT มาตรฐานซิล่า ได้ทันที ใช้บิต MCU เพียงแค่ 3 บิต ควบคุมขา CLOCK, ขา DATA และขา STROBE แต่สามารถต่อพ่วงกันได้ไม่จำกัดหลัก ใช้หลักการแบบส่งข้อมูลแบบอนุกรมให้ครบทุกหลักแล้วส่งสัญญาณ STROBE เพื่อ LATCH ข้อมูลทุกหลักค้างไว้พร้อมกัน ทำให้เขียนโปรแกรมง่ายและไม่ต้องเสียเวลาในการสแกนแต่อย่างใด

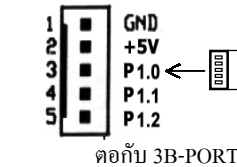
คุณสมบัติทั่วไป

1. ตัวเลขสูง 4 นิ้ว กว้าง 2.4 นิ้ว แต่ละ Segment กว้าง 0.4 นิ้ว
2. ใช้แหล่งจ่ายไฟแยกให้กับ 7 Segment 12 VDC กินกระแส 120 mA/หลัก
3. พอร์ตควบคุมแบบ 3BPORT มาตรฐานซิล่า สามารถต่อพ่วงกันได้ไม่จำกัดหลัก โดยควบคุมจำนวนหลักที่ โปรแกรม
4. ลักษณะบอร์ดออกแบบให้ติดตั้งง่าย สามารถแยกส่วนแต่ละหลักทำให้ง่ายในการตรวจสอบ
5. บอร์ด SIZE 3.5" X 5.3"

การต่อใช้งาน

1. แต่ละบอร์ดจะมีสายติดมากับบอร์ดพร้อมใช้งาน ในการต่อพ่วงกันหลายหลักก็สามารถทำได้ทันที ดังรูป

มาตรฐาน 3B-PORT

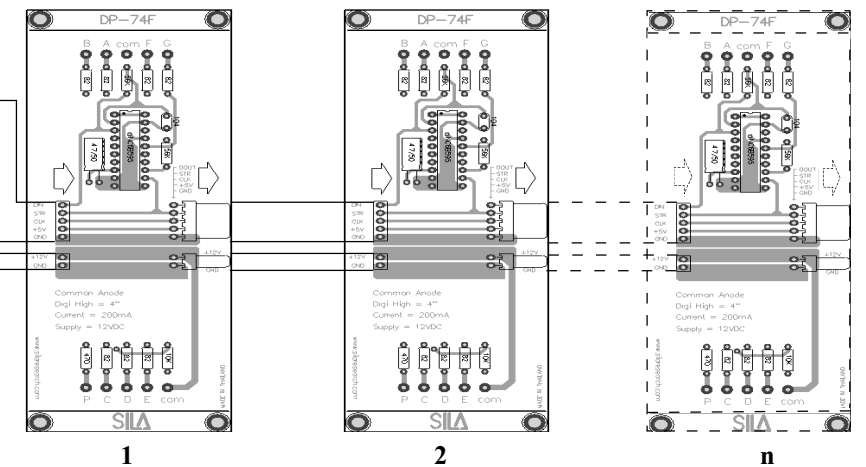


มาตรฐานขั้วต่อ VDC



ต่อกับไฟเลี้ยง 12 VDC

รูปแสดงการต่อพ่วงหลายหลักเป็นภาพด้านหลังบอร์ดให้สังเกตลูกศรบนบอร์ดแสดงทิศทางของข้อมูล

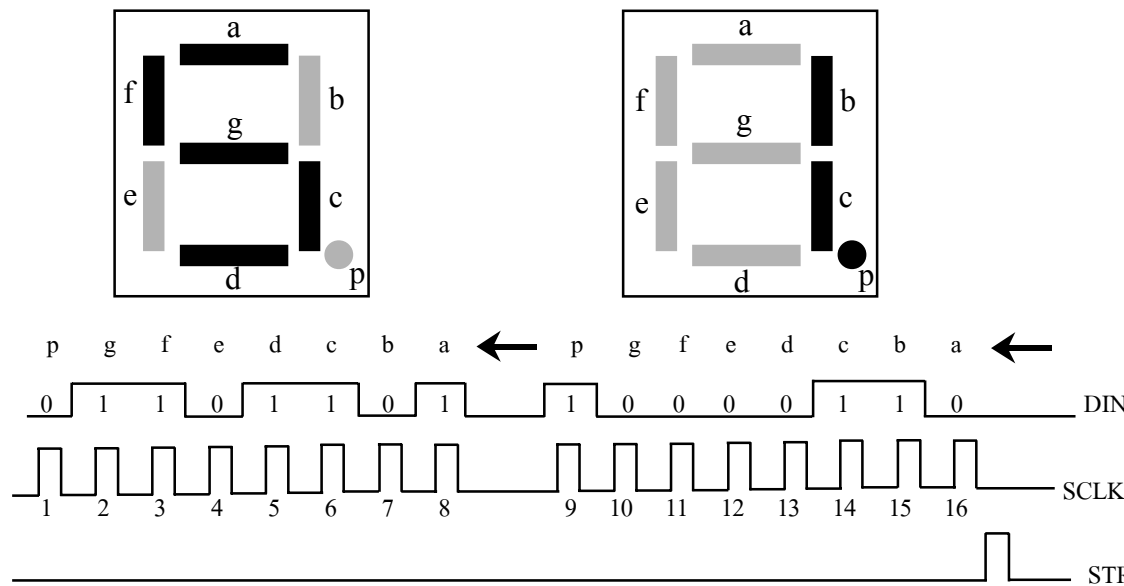


2. วิธีคำนวณกระแสไฟที่ใช้ กรณีที่ต่อ 4 หลัก จะต้องสำรองกระแสไฟประมาณ 400 mA. เพราะการกินกระแสไฟจะขึ้นกับจำนวน Segment ที่ติดในแต่ละหลักนั่นเอง และที่ขั้ว 3BPORT จะต้องมิไฟ +5V ส่งไปเลี้ยงชีพแต่ละหลักด้วย

3. ขาคควบคุมของบอร์ดมี3ขาซึ่งจัดไว้ให้เข้ากับมาตรฐาน 3BPORT ของศิลาแต่ไม่จำเป็นที่จะต้องต่อกับบิทที่กำหนดก็ได้ เพียงใช้ขาของ MCU บิทใดก็ได้ 3 บิท เพื่อควบคุมขา SCLK คือ Serial Clock ขา DIN คือขาข้อมูลเข้าบอร์ด และขา STR คือขา STROBEข้อมูลเพื่อ LATCH แสดงผล ส่วนอีกฝั่งของบอร์ดจะเป็นขา DOUT คือขาข้อมูลออกใช้ต่อไปยังหลักถัดไปไม่สามารถต่อสลับกับขา DIN ได้

ความสัมพันธ์ของขาสัญญาณ

1. แสดงให้เห็นกรณีต่อพ่วง 2 หลัก เพื่อแสดงเลข " 51. "



2. ในการส่งข้อมูลเพื่อให้แสดงเลข " 51. " จะต้องกำหนด Segment คิดค่างโดยที่ติดให้ =1 และดับให้ =0 กรณีแสดง 2 หลัก ก็จะต้องส่งข้อมูลของหลักสิบไปก่อนตามลำดับ CLOCK ที่แสดงลูกที่ 1 ส่งข้อมูล "0" ลูกที่ 2 ส่งข้อมูล "1" ไปเรื่อยๆ ใน 1 หลักจะใช้ CLOCK ทั้งหมด 8 ลูก ดังนั้นข้อมูล 2 หลักจึงใช้ CLOCK ทั้งหมด 16 ลูก ดังภาพ เมื่อส่งครบ 16 ลูกแล้วให้ส่งพัลซ์ที่ขา STR 1 ลูก ข้อมูล ก็จะถูกแสดงทันที

3. สำหรับสินค้า DP-74f นี้มีโปรแกรมตัวอย่างเป็นภาษา C สามารถดาวน์โหลดได้ที่ www.silaresearch.com ชื่อไฟล์คือ xdp72f.c ซึ่งเป็นโปรแกรม Show นาฬิกา 4 หลัก ผู้ใช้สามารถนำเฉพาะส่วนโปรแกรม Show ไปดัดแปลงเพื่อใช้งานอื่นต่อไปซึ่งเพียงแต่กำหนดจำนวนหลักที่ต้องการแสดงที่บรรทัด # define num 4 แล้วในmainโปรแกรมก็เพียงนำตัวเลขที่จะแสดงมาใส่ไว้ที่ DISBUF [4] เพื่อแสดงหลักที่ 4 หรือนำข้อมูลใส่ที่ DISBUF [1] เพื่อแสดงหลักแรก แล้ว call show ก็จะแสดงตัวเลขนั้นบน Display ทันที

โปรแกรมตัวอย่าง xdp72f.c

```
/* Filename      xdf72f.C
Description      Test Board DP-72F Display 2.3" clock 4 digit to
                  3B-PORT
Hardware         Start C51 + DP-72F x 4
Compiler         Keil V5.0
Engineer         Pojchara B.
Company          Sila Research Co.,Ltd. */
#include <reg52.h>
#include <absacc.h>
#include <ctype.h>
#include <intrins.h>
#include <math.h>
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define num 4 //Number DIGIT .....
/***** I/O PORT *****/
sbit CLK  = P3^2;
sbit STR  = P3^3;
sbit DATA = P3^5;
unsigned char data DISBUF[num+1]; // Display Buffer
unsigned char dl,sec;
/***** BASIC FUNCTION *****/
void dmsec (unsigned int count){//Delay mSec
unsigned char i;
while (count) {
for (i=1;i<=226;i++);
count--;
}
}
unsigned char code SEGTAB[10]={0x3F,0x06,0x5B,0x4f,0x66,
                                0x6d,0x7d,0x07,0x7f,0x6f};

//Table 7SEGMENT 0-9
void show (void){ //sub show data from DISBUF to DP-72F
unsigned char i,j,k; //input : DISBUF[x] , num = number digit
for(k=num;k>0;k--){
i = SEGTAB[(DISBUF[k]&0x0f)];
if((DISBUF[k]&0x80)==0x80){i|=0x80;DISBUF[k]=0x7f;}
for(j=0;j<=7;j++){
DATA = i & 0x80;
i = i << 1;
CLK = 1;
_nop_();
CLK = 0;
}
STR = 1;
_nop_();
STR = 0;
}
}
void second (void) interrupt 1{ //interrupt second
TR0 = 0;
dl++;
dmsec(2);
if(dl==27){
sec++;
dl=0;
TR0 = 1;
}
}
}

}
/***** Main Program *****/
void main(void){
unsigned char a;
bit btf;
dmsec(100);
TMOD = 0x11;
TH0 = 0x00;
TL0 = 0x01;

EA = 1;
ET0 = 1;
TR0 = 1; //start timer0
DATA = 0;
CLK = 0;
STR = 0;

btf = 1;
for(a=0;a<num;a++){
DISBUF[a]=0; //default clock 12:00 //Set Hour

DISBUF[4]=0x01;
DISBUF[3]=0x03; //Set Minute
DISBUF[2]=0x00;
DISBUF[1]=0x07;
while(1){ //loop count clock
if(sec==60){
if(btf){
DISBUF[1]++; //increase minute
}
btf=0;
sec=0;
}
}
}
else btf=1;
if(DISBUF[1]>=0x0a){
DISBUF[1]=0;
DISBUF[2]++;
}
if(DISBUF[2]>=0x06){
DISBUF[2]=0;
DISBUF[3]++;
}
if(DISBUF[3]>=0x0a){
DISBUF[3]=0;
DISBUF[4]++;
}
if(DISBUF[4]==0x02){
if(DISBUF[3]==0x04){
DISBUF[4]=0;
DISBUF[3]=0;
}
}
if(dl>=14){ //show dot on
DISBUF[3]=0x80; //add dot
show();
}
if(dl<=14){ //show dot off
show();
}
}
}
```