



Nota de Aplicación: CAN-037

Título: **Utilización de displays LCD color con controladores S1D13706 y Rabbit**

Autor: Sergio R. Caprile, Senior Engineer

Revisiones	Fecha	Comentarios
0	20/07/05	

Modificamos levemente el desarrollo de las CAN-035 y CAN-036, para trabajar con osciladores de 50MHz, comercializados por Cika. Esto nos permite tener un ciclo de acceso más corto, con menos wait-states, y lograr frecuencias de barrido que logran mayor calidad de imagen.

Configuración del S1D13706

Los valores a setear en cada uno de los registros se obtienen de idéntica forma que en la CAN-035 y CAN-036, con la salvedad de indicar 50MHz en la solapa *Clocks*. Exportamos de igual modo los datos en un archivo de tipo *C header file* (s1d13706.h), el cual podremos editar e incluir en el código para Rabbit

A continuación, la inicialización del chip para 8bpp y 141Hz de frecuencia de actualización de trama. Los valores los obtuvimos utilizando el software de configuración provisto por el fabricante, según comentáramos.

```
typedef unsigned short S1D_INDEX;
typedef unsigned char S1D_VALUE;

typedef struct
{
    S1D_INDEX Index;
    S1D_VALUE Value;
} S1D_REGS;

const static S1D_REGS aS1DRegs[] =
{
    {0x04,0x00}, // BUSCLK MEMCLK Config Register
    {0x05,0x32}, // PCLK Config Register
    {0x10,0xD0}, // PANEL Type Register
    {0x11,0x00}, // MOD Rate Register
    {0x12,0x2B}, // Horizontal Total Register
    {0x14,0x27}, // Horizontal Display Period Register
    {0x16,0x00}, // Horizontal Display Period Start Pos Register 0
    {0x17,0x00}, // Horizontal Display Period Start Pos Register 1
    {0x18,0xFA}, // Vertical Total Register 0
    {0x19,0x00}, // Vertical Total Register 1
    {0x1C,0xEF}, // Vertical Display Period Register 0
    {0x1D,0x00}, // Vertical Display Period Register 1
    {0x1E,0x00}, // Vertical Display Period Start Pos Register 0
    {0x1F,0x00}, // Vertical Display Period Start Pos Register 1
    {0x20,0x87}, // Horizontal Sync Pulse Width Register
    {0x22,0x00}, // Horizontal Sync Pulse Start Pos Register 0
    {0x23,0x00}, // Horizontal Sync Pulse Start Pos Register 1
    {0x24,0x80}, // Vertical Sync Pulse Width Register
    {0x26,0x01}, // Vertical Sync Pulse Start Pos Register 0
    {0x27,0x00}, // Vertical Sync Pulse Start Pos Register 1
    {0x70,0x03}, // Display Mode Register
    {0x71,0x00}, // Special Effects Register
    {0x74,0x00}, // Main Window Display Start Address Register 0
    {0x75,0x00}, // Main Window Display Start Address Register 1
    {0x76,0x00}, // Main Window Display Start Address Register 2
    {0x78,0x50}, // Main Window Address Offset Register 0
    {0x79,0x00}, // Main Window Address Offset Register 1
    {0x7C,0x00}, // Sub Window Display Start Address Register 0
    {0x7D,0x00}, // Sub Window Display Start Address Register 1
    {0x7E,0x00}, // Sub Window Display Start Address Register 2
    {0x80,0x50}, // Sub Window Address Offset Register 0
    {0x81,0x00}, // Sub Window Address Offset Register 1
}
```

CAN-037, Utilización de displays LCD color con controladores S1D13706 y Rabbit

```
{0x84,0x00}, // Sub Window X Start Pos Register 0
{0x85,0x00}, // Sub Window X Start Pos Register 1
{0x88,0x00}, // Sub Window Y Start Pos Register 0
{0x89,0x00}, // Sub Window Y Start Pos Register 1
{0x8C,0x4F}, // Sub Window X End Pos Register 0
{0x8D,0x00}, // Sub Window X End Pos Register 1
{0x90,0xEF}, // Sub Window Y End Pos Register 0
{0x91,0x00}, // Sub Window Y End Pos Register 1
{0xA0,0x00}, // Power Save Config Register
{0xA1,0x00}, // CPU Access Control Register
{0xA2,0x00}, // Software Reset Register
{0xA3,0x00}, // BIG Endian Support Register
{0xA4,0x00}, // Scratch Pad Register 0
{0xA5,0x00}, // Scratch Pad Register 1
{0xA8,0x00}, // GPIO Config Register 0
{0xA9,0x80}, // GPIO Config Register 1
{0xAC,0x00}, // GPIO Status Control Register 0
{0xAD,0x00}, // GPIO Status Control Register 1
{0xB0,0x00}, // PWM CV Clock Control Register
{0xB1,0x00}, // PWM CV Clock Config Register
{0xB2,0x00}, // CV Clock Burst Length Register
{0xB3,0x00}, // PWM Clock Duty Cycle Register};
```

Configuración del IOSTROBE

Un clock de frecuencia más alta nos permite disponer de un ciclo de acceso más corto. Al emplear un clock de 24MHz en las CAN-035 y CAN-036, el ciclo de acceso del S1D13706 con la configuración utilizada resultaba de unos 333ns. Un clock de 50MHz, en cambio, resulta en un ciclo de acceso de 160ns. Para un módulo Rabbit rápido como el RCM3360 (44MHz), esto es un total de 7 ciclos de clock.

```
// Use Port E bit 4 for Chip Select with 7 wait-states
#define STROBE          0x10
#define CSREGISTER     IB4CR
#define CSSHADOW       IB4CRShadow
#define CSCONFIG       0x48

// Initialize Port E bit to be a normal I/O pin
WrPortI(PEFR, &PEFRShadow, (PEFRShadow|STROBE));

// Initialize Port E bit to be an output pin
WrPortI(PEDDR, &PEDDRShadow, (PEDDRShadow|STROBE));

// Initialize Port E bit to be a chip select.
WrPortI(CSREGISTER, &CSSHADOW, CSCONFIG);

// Set Port E bit to be clocked by PCLK/2
WrPortI(PECR, &PECRShadow, (PECRShadow & ~0xFF));
```

Calidad de imagen

Según comentáramos en la CAN-031, estos displays reciben tres bits de información (R, G, B) por cada pixel en cada trama. La mayor cantidad de colores se logra mediante la alteración sistemática de esa información trama a trama, tarea que desarrolla el controlador (S1D13706) sin que nos enteremos. A frecuencias de actualización relativamente bajas, el ojo puede notar esa modificación en la forma de ruido o "lluvia" en la imagen, una especie de flickering de puntitos en las zonas donde la modificación es mayor. Es recomendable, entonces, utilizar frecuencias de actualización mayores a los 100Hz.