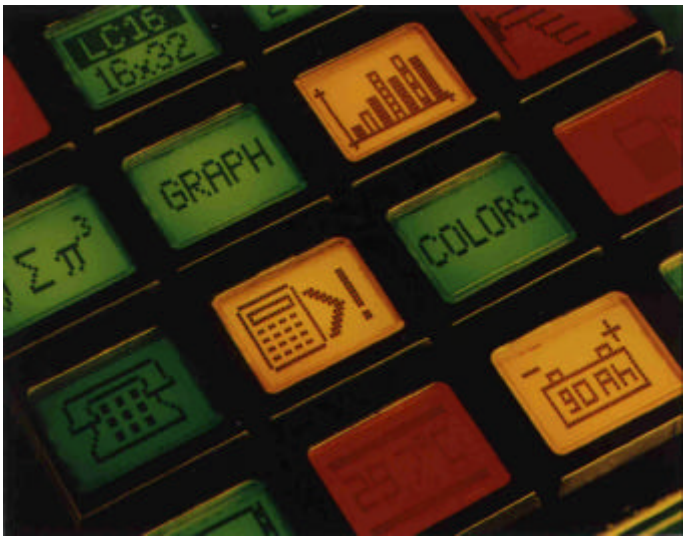
Programming and Interfacing

LCD Key Switches



RAPID TECHNOLOGY INTERFACES LTD.

LCD KEYSWITCH DIVISION

Pottery House, Pottery Road, Dun Laoghaire, Co. Dublin, Ireland.
Tel: +353-1-2350279 **Fax: +353-1-2350361**

Am Gneisenauflöz 6, 66538 Neunkirchen, Germany.
Tel: +49-6821-865460 **Fax: +49-6821-865461**

329 Moore Avenue, Leonia, NJ 07605, USA.
Tel: +1-201-592-7430 **Fax: +1-201-592-7456**

Please visit our website http://www.lcd-keys.com for additional information.

Programming and Interfacing LCD Key Switches

This document provides a guideline for designing a control circuit for programming and interfacing to the LCD key switches LC16.2 Trend and LC24.2 Trend.

For further assistance, a development system called **DevCom** is available for hardware developers. It contains all control circuitry as well as programming software necessary to control 4 LC switches.

What do we want to do?

How to design the interface control circuitry and how to program a LCD switch.

What do we know from the data sheets?

- Only 6 contacts:
  - 2 contacts for switch function
  - 1 serial data line
  - 1 clock line
  - 1 ground contact
  - 1 power line with 5 Vcc
- permanent clock running above 32 kHz (less than 4 MHz)
- Data Input, which is set to high when inactive, to shift in data, commands
- 8 bit data
- 1 bit parity
- 2 bit as spacer

What do we need to learn?

- How to program the keys (configuration, pictogram, illumination)
- How to interface to the keys
- How to scan switch function itself

How can we do this?

- Attach power supply
- Set up clock/data
- Interface to microprocessor/PC
- Program the key in Pseudocode

What do we need?

- Schematic
- Power supply
- Oscilloscope/Multimeter
- Additional components (diodes, resistors, ICs, Crystal) PC
- Programming language
- Interface cable to PC

Note: A complete hardware setup (DevCom) is available including programming software for rapid testing and software development. For more information, please contact your LCD key distributor.

What steps are required to interface with the keys?

- Supply power
- Provide Clock
- Set up parallel port (PC) to feed data to LCD key
- Convert parallel data to serial-link format of the LCD key
- Synchronize data flow with clock

Power Supply

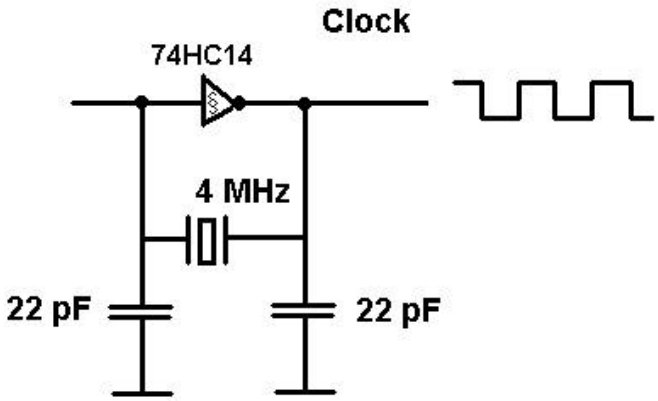
The LCD keys require +4.9 to 5 Volt DC and 100 mA per key switch.

Clock

To operate the LCD keys, a clock must be constantly applied. The frequency must be above 32 kHz and less than 4 MHz.

Warning! The LCD keys will be destroyed if no clock is present when power is supplied to the keys!

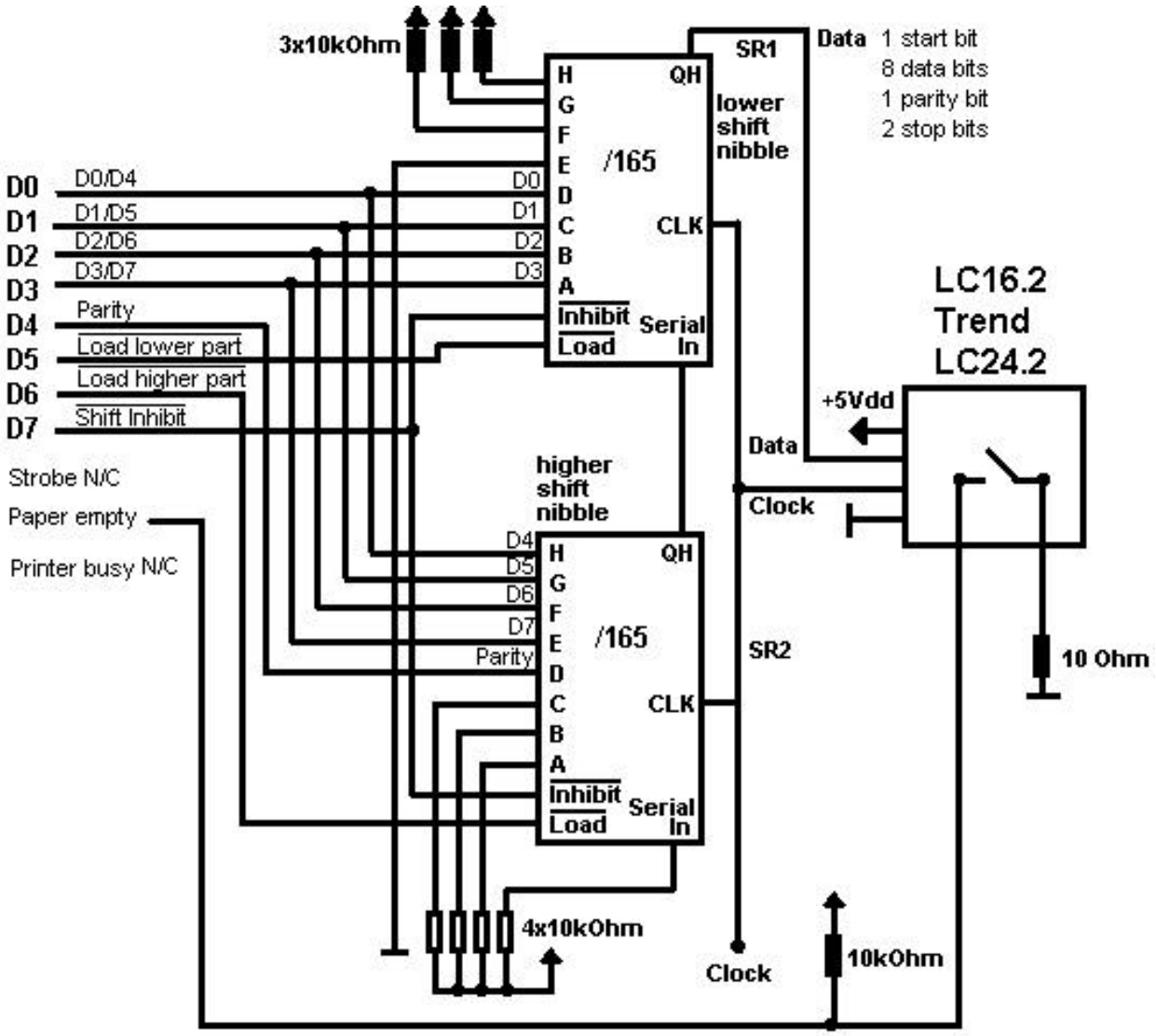
The sample diagram below shows a clock circuit with a 4MHz crystal, 74HC14 Schmitt-trigger inverter, and 22 pF capacitors to generate the clock signal.



Parallel to Serial Data Link

The LCD keys are controlled via a synchronous serial data stream. The data can be easily provided via a parallel or serial interface. If a parallel interface is used, the data must be converted to serial.

The diagram below shows how the parallel input is converted into the required serial data stream by using two 8 bit shift registers HC74(T)165.



Logical Timing Diagram

Clock	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
Shift Inhibit	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Apply D0-D3	X	D	D	D	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Apply D4-D7 + Parity	X	X	X	X	D	D	D	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Load lower part	X	X	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Load higher part	X	X	X	X	X	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Data out	1	1	1	1	1	1	1	1	1	1	0	D0	D1	D2	D3	D4	D5	D6	D7	Parity	1	1	1	1	1

D = valid Data
0 = low voltage
1 = high voltage (Vdd = 5V)
X = don't care

Key Selection

The LCD keys do not have individual key addresses. Instead, a decoder circuit is used to address the keys. Each LCD key waits for a start sequence before it will accept commands and data.

Parity is odd for start sequence
00 = address for all keys
01, 02, …, FE = valid for deselecting keyswitch

Parallel Port Pin Assignments

D0	D0 / D4	hardware description
D1	D1 / D5	
D2	D2 / D6	
D3	D3 / D7	
D4	parity	software related
D5	load lower shift register	
D6	load higher shift register	
D7	clock inhibit	

Programming in Pseudocode

The following section introduces the programming conventions for the LCD keys in Pseudocode. Please refer to the corresponding data sheets for LC16.2 Trend and LC24.2 Trend for the data format for the LCD key commands.

Functions

- Set Port (Data)
- Set Lower Nibble (Data)
- Set Higher Nibble (Data)
- Load Lower Nibble
- Load Higher Nibble
- Inhibit Clock
- Release Clock
- Define Parity (Data, Address)
- Set Parity (Data, Address)
- Get Parity (Data, Address)
- Send Byte (Data, Address)

Set Port (Data)

Write data to LPT1 data_latch
Outp(adr, value)
Data_latch = adr of LPT1 data

Set Lower Nibble (Data)

X=data & 0x0F
Data_latch = data_latch & 0xF0
Data_latch = x | data_latch
Set port(data_latch)

Set Higher Nibble (Data)

X=data & 0x0F
Shift x 4 times to right
[divide by 2;2;2;2!]
Data_latch = data_latch & 0x0F
Data_latch = x | data_latch
Set port(data_latch)

Load Lower Nibble (Data)

X=data_latch & 11011111 ;(= 0xDF)

```
Set port(x)
X=data_latch | 00100000          ;(= 0x20)
Set port(x)
```

Load Higher Nibble

```
X=data_latch & 10111111          ;(= 0xBF)
Set port(x)
X=data_latch & 01000000          ;(= 0x40)
Set port(x)
```

Inhibit Clock

```
X=data_latch & 01111111          ;(= 0x7F)
Set port(x)
```

Release Clock

```
X=data_latch | 10000000          ;(= 0x80)
Set port(x)
```

Define Parity (Data, Address)

```
Parity=0
For x=8; to 0; x - -
Begin
    If (data & 0000001)          ;(= 1)
            Parity = parity + 1
    Data = data/2
End
Parity = parity & 00000001
                                 ;adr = 0 = start byte, address byte
                                 ;adr = 1 = data byte
Adr=adr & 00000001
If parity = 0 set parity to 1
Else set parity to 0
If adr = 0
Begin
    Parity = parity +1
    Parity = parity & 00000001
End
```

Set Parity (Data, Address)

```
X = define parity (data, adr)
If x = 0
```

```
    Data_latch = data_latch & 11101111        ;(= 0xEF)
Else data_latch = data_latch | 00010000       ;(= 0x10)
Set port(data_latch)
```

Send Byte (Data, Address)

```
Data_latch = 0xFF
Inhibit Clock
Set Lower Nibble (Data)
Load Lower Nibble
Set Higher Nibble (Data)
Set Parity (Data, Address)
Load Higher Nibble
Release Clock
```

LCD Key Commands

Send Pictogram

```
Send byte (0,0)                 ;adr = 0; parity 0 to indicate data send
Send byte (80,1)
For x=0; x=108; x+              ;108 bytes for LC24.2 (36x24)
                                ;  64 bytes for LC16.2 (32x16)
Begin
    Send byte(data field(x),1)  ;data field = pictogram data
End
Send byte(AA,0)                 ;deselects key
```

Send Color

```
Send byte(0x0)
Send byte(0xED,1)
Send byte(color,1)
Send byte(AA,0)                 ;deselects key
```

Send Mux (Mutiplex frequency)

```
Send byte(0,0)
Send byte(0xEE,1)
Send byte(mux_frequency,1)    ;mux_frequency from table in data sheet
Send byte(AA,0)               ;deselects key
```

Send Mux Register LC24.2 ***only valid for LC24.2!***

```
Send byte(0,0)
Send byte(0xEF,1)
Send byte(0x07,1)
Send byte(0x00,1)
Send byte(AA,0)               ;deselects key
```

Send Mux Register LC16.2 ***only valid for LC16.2!***

```
Send byte(0,0)
Send byte(0xEF,1)
Send byte(0x02,1)
Send byte(0x05,1)
Send byte(AA,0)               ;deselects key
```