Microprocessors & Interfacing

Input/Output Devices (II)

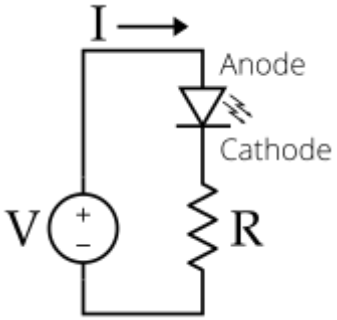
Lecturer : Annie Guo

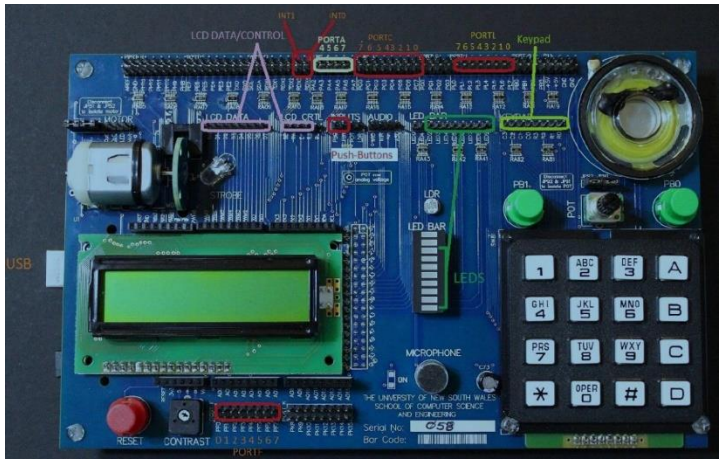
Lecture Overview

- Output devices
  - LED
  - LCD

LED

- Light-Emitting Diode
- Emit light when current flows through it
  - Its brightness increases with the current value
    - Within a limited range
- Can be used to indicate
  - a 1-bit digital output
    - LED on, V=1
    - LED off, V=0
  - an analog output value
    - To be covered later





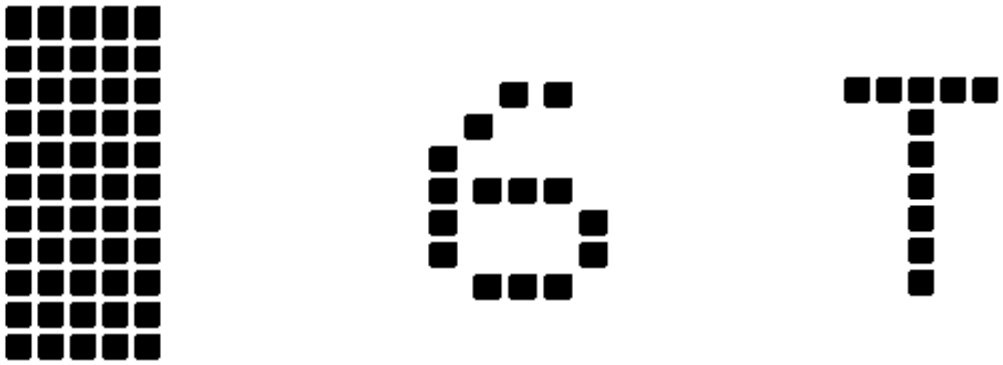
LCD

- Liquid Crystal Display
- Programmable output device

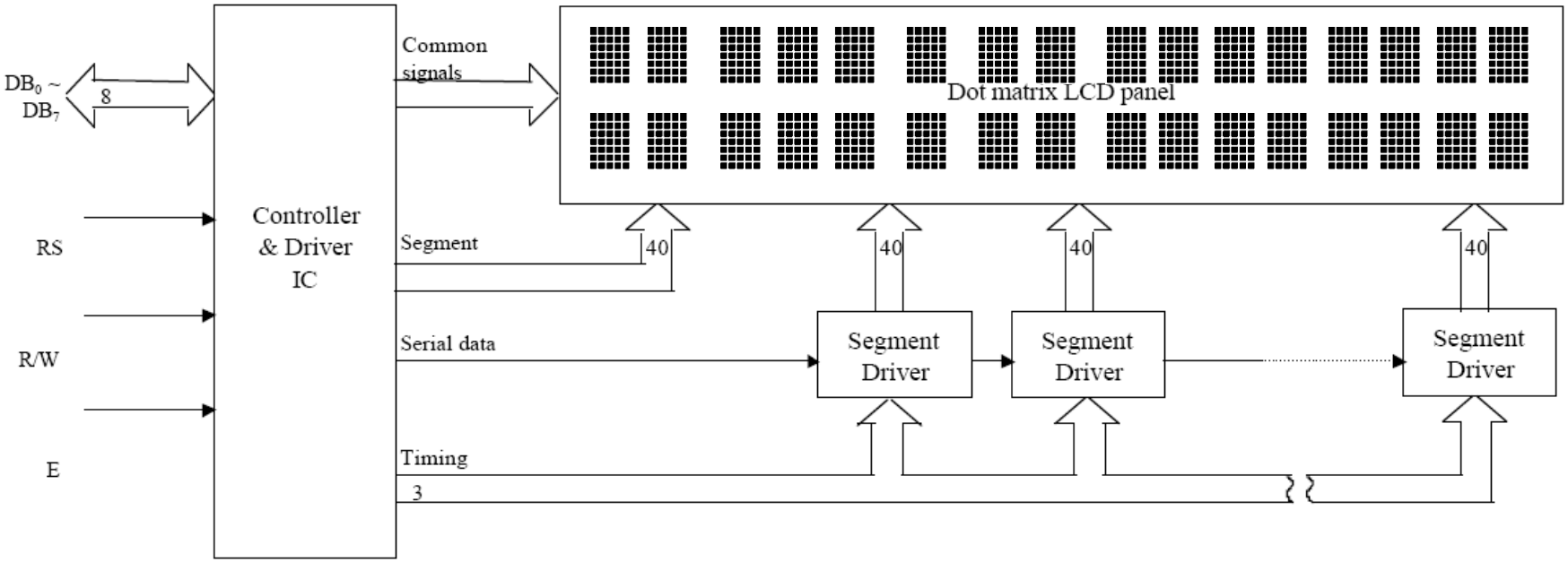
How it works

Dot Matrix LCD

- Characters are displayed using a dot matrix.
  - 5x7, 5x8, and 5x11
- A controller is used for communication between the LCD and other components, e.g. microprocessor unit (MPU)
- The controller has an internal character generator ROM. All display functions are controllable by instructions.



Dot Matrix LCD Diagram*



Note: The diagram and tables are extracted from the LCD Manual available on the course website

Pin Descriptions

Signal name	No. of Lines	Input/Output	Connected to	Function
DB4 ~ DB7	4	Input/Output	MPU	4 lines of high order data bus. Bi-directional transfer of data between MPU and module is done through these lines. Also DB_7 can be used as a busy flag. These lines are used as data in 4 bit operation.
DB0 ~ DB3	4	Input/Output	MPU	4 lines of low order data bus. Bi-directional transfer of data between MPU and module is done through these lines. In 4 bit operation, these are not used and should be grounded.
E	1	Input	MPU	Enable - Operation start signal for data read/write.
R/W	1	Input	MPU	Signal to select Read or Write "0": Write "1": Read
RS	1	Input	MPU	Register Select "0": Instruction register (Write) : Busy flag; Address counter (Read) "1": Data register (Write, Read)
Vee	1		Power Supply	Terminal for LCD drive power source.
Vcc	1		Power Supply	+5V
Vss	1		Power Supply	0V (GND)

Operations

- MPU communicates with LCD through two registers
  - Instruction Register (IR)
    - To store
      - instruction code
        - » e.g Display Clear or Cursor Shift
      - address for the Display Data RAM (DD RAM)
      - etc.
  - Data Register (DR)
    - To store
      - data to be read/written to/from the DD RAM of the display controller.

Operations (cont.)

- The register select (RS) signal determines which of these two registers is selected

- The table below shows the operations by the two control signals

RS	R/W	Operation
0	0	IR write, internal operation (Display Clear etc.)
0	1	Busy flag (DB_7) and Address Counter ($DB_0 \sim DB_6$) read
1	0	DR Write, Internal Operation (DR ~ DD RAM or CG RAM)
1	1	DR Read, Internal Operation (DD RAM or CG RAM)

Operations (cont.)

  - When the busy flag is high or “1”, the LCD is busy with the internal operation.
  - The next instruction must not be written/sent to LCD until the busy flag is low or “0”.
- For details, refer to the LCD USER’S MANUAL.

LCD Instructions

- A list of binary instructions are available for LCD operations
- Some typical ones are explained in the next slides.

Instructions

- Function Set

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	0	0	0	1	DL	N	F	x	x

  - Set the interface data length, the number of lines, and character font.
    - DL = “1”, 8 –bits; otherwise 4 bits
    - N: Sets the number of lines
      - N = “0” : 1 line display
      - N = “1” : 2 line display
    - F: Sets character font.
      - F = “1” : 5 x 10 dots
      - F = “0” : 5 x 7 dots

Instructions

- Entry Mode Set

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	0	1	I/D	S

  - Set the Increment/Decrement and Shift modes
    - I/D = 1: increments the address counter by 1 for each DD RAM access (read or write); I/D = 0: decrements the address counter
    - S=0, no shift
    - S=1, shift the entire display
      - Shift to the left when I/D = 1
      - Shift to the right when I/D = 0

Instructions

- Display ON/OFF Control

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	1	D	C	B

  - Control the display ON/OFF, Cursor (光标) ON/OFF and Cursor Blink (闪烁) function.
    - D: The display is ON when D = 1 and OFF when D = 0.
    - C: The cursor displays when C = 1 and does not display when C = 0.
    - B: The character indicated by the cursor blinks when B = 1.

Instructions

- Clear Display

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	0	0	0	1

  - The display clears and the cursor moves to the upper left corner of the display.

Instructions

- Return Home

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	0	0	0	0	0	0	0	1	x

  - The cursor moves to the upper left corner of the display. Text on the display remains unchanged.

Instructions

- Write Data to DD RAM

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	1	0	D	D	D	D	D	D	D	D

  - Write binary 8-bit data DDDDDDDD to the CG or DD RAM.
  - The previous designation determines whether the CG or DD RAM is to be written (CG RAM address set or DD RAM address set). After a write the entry mode will automatically increase or decrease the address by 1. Display shift will also follow the entry mode.

Instructions

- Set DD RAM Address

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	0	1	A	A	A	A	A	A	A

  - Sets the address counter to DD RAM.
  - The address range:
    - For 1-line display, 0x00-0x4F
    - For 2-line display,
      - 0x00-0x27 for the first line
      - 0x40-0x67 for the second line

Instructions

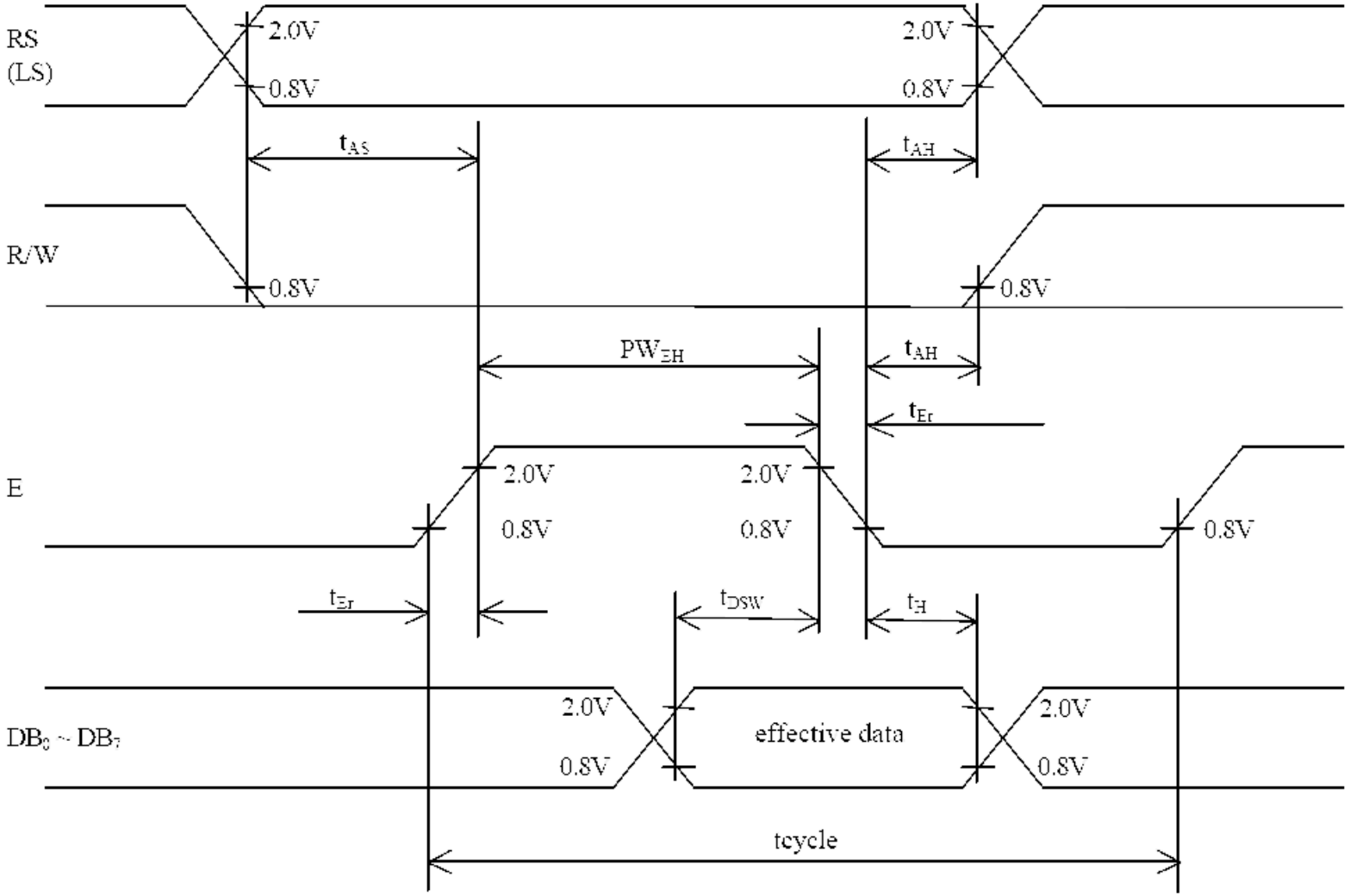
- Read Busy Flag and Address

	RS	R/W	DB7	DB6	DB5	BD4	DB3	DB2	DB1	DB0
Code	0	1	BF	A	A	A	A	A	A	A

  - Read the busy flag (BF) and value of the address counter (AC). BF = 1 indicates that an internal operation is in progress and the next instruction will not be accepted until BF is set to “0”. If the display is written while BF = 1, abnormal operation will occur.

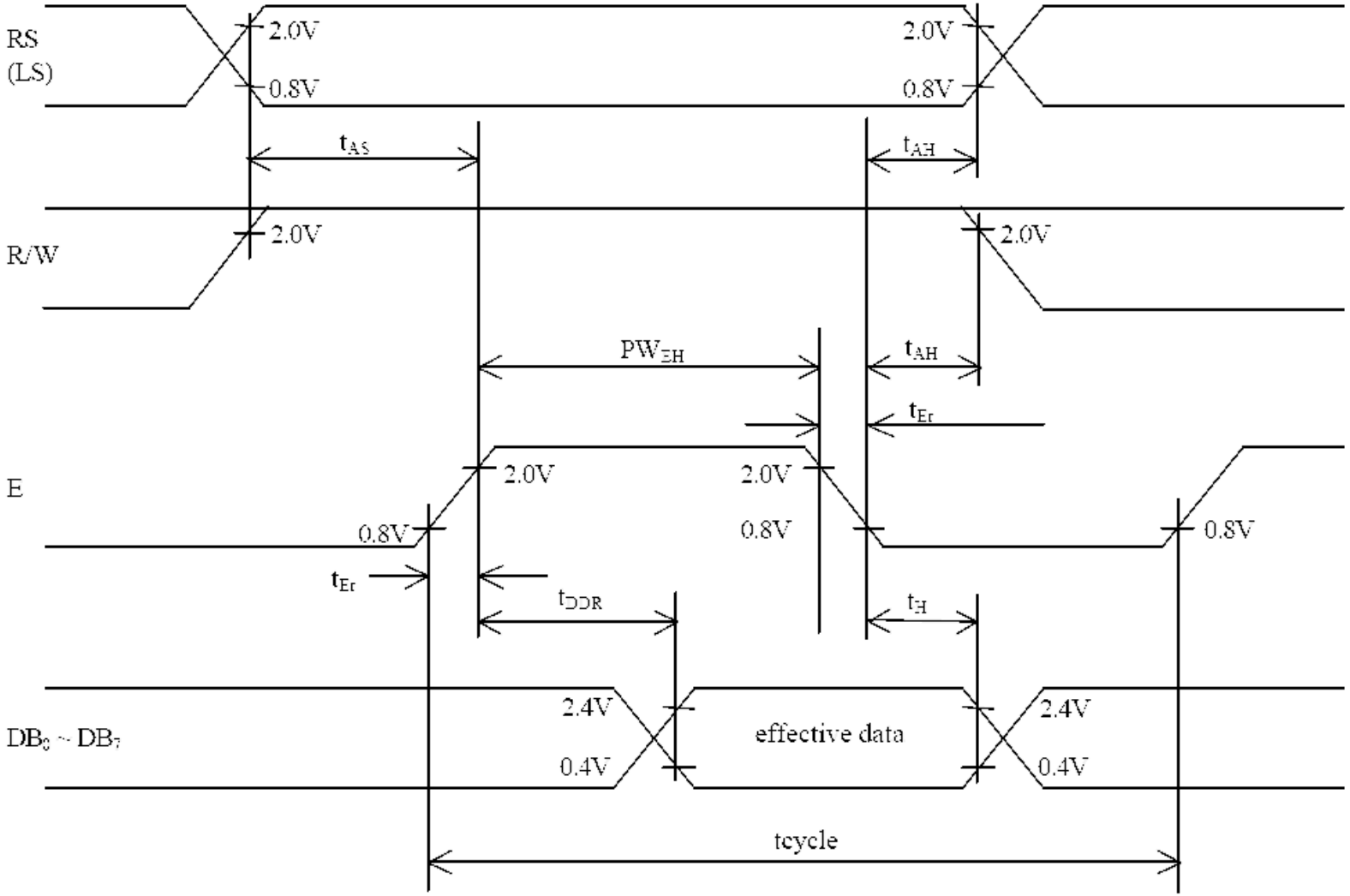
Timing Characteristics*

- For write operation



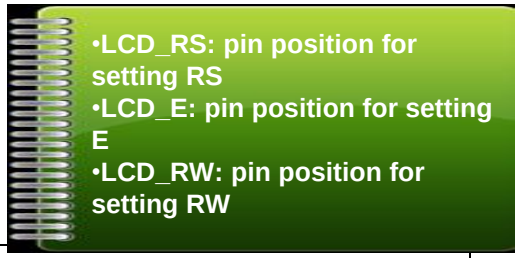
Timing Characteristics*

- For read operation



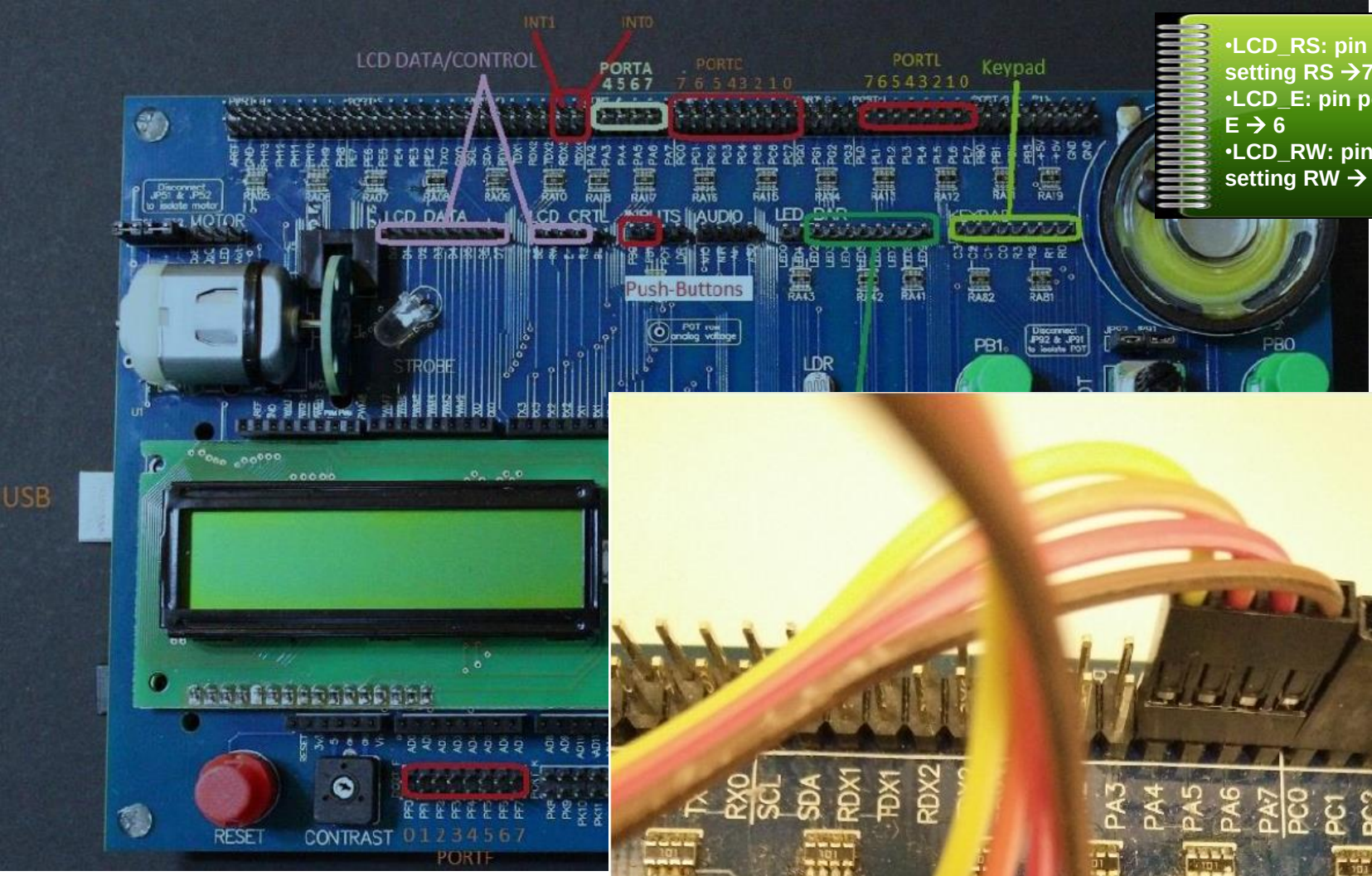
Examples

- Send a command to LCD

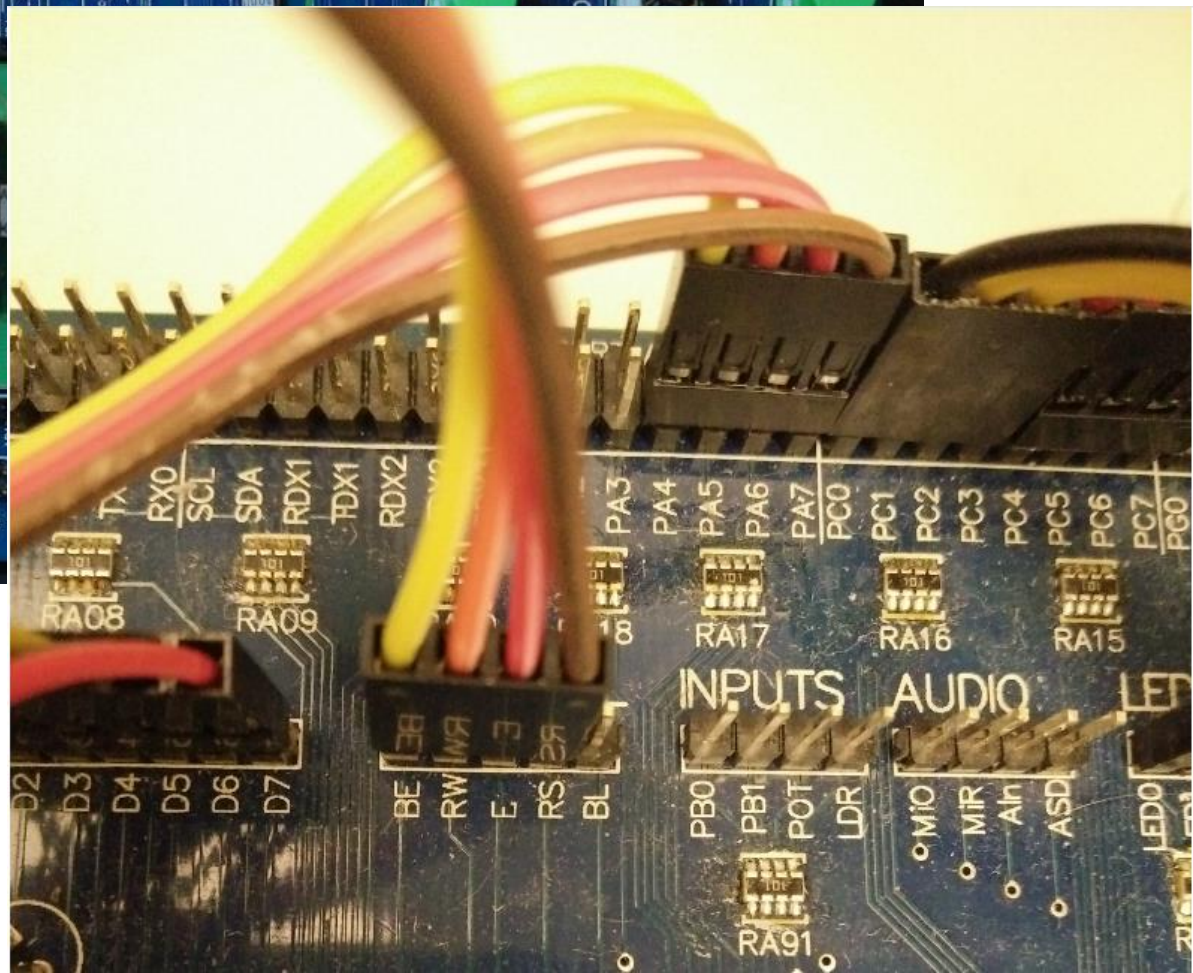



```
; General purpose register data stores value to be written to the LCD
; Port F is output port and connects to LCD data port; Port A controls the LCD (Bit LCD_RS for RS and bit LCD_RW for RW, LCD_E for E). The character to be displayed is stored in register data
; Assume all labels are pre-defined.

.macro lcd_write_com
        out PORTF, data             ; set the data port's value up
        ldi temp, (0<<LCD_RS)|(0<<LCD_RW)
        out PORTA, temp             ; RS = 0, RW = 0 for a command write
        nop                         ; delay to meet timing (Set up time)
        sbi PORTA, LCD_E            ; turn on the enable pin
        nop                         ; delay to meet timing (Enable pulse width)
        nop
        nop
        cbi PORTA, LCD_E            ; turn off the enable pin
        nop                         ; delay to meet timing (Enable cycle time)
        nop
        nop
.endmacro
```



- **LCD_RS: pin position for setting RS → 7**
- **LCD_E: pin position for setting E → 6**
- **LCD_RW: pin position for setting RW → 5**



Examples

- Send data to display

```
; comments are same as in the previous slide.

.macro lcd_write_data
        out PORTF, data             ; set the data port's value up
        ldi temp, (1 << LCD_RS)|(0<<LCD_RW)
        out PORTA, temp             ; RS = 1, RW = 0 for a data write
        nop                         ; delay to meet timing (Set up time)
        sbi PORTA, LCD_E            ; turn on the enable pin
        nop                         ; delay to meet timing (Enable pulse width)
        nop
        nop
        cbi PORTA, LCD_E            ; turn off the enable pin
        nop                         ; delay to meet timing (Enable cycle time)
        nop
        nop
.endmacro
```

Examples

- Check LCD and wait until LCD is not busy

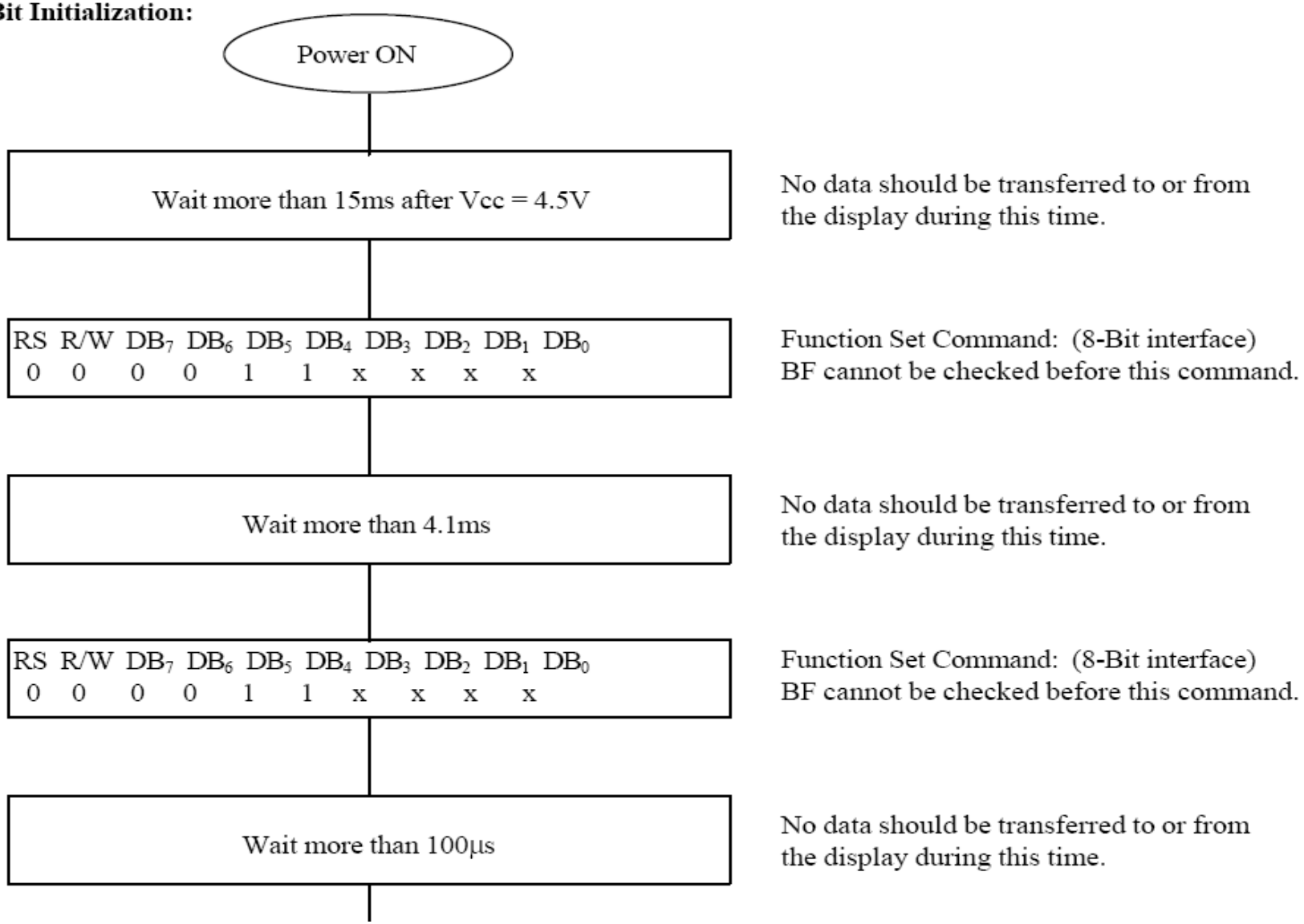
```
; comments are same as in the previous slide
.macro lcd_wait_busy
            clr temp
            out DDRF, temp              ; Make port F as an input port for now
            ldi temp, 1 << LCD_RW
            out PORTA, temp             ; RS = 0, RW = 1 for a command port read
    busy_loop:
            nop                         ; delay to meet set-up time
            sbi PORTA, LCD_E            ; turn on the enable pin
            nop                         ; delay to meet timing (Data delay time)
            nop
            nop
            in temp, PINF               ; read value from LCD
            cbi PORTA, LCD_E            ; turn off the enable pin
            sbrc temp, LCD_BF           ; if the busy flag is set
            rjmp busy_loop              ; repeat command read
            clr temp                    ; else
            out PORTA, temp             ; turn off read mode,
            ser temp                    ;
            out DDRF, temp              ; make port F an output port again
.endmacro
```

LCD Initialization

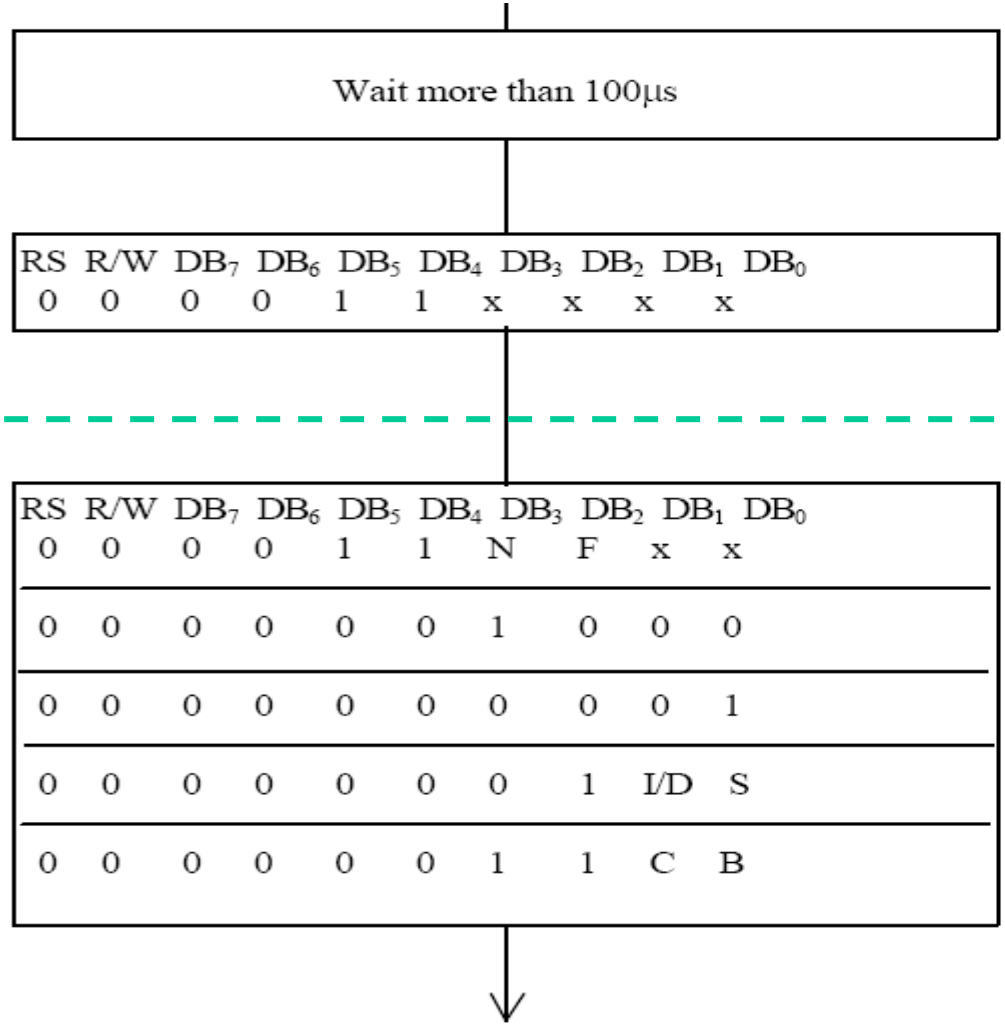
- LCD should be initialized before use
- Internal Reset Circuit can be used, but it is related to power supply loading, may not work properly.
- Therefore, software initialization is recommended.

Software Initialization

8 - Bit Initialization:



Software Initialization



No data should be transferred to or from the display during this time.

Function Set Command: (8-Bit interface)
After this command is written, BF can be checked.

Function Set (Interface = 8 bits, Set No. of lines and display font)

Display OFF

Clear Display

Entry Mode Set:

Display ON (Set C and B for cursor/Blink options.)

Note: BF should be checked before each of the instructions starting with Display OFF.

Example of Initialization Code

```
.include "m2560def.inc"

; The del_hi:del_lo register pair store the loop count
; each iteration of loop1 generates 1 us delay
.macro delay
loop1:
        ldi r16, 0x3
loop2:  dec r16
        nop
        brne loop2

        subi del_lo, 1
        sbci del_hi, 0
        brne loop1          ; taken branch takes two cycles.

.endmacro
```

Example of Initialization Code

```
ldi del_lo, low(15000)                ;delay (>15ms)
ldi del_hi, high(15000)
delay

; Function set command with N = 1 and F = 0
; for 2 line display and 5*7 font. The 1st command
ldi data, LCD_FUNC_SET | (1 << LCD_N)
lcd_write_com

ldi del_lo, low(4100)                 ; delay (>4.1 ms)
ldi del_hi, high(4100)
delay

lcd_write_com              ; 2nd Function set command

                                      ; continued
```

Example of Initialization Code

```
ldi del_lo, low(100)          ; delay (>100 ns)
ldi del_hi, high(100)
delay

lcd_write_com                 ; 3rd Function set command
lcd_write_com                 ; Final Function set command

lcd_wait_busy                 ; Wait until the LCD is ready
ldi data, LCD_DISP_OFF
lcd_write_com                 ; Turn Display off

lcd_wait_busy                 ; Wait until the LCD is ready
ldi data, LCD_DISP_CLR
lcd_write_com                 ; Clear Display

                                        ; continued
```

Example of Initialization Code

```
lcd_wait_busy                          ; Wait until the LCD is ready
; Entry set command with I/D = 1 and S = 0
; Set Entry mode: Increment = yes and Shift = no
ldi data, LCD_ENTRY_SET | (1 << LCD_ID)
lcd_write_com

lcd_wait_busy                          ; Wait until the LCD is ready
 ; Display On command with C = 1 and B = 0
ldi data, LCD_DISP_ON | (1 << LCD_C)
lcd_write_com

; …
```

A working sample code is available on the course website

Reading Material

- DOT Matrix LCD User's Manual
  - Available on the course website.
  - The useful examples of instructions can be found on pages 41-46.

Homework

1. Write an AVR assembly program to display
   - "ComArch" from left to right on the first line of the LCD, and
   - "COMP9032" from right to left on the second line of the LCD.