

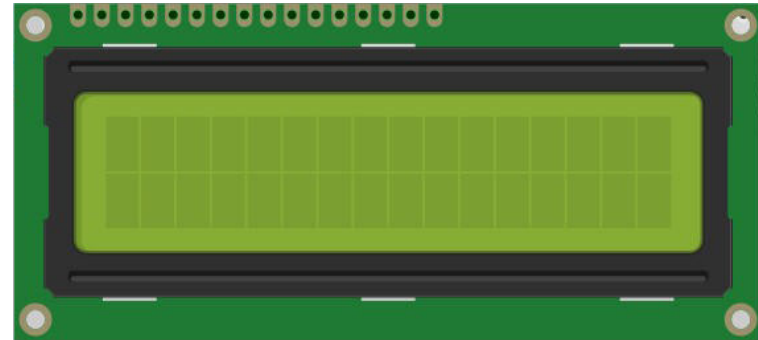


PV198 – One-chip Controllers

LCD Display

Content

1. LCD Display usage
2. LCD 1602A
3. Driver
4. Application





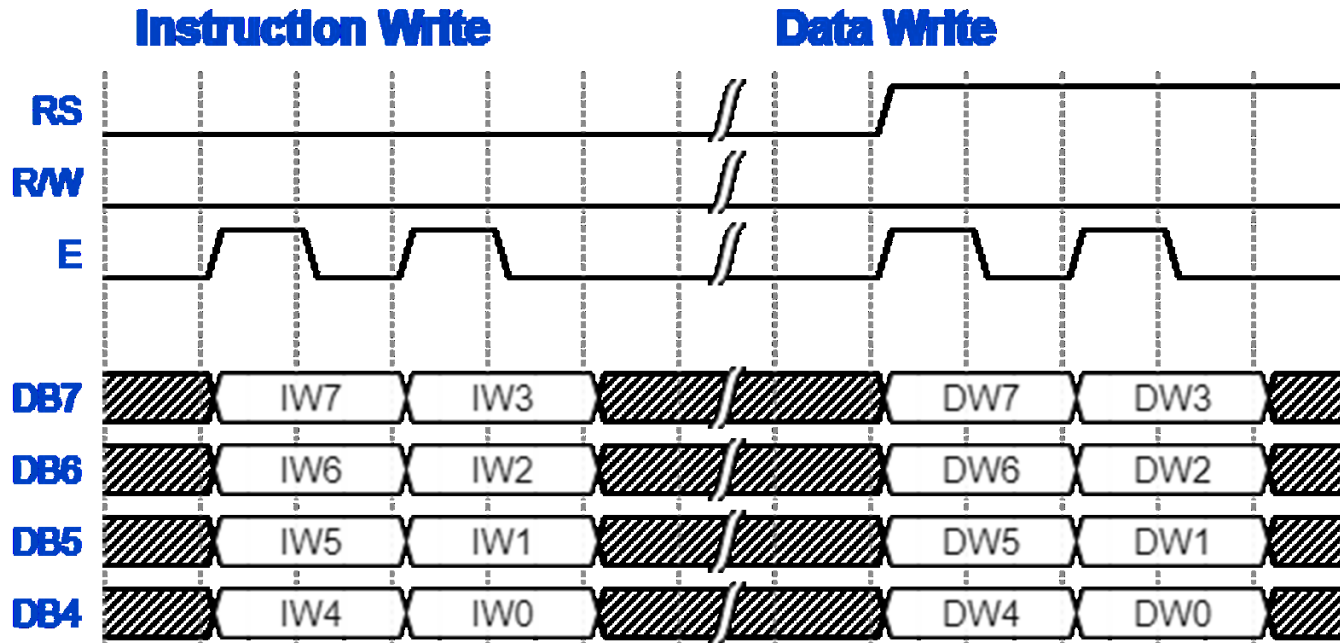
LCD Display usage

- Printers
- Routers
- Industrial equipment
- Consumer equipment, including some washing machines
- Much more...

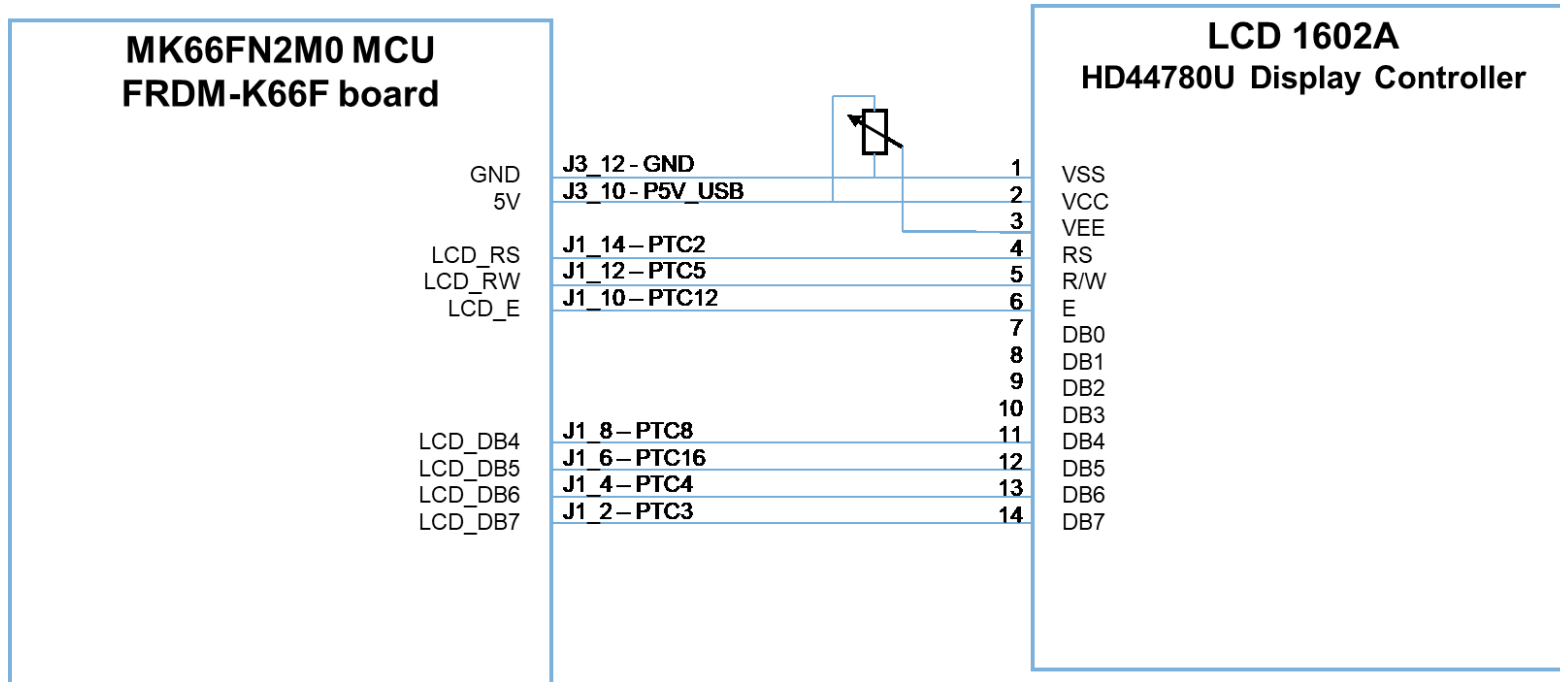
LCD 1602A – Overview

- LCD 1602A display driver
 - HD44780U Display Controller – [Datasheet](#)
 - 4 bits connection
 - Connection via GPIO pins (7 outputs)

LCD 1602A – Message

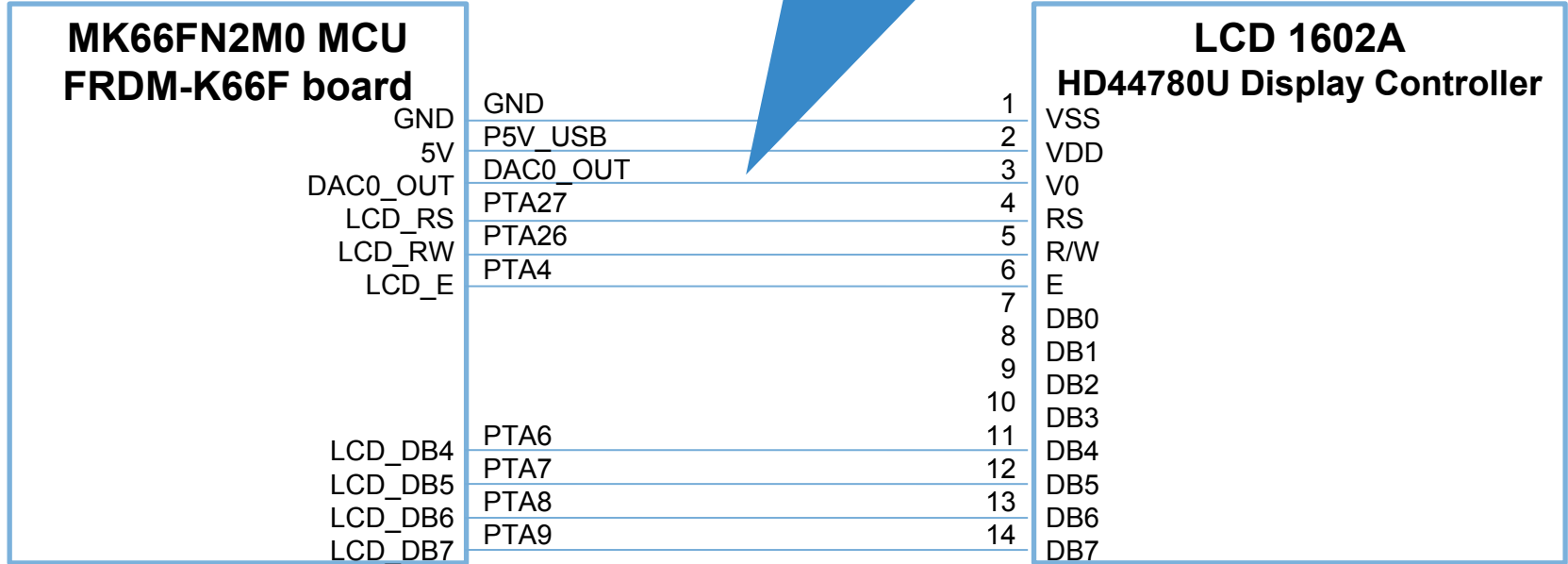


LCD 1602A – Scheme



LCD 1602A – Scheme

We will use DAC output instead of potentiometer and different pins

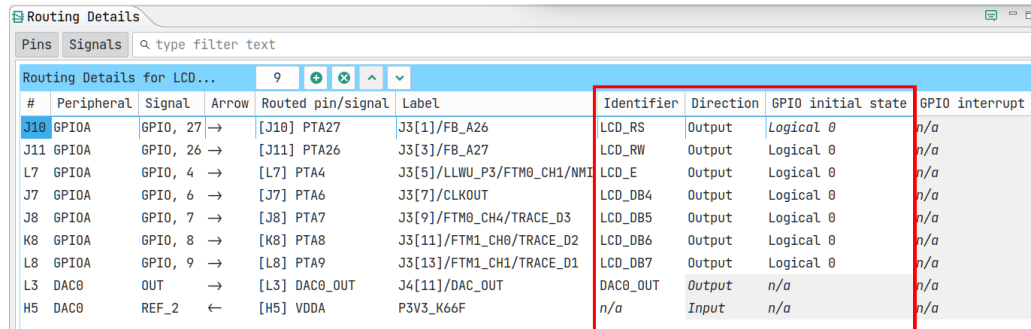


LCD 1602A – Driver

- Support of the 4-bit communication initialization
- SysTick timer usage for delays
- Basic command and data transfer functions
- Does not read busy flag – uses delays instead

LCD 1602A – Driver Interface

- The LCD driver requires to define the following identifiers in the Pins tool for GPIO output pins:
 - LCD_RS
 - LCD_RW
 - LCD_E
 - LCD_DB4
 - LCD_DB5
 - LCD_DB6
 - LCD_DB7

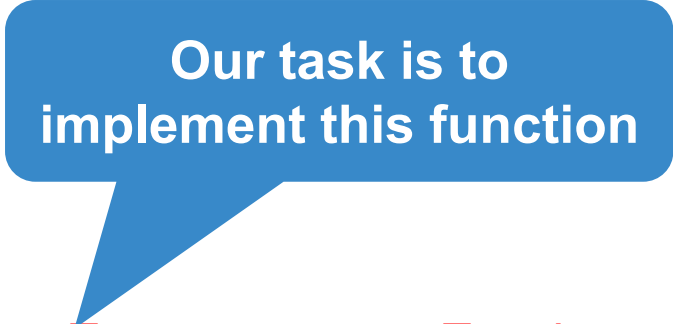


Routing Details for LCD...

#	Peripheral	Signal	Arrow	Routed pin/signal	Label	Identifier	Direction	GPIO initial state	GPIO interrupt
J10	GPIOA	GPIO, 27	→	[J10] PTA27	J3[1]/FB_A26	LCD_RS	Output	Logical 0	n/a
J11	GPIOA	GPIO, 26	→	[J11] PTA26	J3[3]/FB_A27	LCD_RW	Output	Logical 0	n/a
L7	GPIOA	GPIO, 4	→	[L7] PTA4	J3[5]/LLWU_P3/FTM0_CH1/NMI	LCD_E	Output	Logical 0	n/a
J7	GPIOA	GPIO, 6	→	[J7] PTA6	J3[7]/CLKOUT	LCD_DB4	Output	Logical 0	n/a
J8	GPIOA	GPIO, 7	→	[J8] PTA7	J3[9]/FTM0_CH4/TRACE_D3	LCD_DB5	Output	Logical 0	n/a
K8	GPIOA	GPIO, 8	→	[K8] PTA8	J3[11]/FTM1_CH0/TRACE_D2	LCD_DB6	Output	Logical 0	n/a
L8	GPIOA	GPIO, 9	→	[L8] PTA9	J3[13]/FTM1_CH1/TRACE_D1	LCD_DB7	Output	Logical 0	n/a
L3	DAC0	OUT	→	[L3] DAC0_OUT	J4[11]/DAC_OUT	DAC0_OUT	Output	n/a	n/a
H5	DAC0	REF_2	←	[H5] VDDA	P3V3_K66F	n/a	Input	n/a	n/a

LCD 1602A – Driver Interface

- The following LCD driver functions are provided:
 - void LCD_4BitsInit(uint32_t systick_clk_freq, bool cursor_on, bool cursor_blinking)
 - void LCD_Clear()
 - void LCD_Home()
 - void LCD_SetPosition(uint8_t x, uint8_t y)
 - void LCD_PutChar(uint8_t character)
 - void LCD_Print(char s[])
 - void LCD_SendCommand(uint8_t command)
 - **void LCD_SendData(uint8_t data, MessageType_t messageType)**
- SysTick timer is used for implementation of delay



Our task is to
implement this function

Application

- LCD driver initialization
- RTC peripheral initialization
- Real date and time display demo
- Do note the pins are different than on the image :)



Application

- Initialization sequence of the LCD driver

```
BOARD_InitPins();
```

Initialization of GPIO pins (wired interface) with the generated code from the Pins tools

Initialization of the LCD driver (using core clock frequency and without cursor)

```
LCD_4BitsInit(BOARD_BOOTCLOCKRUN_CORE_CLOCK, false, false);
```

- Runtime usage of the driver (example):

```
LCD_SetPosition(0, 0);
```

```
LCD_Print("a string");
```

Displaying "a string" on the LCD screen at position 0,0 (the first column and first line)

Application

- Implement function according to message format picture:
 - `void LCD_SendData(uint8_t data, MessageType_t messageType)`

Bonus:

- Display floating text on the display

Homework

- Generate something else on the LCD string. (Be creative)
- The automatic tests check for initialization and if there is something on the screen
- Avoid adding drivers, will break the project.