Lecture 6
Operating Modes, General Purpose Input/Output and LCD Controller

Texas Instruments Incorporated
University of Beira Interior (PT)

Pedro Dinis Gaspar, António Espírito Santo, Bruno Ribeiro, Humberto Santos
University of Beira Interior, Electromechanical Engineering Department
www.msp430.ubi.pt

Contents

Low power operating modes (1/11)

- **One of the main features of the MSP430 families:**
  - Low power consumption (about 1 mW/MIPS or less);
  - Important in battery operated embedded systems.
- **Low power consumption is only accomplished:**
  - Using low power operating modes design;
  - Depends on several factors such as:
    - Clock frequency;
    - Ambient temperature;
    - Supply voltage;
    - Peripheral selection;
    - Input/output usage;
    - Memory type;
    - ...

Low power operating modes (2/11)

- **Low power modes (LPM):**
  - 6 operating modes;
  - Configured by the SR bits: CPUOFF, OSCOFF, SCG1, SCG0.

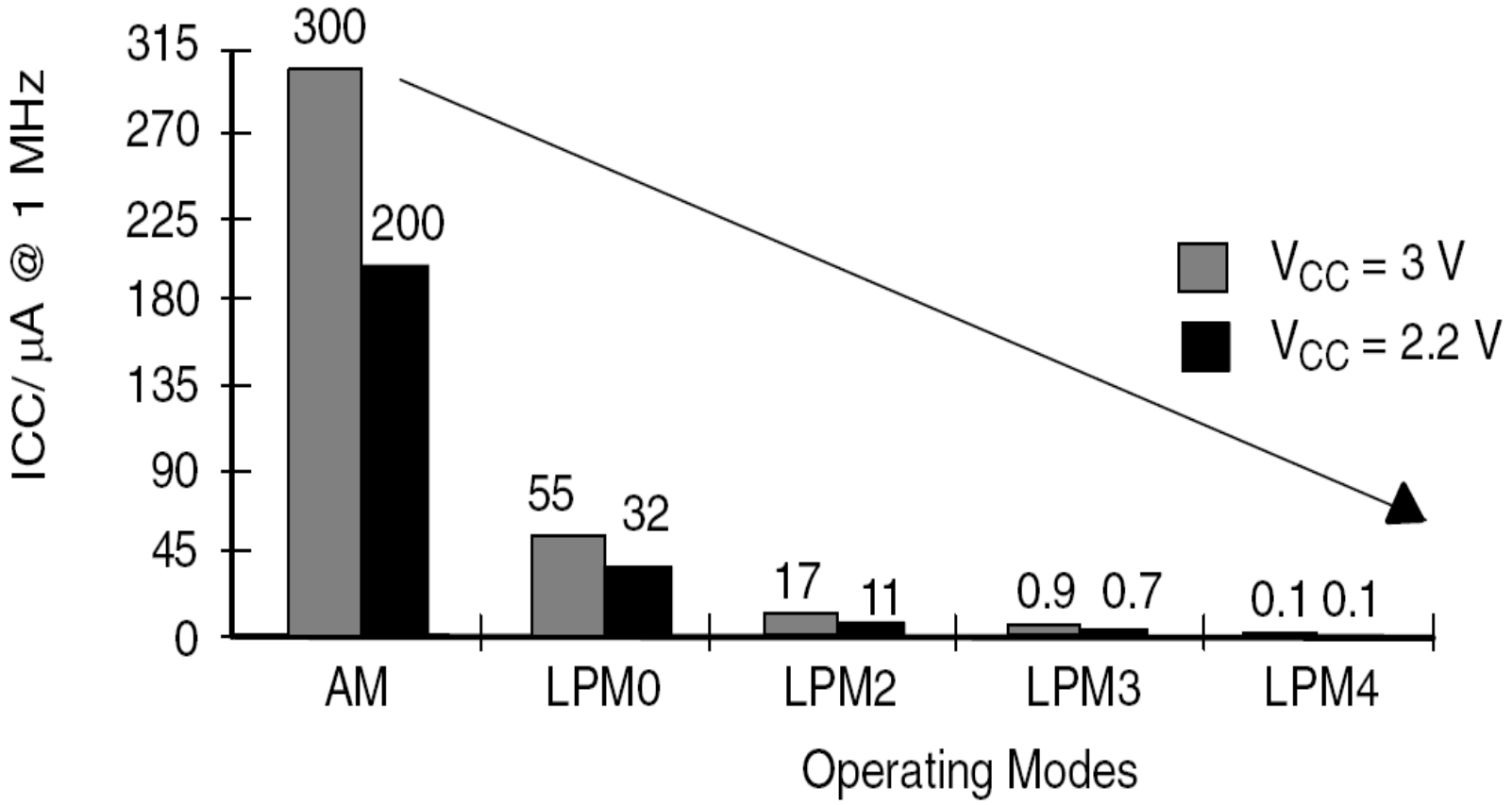
  - **Active mode (AM) - highest power consumption:**
    - Configured by disabling the SR bits described above;
    - CPU is active;
    - All enabled clocks are active;
    - Current consumption: 250 μA.

  - **Software selection up to 5 LPM of operation;**

  - **Operation:**
    - An interrupt event can wake up the CPU from any LPM;
    - Service the interrupt request;
    - Restore back to the LPM.

Low power operating modes (3/11)

- **Low power modes (LPM):**
  - Example: Typical current consumption (41x family).



Low power operating modes (4/11)

❑ Low power modes (LPM):

Mode	Current	SR bits configuration				Clock signals			Oscillator	
	~ [μA]	CPUOFF	OSCOFF	SCG1	SCG0	ACLK	SMCLK	MCLK	DCO	DC gen.
Low-power mode 0 (LPM0)	35	1	0	0	0	1	1	0	1	1
Low-power mode 1 (LPM1)	44	1	0	0	1	1	1	0	1	1*
Low-power mode 2 (LPM2)	19	1	0	1	0	1	0	0	0	1
Low-power mode 3 (LPM3)	0.8	1	0	1	1	1	0	0	0	0
Low-power mode 4 (LPM4)	0.1	1	1	1	1	0	0	0	0	0

*DCO's DC generator is enabled if it is used by peripherals.

Low power operating modes (5/11)

Low power modes (LPM) characteristics:

- **LPM0 to LPM3:**
  - Periodic processing based on a timer interrupt;
  - **LPM0**: Both DCO source signal and DCO's DC gen.;
  - **LPM0** and **LPM1**: Main difference between them is the condition of enable/disable the DCO's DC generator;
  - **LPM2**: DCO's DC generator is active and DCO is disabled;
  - **LPM3**: Only the ACLK is active (< 2 μA).
- **LPM4:**
  - Externally generated interrupts;
  - No clocks are active and available for peripherals.
  - Reduced current consumption (0.1 μA).

Low power operating modes (6/11)

- **Program flow steps:**
  - **Enter Low-power mode:**
    - Enable/disable CPUOFF, OSCOFF, SCG0, SCG1 bits in SR;
    - LPM is active after writing to SR;
    - CPU will suspend the program execution;
    - Disabled peripherals:
      - Operating with any disabled clock;
      - Individual control register settings.
    - All I/O port pins and RAM/registers are unchanged;
    - Wake up is possible through any enabled interrupt.

Low power operating modes (7/11)

- **Program flow steps:**
  - **An enabled interrupt event wakes the MSP430;**
  - **Enter ISR:**
    - The operating mode is saved on the stack during ISR;
    - The PC and SR are stored on the stack;
    - Interrupt vector is moved to the PC;
    - The CPUOFF, SCG1, and OSCOFF bits are automatically reset, enabling normal CPU operation;
    - IFG flag cleared on single source flags.
  - **Returning from the ISR:**
    - The original SR is popped from the stack, restoring the previous operating mode;
    - The SR bits stored in the stack are modified returning to a different operating mode after RETI instruction.

Low power operating modes (8/11)

❑ Examples of applications development using the MSP430 with and without low power modes consideration:

Example	Without low power mode	With low power mode
Toggling the bit 0 of port 1 (P1.0) periodically	Endless loop (100 % CPU load)	LPM0 Watchdog timer interrupt
UART to transmit the received message at a 9600 baud rate	Polling UART receive (100 % CPU load)	UART receive interrupt (0.1 % CPU load)
Set/reset during a time interval, periodically, of the peripheral connected to the bit 2 of port 1 (P1.2)	Endless loop (100 % CPU load)	Setup output unit (Zero CPU load)
Power manage external devices like Op-Amp	Putting the OPA Quiescent (Average current: 1 μA)	Shutdown the Op-Amp between data acquisition (Average current: 0.06 μA)
Power manage internal devices like Comparator A	Always active (Average typical current: 35 μA)	Disable Comparator A between data acquisition
Respond to button-press interrupt in P1.0 and toggle LED on P2.1	Endless loop (100 % CPU load)	Using LPMs while the LED is switch off: LPM3: 1.4 μA LPM4: 0.3 μA Configure unused ports in output direction P1 interrupt service routine

Low power operating modes (9/11)

- **Rules of thumb for the configuration of LP applications:**
  - Extended ultra-low power standby mode. Maximize LPM3;
  - Minimum active duty cycle;
  - Performance on-demand;
  - Use interrupts to control program flow;
  - Replace software with on chip peripherals;
  - Manage the power of external devices;
  - Configure unused pins properly, setting them as outputs to avoid floating gate current.

Low power operating modes (10/11)

Rules of thumb for LP applications configuration:

- Low-power efficient coding techniques:
  - Optimize program flow;
  - Use CPU registers for calculations and dedicated variables;
  - Same code size for word or byte;
  - Use word operations whenever possible;
  - Use the optimizer to reduce code size and cycles;
  - Use local variable (CPU registers) instead of global variables (RAM);
  - Use bit mask instead of bit fields;

Low power operating modes (11/11)

Rules of thumb for LP applications configuration:

- Low-power efficient coding techniques:
  - Use unsigned data types where possible;
  - Use pointers to access structures and unions;
  - Use “static const” class to avoid run-time copying of structures, unions, and arrays;
  - Avoid modulo;
  - Avoid floating point operations;
  - Count down “for” loops;
  - Use short ISRs.

I/O Introduction (1/3)

- **Up to ten 8-bit digital Input/Output (I/O) ports, P1 to P10 (depending on the MSP430 device);**
- **I/O ports P1 and P2 have interrupt capability;**
- **Each interrupt for these I/O lines can be individually configured:**
  - To provide an interrupt on a rising or falling edge;
  - All interruptible I/O lines source a single interrupt vector.
- **The available digital I/O pins for the hardware development tools:**
  - eZ430-F2013: 10 pins - Port P1 (8 bits) and Port P2 (2 bits);
  - eZ430-RF2500: 32 pins - Port P1 to P4 (8 bits);
  - Experimenter's board: 80 pins – Port P1 to P10 (8 bits).

I/O Introduction (2/3)

- **Each I/O port can be:**
  - Programmed independently for each bit;
  - Combine input, output, and interrupt functionality;
  - Edge-selectable input interrupt capability for all 8 bits of ports P1 and P2;
  - Read/write access to port-control registers is supported by all two- or one-address instructions;
  - Individually programmable pull-up/pull-down resistor (2xx family only).

I/O Introduction (3/3)

- **The port pins can be individually configured as I/O for special functions, such as:**
  - USART – Universal Synchronous/Asynchronous Receive/Transmit for serial data;
  - Input comparator for analogue signals;
  - Analogue-to-Digital converter;
  - Others functions (see specific datasheet for details).

Registers (1/6)

- **Independent of the I/O port type (non-interruptible or interruptible), the operation of the ports is configured by user software, as defined by the following registers:**

  - **Direction Registers (PxDIR):**
    - Read/write 8-bit registers;
    - Select the direction of the corresponding I/O pin, regardless of the selected function of the pin (general purpose I/O or as a special function I/O);
    - For other module functions, must be set as required by the other function.

    - **PxDIR configuration:**

      Bit = 1: the individual port pin is set as an output;

      Bit = 0: the individual port pin is set as an input.

Registers (2/6)

- **Input Registers (PxIN):**
  - When pins are configured as GPIO, each bit of these read-only registers reflects the input signal at the corresponding I/O pin;
  - **PxIN configuration:**

    Bit = 1: The input is high;

    Bit = 0: The input is low;
  - Tip: Avoid writing to these read-only registers because it will result in increased current consumption.

Registers (3/6)

- **Output Registers (PxOUT):**
  - Each bit of these registers reflects the value written to the corresponding output pin.

  - **PxOUT configuration:**

    Bit = 1: The output is high;

    Bit = 0: The output is low.

    – Note: the PxOUT Register is read-write. This means that the previous value written to it can be read back and modified to generate the next output signal.

Registers (4/6)

- **Pull-up/down Resistor Enable Registers (PxREN):**

  - Only available for the 2xx family;
  - Each bit of this register enables or disables the pull-up/pull-down resistor of the corresponding I/O pin.

  - **PxREN configuration:**
    - Bit = 1: Pull-up/pull-down resistor enabled;
    - Bit = 0: Pull-up/pull-down resistor disabled.

    - **When pull-up/pull-down resistor is enabled:**
    - In this case Output Registers (PxOUT) select:
      - » Bit = 1: The pin is pulled up;
      - » Bit = 0: The pin is pulled down.

Registers (5/6)

- **Function Select Registers: (PxSEL) and (PxSEL2):**
  - Some port pins are multiplexed with other peripheral module functions (see the device-specific datasheet);
  - These bits: PxSEL and PxSEL2 (see specific device datasheet), are used to select the pin function:
    - I/O general purpose port;
    - Peripheral module function.
  - **PxSEL configuration:**

    Bit = 0: I/O Function is selected for the pin;

    Bit = 1: Peripheral module function enabled for pin.

Registers (6/6)

- **Function Select Registers: (PxSEL) and (PxSEL2):**
  - The 2xx family of devices provide the PxSEL2 bit to configure additional features of the device;
  - The PxSEL and PxSEL2 bits in combination provide the following configuration:
    - Bit = 0: I/O function is selected for the pin;
    - Bit = 1: Peripheral module function is selected for the pin.

PxSEL	PxSEL2	Pin Function
0	0	Selects general purpose I/O function
0	1	Selects the primary peripheral module function
1	0	Reserved (See device-specific data sheet)
1	1	Selects the secondary peripheral module function

Note: P1 and P2 configured as peripheral module function (PxSEL = 1 and/or PxSEL2) -> interrupts disabled.

Interruptible ports (P1 and P2) (1/2)

- **Each pin of ports P1 and P2 is able to make an interrupt request;**
- **Pins are configured with additional registers:**
  - **Interrupt Enable (PxIE):**
    - Read-write register to enable interrupts on individual pins;
    - **PxIE configuration:**
      Bit = 1: The interrupt is enabled;
      Bit = 0: The interrupt is disabled.
    - Each PxIE bit enables the interrupt request associated with the corresponding PxIFG interrupt flag;
    - Writing to PxOUT and/or PxDIR can result in setting PxIFG.

Interruptible ports (P1 and P2) (2/2)

- **Interrupt Edge Select Registers (PxIES):**
  - Selects the transition on which an interrupt occurs (if PxIE and GIE are set);
  - **PxIES configuration:**
    Bit = 1: Interrupt flag is set on a high-to-low transition;
    Bit = 0: Interrupt flag is set on a low-to-high transition.
- **Interrupt Flag Registers (PxIFG)**
  - Set automatically when an the programmed signal transition (edge) occurs;
  - PxIFG flag can be set and must be reset by software.
  - **PxIFG configuration:**
    Bit = 0: No interrupt is pending;
    Bit = 1: An interrupt is pending.

LCD Controller Introduction (1/3)

- **Both the ’3xx and ’4xx families provide controllers for liquid crystal displays (LCDs):**
  - LCD_A controller: MSP430x42x0 and MSP430FG461x;
  - LCD controller: All MSP430x4xx.
- **Example of LCD_A controller: Experimenter’s board;**
- **Features:**
  - Display memory;
  - Automatic signal generation;
  - Configurable frame frequency;
  - Blinking capability;
  - Support for 4 types of LCDs:
    - Static;
    - 2-mux, 1/2 bias (or 1/3 bias for LCD_A controller);
    - 3-mux, 1/3 bias (or 1/2 bias for LCD_A controller);
    - 4-mux, 1/3 bias (or 1/2 bias for LCD_A controller).

LCD Controller Introduction (2/3)

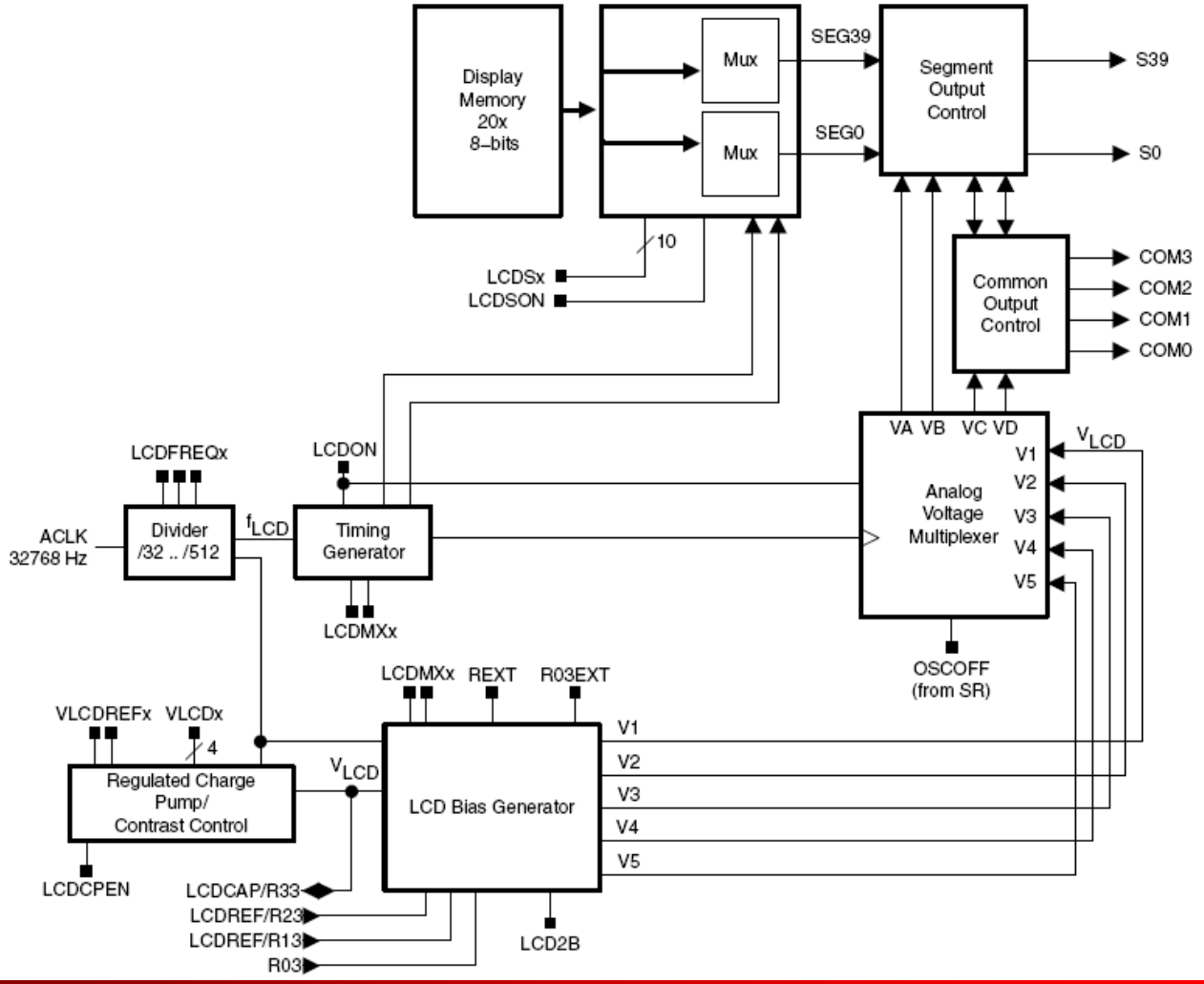
Main differences between LCD and LCD_A controllers:

- **LCD controller:**
  - Requires external circuitry (a resistor-divider network) to generate the 4 externally supplied voltage levels (R03, R13, R23 and R33) that supply the voltage generator;
  - Uses the timing generator derived from Basic Timer 1.

- **LCD_A controller:**
  - Similar features as LCD controller, but in addition:
    - Regulated charge pump and contrast control by software;
    - Fractional LCD biasing voltages (sourced internally or externally);
    - Uses the ACLK to generate the timing for common and segment lines.

LCD Controller Introduction (3/3)

❑ LCD_A controller block diagram:



LCD_A Controller Operation (1/7)

- **Can be configured to:**
  - Use external circuitry to generate the 4 externally supplied voltage levels (R03, R13, R23 and R33), which supply the voltage generator;
  - Use the internal LCD Bias Generator to generate the fractional LCD biasing voltages, V2 – V5 independent of the source for V_{LCD}:

    R33 → V1: full-scale voltage (V_{LCD});
    R23 → V2: 2/3 of full scale;
    V3: 1/2 of full scale;
    R13 → V4: 1/3 of full scale;
    R03 → V5: ground.

LCD_A Controller Operation (2/7)

- **LCD_A voltage and bias generation:**
  - Both the peak output waveform voltage V1, as well as the fractional LCD biasing voltages V2 – V5 can be sourced externally:
    - OSCOFF = 0: Oscillator sourcing ACLK off;
    - LCDON = 0: LCD_A module in inactive;
  - To use the internal voltage generation:
    - OSCOFF = 1: Oscillator sourcing ACLK set;
    - LCDON = 1: LCD_A module active;
    - V_{LCD} may be sourced internally from AV_{CC} or by an internal charge pump.

LCD_A Controller Operation (3/7)

- **LCD voltage and biasing characteristics LCD_A controller:**

Mode	Bias conf.	LCD_A controller							
		LCDMx	LCD2B	COM	V1	V2	V3	V4	V5
Static	Static	00	X	1	X				X
2-mux	1/2	01	1	2	X		X		X
2-mux	1/3	01	0	2	X	X		X	X
3-mux	1/2	10	1	3	X		X		X
3-mux	1/3	10	0	3	X	X		X	X
4-mux	1/2	11	1	4	X		X		X
4-mux	1/3	11	0	4	X	X		X	X

- **The LCD_A controller uses the ACLK (32768 Hz) prescaler selected using the LCDFREQx bits;**
- **LCD frequency, f_{LCD}, depends on:**
  - Framing frequency, f_{frame};
  - Multiplex rate, mux (defined on the LCD specifications).
- **LCD frequency: $f_{LCD} = 2 \times mux \times f_{frame}$**

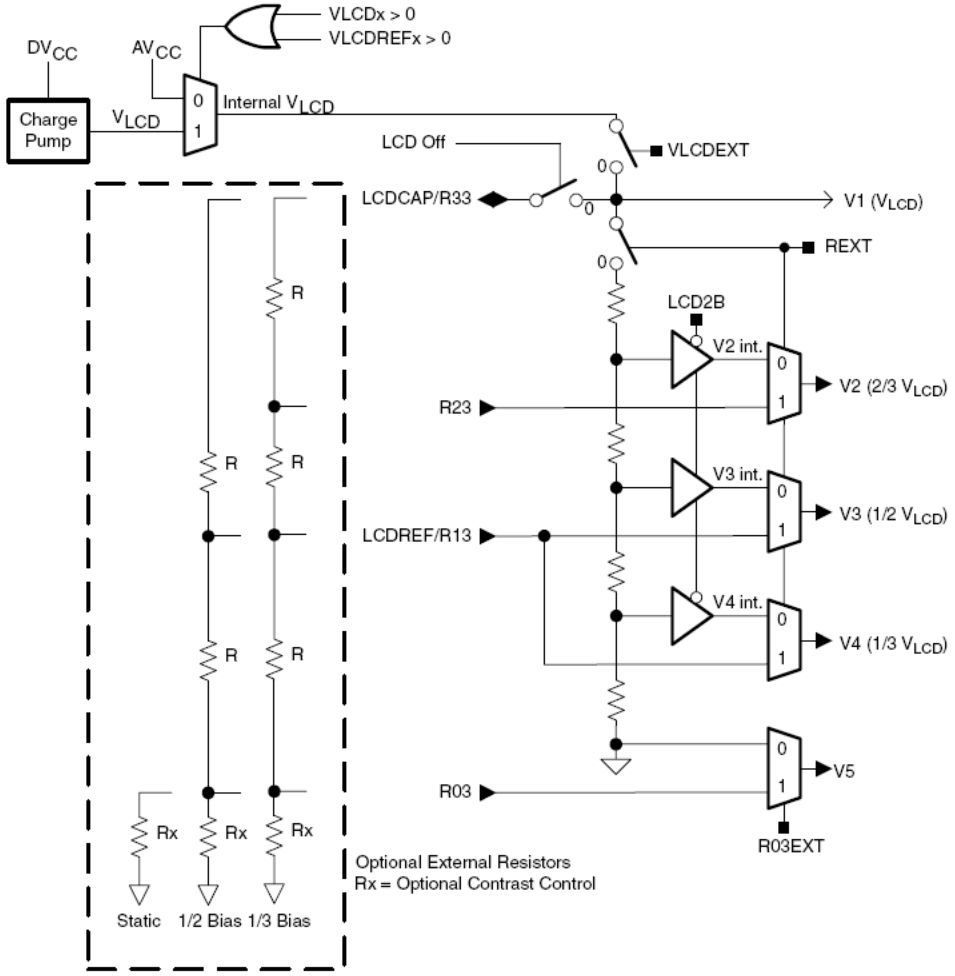
LCD_A Controller Operation (4/7)

LCD_A voltage selection:

- V_{LCD} source:
  - AV_{CC} requires:
    - VLCDEXT = 0;
    - VLCDx = 0;
    - VREFx = 0.
  - Internal charge pump sourced from DV_{CC} requires:
    - VLCDEXT = 0;
    - VLCDPEN = 1;
    - VLCDx > 0 (software selectable LCD voltage from 2.60 V to 3.44 V (typical), independent of DV_{CC};
    - Connect a 4.7 μF capacitor between LCDCAP pin and ground.

LCD_A Controller Operation (5/7)

❑ LCD_A Bias Generation block diagram:



LCD_A Controller Operation (6/7)

LCD_A Bias Generation:

- External fractional LCD biasing voltages, V2 – V5:
  - REXT = 1;
  - External equally weighted resistor divider (100 kΩ to 1 MΩ);
  - VLCDEXT = 0:
    - The V_{LCD} voltage is sourced from the internal charge pump, with R33 providing a switched-V_{LCD} output;
    - Otherwise (VLCDEXT = 1), R33 provides a V_{LCD} input.
  - R03EXT = 1: V5 is sourced externally.

LCD_A Controller Operation (7/7)

LCD_A Bias Generation:

- Internal bias generator:
  - When LCD2B = 1, supports 1/2 bias LCDs;
  - When LCD2B = 0, supports 1/3 bias LCDs in 2-mux, 3-mux, and 4-mux modes. In static mode, the internal divider network is disabled;
  - For LCD devices that share the LCDCAP, R33, and R23 functions, the charge pump cannot be used with an external resistor divider using 1/3 biasing;
  - When R03 is not available externally, V5 is always set to AV_{SS}.

LCD Modes (1/6)

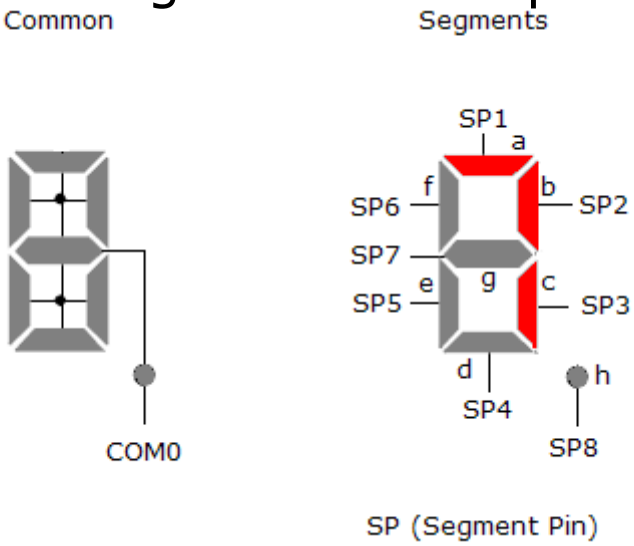
- **LCD_A controller supports 4 types of LCDs:**
  - **Static:**
    - Each MSP430 segment pin drives:
      - One LCD segment.
    - One common line driven by COM0.
    - Capacity to drive 32 segments.
  - **2-mux, 1/2 bias (or 1/3 bias):**
    - Each MSP430 segment pin drives:
      - Two LCD segments;
    - Two common lines driven by COM0 and COM1.
    - Capacity to drive 64 segments.

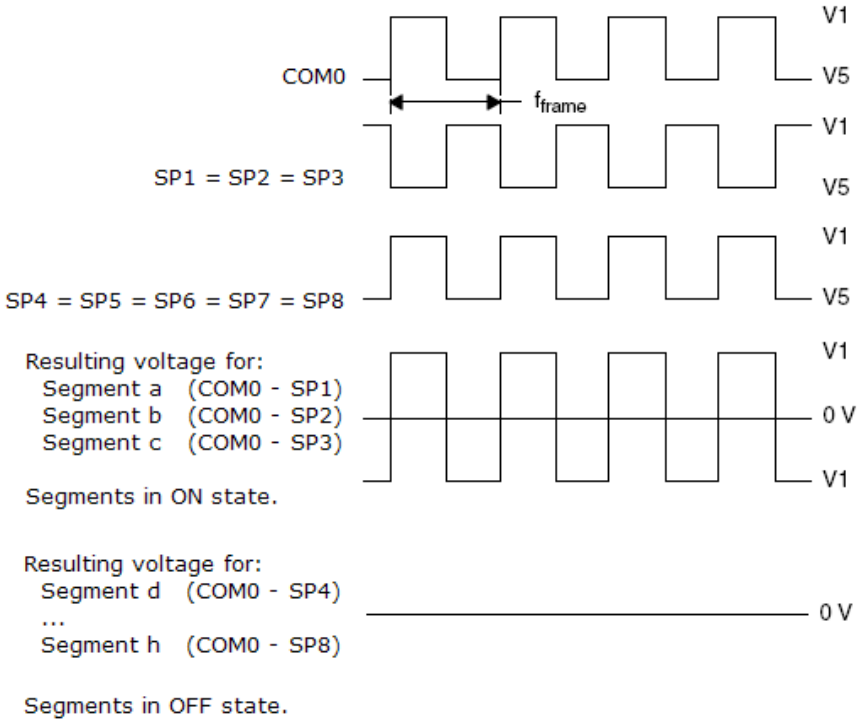
LCD Modes (2/6)

- **LCD_A controller supports 4 types of LCDs:**
  - **3-mux, 1/3 bias (or 1/2 bias):**
    - Each MSP430 segment pin drives:
      - Three LCD segments;
    - Three common lines driven by COM0, COM1, and COM2.
    - Capacity to drive 90 segments.
  - **4-mux, 1/3 bias (or 1/2 bias):**
    - Each MSP430 segment pin drives:
      - Four LCD segments;
    - Four common lines driven by COM0, COM1, COM2, and COM3.
    - Capacity to drive 120 segments.

LCD Modes (3/6)

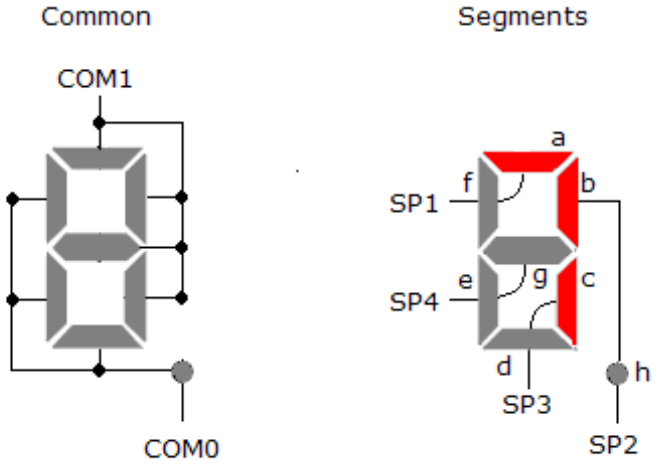
- **Static LCD:**
  - One pin for each segment;
  - One pin for the backplane.
  - Features:
    - High contrast ratio;
    - Large number of pins.

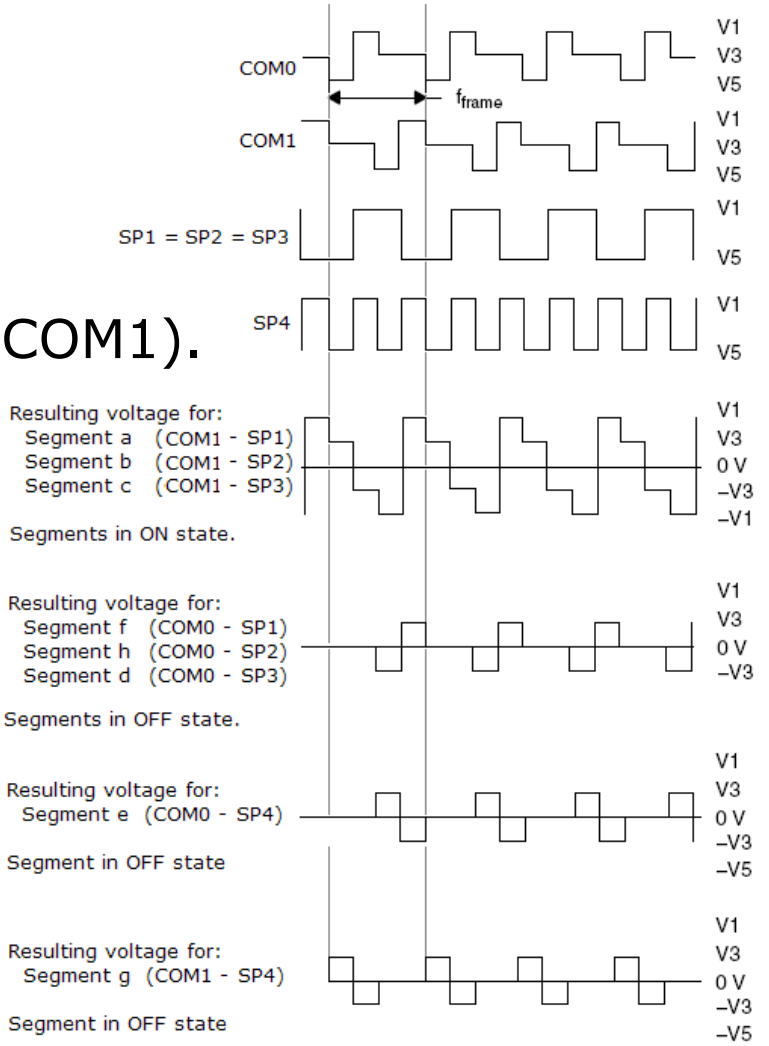




LCD Modes (4/6)

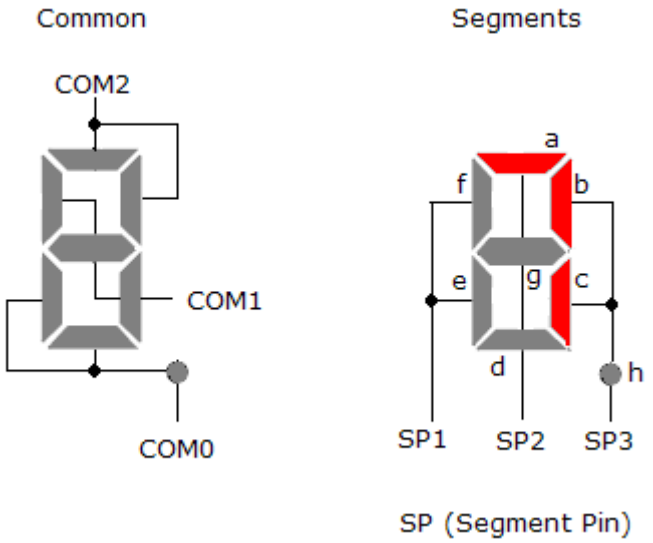
- **2-mux LCD:**
  - Reduced pin count;
  - LCD segments multiplexed:
    - Matrix of segments;
    - Two common pins (COM0 and COM1).
  - Example: 2-mux;

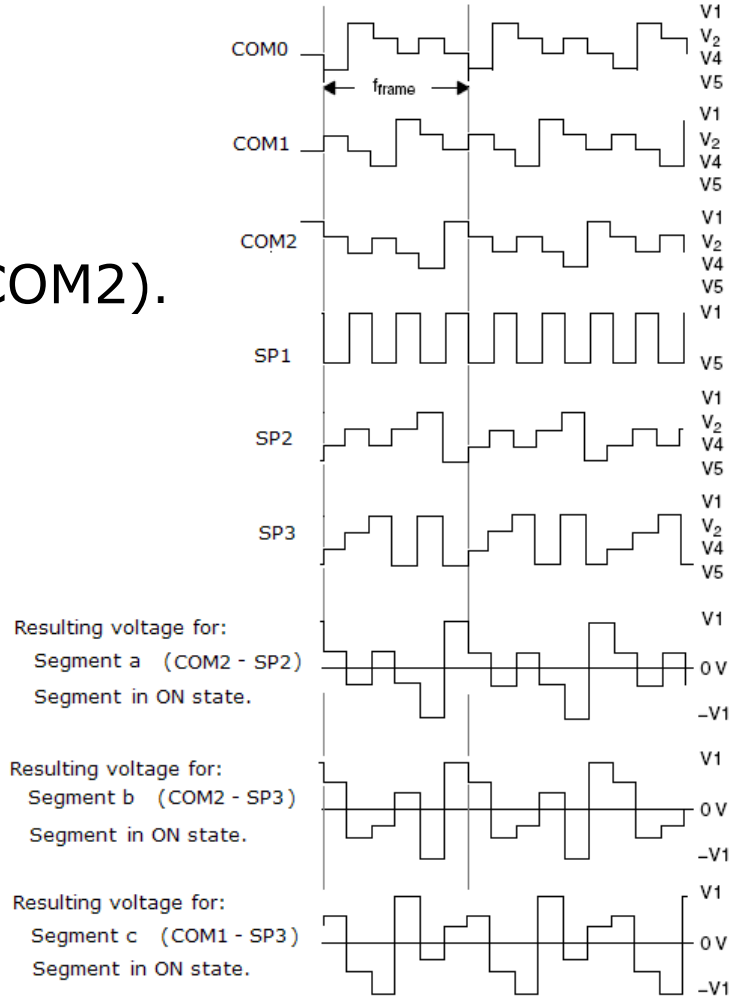




LCD Modes (5/6)

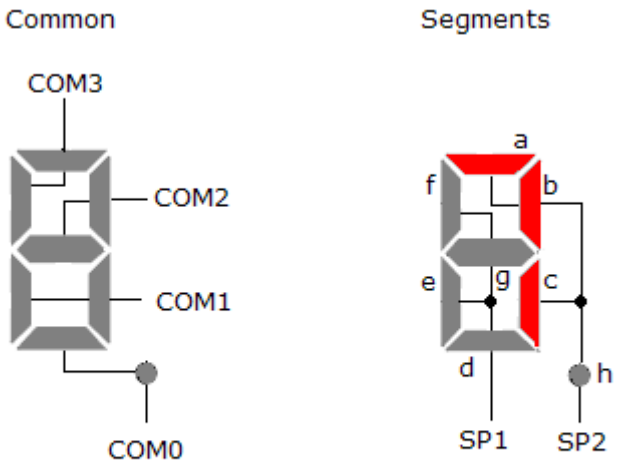
- **3-mux LCD**
  - 1 segment pin to drive:
    - 3 LCD segments;
    - 3 common lines (COM0 to COM2).
  - Example: 3-mux, 1/3 bias.



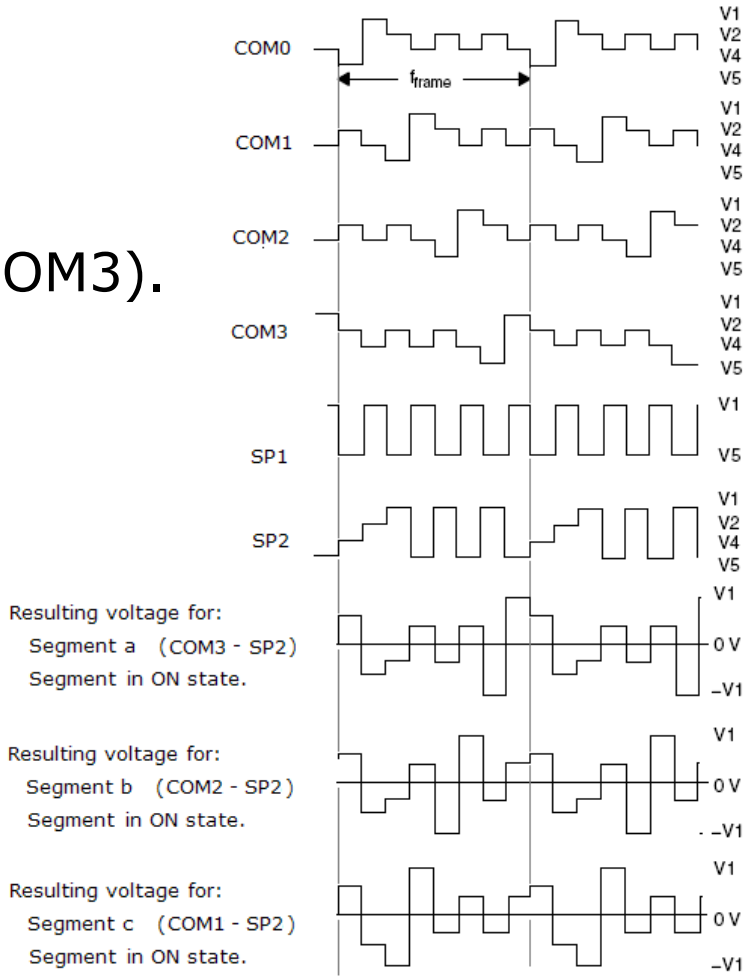


LCD Modes (6/6)

- **4-mux LCD**
  - 1 segment pin to drive:
    - 4 LCD segments;
    - 4 common lines (COM0 to COM3).
  - Example: 4-mux, 1/3 bias.



SP (Segment Pin)



LCD_A Controller Registers (1/4)

❑ LCDACTL, LCD_A Control Register

7	5	4	3	2	1	0
LCDFREQx		LCDMXx		LCDSON	Unused	LCDON

Bit		Description
7 - 5	LCDFREQx	LCD Frequency Select by ACLK divider configuration: LCDFREQ2 LCDFREQ1 LCDFREQ0 = 000 ⇒ ACLK / 32 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 001 ⇒ ACLK / 64 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 010 ⇒ ACLK / 96 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 011 ⇒ ACLK / 128 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 100 ⇒ ACLK / 192 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 101 ⇒ ACLK / 256 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 110 ⇒ ACLK / 384 LCDFREQ2 LCDFREQ1 LCDFREQ0 = 111 ⇒ ACLK / 512
4 - 3	LCDMXx	LCD mux rate for LCD mode setting: LCDMX1 LCDMX0 = 00 ⇒ Static LCDMX1 LCDMX0 = 01 ⇒ 2–mux LCDMX1 LCDMX0 = 10 ⇒ 3–mux LCDMX1 LCDMX0 = 11 ⇒ 4–mux
2	LCDSON	LCD segments on when LCDSON = 1.
0	LCDON	LCD_A module active when LCDON = 1.

LCD_A Controller Registers (2/4)

❑ LCDAPCTL1, LCD_A Port Control Register 1

7	...	2	1	0
Unused			LCDS36	LCDS32

Bit		Description
1	LCDS36	LCD Segment 36 to 39 Enable.
0	LCDS32	LCD Segment 32 to 35 Enable.

❑ LCDAPCTL0, LCD_A Port Control Register 0

7	6	5	4	3	2	1	0
LCDS28	LCDS24	LCDS20	LCDS16	LCDS12	LCDS8	LCDS4	LCDS0

Bit		Description
7	LCDS28	LCD Segment 28 to 31 Enable.
6	LCDS24	LCD Segment 24 to 27 Enable.
5	LCDS20	LCD Segment 20 to 23 Enable.
4	LCDS16	LCD Segment 16 to 19 Enable.
3	LCDS12	LCD Segment 12 to 15 Enable.
2	LCDS8	LCD Segment 8 to 11 Enable.
1	LCDS4	LCD Segment 4 to 7 Enable.
0	LCDS0	LCD Segment 0 to 3 Enable.

LCD_A Controller Registers (3/4)

❑ LCDAVCTL0, LCD_A Voltage Control Register 0

7	6	5	4	3	2	1	0
Unused	R03EXT	REXT	VLCDEXT	LCDCPEN	VLCDREFx		LCD2B

Bit		Description
6	R03EXT	V5 voltage select: R03EXT = 0 ⇒ V5 is AV_{SS}. R03EXT = 1 ⇒ V5 is sourced from the R03 pin.
5	REXT	V2 – V4 voltage source selection: REXT = 0 ⇒ V2 – V4 are generated internally. REXT = 1 ⇒ V2 – V4 are sourced externally
4	VLCDEXT	V_{LCD} source select: VLCDEXT = 0 ⇒ V_{LCD} is generated internally. VLCDEXT = 1 ⇒ V_{LCD} is generated externally.
3	LCDCPEN	Charge pump enable: LCDCPEN = 0 ⇒ Charge pump disabled. LCDCPEN = 1 ⇒ Charge pump enabled when VLCDEXT = 0 and VLCDx > 0 or VLCDREFx > 0.
2 – 1	VLCDREFx	Charge pump reference select: VLCDREF1 VLCDREF0 = 00 ⇒ Internal. VLCDREF1 VLCDREF0 = 01 ⇒ External. VLCDREF1 VLCDREF0 = 10 ⇒ Reserved. VLCDREF1 VLCDREF0 = 11 ⇒ Reserved.
0	LCD2B	Bias select (for any of the mux modes): LCD2B = 0 ⇒ 1/3 bias. LCD2B = 1 ⇒ 1/2 bias.

LCD_A Controller Registers (4/4)

LCDAVCTL1, LCD_A Voltage Control Register 1

7	5	4	3	2	1	0
Unused		VLCDx				Unused

Bit		Description		
4 - 1	VLCDx	Charge pump voltage select:		
		VLCD3 VLCD2 VLCD1 VLCD0 = 0000	⇒	Disable.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0001	⇒	V_{LCD} = 2.60 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0010	⇒	V_{LCD} = 2.66 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0011	⇒	V_{LCD} = 2.72 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0100	⇒	V_{LCD} = 2.78 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0101	⇒	V_{LCD} = 2.84 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0110	⇒	V_{LCD} = 2.90 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 0111	⇒	V_{LCD} = 2.96 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1000	⇒	V_{LCD} = 3.02 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1001	⇒	V_{LCD} = 3.08 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1010	⇒	V_{LCD} = 3.14 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1011	⇒	V_{LCD} = 3.20 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1100	⇒	V_{LCD} = 3.26 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1101	⇒	V_{LCD} = 3.32 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1110	⇒	V_{LCD} = 3.38 V.
		VLCD3 VLCD2 VLCD1 VLCD0 = 1111	⇒	V_{LCD} = 3.44 V.