

Работа с LCD-дисплеем.

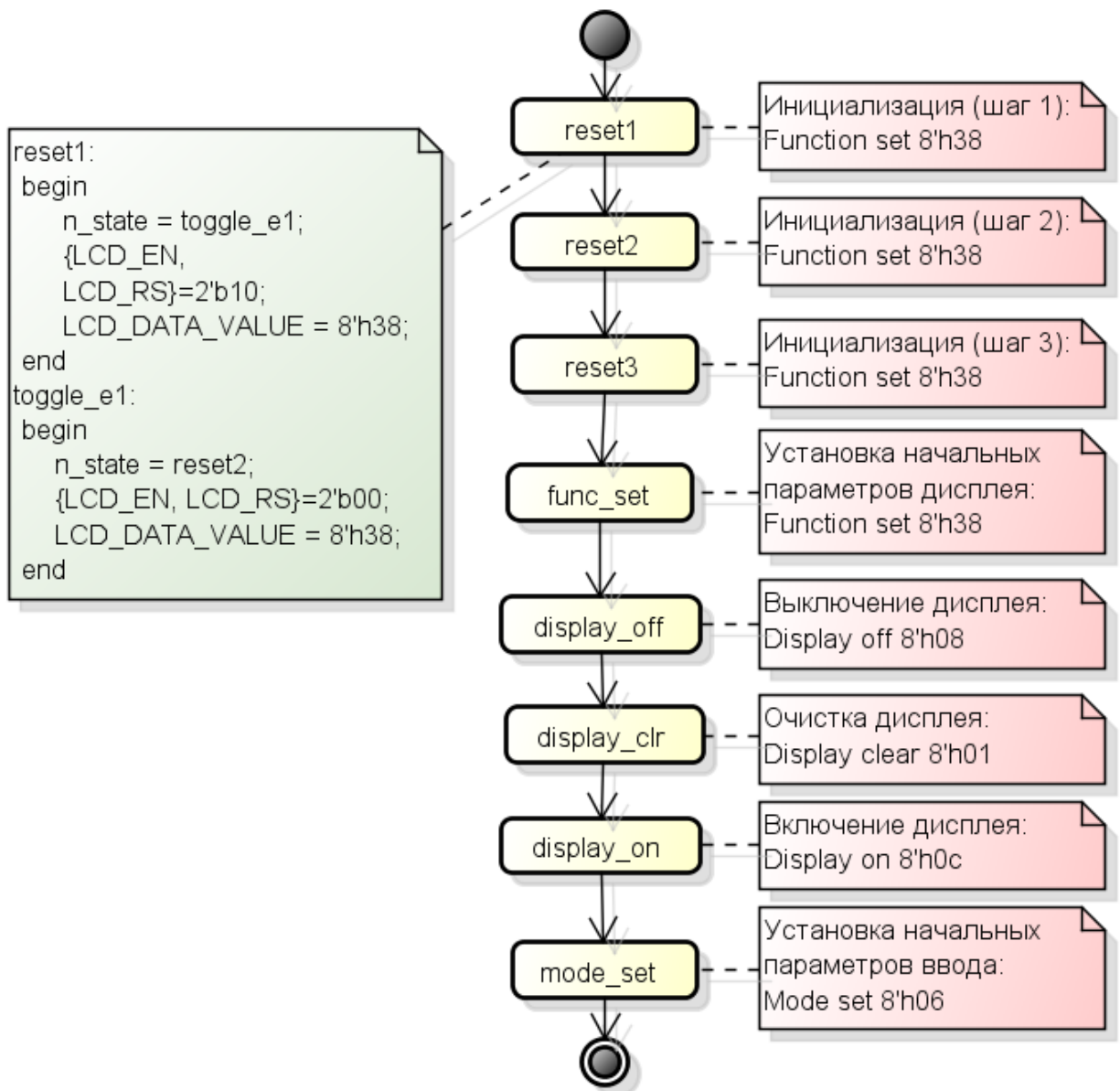
В этой работе рассмотрена программа, производящая вывод в заданную позицию символического ЖКИ с контроллером HD44780 нужный символ. Для управления выводом используются **переключатели**:

- 1) SW[15] – используется для задания номера строки дисплея, в которую производится вывод;
- 2) SW[14..8] – используются для задания позиции вывода в выбранной строке;
- 3) SW[7..0] – используются для задания кода символа, который требуется вывести.

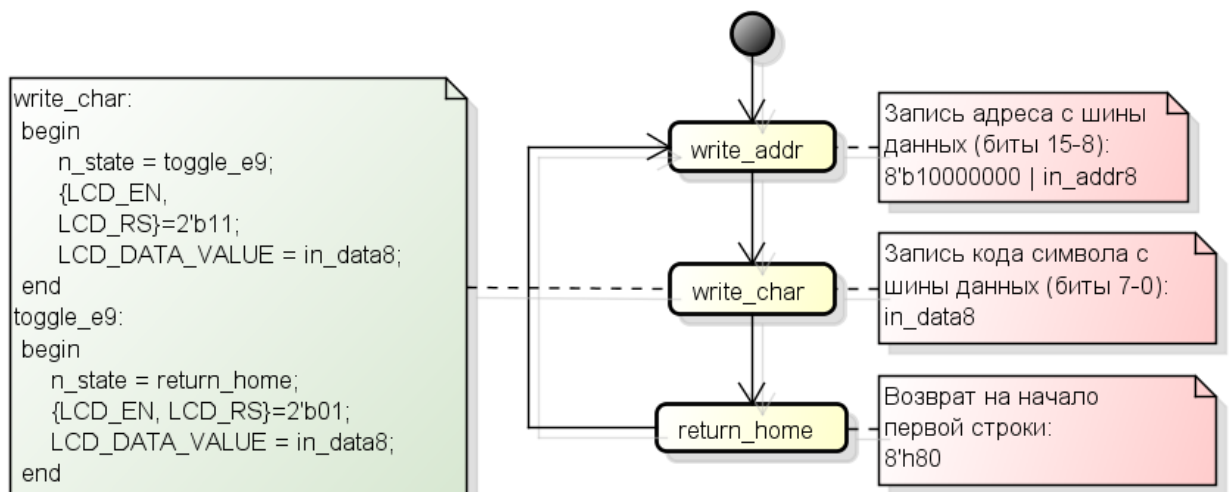
Для работы с дисплеем необходимо использовать следующие **команды**:

Название инструкции	Состояние выводов										Описание инструкции
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Clear display	0	0	0	0	0	0	0	0	0	1	Очистка всего ЖКИ и установка адреса DDRAM в 0
Return home	0	0	0	0	0	0	0	0	1	X	Установка текущего адреса DDRAM в 0 (курсор – домой), данные DDRAM не меняются
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Установка направления движения курсора (I/D: адрес увеличивается/уменьшается) и смещения дисплея (S: сдвиг разрешен/запрещен) при выводе данных
Display on/off control	0	0	0	0	0	0	1	D	C	B	Вкл/выкл. дисплей (D), курсор (C) и его мерцание (B)
Cursor or display shift	0	0	0	0	0	1	S/C	R/L	X	X	Двигает курсор и смещает дисплей (S/C: смещает дисплей/курсор) по DDRAM (R/L: вправо/влево)
Function set	0	0	0	0	1	DL	N	F	X	X	Установка интерфейса (DL: 8 бит/4 бит), числа строк (N: 2/1) и шрифта символов (F: 5x11/8)
Set CGRAM address	0	0	0	1	ACG5	ACG4	ACG3	ACG2	ACG1	ACG0	Установка счетчика адреса CGRAM
Set DDRAM address	0	0	1	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	Установка счетчика адреса DDRAM
Read busy flag & address	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Если BF = 1 то контроллер ЖКИ выполняет внутреннюю операцию (занят). AC6-AC0 – текущее значение адреса DDRAM
Write data to RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Запись данных в RAM
Read data from RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Чтение данных из RAM

Для **инициализации дисплея** и его начальной настройки необходимо выполнить следующую последовательность команд:



Затем в цикле происходит **считывание и вывод** нужного символа в заданную позицию дисплея:



Дополнительное задание: определите, какие параметры дисплея были установлены в ходе инициализации.

После того, как дисплей настроен и выбрана нужная позиция для вывода символа, необходимо задать **код символа**, который требуется вывести:

Lower 4 Bits \ Upper 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)			0	1	P	`	P				-	9	3	α	ρ
xxxx0001	(2)		!	1	A	Q	a	q			。	ア	チ	△	ä	g
xxxx0010	(3)		"	2	B	R	b	r			「	イ	ツ	×	β	θ
xxxx0011	(4)		#	3	C	S	c	s			」	ウ	テ	ε	ε	∞
xxxx0100	(5)		\$	4	D	T	d	t			、	エ	ト	フ	μ	Ω
xxxx0101	(6)		%	5	E	U	e	u			・	オ	ナ	1	ε	ü
xxxx0110	(7)		&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ
xxxx0111	(8)		'	7	G	W	g	w			ア	キ	ヌ	ラ	g	π
xxxx1000	(1)		(8	H	X	h	x			イ	ク	ネ	リ	γ	×
xxxx1001	(2))	9	I	Y	i	y			ウ	ケ	ル	ル	'	γ
xxxx1010	(3)		*	:	J	Z	j	z			エ	コ	ン	レ	j	¥
xxxx1011	(4)		+	;	K	L	k	l			オ	サ	ヒ	ロ	*	¥
xxxx1100	(5)		,	<	L	¥	l	l			カ	シ	フ	ワ	φ	¥
xxxx1101	(6)		-	=	M	J	m	j			ユ	ズ	ン	ン	も	÷
xxxx1110	(7)		.	>	N	^	n	→			ヨ	セ	ホ	°	π	
xxxx1111	(8)		/	?	O	_	o	←			ッ	ッ	マ	°	ö	■

Дополнительное задание: определите, какую последовательность 16-чных чисел необходимо подать дисплею, чтобы на экране появилась строка: «RoboSchool - 2015».

Программа, выполняющая все вышеуказанные действия, представлена в проекте **disp1602simple**. Для его запуска необходимо воспользоваться САПР QUARTUS II v13.1.

В файле **counter_divider.v** представлено описание модуля счетчика – делителя входной частоты в 10 раз:

```
module counter_divider (CLOCK, RESET, OUT); ...
```

Здесь **CLOCK** – это входной сигнал, определяющий исходный тактовый импульс;
RESET – входной сигнал, производящий сброс счетчика;
OUT – выходной сигнал, соответствующий новому тактовому импульсу.

Дополнительное задание: определите, какое значение должен иметь параметр `half_period_m1`, чтобы частота делилась не в 10 раз, а в 50.

В файле **fsm_lcd.v** представлено описание модуля непосредственной работы с дисплеем:

```
module fsm_lcd (CLK_400Hz, RESETN, IN_DATA8, IN_ADDR8, LCD_ON, LCD_RS, LCD_EN, LCD_RW, LCD_DATA);
```

Здесь **CLK_400Hz** – это входной тактовый сигнал;
RESETN – входной сигнал, производящий сброс счетчика;
IN_DATA8 – входная шина, задающая код выводимого символа;
IN_ADDR8 – входная шина, задающая адрес вывода символа;
LCD_ON – выходной сигнал, управляет подачей питания на дисплей;
LCD_RS – выходной сигнал выбора регистра;
LCD_EN – выходной сигнала строба чтения/записи;
LCD_RW – выходной сигнал выбора операции;
LCD_DATA – выходная шина, содержащая данные для вывода на дисплей.

В файле **disp1602back.bdf** представлена схема соединения компонентов. Входы и выходы, соответствующие конкретным выводам ПЛИС, имеют соответствующие названия.

Дополнительное задание: дополните схему так, чтобы код символа, выставляемый на переключателях, отображался на соответствующих диодах (**LEDR[..]** или **LEDG[..]**).

Для **запуска программы** необходимо открыть проект в среде QUARTUS (File – Open Project), скомпилировать его (Processing – Start Compilation), затем выполнить прошивку ПЛИС (Tools – Programmer – Start). Если в списке нет доступных устройств, необходимо проверить подключение платы к компьютеру, обновить драйвера или обратиться к преподавателю ☺.

Дополнительное задание: вывести на дисплей надпись « RoboSchool »
« *2015* ».

Дополнительное задание: после этого вывести на дисплей ФИО любого из участников команды.
