

MHVLib

20111011

Generated by Doxygen 1.7.5.1

Tue Oct 11 2011 22:26:37

Contents

1	Class Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	File Index	5
3.1	File List	5
4	Class Documentation	7
4.1	MHV_ADC Class Reference	7
4.1.1	Detailed Description	7
4.1.2	Constructor & Destructor Documentation	8
4.1.2.1	MHV_ADC	8
4.1.3	Member Function Documentation	8
4.1.3.1	adc	8
4.1.3.2	asyncRead	8
4.1.3.3	busyRead	8
4.1.3.4	deregisterListener	8
4.1.3.5	disable	9
4.1.3.6	enable	9
4.1.3.7	handleEvents	9
4.1.3.8	registerListener	9
4.1.3.9	setPrescaler	9
4.1.4	Member Data Documentation	9
4.1.4.1	_adcChannel	9

4.1.4.2	_adcCount	9
4.1.4.3	_adcs	10
4.1.4.4	_adcValue	10
4.2	MHV_ADCListener Class Reference	10
4.2.1	Detailed Description	10
4.2.2	Member Function Documentation	10
4.2.2.1	adc	10
4.3	mhv_alarm Struct Reference	10
4.3.1	Detailed Description	11
4.3.2	Member Data Documentation	11
4.3.2.1	listener	11
4.3.2.2	repeat	11
4.3.2.3	when	11
4.4	MHV_AlarmListener Class Reference	11
4.4.1	Detailed Description	11
4.4.2	Member Function Documentation	12
4.4.2.1	alarm	12
4.5	MHV_Debounce Class Reference	12
4.5.1	Detailed Description	13
4.5.2	Constructor & Destructor Documentation	13
4.5.2.1	MHV_Debounce	13
4.5.3	Member Function Documentation	13
4.5.3.1	assignKey	13
4.5.3.2	checkHeld	14
4.5.3.3	deassignKey	14
4.5.3.4	initPin	14
4.5.3.5	pinChanged	14
4.5.4	Member Data Documentation	14
4.5.4.1	_debounceTime	14
4.5.4.2	_heldTime	15
4.5.4.3	_pinChangeManager	15
4.5.4.4	_pins	15
4.5.4.5	_repeatTime	15
4.5.4.6	_rtc	15

4.6	MHV_DebounceListener Class Reference	15
4.6.1	Detailed Description	15
4.6.2	Member Function Documentation	16
4.6.2.1	heldDown	16
4.6.2.2	singlePress	16
4.7	mhv_debouncePin Struct Reference	16
4.7.1	Detailed Description	16
4.7.2	Member Data Documentation	16
4.7.2.1	held	16
4.7.2.2	listener	16
4.7.2.3	previous	16
4.7.2.4	timestamp	17
4.8	MHV_Device_RX Class Reference	17
4.8.1	Detailed Description	17
4.8.2	Constructor & Destructor Documentation	18
4.8.2.1	MHV_Device_RX	18
4.8.3	Member Function Documentation	18
4.8.3.1	asyncReadLine	18
4.8.3.2	busyReadLine	18
4.8.3.3	deregisterListener	18
4.8.3.4	flush	18
4.8.3.5	handleEvents	19
4.8.3.6	read	19
4.8.3.7	ready	19
4.8.3.8	registerListener	19
4.8.4	Member Data Documentation	19
4.8.4.1	_listener	19
4.8.4.2	_rxBuffer	19
4.9	MHV_Device_TX Class Reference	20
4.9.1	Detailed Description	20
4.9.2	Constructor & Destructor Documentation	21
4.9.2.1	MHV_Device_TX	21
4.9.3	Member Function Documentation	21
4.9.3.1	canWrite	21

4.9.3.2	moreTX	21
4.9.3.3	nextCharacter	21
4.9.3.4	runTxBuffers	21
4.9.3.5	write	22
4.9.3.6	write	22
4.9.3.7	write	22
4.9.3.8	write	23
4.9.3.9	write_P	23
4.9.3.10	write_P	23
4.9.4	Member Data Documentation	24
4.9.4.1	_currentTx	24
4.9.4.2	_tx	24
4.9.4.3	_txPointers	24
4.10	MHV_Display_Character Class Reference	24
4.10.1	Detailed Description	25
4.10.2	Constructor & Destructor Documentation	25
4.10.2.1	MHV_Display_Character	25
4.10.3	Member Function Documentation	26
4.10.3.1	_readChar	26
4.10.3.2	_setCursor	26
4.10.3.3	_writeChar	26
4.10.3.4	clear	26
4.10.3.5	getHeight	26
4.10.3.6	getWidth	26
4.10.3.7	runTxBuffers	26
4.10.3.8	scrollVertically	26
4.10.3.9	setCursor	27
4.10.3.10	setScroll	27
4.10.3.11	setWrap	27
4.10.3.12	txAnimation	27
4.10.3.13	writeBuffer	28
4.10.3.14	writeBuffer_P	28
4.10.3.15	writeChar	28
4.10.3.16	writeString	29

4.10.3.17	writeString_P	29
4.10.4	Member Data Documentation	29
4.10.4.1	_colCount	29
4.10.4.2	_currentCol	29
4.10.4.3	_currentRow	29
4.10.4.4	_rowCount	30
4.10.4.5	_scroll	30
4.10.4.6	_txOffset	30
4.10.4.7	_wrap	30
4.11	MHV_Display_HD44780 Class Reference	30
4.11.1	Detailed Description	31
4.11.2	Constructor & Destructor Documentation	31
4.11.2.1	MHV_Display_HD44780	31
4.11.3	Member Function Documentation	32
4.11.3.1	_readChar	32
4.11.3.2	_setCursor	32
4.11.3.3	_setCursor	32
4.11.3.4	_writeChar	32
4.11.3.5	addressCGRAM	33
4.11.3.6	addressDDRAM	33
4.11.3.7	clear	33
4.11.3.8	control	33
4.11.3.9	delay	34
4.11.3.10	entryMode	34
4.11.3.11	function	34
4.11.3.12	init	34
4.11.3.13	isBusy	35
4.11.3.14	readByte	35
4.11.3.15	writeByte	35
4.11.3.16	writeCommand	35
4.11.4	Member Data Documentation	35
4.11.4.1	_animateTicks	35
4.11.4.2	_byteMode	35
4.11.4.3	_mustDelay	35

4.11.4.4	_ticks	35
4.12	MHV_Display_HD44780_Direct_Connect Class Reference	36
4.12.1	Detailed Description	37
4.12.2	Constructor & Destructor Documentation	37
4.12.2.1	MHV_Display_HD44780_Direct_Connect	37
4.12.2.2	MHV_Display_HD44780_Direct_Connect	38
4.12.3	Member Function Documentation	39
4.12.3.1	delay	39
4.12.3.2	init	39
4.12.3.3	isBusy	39
4.12.3.4	readByte	40
4.12.3.5	readNibble	40
4.12.3.6	setBacklight	40
4.12.3.7	setContrast	40
4.12.3.8	tickPWM	40
4.12.3.9	writeByte	41
4.12.3.10	writeNibble	41
4.12.4	Member Data Documentation	41
4.12.4.1	_brightness	41
4.12.4.2	_contrast	41
4.12.4.3	_controlOut	41
4.12.4.4	_controlPin	42
4.12.4.5	_dataDir	42
4.12.4.6	_dataIn	42
4.12.4.7	_dataMask	42
4.12.4.8	_dataOut	42
4.12.4.9	_dataPin	42
4.12.4.10	_visualOut	42
4.12.4.11	_visualPin	42
4.13	MHV_Display_HD44780_Shift_Register Class Reference	43
4.13.1	Detailed Description	44
4.13.2	Constructor & Destructor Documentation	44
4.13.2.1	MHV_Display_HD44780_Shift_Register	44
4.13.3	Member Function Documentation	45

4.13.3.1	delay	45
4.13.3.2	init	45
4.13.3.3	isBusy	45
4.13.3.4	pushBits	45
4.13.3.5	readByte	46
4.13.3.6	writeByte	46
4.13.4	Member Data Documentation	46
4.13.4.1	_clockOut	46
4.13.4.2	_clockPin	46
4.13.4.3	_dataOut	46
4.13.4.4	_dataPin	47
4.13.4.5	_enableOut	47
4.13.4.6	_enablePin	47
4.14	MHV_Display_Holtek_HT1632 Class Reference	47
4.14.1	Detailed Description	48
4.14.2	Constructor & Destructor Documentation	48
4.14.2.1	MHV_Display_Holtek_HT1632	48
4.14.3	Member Function Documentation	48
4.14.3.1	brightness	49
4.14.3.2	flush	49
4.14.3.3	getPixel	49
4.14.3.4	poweroff	49
4.14.3.5	poweron	49
4.14.3.6	setPixel	50
4.15	MHV_Display_Monochrome Class Reference	50
4.15.1	Detailed Description	51
4.15.2	Constructor & Destructor Documentation	51
4.15.2.1	MHV_Display_Monochrome	51
4.15.3	Member Function Documentation	51
4.15.3.1	clear	52
4.15.3.2	getHeight	52
4.15.3.3	getPixel	52
4.15.3.4	getWidth	52
4.15.3.5	runTxBuffers	52

4.15.3.6	setPixel	52
4.15.3.7	txAnimation	52
4.15.3.8	writeBuffer	53
4.15.3.9	writeBuffer_P	53
4.15.3.10	writeChar	54
4.15.3.11	writeSeperator	54
4.15.3.12	writeString	55
4.15.3.13	writeString_P	55
4.15.4	Member Data Documentation	56
4.15.4.1	_colCount	56
4.15.4.2	_rowCount	56
4.15.4.3	_txOffset	56
4.16	MHV_Display_Monochrome_Buffered Class Reference	56
4.16.1	Detailed Description	57
4.16.2	Constructor & Destructor Documentation	57
4.16.2.1	MHV_Display_Monochrome_Buffered	57
4.16.3	Member Function Documentation	57
4.16.3.1	getPixel	57
4.16.3.2	setPixel	57
4.16.4	Member Data Documentation	57
4.16.4.1	_frameBuffer	57
4.17	MHV_EEPROM Class Reference	57
4.17.1	Detailed Description	58
4.17.2	Constructor & Destructor Documentation	58
4.17.2.1	MHV_EEPROM	58
4.17.3	Member Function Documentation	58
4.17.3.1	busyRead	58
4.17.3.2	busyRead	58
4.17.3.3	busyWrite	59
4.17.3.4	isBusy	59
4.17.3.5	read	59
4.17.3.6	read	59
4.17.3.7	write	60
4.17.3.8	write	60

4.17.3.9	writeInterrupt	60
4.18	mhv_eventADC Struct Reference	61
4.18.1	Detailed Description	61
4.18.2	Member Data Documentation	61
4.18.2.1	channel	61
4.18.2.2	listener	61
4.19	mhv_eventPin Struct Reference	61
4.19.1	Detailed Description	62
4.19.2	Member Data Documentation	62
4.19.2.1	changed	62
4.19.2.2	listener	62
4.19.2.3	mask	62
4.19.2.4	pclnt	62
4.19.2.5	port	62
4.19.2.6	previous	62
4.20	mhv_font Struct Reference	62
4.20.1	Detailed Description	63
4.20.2	Member Data Documentation	63
4.20.2.1	charCount	63
4.20.2.2	columnBytes	63
4.20.2.3	firstChar	63
4.20.2.4	fontData	63
4.20.2.5	maxHeight	63
4.20.2.6	maxWidth	63
4.20.2.7	offsets	63
4.20.2.8	unknown	64
4.20.2.9	widths	64
4.21	MHV_HardwareSerial Class Reference	64
4.21.1	Detailed Description	65
4.21.2	Constructor & Destructor Documentation	65
4.21.2.1	MHV_HardwareSerial	65
4.21.3	Member Function Documentation	65
4.21.3.1	busy	65
4.21.3.2	busyWrite	66

4.21.3.3	busyWrite	66
4.21.3.4	busyWrite	66
4.21.3.5	busyWrite_P	66
4.21.3.6	busyWrite_P	66
4.21.3.7	canSendBusy	67
4.21.3.8	debug	67
4.21.3.9	echo	67
4.21.3.10	end	67
4.21.3.11	runTxBuffers	68
4.21.3.12	rx	68
4.21.3.13	setSpeed	68
4.21.3.14	tx	68
4.22	MHV_Lock Class Reference	68
4.22.1	Detailed Description	69
4.22.2	Constructor & Destructor Documentation	69
4.22.2.1	MHV_Lock	69
4.22.3	Member Function Documentation	69
4.22.3.1	check	69
4.22.3.2	obtain	69
4.22.3.3	release	69
4.23	MHV_PID Class Reference	69
4.23.1	Detailed Description	70
4.23.2	Constructor & Destructor Documentation	70
4.23.2.1	MHV_PID	70
4.23.3	Member Function Documentation	71
4.23.3.1	clampIntegral	71
4.23.3.2	compute	71
4.23.3.3	enable	71
4.23.3.4	setDirection	71
4.23.3.5	setOutputLimits	72
4.23.3.6	setTuning	72
4.23.4	Member Data Documentation	72
4.23.4.1	_enabled	72
4.23.4.2	_integral	72

4.23.4.3	_kD	72
4.23.4.4	_kl	72
4.23.4.5	_kP	72
4.23.4.6	_lastInput	73
4.23.4.7	_lastOutput	73
4.23.4.8	_reverse	73
4.23.4.9	_setpoint	73
4.23.4.10	outMax	73
4.23.4.11	outMin	73
4.24	mhv_pin Struct Reference	73
4.24.1	Detailed Description	74
4.24.2	Member Data Documentation	74
4.24.2.1	bit	74
4.24.2.2	dir	74
4.24.2.3	input	74
4.24.2.4	output	74
4.24.2.5	pclnt	74
4.25	MHV_PinChangeManager Class Reference	74
4.25.1	Detailed Description	75
4.25.2	Constructor & Destructor Documentation	75
4.25.2.1	MHV_PinChangeManager	75
4.25.3	Member Function Documentation	75
4.25.3.1	deregisterListener	75
4.25.3.2	handleEvents	75
4.25.3.3	pinChange	76
4.25.3.4	pinChange0	76
4.25.3.5	registerListener	76
4.25.4	Member Data Documentation	76
4.25.4.1	_pins	76
4.26	MHV_PinEventListener Class Reference	77
4.26.1	Detailed Description	77
4.26.2	Member Function Documentation	77
4.26.2.1	pinChanged	77
4.27	MHV_PWMMatrix Class Reference	77

4.27.1 Detailed Description	78
4.27.2 Constructor & Destructor Documentation	78
4.27.2.1 MHV_PWMMatrix	78
4.27.3 Member Function Documentation	79
4.27.3.1 tick	79
4.28 MHV_RingBuffer Class Reference	79
4.28.1 Detailed Description	79
4.28.2 Constructor & Destructor Documentation	79
4.28.2.1 MHV_RingBuffer	79
4.28.3 Member Function Documentation	80
4.28.3.1 append	80
4.28.3.2 append	80
4.28.3.3 consume	80
4.28.3.4 consume	80
4.28.3.5 copyLine	81
4.28.3.6 flush	81
4.28.3.7 full	81
4.28.3.8 full	81
4.28.3.9 length	81
4.28.3.10 peekHead	81
4.28.3.11 size	82
4.29 MHV_RTC Class Reference	82
4.29.1 Detailed Description	83
4.29.2 Constructor & Destructor Documentation	83
4.29.2.1 MHV_RTC	83
4.29.3 Member Function Documentation	83
4.29.3.1 addAlarm	83
4.29.3.2 alarmsPending	84
4.29.3.3 current	84
4.29.3.4 elapsed	84
4.29.3.5 handleEvents	84
4.29.3.6 removeAlarm	84
4.29.3.7 setTime	85
4.29.3.8 setTime	85

4.29.3.9	synchronise	85
4.29.3.10	tick	85
4.29.3.11	tick1ms	85
4.29.3.12	tick1msAndRunEvents	86
4.29.3.13	tickAndRunEvents	86
4.29.3.14	toTime	86
4.29.3.15	toTimestamp	86
4.29.4	Member Data Documentation	86
4.29.4.1	_alarmCount	86
4.29.4.2	_alarmMax	86
4.29.4.3	_alarms	87
4.29.4.4	_milliseconds	87
4.29.4.5	_ticks	87
4.29.4.6	_ticksPerMillisecond	87
4.29.4.7	_timer	87
4.29.4.8	_timestamp	87
4.29.4.9	_tzOffset	87
4.30	MHV_RXListener Class Reference	87
4.30.1	Detailed Description	88
4.30.2	Member Function Documentation	88
4.30.2.1	rxReady	88
4.31	mhv_time Struct Reference	88
4.31.1	Detailed Description	88
4.31.2	Member Data Documentation	88
4.31.2.1	day	88
4.31.2.2	hours	88
4.31.2.3	milliseconds	89
4.31.2.4	minutes	89
4.31.2.5	month	89
4.31.2.6	seconds	89
4.31.2.7	timezone	89
4.31.2.8	year	89
4.31.2.9	yday	89
4.32	MHV_Timer8 Class Reference	89

4.32.1	Detailed Description	91
4.32.2	Constructor & Destructor Documentation	91
4.32.2.1	MHV_Timer8	91
4.32.2.2	MHV_Timer8	91
4.32.3	Member Function Documentation	91
4.32.3.1	_setPrescaler	91
4.32.3.2	calculatePrescaler	91
4.32.3.3	calculateTop	91
4.32.3.4	connectOutput1	91
4.32.3.5	connectOutput2	92
4.32.3.6	current	92
4.32.3.7	disable	92
4.32.3.8	enable	92
4.32.3.9	enabled	92
4.32.3.10	getOutput	92
4.32.3.11	getOutput1	92
4.32.3.12	getOutput2	92
4.32.3.13	getPrescaler	92
4.32.3.14	getPrescalerMultiplier	92
4.32.3.15	getTop	93
4.32.3.16	setGenerationMode	93
4.32.3.17	setMode	93
4.32.3.18	setOutput	93
4.32.3.19	setOutput1	93
4.32.3.20	setOutput2	93
4.32.3.21	setPeriods	93
4.32.3.22	setPeriods	93
4.32.3.23	setPrescaler	93
4.32.3.24	setTop	93
4.32.3.25	setTriggers	94
4.32.3.26	trigger1	94
4.32.3.27	trigger2	94
4.32.4	Member Data Documentation	94
4.32.4.1	_controlRegA	94

4.32.4.2	_controlRegB	94
4.32.4.3	_counter	94
4.32.4.4	_counterSize	94
4.32.4.5	_haveTime2	94
4.32.4.6	_interrupt	94
4.32.4.7	_interruptEnableA	94
4.32.4.8	_mode	95
4.32.4.9	_outputCompare1	95
4.32.4.10	_outputCompare2	95
4.32.4.11	_prescaler	95
4.32.4.12	_triggerData1	95
4.32.4.13	_triggerData2	95
4.32.4.14	_triggerFunction1	95
4.32.4.15	_triggerFunction2	95
4.32.4.16	_type	95
4.33	mhv_timestamp Struct Reference	96
4.33.1	Detailed Description	96
4.33.2	Member Data Documentation	96
4.33.2.1	milliseconds	96
4.33.2.2	timestamp	96
4.34	mhv_tx_buffer Struct Reference	96
4.34.1	Detailed Description	96
4.34.2	Member Data Documentation	97
4.34.2.1	completeFunction	97
4.34.2.2	data	97
4.34.2.3	isString	97
4.34.2.4	length	97
4.34.2.5	progmem	97
4.35	MHV_VusbKeyboard Class Reference	97
4.35.1	Detailed Description	98
4.35.2	Constructor & Destructor Documentation	98
4.35.2.1	MHV_VusbKeyboard	98
4.35.3	Member Function Documentation	98
4.35.3.1	alarm	98

4.35.3.2	keyDown	98
4.35.3.3	keyStroke	99
4.35.3.4	keyStroke	99
4.35.3.5	keysUp	99
4.35.3.6	keysUp	100
4.35.4	Member Data Documentation	100
4.35.4.1	_rtc	100
4.36	MHV_VusbTypist Class Reference	100
4.36.1	Detailed Description	101
4.36.2	Constructor & Destructor Documentation	101
4.36.2.1	MHV_VusbTypist	101
4.36.3	Member Function Documentation	101
4.36.3.1	alarm	101
4.36.3.2	runTxBuffers	101
4.36.3.3	typeChar	101
4.36.4	Member Data Documentation	101
4.36.4.1	_isTyping	102
5	File Documentation	103
5.1	A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbKeyboard.cpp File Reference	103
5.1.1	Define Documentation	103
5.1.1.1	MHV_OSCCAL_EEPROM_ADDRESS	103
5.1.2	Function Documentation	103
5.1.2.1	usbFunctionSetup	104
5.1.3	Variable Documentation	104
5.1.3.1	usbHidReportDescriptor	104
5.2	A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbKeyboard.h File Reference	104
5.2.1	Define Documentation	106
5.2.1.1	MHV_MOD_ALT_LEFT	106
5.2.1.2	MHV_MOD_ALT_RIGHT	106
5.2.1.3	MHV_MOD_CONTROL_LEFT	106
5.2.1.4	MHV_MOD_CONTROL_RIGHT	106
5.2.1.5	MHV_MOD_GUI_LEFT	106

5.2.1.6	MHV_MOD_GUI_RIGHT	106
5.2.1.7	MHV_MOD_SHIFT_LEFT	106
5.2.1.8	MHV_MOD_SHIFT_RIGHT	107
5.2.2	Typedef Documentation	107
5.2.2.1	MHV_VUSB_KEYBOARD_KEY	107
5.2.3	Enumeration Type Documentation	107
5.2.3.1	mhv_vusb_keyboard_key	107
5.3	A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbTypist.cpp File Reference	111
5.4	A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbTypist.h File Reference	111
5.5	A:/eclipse/mhvlb/MHV_AD.cpp File Reference	111
5.5.1	Function Documentation	111
5.5.1.1	mhv_ad_asyncRead	111
5.5.1.2	mhv_ad_busyRead	111
5.5.1.3	mhv_ad_setPrescaler	111
5.6	A:/eclipse/mhvlb/MHV_AD.h File Reference	112
5.6.1	Define Documentation	112
5.6.1.1	MHV_AD_ASSIGN_INTERRUPT	112
5.6.1.2	MHV_AD_CHANNEL	113
5.6.1.3	MHV_AD_DISABLE	113
5.6.1.4	MHV_AD_DISABLE_INTERRUPT	113
5.6.1.5	MHV_AD_ENABLE	113
5.6.1.6	MHV_AD_ENABLE_INTERRUPT	113
5.6.1.7	MHV_AD_MAX	113
5.6.1.8	MHV_AD_REFERENCE	113
5.6.2	Typedef Documentation	113
5.6.2.1	MHV_AD_PRESCALER	113
5.6.3	Enumeration Type Documentation	114
5.6.3.1	mhv_ad_prescaler	114
5.6.4	Function Documentation	114
5.6.4.1	mhv_ad_asyncRead	114
5.6.4.2	mhv_ad_busyRead	114
5.6.4.3	mhv_ad_setPrescaler	114
5.7	A:/eclipse/mhvlb/MHV_ADC.cpp File Reference	114
5.8	A:/eclipse/mhvlb/MHV_ADC.h File Reference	114

5.8.1	Define Documentation	115
5.8.1.1	MHV_ADC_ASSIGN_INTERRUPT	115
5.8.1.2	MHV_ADC_BUFFER_CREATE	115
5.8.2	Typedef Documentation	115
5.8.2.1	MHV_EVENT_ADC	115
5.9	A:/eclipse/mhvlb/MHV_Debounce.cpp File Reference	115
5.10	A:/eclipse/mhvlb/MHV_Debounce.h File Reference	115
5.10.1	Define Documentation	116
5.10.1.1	MHV_DEBOUNCE_ASSIGN_INTERRUPTS	116
5.10.1.2	MHV_DEBOUNCE_ASSIGN_PCINT	116
5.10.1.3	MHV_DEBOUNCE_ASSIGN_PCINT0	116
5.10.1.4	MHV_DEBOUNCE_ASSIGN_PCINT1	117
5.10.1.5	MHV_DEBOUNCE_ASSIGN_PCINT2	117
5.10.2	Typedef Documentation	117
5.10.2.1	MHV_DEBOUNCE_PIN	117
5.11	A:/eclipse/mhvlb/MHV_Device_RX.cpp File Reference	117
5.12	A:/eclipse/mhvlb/MHV_Device_RX.h File Reference	117
5.12.1	Define Documentation	118
5.12.1.1	MHV_RX_BUFFER_CREATE	118
5.13	A:/eclipse/mhvlb/MHV_Device_TX.cpp File Reference	118
5.14	A:/eclipse/mhvlb/MHV_Device_TX.h File Reference	118
5.14.1	Define Documentation	119
5.14.1.1	MHV_TX_BUFFER_CREATE	119
5.14.1.2	MHVLIB_NEED_PURE_VIRTUAL	119
5.14.2	Typedef Documentation	119
5.14.2.1	MHV_TX_BUFFER	119
5.15	A:/eclipse/mhvlb/MHV_Display_Character.cpp File Reference	119
5.16	A:/eclipse/mhvlb/MHV_Display_Character.h File Reference	119
5.17	A:/eclipse/mhvlb/MHV_Display_HD44780.cpp File Reference	119
5.17.1	Define Documentation	120
5.17.1.1	HD44780_TINIT	120
5.18	A:/eclipse/mhvlb/MHV_Display_HD44780.h File Reference	120
5.18.1	Typedef Documentation	120
5.18.1.1	MHV_HD44780_COMMAND	120

5.18.2	Enumeration Type Documentation	120
5.18.2.1	mhv_hd44780_command	120
5.19	A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.cpp File - Reference	121
5.19.1	Define Documentation	121
5.19.1.1	HD44780_CONTRAST	121
5.19.1.2	HD44780_DB4	122
5.19.1.3	HD44780_DB5	122
5.19.1.4	HD44780_DB6	122
5.19.1.5	HD44780_DB7	122
5.19.1.6	HD44780_E	122
5.19.1.7	HD44780_LED	122
5.19.1.8	HD44780_RS	122
5.19.1.9	HD44780_RW	122
5.19.1.10	HD44780_TC	122
5.19.1.11	HD44780_TCLEAR	122
5.19.1.12	HD44780_TDH	123
5.19.1.13	HD44780_TINIT	123
5.19.1.14	HD44780_TINSTR	123
5.19.1.15	HD44780_TRAM	123
5.19.1.16	HD44780_TSU1	123
5.19.1.17	HD44780_TSU2	123
5.19.1.18	HD44780_TW	123
5.20	A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.h File - Reference	123
5.21	A:/eclipse/mhvlb/MHV_Display_HD44780_Shift_Register.cpp File - Reference	124
5.21.1	Define Documentation	124
5.21.1.1	HD44780_TSU1	124
5.21.1.2	HD44780_TW	124
5.21.1.3	MHV_SHIFT_ORDER_MSB	124
5.21.1.4	MHV_SHIFT_WRITECLOCK	124
5.21.1.5	MHV_SHIFT_WRITEDATA	124
5.22	A:/eclipse/mhvlb/MHV_Display_HD44780_Shift_Register.h File - Reference	124

5.23	A:/eclipse/mhvlb/MHV_Display_Holtek_HT1632.cpp File Reference	125
5.23.1	Define Documentation	125
5.23.1.1	MHV_SHIFT_ORDER_MSB	125
5.23.1.2	MHV_SHIFT_WRITECLOCK	125
5.23.1.3	MHV_SHIFT_WRITEDATA	125
5.24	A:/eclipse/mhvlb/MHV_Display_Holtek_HT1632.h File Reference	125
5.24.1	Define Documentation	126
5.24.1.1	MHV_HT1632_BRIGHTNESS_MAX	126
5.24.1.2	MHV_HT1632_BRIGHTNESS_MED	126
5.24.1.3	MHV_HT1632_BRIGHTNESS_MIN	126
5.24.2	Typedef Documentation	126
5.24.2.1	MHV_HT1632_COMMAND	126
5.24.2.2	MHV_HT1632_MODE	126
5.24.3	Enumeration Type Documentation	127
5.24.3.1	mhv_ht1632_command	127
5.24.3.2	mhv_ht1632_mode	127
5.25	A:/eclipse/mhvlb/MHV_Display_Monochrome.cpp File Reference	127
5.26	A:/eclipse/mhvlb/MHV_Display_Monochrome.h File Reference	127
5.27	A:/eclipse/mhvlb/MHV_Display_Monochrome_Buffered.cpp File Reference	128
5.27.1	Define Documentation	128
5.27.1.1	pixel	128
5.28	A:/eclipse/mhvlb/MHV_Display_Monochrome_Buffered.h File Reference	128
5.29	A:/eclipse/mhvlb/MHV_EEPROM.cpp File Reference	128
5.30	A:/eclipse/mhvlb/MHV_EEPROM.h File Reference	128
5.30.1	Define Documentation	129
5.30.1.1	MHV_EEPROM_ASSIGN_INTERRUPTS	129
5.30.1.2	MHV_EEPROM_BUSY	129
5.30.1.3	MHV_EEPROM_CREATE	129
5.31	A:/eclipse/mhvlb/MHV_EPP.cpp File Reference	129
5.32	A:/eclipse/mhvlb/MHV_EPP.h File Reference	129
5.33	A:/eclipse/mhvlb/MHV_Font.h File Reference	129
5.33.1	Typedef Documentation	130
5.33.1.1	MHV_FONT	130

5.34	A:/eclipse/mhvlb/MHV_Font_SansSerif_10x8.h File Reference	130
5.35	A:/eclipse/mhvlb/MHV_GammaCorrect.cpp File Reference	130
5.35.1	Function Documentation	130
5.35.1.1	mhv_calculatedGammaCorrect	130
5.35.1.2	mhv_precalculatedGammaCorrect	130
5.35.2	Variable Documentation	131
5.35.2.1	PROGMEM	131
5.36	A:/eclipse/mhvlb/MHV_GammaCorrect.h File Reference	131
5.36.1	Define Documentation	131
5.36.1.1	MHV_PRECALCULATED_GAMMA_CORRECT	131
5.36.2	Function Documentation	131
5.36.2.1	mhv_calculatedGammaCorrect	131
5.36.2.2	mhv_precalculatedGammaCorrect	131
5.37	A:/eclipse/mhvlb/MHV_HardwareSerial.cpp File Reference	131
5.38	A:/eclipse/mhvlb/MHV_HardwareSerial.h File Reference	132
5.38.1	Define Documentation	132
5.38.1.1	_MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS	132
5.38.1.2	MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS	133
5.38.1.3	MHV_HARDWARESERIAL_CREATE	133
5.38.1.4	MHV_HARDWARESERIAL_DEBUG	133
5.39	A:/eclipse/mhvlb/MHV_io.h File Reference	134
5.39.1	Define Documentation	136
5.39.1.1	_mhv_bit	136
5.39.1.2	_mhv_declareExternalInterrupt	136
5.39.1.3	_mhv_dir	136
5.39.1.4	_mhv_enableExternalInterrupt	136
5.39.1.5	_mhv_in	136
5.39.1.6	_mhv_make_pin	136
5.39.1.7	_mhv_out	136
5.39.1.8	_mhv_PCInt	136
5.39.1.9	_mhv_pin	137
5.39.1.10	GCC_VERSION	137
5.39.1.11	mhv_bit	137

5.39.1.12	mhv_declareExternalInterrupt	137
5.39.1.13	mhv_dir	137
5.39.1.14	mhv_enableExternalInterrupt	138
5.39.1.15	mhv_in	138
5.39.1.16	MHV_IO_H	138
5.39.1.17	mhv_make_pin	138
5.39.1.18	mhv_out	139
5.39.1.19	mhv_pcint	139
5.39.1.20	mhv_pin	139
5.39.1.21	NORETURN	139
5.39.1.22	PURE	139
5.39.2	Typedef Documentation	139
5.39.2.1	MHV_INTERRUPTMODE	139
5.39.2.2	MHV_PIN	140
5.39.3	Enumeration Type Documentation	140
5.39.3.1	mhv_interruptMode	140
5.39.4	Function Documentation	140
5.39.4.1	mhv_memClear	140
5.39.4.2	mhv_memClear	140
5.39.4.3	mhv_pinOff	141
5.39.4.4	mhv_pinOff	141
5.39.4.5	mhv_pinOffAtomic	141
5.39.4.6	mhv_pinOffAtomic	141
5.39.4.7	mhv_pinOn	142
5.39.4.8	mhv_pinOn	142
5.39.4.9	mhv_pinOnAtomic	142
5.39.4.10	mhv_pinOnAtomic	142
5.39.4.11	mhv_pinRead	143
5.39.4.12	mhv_pinRead	143
5.39.4.13	mhv_pinSet	143
5.39.4.14	mhv_pinSetAtomic	144
5.39.4.15	mhv_pinToggle	144
5.39.4.16	mhv_pinToggle	144
5.39.4.17	mhv_pinToggleAtomic	144

5.39.4.18	mhv_pinToggleAtomic	145
5.39.4.19	mhv_setInput	145
5.39.4.20	mhv_setInput	145
5.39.4.21	mhv_setInputAtomic	146
5.39.4.22	mhv_setInputAtomic	146
5.39.4.23	mhv_setInputPullup	146
5.39.4.24	mhv_setInputPullup	146
5.39.4.25	mhv_setInputPullupAtomic	147
5.39.4.26	mhv_setInputPullupAtomic	147
5.39.4.27	mhv_setOutput	147
5.39.4.28	mhv_setOutput	147
5.39.4.29	mhv_setOutputAtomic	148
5.39.4.30	mhv_setOutputAtomic	148
5.40	A:/eclipse/mhvlb/MHV_io_ArduinoDiecimilla.h File Reference	148
5.40.1	Define Documentation	149
5.40.1.1	MHV_ARDUINO_PIN_0	149
5.40.1.2	MHV_ARDUINO_PIN_1	149
5.40.1.3	MHV_ARDUINO_PIN_10	149
5.40.1.4	MHV_ARDUINO_PIN_11	149
5.40.1.5	MHV_ARDUINO_PIN_12	149
5.40.1.6	MHV_ARDUINO_PIN_13	149
5.40.1.7	MHV_ARDUINO_PIN_2	150
5.40.1.8	MHV_ARDUINO_PIN_3	150
5.40.1.9	MHV_ARDUINO_PIN_4	150
5.40.1.10	MHV_ARDUINO_PIN_5	150
5.40.1.11	MHV_ARDUINO_PIN_6	150
5.40.1.12	MHV_ARDUINO_PIN_7	150
5.40.1.13	MHV_ARDUINO_PIN_8	150
5.40.1.14	MHV_ARDUINO_PIN_9	150
5.40.1.15	MHV_ARDUINO_PIN_A0	150
5.40.1.16	MHV_ARDUINO_PIN_A1	150
5.40.1.17	MHV_ARDUINO_PIN_A2	151
5.40.1.18	MHV_ARDUINO_PIN_A3	151
5.40.1.19	MHV_ARDUINO_PIN_A4	151

5.40.1.20 MHV_ARDUINO_PIN_A5	151
5.41 A:/eclipse/mhvlb/MHV_io_ArduinoMega.h File Reference	151
5.41.1 Define Documentation	153
5.41.1.1 MHV_ARDUINO_PIN_0	153
5.41.1.2 MHV_ARDUINO_PIN_1	153
5.41.1.3 MHV_ARDUINO_PIN_10	153
5.41.1.4 MHV_ARDUINO_PIN_11	153
5.41.1.5 MHV_ARDUINO_PIN_12	153
5.41.1.6 MHV_ARDUINO_PIN_13	153
5.41.1.7 MHV_ARDUINO_PIN_14	153
5.41.1.8 MHV_ARDUINO_PIN_15	153
5.41.1.9 MHV_ARDUINO_PIN_16	153
5.41.1.10 MHV_ARDUINO_PIN_17	153
5.41.1.11 MHV_ARDUINO_PIN_18	154
5.41.1.12 MHV_ARDUINO_PIN_19	154
5.41.1.13 MHV_ARDUINO_PIN_2	154
5.41.1.14 MHV_ARDUINO_PIN_20	154
5.41.1.15 MHV_ARDUINO_PIN_21	154
5.41.1.16 MHV_ARDUINO_PIN_22	154
5.41.1.17 MHV_ARDUINO_PIN_23	154
5.41.1.18 MHV_ARDUINO_PIN_24	154
5.41.1.19 MHV_ARDUINO_PIN_25	154
5.41.1.20 MHV_ARDUINO_PIN_26	154
5.41.1.21 MHV_ARDUINO_PIN_27	155
5.41.1.22 MHV_ARDUINO_PIN_28	155
5.41.1.23 MHV_ARDUINO_PIN_29	155
5.41.1.24 MHV_ARDUINO_PIN_3	155
5.41.1.25 MHV_ARDUINO_PIN_30	155
5.41.1.26 MHV_ARDUINO_PIN_31	155
5.41.1.27 MHV_ARDUINO_PIN_32	155
5.41.1.28 MHV_ARDUINO_PIN_33	155
5.41.1.29 MHV_ARDUINO_PIN_34	155
5.41.1.30 MHV_ARDUINO_PIN_35	155
5.41.1.31 MHV_ARDUINO_PIN_36	156

5.41.1.32 MHV_ARDUINO_PIN_37	156
5.41.1.33 MHV_ARDUINO_PIN_38	156
5.41.1.34 MHV_ARDUINO_PIN_39	156
5.41.1.35 MHV_ARDUINO_PIN_4	156
5.41.1.36 MHV_ARDUINO_PIN_40	156
5.41.1.37 MHV_ARDUINO_PIN_41	156
5.41.1.38 MHV_ARDUINO_PIN_42	156
5.41.1.39 MHV_ARDUINO_PIN_43	156
5.41.1.40 MHV_ARDUINO_PIN_44	156
5.41.1.41 MHV_ARDUINO_PIN_45	157
5.41.1.42 MHV_ARDUINO_PIN_46	157
5.41.1.43 MHV_ARDUINO_PIN_47	157
5.41.1.44 MHV_ARDUINO_PIN_48	157
5.41.1.45 MHV_ARDUINO_PIN_49	157
5.41.1.46 MHV_ARDUINO_PIN_5	157
5.41.1.47 MHV_ARDUINO_PIN_50	157
5.41.1.48 MHV_ARDUINO_PIN_51	157
5.41.1.49 MHV_ARDUINO_PIN_52	157
5.41.1.50 MHV_ARDUINO_PIN_53	157
5.41.1.51 MHV_ARDUINO_PIN_6	158
5.41.1.52 MHV_ARDUINO_PIN_7	158
5.41.1.53 MHV_ARDUINO_PIN_8	158
5.41.1.54 MHV_ARDUINO_PIN_9	158
5.41.1.55 MHV_ARDUINO_PIN_A0	158
5.41.1.56 MHV_ARDUINO_PIN_A1	158
5.41.1.57 MHV_ARDUINO_PIN_A10	158
5.41.1.58 MHV_ARDUINO_PIN_A11	158
5.41.1.59 MHV_ARDUINO_PIN_A12	158
5.41.1.60 MHV_ARDUINO_PIN_A13	158
5.41.1.61 MHV_ARDUINO_PIN_A14	159
5.41.1.62 MHV_ARDUINO_PIN_A15	159
5.41.1.63 MHV_ARDUINO_PIN_A2	159
5.41.1.64 MHV_ARDUINO_PIN_A3	159
5.41.1.65 MHV_ARDUINO_PIN_A4	159

5.41.1.66 MHV_ARDUINO_PIN_A5	159
5.41.1.67 MHV_ARDUINO_PIN_A6	159
5.41.1.68 MHV_ARDUINO_PIN_A7	159
5.41.1.69 MHV_ARDUINO_PIN_A8	159
5.41.1.70 MHV_ARDUINO_PIN_A9	159
5.42 A:/eclipse/mhvlb/MHV_io_ATmega1280.h File Reference	160
5.42.1 Define Documentation	164
5.42.1.1 MHV_AD_CHANNEL_0	164
5.42.1.2 MHV_AD_CHANNEL_0_X10_0	164
5.42.1.3 MHV_AD_CHANNEL_0_X1_1	164
5.42.1.4 MHV_AD_CHANNEL_0_X1_2	165
5.42.1.5 MHV_AD_CHANNEL_0_X200_0	165
5.42.1.6 MHV_AD_CHANNEL_0V	165
5.42.1.7 MHV_AD_CHANNEL_1	165
5.42.1.8 MHV_AD_CHANNEL_10	165
5.42.1.9 MHV_AD_CHANNEL_10_X10_10	165
5.42.1.10 MHV_AD_CHANNEL_10_X1_10	165
5.42.1.11 MHV_AD_CHANNEL_10_X1_9	165
5.42.1.12 MHV_AD_CHANNEL_10_X200_10	165
5.42.1.13 MHV_AD_CHANNEL_11	165
5.42.1.14 MHV_AD_CHANNEL_11_X10_10	166
5.42.1.15 MHV_AD_CHANNEL_11_X1_10	166
5.42.1.16 MHV_AD_CHANNEL_11_X1_9	166
5.42.1.17 MHV_AD_CHANNEL_11_X200_10	166
5.42.1.18 MHV_AD_CHANNEL_12	166
5.42.1.19 MHV_AD_CHANNEL_12_X1_10	166
5.42.1.20 MHV_AD_CHANNEL_12_X1_9	166
5.42.1.21 MHV_AD_CHANNEL_13	166
5.42.1.22 MHV_AD_CHANNEL_13_X1_10	166
5.42.1.23 MHV_AD_CHANNEL_13_X1_9	166
5.42.1.24 MHV_AD_CHANNEL_14	167
5.42.1.25 MHV_AD_CHANNEL_14_X1_9	167
5.42.1.26 MHV_AD_CHANNEL_15	167
5.42.1.27 MHV_AD_CHANNEL_15_X1_9	167

5.42.1.28 MHV_AD_CHANNEL_1_X10_0	167
5.42.1.29 MHV_AD_CHANNEL_1_X1_1	167
5.42.1.30 MHV_AD_CHANNEL_1_X1_2	167
5.42.1.31 MHV_AD_CHANNEL_1_X200_0	167
5.42.1.32 MHV_AD_CHANNEL_1V1	167
5.42.1.33 MHV_AD_CHANNEL_2	167
5.42.1.34 MHV_AD_CHANNEL_2_X10_2	168
5.42.1.35 MHV_AD_CHANNEL_2_X1_1	168
5.42.1.36 MHV_AD_CHANNEL_2_X1_2	168
5.42.1.37 MHV_AD_CHANNEL_2_X200_2	168
5.42.1.38 MHV_AD_CHANNEL_3	168
5.42.1.39 MHV_AD_CHANNEL_3_X10_2	168
5.42.1.40 MHV_AD_CHANNEL_3_X1_1	168
5.42.1.41 MHV_AD_CHANNEL_3_X1_2	168
5.42.1.42 MHV_AD_CHANNEL_3_X200_2	168
5.42.1.43 MHV_AD_CHANNEL_4	168
5.42.1.44 MHV_AD_CHANNEL_4_X1_1	169
5.42.1.45 MHV_AD_CHANNEL_4_X1_2	169
5.42.1.46 MHV_AD_CHANNEL_5	169
5.42.1.47 MHV_AD_CHANNEL_5_X1_1	169
5.42.1.48 MHV_AD_CHANNEL_5_X1_2	169
5.42.1.49 MHV_AD_CHANNEL_6	169
5.42.1.50 MHV_AD_CHANNEL_6_X1_1	169
5.42.1.51 MHV_AD_CHANNEL_7	169
5.42.1.52 MHV_AD_CHANNEL_7_X1_1	169
5.42.1.53 MHV_AD_CHANNEL_8	169
5.42.1.54 MHV_AD_CHANNEL_8_X10_8	170
5.42.1.55 MHV_AD_CHANNEL_8_X1_10	170
5.42.1.56 MHV_AD_CHANNEL_8_X1_9	170
5.42.1.57 MHV_AD_CHANNEL_8_X200_8	170
5.42.1.58 MHV_AD_CHANNEL_9	170
5.42.1.59 MHV_AD_CHANNEL_9_X10_8	170
5.42.1.60 MHV_AD_CHANNEL_9_X1_10	170
5.42.1.61 MHV_AD_CHANNEL_9_X1_9	170

5.42.1.62 MHV_AD_CHANNEL_9_X200_8	170
5.42.1.63 MHV_AD_PRR	170
5.42.1.64 MHV_AD_REFERENCE_1V1	171
5.42.1.65 MHV_AD_REFERENCE_2V56	171
5.42.1.66 MHV_AD_REFERENCE_AREF	171
5.42.1.67 MHV_AD_REFERENCE_AVCC	171
5.42.1.68 MHV_AD_RESOLUTION	171
5.42.1.69 MHV_EEPROM_VECT	171
5.42.1.70 MHV_PC_INT_COUNT	171
5.42.1.71 MHV_PIN_A0	171
5.42.1.72 MHV_PIN_A1	171
5.42.1.73 MHV_PIN_A2	171
5.42.1.74 MHV_PIN_A3	172
5.42.1.75 MHV_PIN_A4	172
5.42.1.76 MHV_PIN_A5	172
5.42.1.77 MHV_PIN_A6	172
5.42.1.78 MHV_PIN_A7	172
5.42.1.79 MHV_PIN_B0	172
5.42.1.80 MHV_PIN_B1	172
5.42.1.81 MHV_PIN_B2	172
5.42.1.82 MHV_PIN_B3	172
5.42.1.83 MHV_PIN_B4	172
5.42.1.84 MHV_PIN_B5	173
5.42.1.85 MHV_PIN_B6	173
5.42.1.86 MHV_PIN_B7	173
5.42.1.87 MHV_PIN_C0	173
5.42.1.88 MHV_PIN_C1	173
5.42.1.89 MHV_PIN_C2	173
5.42.1.90 MHV_PIN_C3	173
5.42.1.91 MHV_PIN_C4	173
5.42.1.92 MHV_PIN_C5	173
5.42.1.93 MHV_PIN_C6	173
5.42.1.94 MHV_PIN_C7	174
5.42.1.95 MHV_PIN_D0	174

5.42.1.96 MHV_PIN_D1	174
5.42.1.97 MHV_PIN_D2	174
5.42.1.98 MHV_PIN_D3	174
5.42.1.99 MHV_PIN_D4	174
5.42.1.100MHV_PIN_D5	174
5.42.1.101MHV_PIN_D6	174
5.42.1.102MHV_PIN_D7	174
5.42.1.103MHV_PIN_E0	174
5.42.1.104MHV_PIN_E1	175
5.42.1.105MHV_PIN_E2	175
5.42.1.106MHV_PIN_E3	175
5.42.1.107MHV_PIN_E4	175
5.42.1.108MHV_PIN_E5	175
5.42.1.109MHV_PIN_E6	175
5.42.1.110MHV_PIN_E7	175
5.42.1.111MHV_PIN_F0	175
5.42.1.112MHV_PIN_F1	175
5.42.1.113MHV_PIN_F2	175
5.42.1.114MHV_PIN_F3	176
5.42.1.115MHV_PIN_F4	176
5.42.1.116MHV_PIN_F5	176
5.42.1.117MHV_PIN_F6	176
5.42.1.118MHV_PIN_F7	176
5.42.1.119MHV_PIN_G0	176
5.42.1.120MHV_PIN_G1	176
5.42.1.121MHV_PIN_G2	176
5.42.1.122MHV_PIN_G3	176
5.42.1.123MHV_PIN_G4	176
5.42.1.124MHV_PIN_G5	177
5.42.1.125MHV_PIN_H0	177
5.42.1.126MHV_PIN_H1	177
5.42.1.127MHV_PIN_H2	177
5.42.1.128MHV_PIN_H3	177
5.42.1.129MHV_PIN_H4	177

5.42.1.130MHV_PIN_H5	177
5.42.1.131MHV_PIN_H6	177
5.42.1.132MHV_PIN_H7	177
5.42.1.133MHV_PIN_J0	177
5.42.1.134MHV_PIN_J1	178
5.42.1.135MHV_PIN_J2	178
5.42.1.136MHV_PIN_J3	178
5.42.1.137MHV_PIN_J4	178
5.42.1.138MHV_PIN_J5	178
5.42.1.139MHV_PIN_J6	178
5.42.1.140MHV_PIN_J7	178
5.42.1.141MHV_PIN_K0	178
5.42.1.142MHV_PIN_K1	178
5.42.1.143MHV_PIN_K2	178
5.42.1.144MHV_PIN_K3	179
5.42.1.145MHV_PIN_K4	179
5.42.1.146MHV_PIN_K5	179
5.42.1.147MHV_PIN_K6	179
5.42.1.148MHV_PIN_K7	179
5.42.1.149MHV_PIN_L0	179
5.42.1.150MHV_PIN_L1	179
5.42.1.151MHV_PIN_L2	179
5.42.1.152MHV_PIN_L3	179
5.42.1.153MHV_PIN_L4	179
5.42.1.154MHV_PIN_L5	180
5.42.1.155MHV_PIN_L6	180
5.42.1.156MHV_PIN_L7	180
5.42.1.157MHV_PIN_TIMER_0_A	180
5.42.1.158MHV_PIN_TIMER_0_B	180
5.42.1.159MHV_PIN_TIMER_1_A	180
5.42.1.160MHV_PIN_TIMER_1_B	180
5.42.1.161MHV_PIN_TIMER_1_C	180
5.42.1.162MHV_PIN_TIMER_2_A	180
5.42.1.163MHV_PIN_TIMER_2_B	180

5.42.1.164	MHV_PIN_TIMER_3_A	181
5.42.1.165	MHV_PIN_TIMER_3_B	181
5.42.1.166	MHV_PIN_TIMER_3_C	181
5.42.1.167	MHV_PIN_TIMER_4_A	181
5.42.1.168	MHV_PIN_TIMER_4_B	181
5.42.1.169	MHV_PIN_TIMER_4_C	181
5.42.1.170	MHV_PIN_TIMER_5_A	181
5.42.1.171	MHV_PIN_TIMER_5_B	181
5.42.1.172	MHV_PIN_TIMER_5_C	181
5.42.1.173	MHV_TIMER0_INTERRUPTS	181
5.42.1.174	MHV_TIMER16_1	182
5.42.1.175	MHV_TIMER16_3	182
5.42.1.176	MHV_TIMER16_4	182
5.42.1.177	MHV_TIMER16_5	182
5.42.1.178	MHV_TIMER1_INTERRUPTS	182
5.42.1.179	MHV_TIMER2_INTERRUPTS	182
5.42.1.180	MHV_TIMER3_INTERRUPTS	182
5.42.1.181	MHV_TIMER4_INTERRUPTS	182
5.42.1.182	MHV_TIMER5_INTERRUPTS	182
5.42.1.183	MHV_TIMER8_0	183
5.42.1.184	MHV_TIMER8_2	183
5.42.1.185	MHV_USART0	183
5.42.1.186	MHV_USART0_INTERRUPTS	183
5.42.1.187	MHV_USART1	183
5.42.1.188	MHV_USART1_INTERRUPTS	183
5.42.1.189	MHV_USART2	183
5.42.1.190	MHV_USART2_INTERRUPTS	183
5.42.1.191	MHV_USART3	183
5.42.1.192	MHV_USART3_INTERRUPTS	184
5.43	A:/eclipse/mhplib/MHV_io_ATmega168.h File Reference	184
5.43.1	Define Documentation	185
5.43.1.1	MHV_AD_CHANNEL_0	185
5.43.1.2	MHV_AD_CHANNEL_0V	185
5.43.1.3	MHV_AD_CHANNEL_1	185

5.43.1.4	MHV_AD_CHANNEL_1V1	186
5.43.1.5	MHV_AD_CHANNEL_2	186
5.43.1.6	MHV_AD_CHANNEL_3	186
5.43.1.7	MHV_AD_CHANNEL_4	186
5.43.1.8	MHV_AD_CHANNEL_5	186
5.43.1.9	MHV_AD_CHANNEL_6	186
5.43.1.10	MHV_AD_CHANNEL_7	186
5.43.1.11	MHV_AD_CHANNEL_8	186
5.43.1.12	MHV_AD_PRR	186
5.43.1.13	MHV_AD_REFERENCE_1V1	186
5.43.1.14	MHV_AD_REFERENCE_AREF	187
5.43.1.15	MHV_AD_REFERENCE_AVCC	187
5.43.1.16	MHV_AD_RESOLUTION	187
5.43.1.17	MHV_AD_TEMPERATURE	187
5.43.1.18	MHV_EEPROM_VECT	187
5.43.1.19	MHV_INTERRUPT_INT0	187
5.43.1.20	MHV_INTERRUPT_INT1	187
5.43.1.21	MHV_PC_INT_COUNT	187
5.43.1.22	MHV_PIN_B0	187
5.43.1.23	MHV_PIN_B1	187
5.43.1.24	MHV_PIN_B2	188
5.43.1.25	MHV_PIN_B3	188
5.43.1.26	MHV_PIN_B4	188
5.43.1.27	MHV_PIN_B5	188
5.43.1.28	MHV_PIN_B6	188
5.43.1.29	MHV_PIN_B7	188
5.43.1.30	MHV_PIN_C0	188
5.43.1.31	MHV_PIN_C1	188
5.43.1.32	MHV_PIN_C2	188
5.43.1.33	MHV_PIN_C3	188
5.43.1.34	MHV_PIN_C4	189
5.43.1.35	MHV_PIN_C5	189
5.43.1.36	MHV_PIN_C6	189
5.43.1.37	MHV_PIN_D0	189

5.43.1.38 MHV_PIN_D1	189
5.43.1.39 MHV_PIN_D2	189
5.43.1.40 MHV_PIN_D3	189
5.43.1.41 MHV_PIN_D4	189
5.43.1.42 MHV_PIN_D5	189
5.43.1.43 MHV_PIN_D6	189
5.43.1.44 MHV_PIN_D7	190
5.43.1.45 MHV_PIN_TIMER_0_A	190
5.43.1.46 MHV_PIN_TIMER_0_B	190
5.43.1.47 MHV_PIN_TIMER_1_A	190
5.43.1.48 MHV_PIN_TIMER_1_B	190
5.43.1.49 MHV_PIN_TIMER_2_A	190
5.43.1.50 MHV_PIN_TIMER_2_B	190
5.43.1.51 MHV_TIMER0_INTERRUPTS	190
5.43.1.52 MHV_TIMER16_1	190
5.43.1.53 MHV_TIMER1_INTERRUPTS	190
5.43.1.54 MHV_TIMER2_INTERRUPTS	191
5.43.1.55 MHV_TIMER8_0	191
5.43.1.56 MHV_TIMER8_2	191
5.43.1.57 MHV_USART0	191
5.43.1.58 MHV_USART0_INTERRUPTS	191
5.44 A:/eclipse/mhplib/MHV_io_ATtiny2313.h File Reference	191
5.44.1 Define Documentation	192
5.44.1.1 MHV_EEPROM_VECT	192
5.44.1.2 MHV_INTERRUPT_INT0	192
5.44.1.3 MHV_INTERRUPT_INT1	192
5.44.1.4 MHV_PC_INT_COUNT	192
5.44.1.5 MHV_PIN_A0	192
5.44.1.6 MHV_PIN_A1	193
5.44.1.7 MHV_PIN_A2	193
5.44.1.8 MHV_PIN_B0	193
5.44.1.9 MHV_PIN_B1	193
5.44.1.10 MHV_PIN_B2	193
5.44.1.11 MHV_PIN_B3	193

5.44.1.12 MHV_PIN_B4	193
5.44.1.13 MHV_PIN_B5	193
5.44.1.14 MHV_PIN_B6	193
5.44.1.15 MHV_PIN_B7	193
5.44.1.16 MHV_PIN_D0	194
5.44.1.17 MHV_PIN_D1	194
5.44.1.18 MHV_PIN_D2	194
5.44.1.19 MHV_PIN_D3	194
5.44.1.20 MHV_PIN_D4	194
5.44.1.21 MHV_PIN_D5	194
5.44.1.22 MHV_PIN_D6	194
5.44.1.23 MHV_PIN_TIMER_0_A	194
5.44.1.24 MHV_PIN_TIMER_0_B	194
5.44.1.25 MHV_PIN_TIMER_1_A	194
5.44.1.26 MHV_PIN_TIMER_1_B	195
5.44.1.27 MHV_TIMER0_INTERRUPTS	195
5.44.1.28 MHV_TIMER16_1	195
5.44.1.29 MHV_TIMER1_INTERRUPTS	195
5.44.1.30 MHV_TIMER8_0	195
5.44.1.31 MHV_USART0	195
5.44.1.32 MHV_USART0_INTERRUPTS	195
5.45 A:/eclipse/mhvlb/MHV_io_ATtiny85.h File Reference	195
5.45.1 Define Documentation	196
5.45.1.1 MHV_AD_0V	196
5.45.1.2 MHV_AD_CHANNEL_0	197
5.45.1.3 MHV_AD_CHANNEL_0_X1_0	197
5.45.1.4 MHV_AD_CHANNEL_0_X1_1	197
5.45.1.5 MHV_AD_CHANNEL_0_X20_0	197
5.45.1.6 MHV_AD_CHANNEL_0_X20_1	197
5.45.1.7 MHV_AD_CHANNEL_1	197
5.45.1.8 MHV_AD_CHANNEL_2	197
5.45.1.9 MHV_AD_CHANNEL_2_X1_2	197
5.45.1.10 MHV_AD_CHANNEL_2_X1_3	197
5.45.1.11 MHV_AD_CHANNEL_2_X20_2	197

5.45.1.12 MHV_AD_CHANNEL_2_X20_3	198
5.45.1.13 MHV_AD_CHANNEL_3	198
5.45.1.14 MHV_AD_PRR	198
5.45.1.15 MHV_AD_REFERENCE_1V1	198
5.45.1.16 MHV_AD_REFERENCE_2V56	198
5.45.1.17 MHV_AD_REFERENCE_2V56_AREF	198
5.45.1.18 MHV_AD_REFERENCE_AREF	198
5.45.1.19 MHV_AD_REFERENCE_VCC	198
5.45.1.20 MHV_AD_RESOLUTION	198
5.45.1.21 MHV_AD_TEMPERATURE	198
5.45.1.22 MHV_AD_V_BANDGAP	199
5.45.1.23 MHV_EEPROM_VECT	199
5.45.1.24 MHV_INTERRUPT_INT0	199
5.45.1.25 MHV_PC_INT_COUNT	199
5.45.1.26 MHV_PIN_B0	199
5.45.1.27 MHV_PIN_B1	199
5.45.1.28 MHV_PIN_B2	199
5.45.1.29 MHV_PIN_B3	199
5.45.1.30 MHV_PIN_B4	199
5.45.1.31 MHV_PIN_B5	199
5.45.1.32 MHV_PIN_TIMER_0_A	200
5.45.1.33 MHV_PIN_TIMER_0_B	200
5.45.1.34 MHV_PIN_TIMER_1_A	200
5.45.1.35 MHV_PIN_TIMER_1_B	200
5.45.1.36 MHV_TIMER0_INTERRUPTS	200
5.45.1.37 MHV_TIMER1_INTERRUPTS	200
5.45.1.38 MHV_TIMER8_0	200
5.45.1.39 MHV_TIMER8_1	200
5.46 A:/eclipse/mhvlb/MHV_Lock.cpp File Reference	200
5.47 A:/eclipse/mhvlb/MHV_Lock.h File Reference	201
5.48 A:/eclipse/mhvlb/MHV_PID.cpp File Reference	201
5.49 A:/eclipse/mhvlb/MHV_PID.h File Reference	201
5.50 A:/eclipse/mhvlb/MHV_PinChangeManager.cpp File Reference	201
5.51 A:/eclipse/mhvlb/MHV_PinChangeManager.h File Reference	201

5.51.1	Define Documentation	202
5.51.1.1	MHV_PINCHANGE_MANAGER_ASSIGN_INTERRUPTS	202
5.51.1.2	MHV_PINCHANGE_MANAGER_ASSIGN_PCINT	202
5.51.1.3	MHV_PINCHANGE_MANAGER_ASSIGN_PCINT0	202
5.51.1.4	MHV_PINCHANGE_MANAGER_ASSIGN_PCINT1	202
5.51.1.5	MHV_PINCHANGE_MANAGER_ASSIGN_PCINT2	203
5.51.2	Typedef Documentation	203
5.51.2.1	MHV_EVENT_PIN	203
5.52	A:/eclipse/mhvlb/MHV_PWMMatrix.cpp File Reference	203
5.52.1	Define Documentation	203
5.52.1.1	pixel	203
5.53	A:/eclipse/mhvlb/MHV_PWMMatrix.h File Reference	203
5.53.1	Typedef Documentation	204
5.53.1.1	MHV_PWMMATRIX_MODE	204
5.53.2	Enumeration Type Documentation	204
5.53.2.1	MHV_PWMMatrix_Mode	204
5.54	A:/eclipse/mhvlb/MHV_RingBuffer.cpp File Reference	204
5.55	A:/eclipse/mhvlb/MHV_RingBuffer.h File Reference	204
5.56	A:/eclipse/mhvlb/MHV_RTC.cpp File Reference	205
5.56.1	Function Documentation	205
5.56.1.1	mhv_daysInMonth	205
5.56.1.2	mhv_isLeapYear	205
5.56.1.3	mhv_timestampGreaterThanOrEqual	206
5.56.1.4	mhv_timestampIncrement	206
5.56.1.5	mhv_timestampIncrement	206
5.56.1.6	mhv_timestampLessThan	206
5.56.2	Variable Documentation	207
5.56.2.1	PROGMEM	207
5.57	A:/eclipse/mhvlb/MHV_RTC.h File Reference	207
5.57.1	Typedef Documentation	208
5.57.1.1	MHV_ALARM	208
5.57.1.2	MHV_MONTH	208
5.57.1.3	MHV_TIME	208

5.57.1.4	MHV_TIMESTAMP	209
5.57.1.5	MHV_WEEKDAY	209
5.57.2	Enumeration Type Documentation	209
5.57.2.1	mhv_month	209
5.57.2.2	mhv_weekday	209
5.57.3	Function Documentation	210
5.57.3.1	mhv_daysInMonth	210
5.57.3.2	mhv_isLeapYear	210
5.57.3.3	mhv_timestampGreaterThanOrEqual	210
5.57.3.4	mhv_timestampIncrement	210
5.57.3.5	mhv_timestampIncrement	211
5.57.3.6	mhv_timestampLessThan	211
5.58	A:/eclipse/mhvlb/MHV_ServoControl.cpp File Reference	211
5.59	A:/eclipse/mhvlb/MHV_ServoControl.h File Reference	211
5.60	A:/eclipse/mhvlb/MHV_Shifter.cpp File Reference	212
5.60.1	Function Documentation	212
5.60.1.1	mhv_shiftout_byte_lsb	212
5.60.1.2	mhv_shiftout_byte_msb	212
5.61	A:/eclipse/mhvlb/MHV_Shifter.h File Reference	212
5.61.1	Define Documentation	213
5.61.1.1	MHV_BIT_1	213
5.61.1.2	MHV_BIT_2	213
5.61.1.3	MHV_BIT_3	213
5.61.1.4	MHV_BIT_4	213
5.61.1.5	MHV_BIT_5	213
5.61.1.6	MHV_BIT_6	213
5.61.1.7	MHV_BIT_7	213
5.61.1.8	MHV_BIT_8	213
5.61.1.9	MHV_SHIFTOUT_ARRAY_CLOCKED_FALLING	213
5.61.1.10	MHV_SHIFTOUT_ARRAY_CLOCKED_RISING	214
5.61.1.11	MHV_SHIFTOUT_BYTE	214
5.61.1.12	SHIFTOUT_BYTE_LOOP	214
5.61.2	Function Documentation	214
5.61.2.1	mhv_shiftout_byte_lsb	214

5.61.2.2	mhv_shiftout_byte_msb	214
5.62	A:/eclipse/mhvlb/MHV_SoftwareHBridge.cpp File Reference	214
5.63	A:/eclipse/mhvlb/MHV_SoftwareHBridge.h File Reference	214
5.64	A:/eclipse/mhvlb/MHV_Timer16.cpp File Reference	214
5.65	A:/eclipse/mhvlb/MHV_Timer16.h File Reference	215
5.66	A:/eclipse/mhvlb/MHV_Timer8.cpp File Reference	215
5.67	A:/eclipse/mhvlb/MHV_Timer8.h File Reference	215
5.67.1	Define Documentation	216
5.67.1.1	_MHV_TIMER_ASSIGN_1INTERRUPT	216
5.67.1.2	_MHV_TIMER_ASSIGN_2INTERRUPTS	216
5.67.1.3	MHV_TIMER_ASSIGN_1INTERRUPT	216
5.67.1.4	MHV_TIMER_ASSIGN_2INTERRUPTS	217
5.67.2	Typedef Documentation	217
5.67.2.1	MHV_TIMER_CONNECT_TYPE	217
5.67.2.2	MHV_TIMER_MODE	217
5.67.2.3	MHV_TIMER_PRESCALER	217
5.67.2.4	MHV_TIMER_TYPE	217
5.67.3	Enumeration Type Documentation	217
5.67.3.1	mhv_timer_connect_type	217
5.67.3.2	mhv_timer_mode	217
5.67.3.3	mhv_timer_prescaler	218
5.67.3.4	mhv_timer_type	218
5.68	A:/eclipse/mhvlb/MHV_VoltageRegulator.cpp File Reference	218
5.69	A:/eclipse/mhvlb/MHV_VoltageRegulator.h File Reference	219

Chapter 1

Class Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

MHV_ADC	7
MHV_ADCListener	10
mhv_alarm	10
MHV_AlarmListener	11
MHV_VusbKeyboard	97
MHV_VusbTypist	100
MHV_DebounceListener	15
mhv_debouncePin	16
MHV_Device_RX	17
MHV_HardwareSerial	64
MHV_Device_TX	20
MHV_Display_Character	24
MHV_Display_HD44780	30
MHV_Display_HD44780_Direct_Connect	36
MHV_Display_HD44780_Shift_Register	43
MHV_Display_Monochrome	50
MHV_Display_Holtek_HT1632	47
MHV_Display_Monochrome_Buffered	56
MHV_PWMMatrix	77
MHV_HardwareSerial	64
MHV_VusbTypist	100
MHV_EEPROM	57
mhv_eventADC	61
mhv_eventPin	61
mhv_font	62
MHV_Lock	68
MHV_PID	69
mhv_pin	73

MHV_PinChangeManager	74
MHV_PinEventListener	77
MHV_Debounce	12
MHV_RingBuffer	79
MHV_RTC	82
MHV_RXListener	87
mhv_time	88
MHV_Timer8	89
mhv_timestamp	96
mhv_tx_buffer	96

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

MHV_ADC	7
MHV_ADCListener	10
mhv_alarm	10
MHV_AlarmListener	11
MHV_Debounce	12
MHV_DebounceListener	15
mhv_debouncePin	16
MHV_Device_RX	17
MHV_Device_TX	20
MHV_Display_Character	24
MHV_Display_HD44780	30
MHV_Display_HD44780_Direct_Connect	36
MHV_Display_HD44780_Shift_Register	43
MHV_Display_Holtek_HT1632	47
MHV_Display_Monochrome	50
MHV_Display_Monochrome_Buffered	56
MHV_EEPROM	57
mhv_eventADC	61
mhv_eventPin	61
mhv_font	62
MHV_HardwareSerial	64
MHV_Lock	68
MHV_PID	69
mhv_pin	73
MHV_PinChangeManager	74
MHV_PinEventListener	77
MHV_PWMMatrix	77
MHV_RingBuffer	79
MHV_RTC	82

MHV_RXListener	87
mhv_time	88
MHV_Timer8	89
mhv_timestamp	96
mhv_tx_buffer	96
MHV_VusbKeyboard	97
MHV_VusbTypist	100

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

A:/eclipse/mhplib-Vusb-KeyBoard/MHV_VusbKeyboard.cpp	103
A:/eclipse/mhplib-Vusb-KeyBoard/MHV_VusbKeyboard.h	104
A:/eclipse/mhplib-Vusb-KeyBoard/MHV_VusbTypist.cpp	111
A:/eclipse/mhplib-Vusb-KeyBoard/MHV_VusbTypist.h	111
A:/eclipse/mhplib/MHV_AD.cpp	111
A:/eclipse/mhplib/MHV_AD.h	112
A:/eclipse/mhplib/MHV_ADC.cpp	114
A:/eclipse/mhplib/MHV_ADC.h	114
A:/eclipse/mhplib/MHV_Debounce.cpp	115
A:/eclipse/mhplib/MHV_Debounce.h	115
A:/eclipse/mhplib/MHV_Device_RX.cpp	117
A:/eclipse/mhplib/MHV_Device_RX.h	117
A:/eclipse/mhplib/MHV_Device_TX.cpp	118
A:/eclipse/mhplib/MHV_Device_TX.h	118
A:/eclipse/mhplib/MHV_Display_Character.cpp	119
A:/eclipse/mhplib/MHV_Display_Character.h	119
A:/eclipse/mhplib/MHV_Display_HD44780.cpp	119
A:/eclipse/mhplib/MHV_Display_HD44780.h	120
A:/eclipse/mhplib/MHV_Display_HD44780_Direct_Connect.cpp	121
A:/eclipse/mhplib/MHV_Display_HD44780_Direct_Connect.h	123
A:/eclipse/mhplib/MHV_Display_HD44780_Shift_Register.cpp	124
A:/eclipse/mhplib/MHV_Display_HD44780_Shift_Register.h	124
A:/eclipse/mhplib/MHV_Display_Holtek_HT1632.cpp	125
A:/eclipse/mhplib/MHV_Display_Holtek_HT1632.h	125
A:/eclipse/mhplib/MHV_Display_Monochrome.cpp	127
A:/eclipse/mhplib/MHV_Display_Monochrome.h	127
A:/eclipse/mhplib/MHV_Display_Monochrome_Buffered.cpp	128
A:/eclipse/mhplib/MHV_Display_Monochrome_Buffered.h	128
A:/eclipse/mhplib/MHV_EEPROM.cpp	128

A:/eclipse/mhvlb/MHV_EEPROM.h	128
A:/eclipse/mhvlb/MHV_EPP.cpp	129
A:/eclipse/mhvlb/MHV_EPP.h	129
A:/eclipse/mhvlb/MHV_Font.h	129
A:/eclipse/mhvlb/MHV_Font_SansSerif_10x8.h	130
A:/eclipse/mhvlb/MHV_GammaCorrect.cpp	130
A:/eclipse/mhvlb/MHV_GammaCorrect.h	131
A:/eclipse/mhvlb/MHV_HardwareSerial.cpp	131
A:/eclipse/mhvlb/MHV_HardwareSerial.h	132
A:/eclipse/mhvlb/MHV_io.h	134
A:/eclipse/mhvlb/MHV_io_ArduinoDiecimilla.h	148
A:/eclipse/mhvlb/MHV_io_ArduinoMega.h	151
A:/eclipse/mhvlb/MHV_io_ATmega1280.h	160
A:/eclipse/mhvlb/MHV_io_ATmega168.h	184
A:/eclipse/mhvlb/MHV_io_ATtiny2313.h	191
A:/eclipse/mhvlb/MHV_io_ATtiny85.h	195
A:/eclipse/mhvlb/MHV_Lock.cpp	200
A:/eclipse/mhvlb/MHV_Lock.h	201
A:/eclipse/mhvlb/MHV_PID.cpp	201
A:/eclipse/mhvlb/MHV_PID.h	201
A:/eclipse/mhvlb/MHV_PinChangeManager.cpp	201
A:/eclipse/mhvlb/MHV_PinChangeManager.h	201
A:/eclipse/mhvlb/MHV_PWMMatrix.cpp	203
A:/eclipse/mhvlb/MHV_PWMMatrix.h	203
A:/eclipse/mhvlb/MHV_RingBuffer.cpp	204
A:/eclipse/mhvlb/MHV_RingBuffer.h	204
A:/eclipse/mhvlb/MHV_RTC.cpp	205
A:/eclipse/mhvlb/MHV_RTC.h	207
A:/eclipse/mhvlb/MHV_ServoControl.cpp	211
A:/eclipse/mhvlb/MHV_ServoControl.h	211
A:/eclipse/mhvlb/MHV_Shifter.cpp	212
A:/eclipse/mhvlb/MHV_Shifter.h	212
A:/eclipse/mhvlb/MHV_SoftwareHBridge.cpp	214
A:/eclipse/mhvlb/MHV_SoftwareHBridge.h	214
A:/eclipse/mhvlb/MHV_Timer16.cpp	214
A:/eclipse/mhvlb/MHV_Timer16.h	215
A:/eclipse/mhvlb/MHV_Timer8.cpp	215
A:/eclipse/mhvlb/MHV_Timer8.h	215
A:/eclipse/mhvlb/MHV_VoltageRegulator.cpp	218
A:/eclipse/mhvlb/MHV_VoltageRegulator.h	219

Chapter 4

Class Documentation

4.1 MHV_ADC Class Reference

```
#include <MHV_ADC.h>
```

Public Member Functions

- [MHV_ADC](#) ([MHV_EVENT_ADC](#) *adcs, uint8_t adcCount)
- void [adc](#) ()
- void [registerListener](#) (uint8_t channel, [MHV_ADCListener](#) *listener)
- void [deregisterListener](#) (uint8_t channel)
- void [enable](#) ()
- void [disable](#) ()
- uint16_t [busyRead](#) (uint8_t channel, uint8_t reference)
- void [asyncRead](#) (uint8_t channel, uint8_t reference)
- void [setPrescaler](#) ([MHV_AD_PRESCALER](#) prescaler)
- void [handleEvents](#) ()

Protected Attributes

- uint16_t [_adcValue](#)
- int8_t [_adcChannel](#)
- [MHV_EVENT_ADC](#) * [_adcs](#)
- uint8_t [_adcCount](#)

4.1.1 Detailed Description

Definition at line 57 of file MHV_ADC.h.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 MHV_ADC::MHV_ADC (MHV_EVENT_ADC * *adcs*, uint8_t *adcCount*)

An event manager for ADC events

Parameters

<i>adcs</i>	ADC event handles
<i>adcCount</i>	The number of ADC events we can handle (must match <i>adcs</i>)

Definition at line 35 of file MHV_ADC.cpp.

4.1.3 Member Function Documentation

4.1.3.1 void MHV_ADC::adc ()

Interrupt handler to read the ADC

Definition at line 47 of file MHV_ADC.cpp.

4.1.3.2 void MHV_ADC::asyncRead (uint8_t *channel*, uint8_t *reference*)

Trigger an ADC channel event

Parameters

<i>channel</i>	the channel to read
<i>reference</i>	the voltage reference to use

Definition at line 107 of file MHV_ADC.cpp.

4.1.3.3 uint16_t MHV_ADC::busyRead (uint8_t *channel*, uint8_t *reference*)

Read an ADC channel

Parameters

<i>channel</i>	the channel to read
<i>reference</i>	the voltage reference to use

Definition at line 87 of file MHV_ADC.cpp.

4.1.3.4 void MHV_ADC::deregisterListener (uint8_t *channel*)

Deregister interest for an ADC channel

Parameters

<i>channel</i>	the ADC channel
----------------	-----------------

Definition at line 73 of file MHV_ADC.cpp.

4.1.3.5 void MHV_ADC::disable ()

4.1.3.6 void MHV_ADC::enable ()

4.1.3.7 void MHV_ADC::handleEvents ()

Call from the main loop to handle any events

Definition at line 132 of file MHV_ADC.cpp.

4.1.3.8 void MHV_ADC::registerListener (uint8_t *channel*, MHV_ADCListener * *listener*)

Register interest for an ADC channel

Parameters

<i>channel</i>	the ADC channel
<i>listener</i>	an MHV_ADCListener to notify when an ADC reading has been completed

Definition at line 59 of file MHV_ADC.cpp.

4.1.3.9 void MHV_ADC::setPrescaler (MHV_AD_PRESCALER *prescaler*)

Set the ADC clock prescaler

Parameters

<i>prescaler</i>	the prescaler to use
------------------	----------------------

Definition at line 124 of file MHV_ADC.cpp.

4.1.4 Member Data Documentation

4.1.4.1 int8_t MHV_ADC::_adcChannel [protected]

Definition at line 60 of file MHV_ADC.h.

4.1.4.2 uint8_t MHV_ADC::_adcCount [protected]

Definition at line 62 of file MHV_ADC.h.

4.1.4.3 MHV_EVENT_ADC* MHV_ADC::_adcs [protected]

Definition at line 61 of file MHV_ADC.h.

4.1.4.4 uint16_t MHV_ADC::_adcValue [protected]

Definition at line 59 of file MHV_ADC.h.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_ADC.h](#)
- [A:/eclipse/mhvlb/MHV_ADC.cpp](#)

4.2 MHV_ADCListener Class Reference

```
#include <MHV_ADC.h>
```

Public Member Functions

- virtual void [adc](#) (uint8_t adcChannel, uint16_t adcValue)=0

4.2.1 Detailed Description

Definition at line 46 of file MHV_ADC.h.

4.2.2 Member Function Documentation

4.2.2.1 virtual void MHV_ADCListener::adc (uint8_t *adcChannel*, uint16_t *adcValue*) [pure virtual]

The documentation for this class was generated from the following file:

- [A:/eclipse/mhvlb/MHV_ADC.h](#)

4.3 mhv_alarm Struct Reference

```
#include <MHV_RTC.h>
```

Public Attributes

- [MHV_TIMESTAMP](#) when
- [MHV_TIMESTAMP](#) repeat
- [MHV_AlarmListener](#) * listener

4.3.1 Detailed Description

Definition at line 82 of file MHV_RTC.h.

4.3.2 Member Data Documentation

4.3.2.1 MHV_AlarmListener* mhv_alarm::listener

Definition at line 85 of file MHV_RTC.h.

4.3.2.2 MHV_TIMESTAMP mhv_alarm::repeat

Definition at line 84 of file MHV_RTC.h.

4.3.2.3 MHV_TIMESTAMP mhv_alarm::when

Definition at line 83 of file MHV_RTC.h.

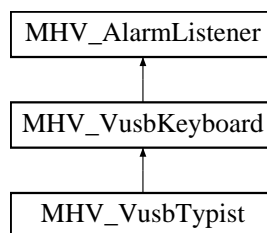
The documentation for this struct was generated from the following file:

- A:/eclipse/mhvlb/[MHV_RTC.h](#)

4.4 MHV_AlarmListener Class Reference

```
#include <MHV_RTC.h>
```

Inheritance diagram for MHV_AlarmListener:



Public Member Functions

- virtual void [alarm](#) (MHV_ALARM *alarm)=0

4.4.1 Detailed Description

Definition at line 89 of file MHV_RTC.h.

4.4.2 Member Function Documentation

4.4.2.1 `virtual void MHV_AlarmListener::alarm (MHV_ALARM * alarm)` [pure virtual]

Implemented in [MHV_VusbKeyboard](#), and [MHV_VusbTypist](#).

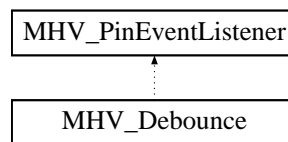
The documentation for this class was generated from the following file:

- [A:/eclipse/mhvlb/MHV_RTC.h](#)

4.5 MHV_Debounce Class Reference

```
#include <MHV_Debounce.h>
```

Inheritance diagram for MHV_Debounce:



Public Member Functions

- [MHV_Debounce](#) ([MHV_PinChangeManager](#) *pinChangeManager, [MHV_RTC](#) *rtc, uint16_t debounceTime, uint16_t heldTime, uint16_t repeatTime)
- void [assignKey](#) (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t pin, int8_t pinchangeInterrupt, [MHV_DebounceListener](#) *listener)
- void [deassignKey](#) (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t pin, int8_t pinchangeInterrupt)
- void [checkHeld](#) ()

Protected Member Functions

- void [pinChanged](#) (uint8_t pcInt, bool newState)
- void [initPin](#) (uint8_t pinchangeInterrupt)

Protected Attributes

- [MHV_RTC](#) * _rtc
- [MHV_DEBOUNCE_PIN](#) _pins [MHV_PC_INT_COUNT]
- [MHV_TIMESTAMP](#) _debounceTime
- [MHV_TIMESTAMP](#) _heldTime
- [MHV_TIMESTAMP](#) _repeatTime
- [MHV_PinChangeManager](#) * _pinChangeManager

4.5.1 Detailed Description

Definition at line 70 of file MHV_Debounce.h.

4.5.2 Constructor & Destructor Documentation

4.5.2.1 MHV_Debounce::MHV_Debounce (MHV_PinChangeManager * *pinChangeManager*, MHV_RTC * *rtc*, uint16_t *debounceTime*, uint16_t *heldTime*, uint16_t *repeatTime*)

Debouncing helper for buttons connected directly to PCINT capable pins The user must pull the pin up, either externally (and initing the pin by calling `mhv_setInput`), or internally by calling `mhv_setInputPullup`

Parameters

<i>pinChange-Manager</i>	the pin change manager
<i>rtc</i>	the realtime clock we will use for timing
<i>debounce-Time</i>	the minimum amount of time to count as a button press (in milliseconds)
<i>heldTime</i>	the minimum amount of time to consider a button held down
<i>repeatTime</i>	the time after which the held call repeats

Definition at line 40 of file MHV_Debounce.cpp.

4.5.3 Member Function Documentation

4.5.3.1 void MHV_Debounce::assignKey (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pinchangeInterrupt*, MHV_DebounceListener * *listener*)

Assign a pin to debounce

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>listener</i>	a class to call when the button is pressed or held down

Definition at line 126 of file MHV_Debounce.cpp.

4.5.3.2 void MHV_Debounce::checkHeld ()

Called periodically to check if pins have been held Ideally, this should be called from the main loop, rather than the interrupt context

Definition at line 68 of file MHV_Debounce.cpp.

4.5.3.3 void MHV_Debounce::deassignKey (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pinchangeInterrupt*)

Deassign a pin

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pinchange-Interrupt</i>	A member of the MHV_PIN_* macro

Definition at line 150 of file MHV_Debounce.cpp.

4.5.3.4 void MHV_Debounce::initPin (uint8_t *pinchangeInterrupt*) [protected]

Definition at line 60 of file MHV_Debounce.cpp.

4.5.3.5 void MHV_Debounce::pinChanged (uint8_t *pcInt*, bool *newState*) [protected, virtual]

Trigger for pin change interrupts - scans through 8 pins starting at the offset

Parameters

<i>pcInt</i>	the pin change interrupt that was triggered
<i>newState</i>	the new state of the pin

Implements [MHV_PinEventListener](#).

Definition at line 95 of file MHV_Debounce.cpp.

4.5.4 Member Data Documentation

4.5.4.1 MHV_TIMESTAMP MHV_Debounce::_debounceTime [protected]

Definition at line 74 of file MHV_Debounce.h.

4.5.4.2 MHV_TIMESTAMP MHV_Debounce::_heldTime [protected]

Definition at line 75 of file MHV_Debounce.h.

4.5.4.3 MHV_PinChangeManager* MHV_Debounce::_pinChangeManager [protected]

Definition at line 77 of file MHV_Debounce.h.

4.5.4.4 MHV_DEBOUNCE_PIN MHV_Debounce::_pins[MHV_PC_INT_COUNT] [protected]

Definition at line 73 of file MHV_Debounce.h.

4.5.4.5 MHV_TIMESTAMP MHV_Debounce::_repeatTime [protected]

Definition at line 76 of file MHV_Debounce.h.

4.5.4.6 MHV_RTC* MHV_Debounce::_rtc [protected]

Definition at line 72 of file MHV_Debounce.h.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Debounce.h](#)
- [A:/eclipse/mhvlb/MHV_Debounce.cpp](#)

4.6 MHV_DebounceListener Class Reference

```
#include <MHV_Debounce.h>
```

Public Member Functions

- virtual void [singlePress](#) (uint8_t pcInt, [MHV_TIMESTAMP](#) *heldFor)=0
- virtual void [heldDown](#) (uint8_t pcInt, [MHV_TIMESTAMP](#) *heldFor)=0

4.6.1 Detailed Description

Definition at line 56 of file MHV_Debounce.h.

4.6.2 Member Function Documentation

4.6.2.1 `virtual void MHV_DebounceListener::heldDown (uint8_t pcInt, MHV_TIMESTAMP * heldFor) [pure virtual]`

4.6.2.2 `virtual void MHV_DebounceListener::singlePress (uint8_t pcInt, MHV_TIMESTAMP * heldFor) [pure virtual]`

The documentation for this class was generated from the following file:

- [A:/eclipse/mhvlb/MHV_Debounce.h](#)

4.7 mhv_debouncePin Struct Reference

```
#include <MHV_Debounce.h>
```

Public Attributes

- `uint8_t` [previous](#)
- `MHV_TIMESTAMP` [timestamp](#)
- `MHV_DebounceListener *` [listener](#)
- `bool` [held](#)

4.7.1 Detailed Description

Definition at line 62 of file `MHV_Debounce.h`.

4.7.2 Member Data Documentation

4.7.2.1 `bool mhv_debouncePin::held`

Definition at line 66 of file `MHV_Debounce.h`.

4.7.2.2 `MHV_DebounceListener* mhv_debouncePin::listener`

Definition at line 65 of file `MHV_Debounce.h`.

4.7.2.3 `uint8_t mhv_debouncePin::previous`

Definition at line 63 of file `MHV_Debounce.h`.

4.7.2.4 MHV_TIMESTAMP mhv_debouncePin::timestamp

Definition at line 64 of file MHV_Debounce.h.

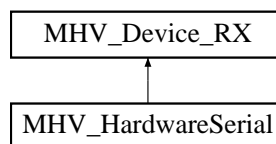
The documentation for this struct was generated from the following file:

- [A:/eclipse/mhvlb/MHV_Debounce.h](#)

4.8 MHV_Device_RX Class Reference

```
#include <MHV_Device_RX.h>
```

Inheritance diagram for MHV_Device_RX:



Public Member Functions

- int [asyncReadLine](#) (char *buffer, uint8_t bufferLength)
- int [busyReadLine](#) (char *buffer, uint8_t bufferLength)
- int [read](#) ()
- void [flush](#) ()
- bool [ready](#) ()
- void [registerListener](#) (MHV_RXListener *listener)
- void [deregisterListener](#) ()
- void [handleEvents](#) ()

Protected Member Functions

- [MHV_Device_RX](#) (MHV_RingBuffer *rxBuffer)

Protected Attributes

- [MHV_RingBuffer](#) * [_rxBuffer](#)
- [MHV_RXListener](#) * [_listener](#)

4.8.1 Detailed Description

Definition at line 48 of file MHV_Device_RX.h.

4.8.2 Constructor & Destructor Documentation

4.8.2.1 MHV_Device_RX::MHV_Device_RX (MHV_RingBuffer * rxBuffer) [protected]

Constructor

Parameters

<i>rxBuffer</i>	a buffer to read into
-----------------	-----------------------

Definition at line 41 of file MHV_Device_RX.cpp.

4.8.3 Member Function Documentation

4.8.3.1 int MHV_Device_RX::asyncReadLine (char * buffer, uint8_t bufferSize)

If we have a line, copy it into a buffer & null terminate, stripping CR/LF returns 0 if we have successfully copied a line returns -1 if there was no line available returns -2 if the buffer was too small returns -3 if we have reached the end of the ringbuffer with no line terminator

Definition at line 53 of file MHV_Device_RX.cpp.

4.8.3.2 int MHV_Device_RX::busyReadLine (char * buffer, uint8_t bufferSize)

If we have a line, copy it into a buffer & null terminate, stripping CR/LF Blocks until a line is available

Returns

- 0 if we have successfully copied a line
- 2 if the buffer was too small
- 3 if we have reached the end of the ringbuffer with no line terminator

Definition at line 91 of file MHV_Device_RX.cpp.

4.8.3.3 void MHV_Device_RX::deregisterListener ()

Deregister interest for lines/overflows from an RX device

Definition at line 141 of file MHV_Device_RX.cpp.

4.8.3.4 void MHV_Device_RX::flush ()

Discard remaining data in the receive buffer

Definition at line 108 of file MHV_Device_RX.cpp.

4.8.3.5 void MHV_Device_RX::handleEvents ()

Call from the main loop to handle any events

Definition at line 149 of file MHV_Device_RX.cpp.

4.8.3.6 int MHV_Device_RX::read (void)

Read a byte from the receive buffer

Returns

the byte, or -1 if there is nothing to read

Definition at line 101 of file MHV_Device_RX.cpp.

4.8.3.7 bool MHV_Device_RX::ready ()

Check if a line is ready, or the ringbuffer is full

Returns

true if either of the situations occur0

Definition at line 116 of file MHV_Device_RX.cpp.

4.8.3.8 void MHV_Device_RX::registerListener (MHV_RXListener * listener)

Register interest for lines/overflows from an RX device

Parameters

<i>listener</i>	an MHV_RXListener to notify that the device is ready
-----------------	----------------------------------------------------------------------

Definition at line 134 of file MHV_Device_RX.cpp.

4.8.4 Member Data Documentation**4.8.4.1 MHV_RXListener* MHV_Device_RX::_listener** [protected]

Definition at line 51 of file MHV_Device_RX.h.

4.8.4.2 MHV_RingBuffer* MHV_Device_RX::_rxBuffer [protected]

Definition at line 50 of file MHV_Device_RX.h.

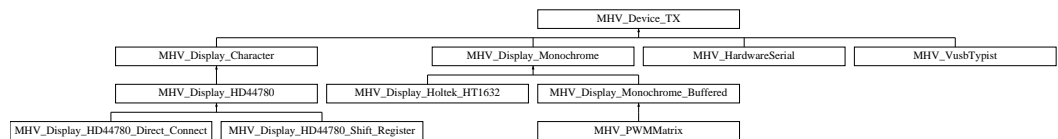
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Device_RX.h](#)
- [A:/eclipse/mhvlb/MHV_Device_RX.cpp](#)

4.9 MHV_Device_TX Class Reference

```
#include <MHV_Device_TX.h>
```

Inheritance diagram for MHV_Device_TX:



Public Member Functions

- bool [canWrite](#) ()
- bool [write](#) (const char *buffer)
- bool [write](#) (const char *buffer, uint16_t length)
- bool [write](#) (const char *buffer, void(*completeFunction)(const char *))
- bool [write](#) (const char *buffer, uint16_t length, void(*completeFunction)(const char *))
- bool [write_P](#) (PGM_P buffer)
- bool [write_P](#) (PGM_P buffer, uint16_t length)

Protected Member Functions

- [MHV_Device_TX](#) ([MHV_RingBuffer](#) *txPointers)
- virtual void [runTxBuffers](#) ()=0
- bool [moreTX](#) ()
- int [nextCharacter](#) ()

Protected Attributes

- [MHV_TX_BUFFER](#) _currentTx
- [MHV_RingBuffer](#) * _txPointers
- const char * _tx

4.9.1 Detailed Description

Definition at line 54 of file MHV_Device_TX.h.

4.9.2 Constructor & Destructor Documentation

4.9.2.1 MHV_Device_TX::MHV_Device_TX (MHV_RingBuffer * txPointers) [protected]

Constructor

Parameters

<i>txPointers</i>	A ringbuffer to store tx pointers in
-------------------	--------------------------------------

Definition at line 41 of file MHV_Device_TX.cpp.

4.9.3 Member Function Documentation

4.9.3.1 bool MHV_Device_TX::canWrite ()

Definition at line 73 of file MHV_Device_TX.cpp.

4.9.3.2 bool MHV_Device_TX::moreTX () [protected]

Called when a buffer has been processed

Returns

true if there is another buffer to process

Definition at line 55 of file MHV_Device_TX.cpp.

4.9.3.3 int MHV_Device_TX::nextCharacter () [protected]

Called by children to get a character to transmit

Returns

the character, or -1 if there is nothing left

Definition at line 81 of file MHV_Device_TX.cpp.

4.9.3.4 virtual void MHV_Device_TX::runTxBuffers () [protected, pure virtual]

Implemented in [MHV_HardwareSerial](#), [MHV_Display_Character](#), [MHV_Display_Monochrome](#), and [MHV_VusbTypist](#).

4.9.3.5 `bool MHV_Device_TX::write (const char * buffer)`

Write a string asynchronously

Parameters

<i>buffer</i>	the string
---------------	------------

Returns

false on success, true on failure

Definition at line 142 of file MHV_Device_TX.cpp.

4.9.3.6 `bool MHV_Device_TX::write (const char * buffer, uint16_t length)`

Write a buffer asynchronously

Parameters

<i>buffer</i>	the buffer
<i>length</i>	the length of the buffer

Returns

0 on success 1 if there is already a string being sent

Definition at line 225 of file MHV_Device_TX.cpp.

4.9.3.7 `bool MHV_Device_TX::write (const char * buffer, void(*) (const char *) completeFunction)`

Write a string asynchronously

Parameters

<i>buffer</i>	the string
<i>complete-Function</i>	a function to call when the string has been written (the string is passed as a parameter)

Returns

false on success, true on failure

Definition at line 169 of file MHV_Device_TX.cpp.

4.9.3.8 `bool MHV_Device_TX::write (const char * buffer, uint16_t length, void(*) (const char *) completeFunction)`

Write a buffer asynchronously

Parameters

<i>buffer</i>	the buffer
<i>length</i>	the length of the buffer
<i>complete-Function</i>	a function to call when the string has been written (the string is passed as a parameter)

Returns

0 on success 1 if there is already a string being sent

Definition at line 254 of file MHV_Device_TX.cpp.

4.9.3.9 `bool MHV_Device_TX::write_P (PGM_P buffer)`

Write a progmem string asynchronously

Parameters

<i>buffer</i>	the progmem string
---------------	--------------------

Returns

false on success, true on failure

Definition at line 116 of file MHV_Device_TX.cpp.

4.9.3.10 `bool MHV_Device_TX::write_P (PGM_P buffer, uint16_t length)`

Write a buffer asynchronously

Parameters

<i>buffer</i>	the buffer
<i>length</i>	the length of the buffer

Returns

0 on success 1 if there is already a string being sent

Definition at line 197 of file MHV_Device_TX.cpp.

4.9.4 Member Data Documentation

4.9.4.1 MHV_TX_BUFFER MHV_Device_TX::_currentTx [protected]

Definition at line 56 of file MHV_Device_TX.h.

4.9.4.2 const char* MHV_Device_TX::_tx [protected]

Definition at line 58 of file MHV_Device_TX.h.

4.9.4.3 MHV_RingBuffer* MHV_Device_TX::_txPointers [protected]

Definition at line 57 of file MHV_Device_TX.h.

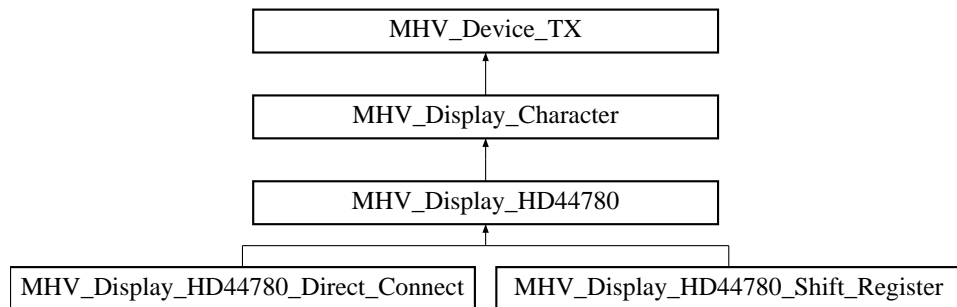
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Device_TX.h](#)
- [A:/eclipse/mhvlb/MHV_Device_TX.cpp](#)

4.10 MHV_Display_Character Class Reference

```
#include <MHV_Display_Character.h>
```

Inheritance diagram for MHV_Display_Character:



Public Member Functions

- [MHV_Display_Character](#) (uint16_t cols, uint16_t rows, [MHV_RingBuffer](#) *tx-Buffer)
- void [writeChar](#) (char character)
- void [scrollVertically](#) ()
- void [setCursor](#) (uint16_t col, uint16_t row)
- void [runTxBuffers](#) ()
- bool [txAnimation](#) (uint16_t row)

- uint8_t [getWidth](#) ()
- uint8_t [getHeight](#) ()
- bool [writeString](#) (int16_t *offsetX, uint16_t offsetY, const char *string)
- bool [writeBuffer](#) (int16_t *offsetX, uint16_t offsetY, const char *buffer, uint16_t length)
- bool [writeString_P](#) (int16_t *offsetX, uint16_t offsetY, PGM_P string)
- bool [writeBuffer_P](#) (int16_t *offsetX, uint16_t offsetY, PGM_P buffer, uint16_t length)
- void [setWrap](#) (bool shouldWrap)
- void [setScroll](#) (bool shouldScroll)
- virtual void [_writeChar](#) (char character)=0
- virtual char [_readChar](#) ()=0
- virtual void [_setCursor](#) (uint16_t col, uint16_t row)=0
- virtual void [clear](#) ()=0

Protected Attributes

- uint16_t [_rowCount](#)
- uint16_t [_colCount](#)
- int16_t [_txOffset](#)
- uint16_t [_currentRow](#)
- uint16_t [_currentCol](#)
- bool [_wrap](#)
- bool [_scroll](#)

4.10.1 Detailed Description

Definition at line 32 of file MHV_Display_Character.h.

4.10.2 Constructor & Destructor Documentation

4.10.2.1 **MHV_Display_Character::MHV_Display_Character** (uint16_t *colCount*, uint16_t *rowCount*, MHV_RingBuffer * *txBuffers*)

A Character charcter display Origin (0,0) is bottom left Create a new character display

Parameters

<i>colCount</i>	the number of columns
<i>rowCount</i>	the number of rows
<i>txBuffers</i>	buffers to use for text writing

Definition at line 40 of file MHV_Display_Character.cpp.

4.10.3 Member Function Documentation

4.10.3.1 `virtual char MHV_Display_Character::readChar ()` [pure virtual]

Implemented in [MHV_Display_HD44780](#).

4.10.3.2 `virtual void MHV_Display_Character::setCursor (uint16_t col, uint16_t row)` [pure virtual]

Implemented in [MHV_Display_HD44780](#).

4.10.3.3 `virtual void MHV_Display_Character::writeChar (char character)` [pure virtual]

Implemented in [MHV_Display_HD44780](#).

4.10.3.4 `virtual void MHV_Display_Character::clear ()` [pure virtual]

Implemented in [MHV_Display_HD44780](#).

4.10.3.5 `uint8_t MHV_Display_Character::getHeight ()`

Get the width of the display

Definition at line 61 of file `MHV_Display_Character.cpp`.

4.10.3.6 `uint8_t MHV_Display_Character::getWidth ()`

Get the width of the display

Definition at line 54 of file `MHV_Display_Character.cpp`.

4.10.3.7 `void MHV_Display_Character::runTxBuffers ()` [virtual]

Start rendering TX buffers

Implements [MHV_Device_TX](#).

Definition at line 186 of file `MHV_Display_Character.cpp`.

4.10.3.8 `void MHV_Display_Character::scrollVertically ()`

Scroll the display up, leaving a blank line at the bottom

Definition at line 325 of file `MHV_Display_Character.cpp`.

4.10.3.9 void MHV_Display_Character::setCursor (uint16_t *col*, uint16_t *row*)

Position the cursor

Parameters

<i>col</i>	the column to set
<i>row</i>	the row to set

Definition at line 253 of file MHV_Display_Character.cpp.

4.10.3.10 void MHV_Display_Character::setScroll (bool *shouldScroll*)

Should the automatically scroll characters vertically?

Parameters

<i>shouldScroll</i>	true to scroll
---------------------	----------------

Definition at line 244 of file MHV_Display_Character.cpp.

4.10.3.11 void MHV_Display_Character::setWrap (bool *shouldWrap*)

Should the display automatically wrap characters?

Parameters

<i>shouldWrap</i>	true to wrap
-------------------	--------------

Definition at line 236 of file MHV_Display_Character.cpp.

4.10.3.12 bool MHV_Display_Character::txAnimation (uint16_t *row*)

Render a frame of TX buffer animation - scrolls text from right to left, before moving to the next buffer

Parameters

<i>row</i>	the vertical pixel offset to start writing at
------------	-----------------------------------------------

Returns

true if there are more frames to be rendered

Definition at line 196 of file MHV_Display_Character.cpp.

4.10.3.13 `bool MHV_Display_Character::writeBuffer (int16_t * offsetX, uint16_t offsetY, const char * buffer, uint16_t length)`

Write a buffer to the display

Parameters

<i>offsetX</i>	the horizontal offset to start writing at
<i>offsetY</i>	the vertical offset to start writing at
<i>buffer</i>	the buffer to write
<i>length</i>	the length of the buffer

Returns

true if anything was written

Definition at line 104 of file MHV_Display_Character.cpp.

4.10.3.14 `bool MHV_Display_Character::writeBuffer_P (int16_t * offsetX, uint16_t offsetY, PGM_P buffer, uint16_t length)`

Write a PROGMEM buffer to the display

Parameters

<i>offsetX</i>	the horizontal offset to start writing at
<i>offsetY</i>	the vertical offset to start writing at
<i>buffer</i>	the buffer to write
<i>length</i>	the length of the buffer

Returns

true if anything was written

Definition at line 164 of file MHV_Display_Character.cpp.

4.10.3.15 `void MHV_Display_Character::writeChar (char character)`

Write a character to the display Will interpret the following special characters: \b - Backspace \t Tab \n Newline

Parameters

<i>character</i>	the character to write
------------------	------------------------

Definition at line 268 of file MHV_Display_Character.cpp.

4.10.3.16 `bool MHV_Display_Character::writeString (int16_t * offsetX, uint16_t offsetY, const char * string)`

Write a string to the display

Parameters

<i>offsetX</i>	the horizontal offset to start writing at
<i>offsetY</i>	the vertical offset to start writing at
<i>string</i>	the string to write

Returns

true if anything was written

Definition at line 72 of file MHV_Display_Character.cpp.

4.10.3.17 `bool MHV_Display_Character::writeString_P (int16_t * offsetX, uint16_t offsetY, PGM_P string)`

Write a PROGMEM string to the display

Parameters

<i>offsetX</i>	the horizontal offset to start writing at
<i>offsetY</i>	the vertical offset to start writing at
<i>string</i>	the string to write

Returns

true if anything was written

Definition at line 132 of file MHV_Display_Character.cpp.

4.10.4 Member Data Documentation

4.10.4.1 `uint16_t MHV_Display_Character::_colCount` [protected]

Definition at line 35 of file MHV_Display_Character.h.

4.10.4.2 `uint16_t MHV_Display_Character::_currentCol` [protected]

Definition at line 38 of file MHV_Display_Character.h.

4.10.4.3 `uint16_t MHV_Display_Character::_currentRow` [protected]

Definition at line 37 of file MHV_Display_Character.h.

4.10.4.4 `uint16_t MHV_Display_Character::_rowCount` [protected]

Definition at line 34 of file `MHV_Display_Character.h`.

4.10.4.5 `bool MHV_Display_Character::_scroll` [protected]

Definition at line 40 of file `MHV_Display_Character.h`.

4.10.4.6 `int16_t MHV_Display_Character::_txOffset` [protected]

Definition at line 36 of file `MHV_Display_Character.h`.

4.10.4.7 `bool MHV_Display_Character::_wrap` [protected]

Definition at line 39 of file `MHV_Display_Character.h`.

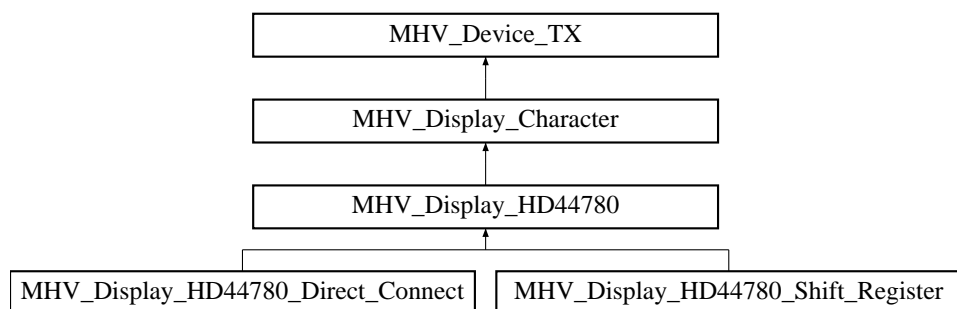
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Display_Character.h](#)
- [A:/eclipse/mhvlb/MHV_Display_Character.cpp](#)

4.11 MHV_Display_HD44780 Class Reference

```
#include <MHV_Display_HD44780.h>
```

Inheritance diagram for `MHV_Display_HD44780`:



Public Member Functions

- `MHV_Display_HD44780` (`uint8_t colCount`, `uint16_t rowCount`, `MHV_RingBuffer *txBuffers`)
- `void init` (`bool byteMode`, `bool multiLine`, `bool bigFont`, `bool cursorOn`, `bool cursorBlink`, `bool left2right`, `bool scroll`)
- `void clear` ()

- void [entryMode](#) (bool left2Right, bool scroll)
- void [control](#) (bool displayOn, bool cursorOn, bool cursorBlink)

Protected Member Functions

- void [writeCommand](#) (MHV_HD44780_COMMAND command, uint8_t data)
- void [function](#) (bool byteMode, bool multiLine, bool bigFont)
- void [addressCGRAM](#) (uint8_t address)
- void [addressDDRAM](#) (uint8_t address)
- virtual void [writeByte](#) (uint8_t byte, bool rs)=0
- virtual uint8_t [readByte](#) (bool rs)=0
- void [_setCursor](#) (uint8_t col, uint8_t row)
- void [_setCursor](#) (uint16_t col, uint16_t row)
- void [_writeChar](#) (char character)
- char [_readChar](#) ()
- virtual bool [isBusy](#) ()=0
- virtual void [delay](#) (MHV_HD44780_COMMAND command)=0

Protected Attributes

- uint16_t [_ticks](#)
- uint16_t [_animateTicks](#)
- bool [_mustDelay](#)
- bool [_byteMode](#)

4.11.1 Detailed Description

Definition at line 45 of file MHV_Display_HD44780.h.

4.11.2 Constructor & Destructor Documentation

4.11.2.1 MHV_Display_HD44780::MHV_Display_HD44780 (uint8_t *colCount*, uint16_t *rowCount*, MHV_RingBuffer * *txBuffers*)

A class for operating HD44780 based LCD displays (and compatible) Data port layout:
Bit description n DB4 n+1 DB5 n+2 DB6 n+3 DB7

Control port layout: n RS Register Select n+1 R/W Read/Write n+2 E Enable

Visual port layout: n Contrast (V0) n+1 LED Positive

Parameters

<i>colCount</i>	the number of columns on the display
<i>rowCount</i>	the number of rows on the display
<i>txBuffers</i>	buffers for async writing

Definition at line 54 of file MHV_Display_HD44780.cpp.

4.11.3 Member Function Documentation

4.11.3.1 `char MHV_Display_HD44780::readChar ()` [protected, virtual]

Read a character from the display at the current location, incrementing the location by 1

Returns

the character at the current location

Implements [MHV_Display_Character](#).

Definition at line 75 of file MHV_Display_HD44780.cpp.

4.11.3.2 `void MHV_Display_HD44780::setCursor (uint8_t col, uint8_t row)` [protected]

Move the cursor to a location, so the next writeChar will write a character at that location (Origin is at the bottom left)

Parameters

<i>col</i>	the column to put the character
<i>row</i>	the row to put the character

Definition at line 101 of file MHV_Display_HD44780.cpp.

4.11.3.3 `void MHV_Display_HD44780::setCursor (uint16_t col, uint16_t row)` [protected, virtual]

Move the cursor to a location, so the next writeChar will write a character at that location (Origin is at the bottom left)

Parameters

<i>col</i>	the column to put the character
<i>row</i>	the row to put the character

Implements [MHV_Display_Character](#).

Definition at line 86 of file MHV_Display_HD44780.cpp.

4.11.3.4 `void MHV_Display_HD44780::writeChar (char character)` [protected, virtual]

Write a character to the display at the current location, incrementing the location by 1

Parameters

<i>character</i>	the character to write
------------------	------------------------

Implements [MHV_Display_Character](#).

Definition at line 65 of file MHV_Display_HD44780.cpp.

4.11.3.5 void MHV_Display_HD44780::addressCGRAM (uint8_t *address*) [protected]

Set the CGRAM address

Parameters

<i>address</i>	the CGRAM address
----------------	-------------------

Definition at line 194 of file MHV_Display_HD44780.cpp.

4.11.3.6 void MHV_Display_HD44780::addressDDRAM (uint8_t *address*) [protected]

Set the DDRAM address

Parameters

<i>address</i>	the DDRAM address
----------------	-------------------

Definition at line 204 of file MHV_Display_HD44780.cpp.

4.11.3.7 void MHV_Display_HD44780::clear () [virtual]

Clear the display

Implements [MHV_Display_Character](#).

Definition at line 137 of file MHV_Display_HD44780.cpp.

4.11.3.8 void MHV_Display_HD44780::control (bool *displayOn*, bool *cursorOn*, bool *cursorBlink*)

Set parameters on the display

Parameters

<i>displayOn</i>	turn the display on
<i>cursorOn</i>	turn the cursor on
<i>cursorBlink</i>	blink the cursor

Definition at line 164 of file MHV_Display_HD44780.cpp.

4.11.3.9 `virtual void MHV_Display_HD44780::delay (MHV_HD44780_COMMAND command)` [protected, pure virtual]

Implemented in [MHV_Display_HD44780_Direct_Connect](#), and [MHV_Display_HD44780_Shift_Register](#).

4.11.3.10 `void MHV_Display_HD44780::entryMode (bool left2Right, bool scroll)`

Set the entry mode - allows for left or right printing, allows for scrolling display or moving cursor

Parameters

<i>left2Right</i>	true for text reading left to right
<i>scroll</i>	true to scroll text rather than moving the cursor

Definition at line 150 of file `MHV_Display_HD44780.cpp`.

4.11.3.11 `void MHV_Display_HD44780::function (bool byteMode, bool multiLine, bool bigFont)` [protected]

Initialise the display

Parameters

<i>byteMode</i>	true to use 8 bit protocol
<i>multiLine</i>	true if there is more than 1 line
<i>bigFont</i>	true to use 5x11 fonts, false for 5x8

Definition at line 178 of file `MHV_Display_HD44780.cpp`.

4.11.3.12 `void MHV_Display_HD44780::init (bool byteMode, bool multiLine, bool bigFont, bool cursorOn, bool cursorBlink, bool left2right, bool scroll)`

Initialise the display

Parameters

<i>byteMode</i>	true to use 8 bit transfers
<i>multiLine</i>	true if there is more than 1 line
<i>bigFont</i>	true to use 5x11 fonts, false for 5x8
<i>cursorOn</i>	turn the cursor on
<i>cursorBlink</i>	blink the cursor
<i>left2right</i>	true for text reading left to right
<i>scroll</i>	true to scroll text rather than moving the cursor

Definition at line 221 of file `MHV_Display_HD44780.cpp`.

4.11.3.13 `virtual bool MHV_Display_HD44780::isBusy ()` [protected, pure virtual]

Implemented in [MHV_Display_HD44780_Direct_Connect](#), and [MHV_Display_HD44780_Shift_Register](#).

4.11.3.14 `virtual uint8_t MHV_Display_HD44780::readByte (bool rs)` [protected, pure virtual]

Implemented in [MHV_Display_HD44780_Direct_Connect](#), and [MHV_Display_HD44780_Shift_Register](#).

4.11.3.15 `virtual void MHV_Display_HD44780::writeByte (uint8_t byte, bool rs)` [protected, pure virtual]

Implemented in [MHV_Display_HD44780_Direct_Connect](#), and [MHV_Display_HD44780_Shift_Register](#).

4.11.3.16 `void MHV_Display_HD44780::writeCommand (MHV_HD44780_COMMAND command, uint8_t data)` [protected]

Send a command to the display

Definition at line 121 of file `MHV_Display_HD44780.cpp`.

4.11.4 Member Data Documentation

4.11.4.1 `uint16_t MHV_Display_HD44780::_animateTicks` [protected]

Definition at line 48 of file `MHV_Display_HD44780.h`.

4.11.4.2 `bool MHV_Display_HD44780::_byteMode` [protected]

Definition at line 50 of file `MHV_Display_HD44780.h`.

4.11.4.3 `bool MHV_Display_HD44780::_mustDelay` [protected]

Definition at line 49 of file `MHV_Display_HD44780.h`.

4.11.4.4 `uint16_t MHV_Display_HD44780::_ticks` [protected]

Definition at line 47 of file `MHV_Display_HD44780.h`.

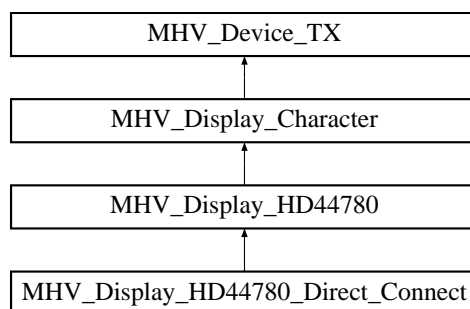
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Display_HD44780.h](#)
- [A:/eclipse/mhvlb/MHV_Display_HD44780.cpp](#)

4.12 MHV_Display_HD44780_Direct_Connect Class Reference

```
#include <MHV_Display_HD44780_Direct_Connect.h>
```

Inheritance diagram for MHV_Display_HD44780_Direct_Connect:



Public Member Functions

- [MHV_Display_HD44780_Direct_Connect](#) (volatile uint8_t *dataDir, volatile uint8_t *dataOut, volatile uint8_t *dataIn, uint8_t dataPin, int8_t dataPinchangeInterrupt, volatile uint8_t *controlDir, volatile uint8_t *controlOut, volatile uint8_t *controlIn, uint8_t controlPin, int8_t controlPinchangeInterrupt, volatile uint8_t *visualDir, volatile uint8_t *visualOut, volatile uint8_t *visualIn, uint8_t visualPin, int8_t visualPinchangeInterrupt, uint8_t colCount, uint16_t rowCount, [MHV_RingBuffer](#) *txBuffers)
- [MHV_Display_HD44780_Direct_Connect](#) (volatile uint8_t *dataDir, volatile uint8_t *dataOut, volatile uint8_t *dataIn, uint8_t dataPin, int8_t dataPinchangeInterrupt, volatile uint8_t *controlDir, volatile uint8_t *controlOut, volatile uint8_t *controlIn, uint8_t controlPin, int8_t controlPinchangeInterrupt, uint8_t colCount, uint16_t rowCount, [MHV_RingBuffer](#) *txBuffers)
- void [setBacklight](#) (uint8_t value)
- void [setContrast](#) (uint8_t value)
- void [tickPWM](#) ()
- void [init](#) (bool multiLine, bool bigFont, bool cursorOn, bool cursorBlink, bool left2right, bool scroll)

Protected Member Functions

- void [writeByte](#) (uint8_t byte, bool rs)
- void [writeNibble](#) (uint8_t nibble, bool rs)
- uint8_t [readByte](#) (bool rs)
- uint8_t [readNibble](#) (bool rs)

- bool [isBusy](#) ()
- void [delay](#) (MHV_HD44780_COMMAND command)

Protected Attributes

- volatile uint8_t * [_dataDir](#)
- volatile uint8_t * [_dataOut](#)
- volatile uint8_t * [_dataIn](#)
- uint8_t [_dataPin](#)
- uint8_t [_dataMask](#)
- volatile uint8_t * [_controlOut](#)
- uint8_t [_controlPin](#)
- volatile uint8_t * [_visualOut](#)
- uint8_t [_visualPin](#)
- uint8_t [_brightness](#)
- uint8_t [_contrast](#)

4.12.1 Detailed Description

Definition at line 32 of file MHV_Display_HD44780_Direct_Connect.h.

4.12.2 Constructor & Destructor Documentation

4.12.2.1 MHV_Display_HD44780_Direct_Connect::MHV_Display_HD44780_Direct_Connect (volatile uint8_t * *dataDir*, volatile uint8_t * *dataOut*, volatile uint8_t * *dataIn*, uint8_t *dataPin*, int8_t *dataPinchangeInterrupt*, volatile uint8_t * *controlDir*, volatile uint8_t * *controlOut*, volatile uint8_t * *controlIn*, uint8_t *controlPin*, int8_t *controlPinchangeInterrupt*, volatile uint8_t * *visualDir*, volatile uint8_t * *visualOut*, volatile uint8_t * *visualIn*, uint8_t *visualPin*, int8_t *visualPinchangeInterrupt*, uint8_t *colCount*, uint16_t *rowCount*, MHV_RingBuffer * *txBuffers*)

A class for operating HD44780 based LCD displays (and compatible) in 4 bit mode Data port layout: Bit description n DB4 n+1 DB5 n+2 DB6 n+3 DB7

Control port layout: n RS Register Select n+1 R/W Read/Write n+2 E Enable

Visual port layout: n Contrast (V0) n+1 LED Positive

Parameters

<i>dataDir</i>	A member of the MHV_PIN_* macro pin declaration for the first bit of the data port DB4..DB7 (will use a nibble starting at this bit)
<i>dataOut</i>	A member of the MHV_PIN_* macro
<i>dataIn</i>	A member of the MHV_PIN_* macro
<i>dataPin</i>	A member of the MHV_PIN_* macro
<i>data-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro

<i>controlDir</i>	A member of the MHV_PIN_* macro pin declaration for the first bit of the control port (will use 3 bits)
<i>controlOut</i>	A member of the MHV_PIN_* macro
<i>controlIn</i>	A member of the MHV_PIN_* macro
<i>controlPin</i>	A member of the MHV_PIN_* macro
<i>control-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>visualDir</i>	A member of the MHV_PIN_* macro pin declaration for the first bit of the visual port (will use 2 bits)
<i>visualOut</i>	A member of the MHV_PIN_* macro
<i>visualIn</i>	A member of the MHV_PIN_* macro
<i>visualPin</i>	A member of the MHV_PIN_* macro
<i>visual-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>colCount</i>	the number of columns on the display
<i>rowCount</i>	the number of rows on the display
<i>txBuffers</i>	buffers for async writing

Definition at line 89 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.2.2 MHV_Display_HD44780_Direct_Connect::MHV_Display_HD44780_Direct_Connect
 (volatile uint8_t * *dataDir*, volatile uint8_t * *dataOut*, volatile uint8_t * *dataIn*,
 uint8_t *dataPin*, int8_t *dataPinchangeInterrupt*, volatile uint8_t * *controlDir*,
 volatile uint8_t * *controlOut*, volatile uint8_t * *controlIn*, uint8_t *controlPin*, int8_t
controlPinchangeInterrupt, uint8_t *colCount*, uint16_t *rowCount*, MHV_RingBuffer
 * *txBuffers*)

An alternate constructor without visual pins - if this constructor is used, tickPWM behaviour is undefined and will like overwrite random bits of memory, so don't call it

Parameters

<i>dataDir</i>	A member of the MHV_PIN_* macro pin declaration for the first bit of the data port DB4..DB7 (will use a nibble starting at this bit)
<i>dataOut</i>	A member of the MHV_PIN_* macro
<i>dataIn</i>	A member of the MHV_PIN_* macro
<i>dataPin</i>	A member of the MHV_PIN_* macro
<i>data-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>controlDir</i>	A member of the MHV_PIN_* macro pin declaration for the first bit of the control port (will use 3 bits)
<i>controlOut</i>	A member of the MHV_PIN_* macro
<i>controlIn</i>	A member of the MHV_PIN_* macro
<i>controlPin</i>	A member of the MHV_PIN_* macro

<i>control-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>colCount</i>	the number of columns on the display
<i>rowCount</i>	the number of rows on the display
<i>txBuffers</i>	buffers for async writing

Definition at line 139 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.3 Member Function Documentation

4.12.3.1 void MHV_Display_HD44780_Direct_Connect::delay (MHV_HD44780_COMMAND *command*) [protected, virtual]

Delay function No delays required as we can check whether the display is busy

Implements [MHV_Display_HD44780](#).

Definition at line 299 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.3.2 void MHV_Display_HD44780_Direct_Connect::init (bool *multiLine*, bool *bigFont*, bool *cursorOn*, bool *cursorBlink*, bool *left2right*, bool *scroll*)

Initialise the display

Parameters

<i>multiLine</i>	true if there is more than 1 line
<i>bigFont</i>	true to use 5x11 fonts, false for 5x8
<i>cursorOn</i>	turn the cursor on
<i>cursorBlink</i>	blink the cursor
<i>left2right</i>	true for text reading left to right
<i>scroll</i>	true to scroll text rather than moving the cursor

Definition at line 312 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.3.3 bool MHV_Display_HD44780_Direct_Connect::isBusy () [protected, virtual]

Check if the display is busy

Returns

true if the display is busy

Implements [MHV_Display_HD44780](#).

Definition at line 234 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.3.4 `uint8_t MHV_Display_HD44780_Direct_Connect::readByte (bool rs)`
[protected, virtual]

Read a byte from the display

Parameters

<i>rs</i>	true to set the RS pin
-----------	------------------------

Implements [MHV_Display_HD44780](#).

Definition at line 200 of file `MHV_Display_HD44780_Direct_Connect.cpp`.

4.12.3.5 `uint8_t MHV_Display_HD44780_Direct_Connect::readNibble (bool rs)`
[protected]

Read a nibble from the display

Parameters

<i>rs</i>	true to set the RS pin
-----------	------------------------

Definition at line 212 of file `MHV_Display_HD44780_Direct_Connect.cpp`.

4.12.3.6 `void MHV_Display_HD44780_Direct_Connect::setBacklight (uint8_t value)`

Manipulate the backlight

Parameters

<i>value</i>	the brightness of the backlight (0..15)
--------------	-----------------------------------------

Definition at line 255 of file `MHV_Display_HD44780_Direct_Connect.cpp`.

4.12.3.7 `void MHV_Display_HD44780_Direct_Connect::setContrast (uint8_t value)`

Set the contrast

Parameters

<i>value</i>	the contrast of the display(0..15)
--------------	------------------------------------

Definition at line 266 of file `MHV_Display_HD44780_Direct_Connect.cpp`.

4.12.3.8 `void MHV_Display_HD44780_Direct_Connect::tickPWM ()`

Tick the display for PWM - this should be called every 500 microseconds

Definition at line 278 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.3.9 void MHV_Display_HD44780_Direct_Connect::writeByte (uint8_t *byte*, bool *rs*)
[protected, virtual]

Write a byte to the display

Parameters

<i>byte</i>	the data to write
<i>rs</i>	true to set the RS pin

Implements [MHV_Display_HD44780](#).

Definition at line 167 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.3.10 void MHV_Display_HD44780_Direct_Connect::writeNibble (uint8_t *nibble*, bool *rs*)
[protected]

Write a nibble to the display

Parameters

<i>nibble</i>	the data to write (lower 4 bits)
<i>rs</i>	true to set the RS pin

Definition at line 179 of file MHV_Display_HD44780_Direct_Connect.cpp.

4.12.4 Member Data Documentation

4.12.4.1 uint8_t MHV_Display_HD44780_Direct_Connect::_brightness
[protected]

Definition at line 44 of file MHV_Display_HD44780_Direct_Connect.h.

4.12.4.2 uint8_t MHV_Display_HD44780_Direct_Connect::_contrast
[protected]

Definition at line 45 of file MHV_Display_HD44780_Direct_Connect.h.

4.12.4.3 volatile uint8_t* MHV_Display_HD44780_Direct_Connect::_controlOut
[protected]

Definition at line 39 of file MHV_Display_HD44780_Direct_Connect.h.

4.12.4.4 `uint8_t MHV_Display_HD44780_Direct_Connect::_controlPin`
[protected]

Definition at line 40 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.5 `volatile uint8_t* MHV_Display_HD44780_Direct_Connect::_dataDir`
[protected]

Definition at line 34 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.6 `volatile uint8_t* MHV_Display_HD44780_Direct_Connect::_dataIn`
[protected]

Definition at line 36 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.7 `uint8_t MHV_Display_HD44780_Direct_Connect::_dataMask`
[protected]

Definition at line 38 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.8 `volatile uint8_t* MHV_Display_HD44780_Direct_Connect::_dataOut`
[protected]

Definition at line 35 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.9 `uint8_t MHV_Display_HD44780_Direct_Connect::_dataPin`
[protected]

Definition at line 37 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.10 `volatile uint8_t* MHV_Display_HD44780_Direct_Connect::_visualOut`
[protected]

Definition at line 41 of file `MHV_Display_HD44780_Direct_Connect.h`.

4.12.4.11 `uint8_t MHV_Display_HD44780_Direct_Connect::_visualPin`
[protected]

Definition at line 42 of file `MHV_Display_HD44780_Direct_Connect.h`.

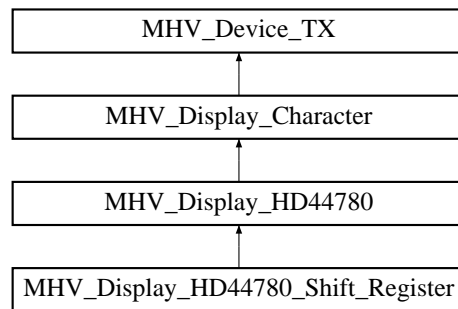
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.h](#)
- [A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.cpp](#)

4.13 MHV_Display_HD44780_Shift_Register Class Reference

```
#include <MHV_Display_HD44780_Shift_Register.h>
```

Inheritance diagram for MHV_Display_HD44780_Shift_Register:



Public Member Functions

- [MHV_Display_HD44780_Shift_Register](#) (volatile uint8_t *dataDir, volatile uint8_t *dataOut, volatile uint8_t *dataIn, uint8_t dataPin, int8_t dataPinchangeInterrupt, volatile uint8_t *enableDir, volatile uint8_t *enableOut, volatile uint8_t *enableIn, uint8_t enablePin, int8_t enablePinchangeInterrupt, volatile uint8_t *clockDir, volatile uint8_t *clockOut, volatile uint8_t *clockIn, uint8_t clockPin, int8_t clockPinchangeInterrupt, uint8_t colCount, uint16_t rowCount, [MHV_RingBuffer](#) *txBuffers)
- void [init](#) (bool multiLine, bool bigFont, bool cursorOn, bool cursorBlink, bool left2right, bool scroll)

Protected Member Functions

- void [pushBits](#) (uint8_t byte, bool rs)
- void [writeByte](#) (uint8_t byte, bool rs)
- uint8_t [readByte](#) (bool rs)
- bool [isBusy](#) ()
- void [delay](#) ([MHV_HD44780_COMMAND](#) command)

Protected Attributes

- volatile uint8_t * [_dataOut](#)
- uint8_t [_dataPin](#)
- volatile uint8_t * [_enableOut](#)
- uint8_t [_enablePin](#)
- volatile uint8_t * [_clockOut](#)
- uint8_t [_clockPin](#)

4.13.1 Detailed Description

Definition at line 32 of file MHV_Display_HD44780_Shift_Register.h.

4.13.2 Constructor & Destructor Documentation

4.13.2.1 **MHV_Display_HD44780_Shift_Register::MHV_Display_HD44780_Shift_Register** (volatile uint8_t * *dataDir*, volatile uint8_t * *dataOut*, volatile uint8_t * *dataIn*, uint8_t *dataPin*, int8_t *dataPinchangeInterrupt*, volatile uint8_t * *enableDir*, volatile uint8_t * *enableOut*, volatile uint8_t * *enableIn*, uint8_t *enablePin*, int8_t *enablePinchangeInterrupt*, volatile uint8_t * *clockDir*, volatile uint8_t * *clockOut*, volatile uint8_t * *clockIn*, uint8_t *clockPin*, int8_t *clockPinchangeInterrupt*, uint8_t *colCount*, uint16_t *rowCount*, MHV_RingBuffer * *txBuffers*)

A class for operating HD44780 based LCD displays via a shift register such as a 74HC164

Parameters

<i>dataDir</i>	A member of the MHV_PIN_* macro pin declaration for the data line of the shift register
<i>dataOut</i>	A member of the MHV_PIN_* macro
<i>dataIn</i>	A member of the MHV_PIN_* macro
<i>dataPin</i>	A member of the MHV_PIN_* macro
<i>data-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>enableDir</i>	A member of the MHV_PIN_* macro pin declaration for the enable line of the shift register
<i>enableOut</i>	A member of the MHV_PIN_* macro
<i>enableIn</i>	A member of the MHV_PIN_* macro
<i>enablePin</i>	A member of the MHV_PIN_* macro
<i>enable-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>clockDir</i>	A member of the MHV_PIN_* macro pin declaration for the clock line of the shift register
<i>clockOut</i>	A member of the MHV_PIN_* macro
<i>clockIn</i>	A member of the MHV_PIN_* macro
<i>clockPin</i>	A member of the MHV_PIN_* macro
<i>clock-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>colCount</i>	the number of columns on the display
<i>rowCount</i>	the number of rows on the display
<i>txBuffers</i>	buffers for async writing

Definition at line 64 of file MHV_Display_HD44780_Shift_Register.cpp.

4.13.3 Member Function Documentation

4.13.3.1 `void MHV_Display_HD44780_Shift_Register::delay (MHV_HD44780_COMMAND
command) [protected, virtual]`

Post-command delays

Implements [MHV_Display_HD44780](#).

Definition at line 144 of file `MHV_Display_HD44780_Shift_Register.cpp`.

4.13.3.2 `void MHV_Display_HD44780_Shift_Register::init (bool multiLine, bool bigFont, bool
cursorOn, bool cursorBlink, bool left2right, bool scroll)`

Initialise the display

Parameters

<i>multiLine</i>	true if there is more than 1 line
<i>bigFont</i>	true to use 5x11 fonts, false for 5x8
<i>cursorOn</i>	turn the cursor on
<i>cursorBlink</i>	blink the cursor
<i>left2right</i>	true for text reading left to right
<i>scroll</i>	true to scroll text rather than moving the cursor

Definition at line 171 of file `MHV_Display_HD44780_Shift_Register.cpp`.

4.13.3.3 `bool MHV_Display_HD44780_Shift_Register::isBusy () [protected,
virtual]`

Check if the display is busy

Returns

true if the display is busy

Implements [MHV_Display_HD44780](#).

Definition at line 137 of file `MHV_Display_HD44780_Shift_Register.cpp`.

4.13.3.4 `void MHV_Display_HD44780_Shift_Register::pushBits (uint8_t byte, bool rs)
[protected]`

Write 8 data bits to the display

Parameters

<i>byte</i>	the data to write (nibble or true byte)
<i>rs</i>	true to set the RS pin

Definition at line 92 of file MHV_Display_HD44780_Shift_Register.cpp.

4.13.3.5 `uint8_t MHV_Display_HD44780_Shift_Register::readByte (bool rs)` `[protected, virtual]`

Read a byte from the display

Parameters

<i>rs</i>	true to set the RS pin
-----------	------------------------

Implements [MHV_Display_HD44780](#).

Definition at line 129 of file MHV_Display_HD44780_Shift_Register.cpp.

4.13.3.6 `void MHV_Display_HD44780_Shift_Register::writeByte (uint8_t byte, bool rs)` `[protected, virtual]`

Write a byte to the display

Parameters

<i>byte</i>	the data to write
<i>rs</i>	true to set the RS pin (aka data pin)

Implements [MHV_Display_HD44780](#).

Definition at line 112 of file MHV_Display_HD44780_Shift_Register.cpp.

4.13.4 Member Data Documentation

4.13.4.1 `volatile uint8_t* MHV_Display_HD44780_Shift_Register::_clockOut` `[protected]`

Definition at line 40 of file MHV_Display_HD44780_Shift_Register.h.

4.13.4.2 `uint8_t MHV_Display_HD44780_Shift_Register::_clockPin` `[protected]`

Definition at line 41 of file MHV_Display_HD44780_Shift_Register.h.

4.13.4.3 `volatile uint8_t* MHV_Display_HD44780_Shift_Register::_dataOut` `[protected]`

Definition at line 34 of file MHV_Display_HD44780_Shift_Register.h.

4.13.4.4 `uint8_t MHV_Display_HD44780_Shift_Register::_dataPin` `[protected]`

Definition at line 35 of file `MHV_Display_HD44780_Shift_Register.h`.

4.13.4.5 `volatile uint8_t* MHV_Display_HD44780_Shift_Register::_enableOut`
`[protected]`

Definition at line 37 of file `MHV_Display_HD44780_Shift_Register.h`.

4.13.4.6 `uint8_t MHV_Display_HD44780_Shift_Register::_enablePin`
`[protected]`

Definition at line 38 of file `MHV_Display_HD44780_Shift_Register.h`.

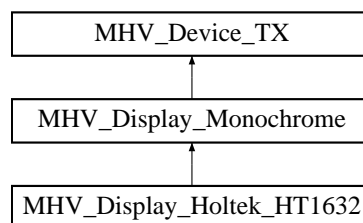
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Display_HD44780_Shift_Register.h](#)
- [A:/eclipse/mhvlb/MHV_Display_HD44780_Shift_Register.cpp](#)

4.14 MHV_Display_Holtek_HT1632 Class Reference

```
#include <MHV_Display_Holtek_HT1632.h>
```

Inheritance diagram for `MHV_Display_Holtek_HT1632`:



Public Member Functions

- [MHV_Display_Holtek_HT1632](#) (`volatile uint8_t *dataDir`, `volatile uint8_t *dataOut`, `volatile uint8_t *dataIn`, `uint8_t dataPin`, `int8_t dataPinchangeInterrupt`, `volatile uint8_t *writeDir`, `volatile uint8_t *writeOut`, `volatile uint8_t *writeIn`, `uint8_t writePin`, `int8_t writePinchangeInterrupt`, [MHV_HT1632_MODE](#) `mode`, `uint8_t arrayX`, `uint8_t arrayY`, `void(*csCallback)(uint8_t x, uint8_t y, bool active)`, `uint8_t *frameBuffer`, [MHV_RingBuffer](#) `*txBuffers`)
- `void brightness` (`uint8_t brightness`)
- `void poweroff` ()
- `void poweron` ()
- `void flush` ()

- void [setPixel](#) (uint16_t row, uint16_t col, uint8_t value)
- uint8_t [getPixel](#) (uint16_t row, uint16_t col)

4.14.1 Detailed Description

Definition at line 50 of file MHV_Display_Holtek_HT1632.h.

4.14.2 Constructor & Destructor Documentation

4.14.2.1 MHV_Display_Holtek_HT1632::MHV_Display_Holtek_HT1632 (volatile uint8_t * *dataDir*, volatile uint8_t * *dataOut*, volatile uint8_t * *dataIn*, uint8_t *dataPin*, int8_t *dataPinchangeInterrupt*, volatile uint8_t * *writeDir*, volatile uint8_t * *writeOut*, volatile uint8_t * *writeln*, uint8_t *writePin*, int8_t *writePinchangeInterrupt*, MHV_HT1632_MODE *mode*, uint8_t *arrayX*, uint8_t *arrayY*, void(*) (uint8_t x, uint8_t y, bool active) *csCallback*, uint8_t * *frameBuffer*, MHV_RingBuffer * *txBuffers*)

Initialise the library

Parameters

<i>dataDir</i>	A member of the MHV_PIN_* macro the data pin
<i>dataOut</i>	A member of the MHV_PIN_* macro
<i>dataIn</i>	A member of the MHV_PIN_* macro
<i>dataPin</i>	A member of the MHV_PIN_* macro
<i>data-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>writeDir</i>	A member of the MHV_PIN_* macro the write pin
<i>writeOut</i>	A member of the MHV_PIN_* macro
<i>writeln</i>	A member of the MHV_PIN_* macro
<i>writePin</i>	A member of the MHV_PIN_* macro
<i>write-Pinchange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>mode</i>	What mode the displays should be run in
<i>arrayX</i>	the width of the array in number of displays
<i>arrayY</i>	the height of the array in number of displays
<i>csCallback</i>	A callback to select which display is active (lines must be active low, x & y select the display)
<i>frameBuffer</i>	memory for a framebuffer, must be at least arrayX * arrayY * displayX * displayY / 8 bytes long
<i>txBuffers</i>	A ringbuffer used for text printing

Definition at line 59 of file MHV_Display_Holtek_HT1632.cpp.

4.14.3 Member Function Documentation

4.14.3.1 void MHV_Display_Holtek_HT1632::brightness (uint8_t *brightness_in*)

Set the brightness of all modules

Parameters

<i>brightness_in</i>	the brightness (from 0 to 15)
----------------------	-------------------------------

Definition at line 336 of file MHV_Display_Holtek_HT1632.cpp.

4.14.3.2 void MHV_Display_Holtek_HT1632::flush ()

Flush the framebuffer to the displays

Definition at line 273 of file MHV_Display_Holtek_HT1632.cpp.

4.14.3.3 uint8_t MHV_Display_Holtek_HT1632::getPixel (uint16_t *col*, uint16_t *row*) [virtual]

Get a pixel

Parameters

<i>col</i>	the column of the pixel
<i>row</i>	the row of the pixel

Returns

the value of the pixel

Implements [MHV_Display_Monochrome](#).

Definition at line 181 of file MHV_Display_Holtek_HT1632.cpp.

4.14.3.4 void MHV_Display_Holtek_HT1632::poweroff ()

Put all modules to sleep

Definition at line 396 of file MHV_Display_Holtek_HT1632.cpp.

4.14.3.5 void MHV_Display_Holtek_HT1632::poweron ()

Wake all modules up

Definition at line 409 of file MHV_Display_Holtek_HT1632.cpp.

4.14.3.6 void MHV_Display_Holtek_HT1632::setPixel (uint16_t col, uint16_t row, uint8_t value)
[virtual]

Set a pixel

Parameters

<i>col</i>	the column of the pixel
<i>row</i>	the row of the pixel
<i>value</i>	the value of the pixel

Implements [MHV_Display_Monochrome](#).

Definition at line 135 of file MHV_Display_Holtek_HT1632.cpp.

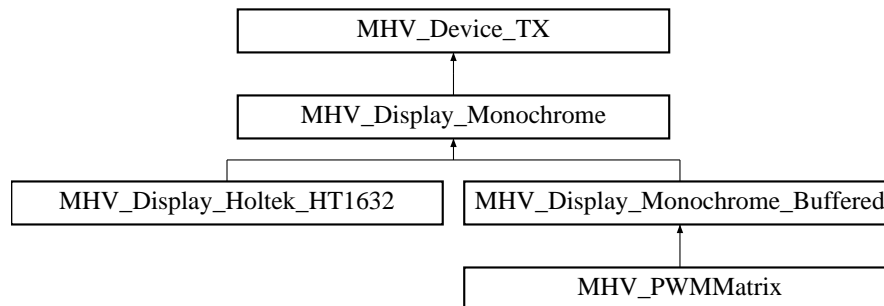
The documentation for this class was generated from the following files:

- A:/eclipse/mhvlb/MHV_Display_Holtek_HT1632.h
- A:/eclipse/mhvlb/MHV_Display_Holtek_HT1632.cpp

4.15 MHV_Display_Monochrome Class Reference

```
#include <MHV_Display_Monochrome.h>
```

Inheritance diagram for MHV_Display_Monochrome:



Public Member Functions

- [MHV_Display_Monochrome](#) (uint16_t colCount, uint16_t rowCount, [MHV_Ring-Buffer](#) *txBuffers)
- uint16_t [getWidth](#) ()
- uint16_t [getHeight](#) ()
- void [clear](#) (uint8_t value)
- bool [writeString](#) (const [MHV_FONT](#) *font, int16_t *offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, const char *string)
- bool [writeString_P](#) (const [MHV_FONT](#) *font, int16_t *offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, PGM_P string)

- bool [writeBuffer](#) (const [MHV_FONT](#) *font, int16_t *offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, const char *buffer, uint16_t length)
- bool [writeBuffer_P](#) (const [MHV_FONT](#) *font, int16_t *offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, PGM_P buffer, uint16_t length)
- bool [txAnimation](#) (const [MHV_FONT](#) *font, int16_t offsetY, uint8_t onValue, uint8_t offValue)
- virtual void [setPixel](#) (uint16_t row, uint16_t col, uint8_t value)=0
- virtual uint8_t [getPixel](#) (uint16_t row, uint16_t col)=0

Protected Member Functions

- bool [writeChar](#) (const [MHV_FONT](#) *font, int16_t *offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, char character)
- bool [writeSeperator](#) (const [MHV_FONT](#) *font, int16_t *offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue)
- void [runTxBuffers](#) ()

Protected Attributes

- uint16_t [_colCount](#)
- uint16_t [_rowCount](#)
- int16_t [_txOffset](#)

4.15.1 Detailed Description

Definition at line 36 of file MHV_Display_Monochrome.h.

4.15.2 Constructor & Destructor Documentation

4.15.2.1 MHV_Display_Monochrome::MHV_Display_Monochrome (uint16_t *colCount*, uint16_t *rowCount*, [MHV_RingBuffer](#) * *txBuffers*)

A monochrome bitmap display Origin (0,0) is bottom left Create a new monochrome display

Parameters

<i>colCount</i>	the number of columns
<i>rowCount</i>	the number of rows
<i>txBuffers</i>	buffers to use for text writing

Definition at line 42 of file MHV_Display_Monochrome.cpp.

4.15.3 Member Function Documentation

4.15.3.1 void MHV_Display_Monochrome::clear (uint8_t *value*)

Clear the display to a particular value

Parameters

<i>value</i>	the value to fill the display with
--------------	------------------------------------

Definition at line 67 of file MHV_Display_Monochrome.cpp.

4.15.3.2 uint16_t MHV_Display_Monochrome::getHeight ()

Get the width of the display

Definition at line 59 of file MHV_Display_Monochrome.cpp.

4.15.3.3 virtual uint8_t MHV_Display_Monochrome::getPixel (uint16_t *row*, uint16_t *col*) [pure virtual]

Implemented in [MHV_Display_Holtek_HT1632](#), and [MHV_Display_Monochrome_Buffered](#).

4.15.3.4 uint16_t MHV_Display_Monochrome::getWidth ()

Get the width of the display

Definition at line 52 of file MHV_Display_Monochrome.cpp.

4.15.3.5 void MHV_Display_Monochrome::runTxBuffers () [protected, virtual]

Start rendering TX buffers

Implements [MHV_Device_TX](#).

Definition at line 271 of file MHV_Display_Monochrome.cpp.

4.15.3.6 virtual void MHV_Display_Monochrome::setPixel (uint16_t *row*, uint16_t *col*, uint8_t *value*) [pure virtual]

Implemented in [MHV_Display_Holtek_HT1632](#), and [MHV_Display_Monochrome_Buffered](#).

4.15.3.7 bool MHV_Display_Monochrome::txAnimation (const MHV_FONT * *font*, int16_t *offsetY*, uint8_t *onValue*, uint8_t *offValue*)

Render a frame of TX buffer animation - scrolls text from right to left, before moving to the next buffer

Parameters

<i>font</i>	the font to use
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value for on pixels
<i>offValue</i>	the pixel value for off pixels

Returns

true if there are more frames to be rendered

Definition at line 284 of file MHV_Display_Monochrome.cpp.

4.15.3.8 `bool MHV_Display_Monochrome::writeBuffer (const MHV_FONT * font, int16_t * offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, const char * buffer, uint16_t length)`

Write a buffer to the display

Parameters

<i>font</i>	the font to use
<i>offsetX</i>	the horizontal pixel offset to start writing at (left side of char) will increment to the next position on return)
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value to use for on
<i>offValue</i>	the pixel value to use for off
<i>buffer</i>	the buffer to write
<i>length</i>	the length of the buffer

Returns

true if anything was written

Definition at line 199 of file MHV_Display_Monochrome.cpp.

4.15.3.9 `bool MHV_Display_Monochrome::writeBuffer_P (const MHV_FONT * font, int16_t * offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, PGM_P buffer, uint16_t length)`

Write a PROGMEM buffer to the display

Parameters

<i>font</i>	the font to use
<i>offsetX</i>	the horizontal pixel offset to start writing at (left side of char) will increment to the next position on return)
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value to use for on

<i>offValue</i>	the pixel value to use for off
<i>buffer</i>	the buffer to write
<i>length</i>	the length of the buffer

Returns

true if anything was written

Definition at line 253 of file MHV_Display_Monochrome.cpp.

4.15.3.10 `bool MHV_Display_Monochrome::writeChar (const MHV_FONT * font, int16_t * offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, char character)`
`[protected]`

Write a character to the display

Parameters

<i>font</i>	the font to use
<i>offsetX</i>	the horizontal pixel offset to start writing at (left side of char) will increment to the next position on return)
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value to use for on
<i>offValue</i>	the pixel value to use for off
<i>character</i>	the character to write

Returns

true if a character was written

Definition at line 87 of file MHV_Display_Monochrome.cpp.

4.15.3.11 `bool MHV_Display_Monochrome::writeSeperator (const MHV_FONT * font, int16_t * offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue)`
`[protected]`

Write a character separator (a single column of off pixels) to the display

Parameters

<i>font</i>	the font to use
<i>offsetX</i>	the horizontal pixel offset to start writing at (left side of char) will increment to the next position on return)
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value to use for on
<i>offValue</i>	the pixel value to use for off

Returns

true if the seperator was written

Definition at line 141 of file MHV_Display_Monochrome.cpp.

4.15.3.12 `bool MHV_Display_Monochrome::writeString (const MHV_FONT * font, int16_t * offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, const char * string)`

Write a string to the display

Parameters

<i>font</i>	the font to use
<i>offsetX</i>	the horizontal pixel offset to start writing at (left side of char) will increment to the next position on return)
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value to use for on
<i>offValue</i>	the pixel value to use for off
<i>string</i>	the string to write

Returns

true if anything was written

Definition at line 173 of file MHV_Display_Monochrome.cpp.

4.15.3.13 `bool MHV_Display_Monochrome::writeString_P (const MHV_FONT * font, int16_t * offsetX, int16_t offsetY, uint8_t onValue, uint8_t offValue, PGM_P string)`

Write a PROGMEM string to the display

Parameters

<i>font</i>	the font to use
<i>offsetX</i>	the horizontal pixel offset to start writing at (left side of char) will increment to the next position on return)
<i>offsetY</i>	the vertical pixel offset to start writing at (bottom of char)
<i>onValue</i>	the pixel value to use for on
<i>offValue</i>	the pixel value to use for off
<i>string</i>	the string to write

Returns

true if anything was written

Definition at line 224 of file MHV_Display_Monochrome.cpp.

4.15.4 Member Data Documentation

4.15.4.1 `uint16_t MHV_Display_Monochrome::_colCount` [protected]

Definition at line 38 of file `MHV_Display_Monochrome.h`.

4.15.4.2 `uint16_t MHV_Display_Monochrome::_rowCount` [protected]

Definition at line 39 of file `MHV_Display_Monochrome.h`.

4.15.4.3 `int16_t MHV_Display_Monochrome::_txOffset` [protected]

Definition at line 40 of file `MHV_Display_Monochrome.h`.

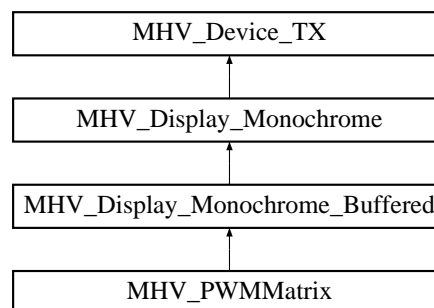
The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Display_Monochrome.h](#)
- [A:/eclipse/mhvlb/MHV_Display_Monochrome.cpp](#)

4.16 MHV_Display_Monochrome_Buffered Class Reference

```
#include <MHV_Display_Monochrome_Buffered.h>
```

Inheritance diagram for `MHV_Display_Monochrome_Buffered`:



Public Member Functions

- [MHV_Display_Monochrome_Buffered](#) (`uint16_t colCount`, `uint16_t rowCount`, `uint8_t *frameBuffer`, [MHV_RingBuffer](#) *txBuffers)
- void [setPixel](#) (`uint16_t col`, `uint16_t row`, `uint8_t value`)
- `uint8_t` [getPixel](#) (`uint16_t col`, `uint16_t row`)

Protected Attributes

- `uint8_t *` [_frameBuffer](#)

4.16.1 Detailed Description

Definition at line 34 of file MHV_Display_Monochrome_Buffered.h.

4.16.2 Constructor & Destructor Documentation

4.16.2.1 `MHV_Display_Monochrome_Buffered::MHV_Display_Monochrome_Buffered (uint16_t colCount, uint16_t rowCount, uint8_t * frameBuffer, MHV_RingBuffer * txBuffers)`

Create a new monochrome display param: colCount the number of columns param: rowCount the number of rows param: frameBuffer memory to use for the framebuffer, must be at least rows * cols * uint8_t

Definition at line 39 of file MHV_Display_Monochrome_Buffered.cpp.

4.16.3 Member Function Documentation

4.16.3.1 `uint8_t MHV_Display_Monochrome_Buffered::getPixel (uint16_t col, uint16_t row)`
[virtual]

Implements [MHV_Display_Monochrome](#).

Definition at line 63 of file MHV_Display_Monochrome_Buffered.cpp.

4.16.3.2 `void MHV_Display_Monochrome_Buffered::setPixel (uint16_t col, uint16_t row, uint8_t value)` [virtual]

Implements [MHV_Display_Monochrome](#).

Definition at line 52 of file MHV_Display_Monochrome_Buffered.cpp.

4.16.4 Member Data Documentation

4.16.4.1 `uint8_t* MHV_Display_Monochrome_Buffered::_frameBuffer`
[protected]

Definition at line 36 of file MHV_Display_Monochrome_Buffered.h.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Display_Monochrome_Buffered.h](#)
- [A:/eclipse/mhvlb/MHV_Display_Monochrome_Buffered.cpp](#)

4.17 MHV_EEPROM Class Reference

```
#include <MHV_EEPROM.h>
```

Public Member Functions

- [MHV_EEPROM](#) ()
- [int16_t read](#) ([uint16_t address](#))
- [uint8_t busyRead](#) ([uint16_t address](#))
- [int8_t read](#) ([void *buffer](#), [uint16_t address](#), [uint16_t length](#))
- [void busyRead](#) ([void *buffer](#), [uint16_t address](#), [uint16_t length](#))
- [int8_t write](#) ([uint16_t address](#), [uint8_t data](#))
- [int8_t busyWrite](#) ([void *buffer](#), [uint16_t address](#), [uint16_t length](#))
- [int8_t write](#) ([void *buffer](#), [uint16_t address](#), [uint16_t length](#), [void\(*done-Callback\)\(void *buffer, void *data\), void *doneCallbackData](#))
- [void writeInterrupt](#) ()
- [bool isBusy](#) ()

4.17.1 Detailed Description

Definition at line 46 of file `MHV_EEPROM.h`.

4.17.2 Constructor & Destructor Documentation

4.17.2.1 `MHV_EEPROM::MHV_EEPROM ()`

Create a new EEPROM access class

Definition at line 33 of file `MHV_EEPROM.cpp`.

4.17.3 Member Function Documentation

4.17.3.1 `uint8_t MHV_EEPROM::busyRead (uint16_t address)`

Read a byte from EEPROM, waiting until the EEPROM is available

Parameters

<i>address</i>	the address to read from
----------------	--------------------------

Returns

the byte

Definition at line 66 of file `MHV_EEPROM.cpp`.

4.17.3.2 `void MHV_EEPROM::busyRead (void * buffer, uint16_t address, uint16_t length)`

Read a buffer from EEPROM, waiting until the EEPROM is available

Parameters

<i>buffer</i>	the buffer to populate
<i>address</i>	the address to read from
<i>length</i>	the number of bytes to read

Definition at line 103 of file MHV_EEPROM.cpp.

4.17.3.3 int8_t MHV_EEPROM::busyWrite (void * *buffer*, uint16_t *address*, uint16_t *length*)

Write a buffer to EEPROM

Parameters

<i>buffer</i>	the buffer to read from
<i>address</i>	the address to write to
<i>length</i>	the number of bytes to write

Definition at line 134 of file MHV_EEPROM.cpp.

4.17.3.4 bool MHV_EEPROM::isBusy ()

Check if the EEPROM hardware is busy

Returns

true if the EEPROM hardware is busy

Definition at line 216 of file MHV_EEPROM.cpp.

4.17.3.5 int16_t MHV_EEPROM::read (uint16_t *address*)

Read a byte from EEPROM

Parameters

<i>address</i>	the address to read from
----------------	--------------------------

Returns

the byte, or MHV_EEPROM_BUSY if the EEPROM is busy

Definition at line 47 of file MHV_EEPROM.cpp.

4.17.3.6 int8_t MHV_EEPROM::read (void * *buffer*, uint16_t *address*, uint16_t *length*)

Read a buffer from EEPROM

Parameters

<i>buffer</i>	the buffer to populate
<i>address</i>	the address to read from
<i>length</i>	the number of bytes to read

Definition at line 79 of file MHV_EEPROM.cpp.

4.17.3.7 int8_t MHV_EEPROM::write (uint16_t address, uint8_t data)

Write a byte to EEPROM

Parameters

<i>address</i>	the address to write to
<i>data</i>	the byte to write

Returns

MHV_EEPROM_BUSY if the EEPROM is busy

Definition at line 113 of file MHV_EEPROM.cpp.

4.17.3.8 int8_t MHV_EEPROM::write (void * buffer, uint16_t address, uint16_t length, void(*) (void *buffer, void *data) doneCallback, void * doneCallbackData)

Write a buffer to EEPROM

Parameters

<i>buffer</i>	the buffer to read from
<i>address</i>	the address to write to
<i>length</i>	the number of bytes to write (must be greater than 0)
<i>done-Callback</i>	A callback to call when the buffer has been written (can be NULL)
<i>done-Callback-Data</i>	A pointer to pass to the callback

Definition at line 164 of file MHV_EEPROM.cpp.

4.17.3.9 void MHV_EEPROM::writeInterrupt ()

Interrupt handler for async writes

Definition at line 192 of file MHV_EEPROM.cpp.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_EEPROM.h](#)
- [A:/eclipse/mhvlb/MHV_EEPROM.cpp](#)

4.18 mhv_eventADC Struct Reference

```
#include <MHV_ADC.h>
```

Public Attributes

- `uint8_t` [channel](#)
- [MHV_ADCListener](#) * [listener](#)

4.18.1 Detailed Description

Definition at line 51 of file MHV_ADC.h.

4.18.2 Member Data Documentation

4.18.2.1 `uint8_t mhv_eventADC::channel`

Definition at line 52 of file MHV_ADC.h.

4.18.2.2 `MHV_ADCListener* mhv_eventADC::listener`

Definition at line 53 of file MHV_ADC.h.

The documentation for this struct was generated from the following file:

- [A:/eclipse/mhvlb/MHV_ADC.h](#)

4.19 mhv_eventPin Struct Reference

```
#include <MHV_PinChangeManager.h>
```

Public Attributes

- `volatile uint8_t` * [port](#)
- `uint8_t` [mask](#)
- `uint8_t` [pclnt](#)
- [MHV_PinEventListener](#) * [listener](#)
- `bool` [previous](#)
- `bool` [changed](#)

4.19.1 Detailed Description

Definition at line 79 of file MHV_PinChangeManager.h.

4.19.2 Member Data Documentation

4.19.2.1 `bool mhv_eventPin::changed`

Definition at line 85 of file MHV_PinChangeManager.h.

4.19.2.2 `MHV_PinEventListener* mhv_eventPin::listener`

Definition at line 83 of file MHV_PinChangeManager.h.

4.19.2.3 `uint8_t mhv_eventPin::mask`

Definition at line 81 of file MHV_PinChangeManager.h.

4.19.2.4 `uint8_t mhv_eventPin::pcInt`

Definition at line 82 of file MHV_PinChangeManager.h.

4.19.2.5 `volatile uint8_t* mhv_eventPin::port`

Definition at line 80 of file MHV_PinChangeManager.h.

4.19.2.6 `bool mhv_eventPin::previous`

Definition at line 84 of file MHV_PinChangeManager.h.

The documentation for this struct was generated from the following file:

- [A:/eclipse/mhvlb/MHV_PinChangeManager.h](#)

4.20 mhv_font Struct Reference

```
#include <MHV_Font.h>
```

Public Attributes

- `uint8_t` [maxWidth](#)
- `uint8_t` [maxHeight](#)

- char [firstChar](#)
- uint8_t [charCount](#)
- char [unknown](#)
- uint8_t [columnBytes](#)
- const uint8_t * [widths](#)
- const uint16_t * [offsets](#)
- const uint8_t * [fontData](#)

4.20.1 Detailed Description

Definition at line 15 of file MHV_Font.h.

4.20.2 Member Data Documentation

4.20.2.1 uint8_t mhv_font::charCount

Definition at line 19 of file MHV_Font.h.

4.20.2.2 uint8_t mhv_font::columnBytes

Definition at line 21 of file MHV_Font.h.

4.20.2.3 char mhv_font::firstChar

Definition at line 18 of file MHV_Font.h.

4.20.2.4 const uint8_t* mhv_font::fontData

Definition at line 24 of file MHV_Font.h.

4.20.2.5 uint8_t mhv_font::maxHeight

Definition at line 17 of file MHV_Font.h.

4.20.2.6 uint8_t mhv_font::maxWidth

Definition at line 16 of file MHV_Font.h.

4.20.2.7 const uint16_t* mhv_font::offsets

Definition at line 23 of file MHV_Font.h.

4.20.2.8 char mhv_font::unknown

Definition at line 20 of file MHV_Font.h.

4.20.2.9 const uint8_t* mhv_font::widths

Definition at line 22 of file MHV_Font.h.

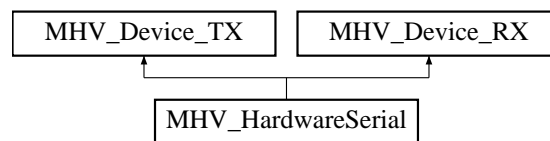
The documentation for this struct was generated from the following file:

- A:/eclipse/mhvlb/[MHV_Font.h](#)

4.21 MHV_HardwareSerial Class Reference

```
#include <MHV_HardwareSerial.h>
```

Inheritance diagram for MHV_HardwareSerial:



Public Member Functions

- [MHV_HardwareSerial](#) ([MHV_RingBuffer](#) *rxBuffer, [MHV_RingBuffer](#) *txBuffer, volatile uint16_t *ubrr, volatile uint8_t *ucsra, volatile uint8_t *ucsrb, volatile uint8_t *udr, uint8_t rxen, uint8_t txen, uint8_t rxcie, uint8_t txcie, uint8_t udre, uint8_t u2x, unsigned long baud)
- void [setSpeed](#) (unsigned long baud)
- void [end](#) ()
- void [busyWrite](#) (char c)
- void [busyWrite](#) (const char *buffer)
- void [busyWrite](#) (const char *buffer, uint16_t length)
- void [busyWrite_P](#) (PGM_P buffer)
- void [busyWrite_P](#) (PGM_P buffer, uint16_t length)
- bool [canSendBusy](#) ()
- void [rx](#) ()
- void [tx](#) ()
- void [echo](#) (bool echoOn)
- bool [busy](#) ()
- void [debug](#) (const char *file, int line, const char *function, PGM_P format,...)

Protected Member Functions

- void [runTxBuffers](#) ()

4.21.1 Detailed Description

Definition at line 71 of file MHV_HardwareSerial.h.

4.21.2 Constructor & Destructor Documentation

4.21.2.1 MHV_HardwareSerial::MHV_HardwareSerial (MHV_RingBuffer * *rxBuffer*, MHV_RingBuffer * *txBuffer*, volatile uint16_t * *ubrr*, volatile uint8_t * *ucsrA*, volatile uint8_t * *ucsrB*, volatile uint8_t * *udr*, uint8_t *rxen*, uint8_t *txen*, uint8_t *rxcie*, uint8_t *txcie*, uint8_t *udre*, uint8_t *u2x*, unsigned long *baud*)

Constructor

Parameters

<i>rxBuffer</i>	the receive buffer
<i>txBuffer</i>	the transmit buffer
<i>ubrr</i>	A member of the MHV_USART_* macro
<i>ucsrA</i>	See ubrr
<i>ucsrB</i>	See ubrr
<i>udr</i>	See ubrr
<i>rxen</i>	See ubrr
<i>txen</i>	See ubrr
<i>rxcie</i>	See ubrr
<i>txcie</i>	See ubrr
<i>udre</i>	See ubrr
<i>u2x</i>	See ubrr
<i>baud</i>	the baud rate

Definition at line 54 of file MHV_HardwareSerial.cpp.

4.21.3 Member Function Documentation

4.21.3.1 bool MHV_HardwareSerial::busy (void)

Check if the hardware is busy - note that this should not be used to determine if you can actually write - use canSend instead

Returns

true if the hardware is busy

Definition at line 266 of file MHV_HardwareSerial.cpp.

4.21.3.2 void MHV_HardwareSerial::busyWrite (char *c*)

Write a character

Parameters

<i>c</i>	the character to write
----------	------------------------

Definition at line 173 of file MHV_HardwareSerial.cpp.

4.21.3.3 void MHV_HardwareSerial::busyWrite (const char * *buffer*)

Write a null terminated string to the serial port

Parameters

<i>buffer</i>	the string to write
---------------	---------------------

Definition at line 209 of file MHV_HardwareSerial.cpp.

4.21.3.4 void MHV_HardwareSerial::busyWrite (const char * *buffer*, uint16_t *length*)

Write a buffer

Parameters

<i>buffer</i>	the buffer to write
<i>length</i>	the length of the buffer

Definition at line 246 of file MHV_HardwareSerial.cpp.

4.21.3.5 void MHV_HardwareSerial::busyWrite_P (PGM_P *buffer*)

Write a null terminated progmem string

Parameters

<i>buffer</i>	the string to write
---------------	---------------------

Definition at line 187 of file MHV_HardwareSerial.cpp.

4.21.3.6 void MHV_HardwareSerial::busyWrite_P (PGM_P *buffer*, uint16_t *length*)

Write a buffer from PROGMEM

Parameters

<i>buffer</i>	the buffer to write
<i>length</i>	the length of the buffer

Definition at line 227 of file MHV_HardwareSerial.cpp.

4.21.3.7 bool MHV_HardwareSerial::canSendBusy ()

Can we send something via busywrite

Returns

true if we can send something

Definition at line 164 of file MHV_HardwareSerial.cpp.

4.21.3.8 void MHV_HardwareSerial::debug (const char * *file*, int *line*, const char * *function*, PGM_P *format*, ...)

Print a debug message

Parameters

<i>file</i>	the filename
<i>line</i>	the line number
<i>function</i>	the function name
<i>format</i>	a printf format
...	the printf parms

Definition at line 278 of file MHV_HardwareSerial.cpp.

4.21.3.9 void MHV_HardwareSerial::echo (bool *echoOn*)

Enable echoing data received by us back to the sender (useful for terminal interaction)

Parameters

<i>echoOn</i>	true to enable echo
---------------	---------------------

Definition at line 155 of file MHV_HardwareSerial.cpp.

4.21.3.10 void MHV_HardwareSerial::end ()

Halt the serial port

Definition at line 145 of file MHV_HardwareSerial.cpp.

4.21.3.11 `void MHV_HardwareSerial::runTxBuffers ()` [protected, virtual]

Start sending async data

Implements [MHV_Device_TX](#).

Definition at line 106 of file `MHV_HardwareSerial.cpp`.

4.21.3.12 `void MHV_HardwareSerial::rx ()`

RX interrupt handler

Definition at line 78 of file `MHV_HardwareSerial.cpp`.

4.21.3.13 `void MHV_HardwareSerial::setSpeed (unsigned long baud)`

Configure the serial port for a specific baud rate

Parameters

<i>baud</i>	the baud rate to set
-------------	----------------------

Definition at line 126 of file `MHV_HardwareSerial.cpp`.

4.21.3.14 `void MHV_HardwareSerial::tx ()`

TX interrupt handler

Definition at line 90 of file `MHV_HardwareSerial.cpp`.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_HardwareSerial.h](#)
- [A:/eclipse/mhvlb/MHV_HardwareSerial.cpp](#)

4.22 MHV_Lock Class Reference

```
#include <MHV_Lock.h>
```

Public Member Functions

- [MHV_Lock \(\)](#)
- bool [obtain \(\)](#)
- void [release \(\)](#)
- bool [check \(\)](#)

4.22.1 Detailed Description

Definition at line 10 of file MHV_Lock.h.

4.22.2 Constructor & Destructor Documentation

4.22.2.1 MHV_Lock::MHV_Lock ()

Create a new lock

Definition at line 13 of file MHV_Lock.cpp.

4.22.3 Member Function Documentation

4.22.3.1 bool MHV_Lock::check ()

Check if the lock is currently held

Returns

true if the lock is held

Definition at line 44 of file MHV_Lock.cpp.

4.22.3.2 bool MHV_Lock::obtain ()

Obtain the lock

Returns

true if the lock was successfully obtained, false otherwise

Definition at line 21 of file MHV_Lock.cpp.

4.22.3.3 void MHV_Lock::release ()

Release the lock

Definition at line 34 of file MHV_Lock.cpp.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Lock.h](#)
- [A:/eclipse/mhvlb/MHV_Lock.cpp](#)

4.23 MHV_PID Class Reference

```
#include <MHV_PID.h>
```

Public Member Functions

- [MHV_PID](#) (float setpoint, float kP, float kI, float kD, uint16_t period, bool reverse, uint16_t min, uint16_t max)
- void [setDirection](#) (bool reverse)
- void [enable](#) (bool enable)
- float [compute](#) (float input)
- void [setTuning](#) (float kP, float kI, float kD, uint16_t period)
- void [setOutputLimits](#) (float, float)

Protected Member Functions

- void [clampIntegral](#) ()

Protected Attributes

- float [_kP](#)
- float [_kI](#)
- float [_kD](#)
- bool [_reverse](#)
- bool [_enabled](#)
- float [_setpoint](#)
- float [_integral](#)
- float [_lastInput](#)
- float [_lastOutput](#)
- float [outMin](#)
- float [outMax](#)

4.23.1 Detailed Description

Definition at line 38 of file MHV_PID.h.

4.23.2 Constructor & Destructor Documentation

- 4.23.2.1 [MHV_PID::MHV_PID](#) (float *setpoint*, float *kP*, float *kI*, float *kD*, uint16_t *period*, bool *reverse*, uint16_t *min*, uint16_t *max*)

Create a new PID

Parameters

<i>setpoint</i>	the target value
<i>kP</i>	the proportional constant
<i>kI</i>	the integral constant
<i>kD</i>	the derivative constant
<i>period</i>	the period that compute() is called, in ms
<i>reverse</i>	true if there is an inverse relationship between the output and the input (eg. running a cooler to reduce temperature)
<i>min</i>	the minimum value for output
<i>max</i>	the maximum value four output

Definition at line 46 of file MHV_PID.cpp.

4.23.3 Member Function Documentation

4.23.3.1 void MHV_PID::clampIntegral () [inline, protected]

Definition at line 57 of file MHV_PID.cpp.

4.23.3.2 float MHV_PID::compute (float *input*)

Calculate the next output

Parameters

<i>input</i>	the latest sample
--------------	-------------------

Definition at line 69 of file MHV_PID.cpp.

4.23.3.3 void MHV_PID::enable (bool *enable*)

Enable/Disable the PID

Parameters

<i>enable</i>	true to enable the pid
---------------	------------------------

Definition at line 129 of file MHV_PID.cpp.

4.23.3.4 void MHV_PID::setDirection (bool *reverse*)

Set the direction for the PID.

Setting reverse means that increasing output reduces the input (eg. increasing power to a cooler, with the input being temperature)

Definition at line 145 of file MHV_PID.cpp.

4.23.3.5 void MHV_PID::setOutputLimits (float *min*, float *max*)

Alter the minimum & maximum values for the output

Parameters

<i>min</i>	the new minimum output value
<i>max</i>	the new maximum output value

Definition at line 117 of file MHV_PID.cpp.

4.23.3.6 void MHV_PID::setTuning (float *kP*, float *kI*, float *kD*, uint16_t *period*)

Adjust PID parameters

Parameters

<i>kP</i>	the proportional constant
<i>kI</i>	the integral constant
<i>kD</i>	the derivative constant
<i>period</i>	the period that compute() is called, in ms

Definition at line 97 of file MHV_PID.cpp.

4.23.4 Member Data Documentation

4.23.4.1 bool MHV_PID::_enabled [protected]

Definition at line 45 of file MHV_PID.h.

4.23.4.2 float MHV_PID::_integral [protected]

Definition at line 48 of file MHV_PID.h.

4.23.4.3 float MHV_PID::_kD [protected]

Definition at line 42 of file MHV_PID.h.

4.23.4.4 float MHV_PID::_kI [protected]

Definition at line 41 of file MHV_PID.h.

4.23.4.5 float MHV_PID::_kP [protected]

Definition at line 40 of file MHV_PID.h.

4.23.4.6 float MHV_PID::_lastInput [protected]

Definition at line 49 of file MHV_PID.h.

4.23.4.7 float MHV_PID::_lastOutput [protected]

Definition at line 50 of file MHV_PID.h.

4.23.4.8 bool MHV_PID::_reverse [protected]

Definition at line 44 of file MHV_PID.h.

4.23.4.9 float MHV_PID::_setpoint [protected]

Definition at line 47 of file MHV_PID.h.

4.23.4.10 float MHV_PID::outMax [protected]

Definition at line 53 of file MHV_PID.h.

4.23.4.11 float MHV_PID::outMin [protected]

Definition at line 52 of file MHV_PID.h.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_PID.h](#)
- [A:/eclipse/mhvlb/MHV_PID.cpp](#)

4.24 mhv_pin Struct Reference

```
#include <MHV_io.h>
```

Public Attributes

- volatile uint8_t * [dir](#)
- volatile uint8_t * [input](#)
- volatile uint8_t * [output](#)
- uint8_t [bit](#)
- int8_t [pclnt](#)

4.24.1 Detailed Description

Definition at line 76 of file MHV_io.h.

4.24.2 Member Data Documentation

4.24.2.1 uint8_t mhv_pin::bit

Definition at line 81 of file MHV_io.h.

4.24.2.2 volatile uint8_t* mhv_pin::dir

Definition at line 77 of file MHV_io.h.

4.24.2.3 volatile uint8_t* mhv_pin::input

Definition at line 78 of file MHV_io.h.

4.24.2.4 volatile uint8_t* mhv_pin::output

Definition at line 79 of file MHV_io.h.

4.24.2.5 int8_t mhv_pin::pcInt

Definition at line 82 of file MHV_io.h.

The documentation for this struct was generated from the following file:

- [A:/eclipse/mhvlb/MHV_io.h](#)

4.25 MHV_PinChangeManager Class Reference

```
#include <MHV_PinChangeManager.h>
```

Public Member Functions

- [MHV_PinChangeManager](#) ()
- void [pinChange0](#) ()
- void [pinChange](#) (uint8_t offset)
- void [registerListener](#) (volatile uint8_t *pinDir, volatile uint8_t *pinOut, volatile uint8_t *pinIn, uint8_t pinBit, int8_t pinChangeInterrupt, [MHV_PinEventListener](#) *listener)

- void [deregisterListener](#) (volatile uint8_t *pinDir, volatile uint8_t *pinOut, volatile uint8_t *pinIn, uint8_t pinBit, int8_t pinChangeInterrupt)
- void [handleEvents](#) ()

Protected Attributes

- [MHV_EVENT_PIN_pins](#) [MHV_PC_INT_COUNT]

4.25.1 Detailed Description

Definition at line 89 of file MHV_PinChangeManager.h.

4.25.2 Constructor & Destructor Documentation

4.25.2.1 MHV_PinChangeManager::MHV_PinChangeManager ()

An event manager for handling pinchange events

Definition at line 33 of file MHV_PinChangeManager.cpp.

4.25.3 Member Function Documentation

4.25.3.1 void MHV_PinChangeManager::deregisterListener (volatile uint8_t * *pinDir*, volatile uint8_t * *pinOut*, volatile uint8_t * *pinIn*, uint8_t *pinBit*, int8_t *pinChangeInterrupt*)

Deregister interest for pinchange events

Parameters

<i>pinDir</i>	A member of the MHV_PIN_* macro, must have a valid pinchange-Interrupt
<i>pinOut</i>	A member of the MHV_PIN_* macro
<i>pinIn</i>	A member of the MHV_PIN_* macro
<i>pinBit</i>	A member of the MHV_PIN_* macro
<i>pinChange-Interrupt</i>	A member of the MHV_PIN_* macro

Definition at line 145 of file MHV_PinChangeManager.cpp.

4.25.3.2 void MHV_PinChangeManager::handleEvents ()

Call from the main loop to handle any events

Definition at line 179 of file MHV_PinChangeManager.cpp.

4.25.3.3 void MHV_PinChangeManager::pinChange (uint8_t offset)

Trigger for pin change interrupts - scans through 8 pins starting at the offset

Parameters

<i>offset</i>	the number of pins to skip before scanning
---------------	--------------------------------------------

Definition at line 68 of file MHV_PinChangeManager.cpp.

4.25.3.4 void MHV_PinChangeManager::pinChange0 ()

Trigger for interrupt PCINT0

Definition at line 42 of file MHV_PinChangeManager.cpp.

4.25.3.5 void MHV_PinChangeManager::registerListener (volatile uint8_t * pinDir, volatile uint8_t * pinOut, volatile uint8_t * pinIn, uint8_t pinBit, int8_t pinChangeInterrupt, MHV_PinEventListener * listener)

Register interest for pinchange events

Parameters

<i>pinDir</i>	A member of the MHV_PIN_* macro, must have a valid pinchange-Interrupt
<i>pinOut</i>	A member of the MHV_PIN_* macro
<i>pinIn</i>	A member of the MHV_PIN_* macro
<i>pinBit</i>	A member of the MHV_PIN_* macro
<i>pinChange-Interrupt</i>	A member of the MHV_PIN_* macro
<i>listener</i>	a MHV_PinEventListener to notify when the pin changes

Definition at line 96 of file MHV_PinChangeManager.cpp.

4.25.4 Member Data Documentation

4.25.4.1 MHV_EVENT_PIN MHV_PinChangeManager::_pins[MHV_PC.INT_COUNT] [protected]

Definition at line 92 of file MHV_PinChangeManager.h.

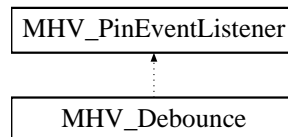
The documentation for this class was generated from the following files:

- A:/eclipse/mhvlb/[MHV_PinChangeManager.h](#)
- A:/eclipse/mhvlb/[MHV_PinChangeManager.cpp](#)

4.26 MHV_PinEventListener Class Reference

```
#include <MHV_PinChangeManager.h>
```

Inheritance diagram for MHV_PinEventListener:



Public Member Functions

- virtual void [pinChanged](#) (uint8_t *pcInt*, bool *newState*)=0

4.26.1 Detailed Description

Definition at line 74 of file MHV_PinChangeManager.h.

4.26.2 Member Function Documentation

4.26.2.1 virtual void MHV_PinEventListener::pinChanged (uint8_t *pcInt*, bool *newState*)
[pure virtual]

Implemented in [MHV_Debounce](#).

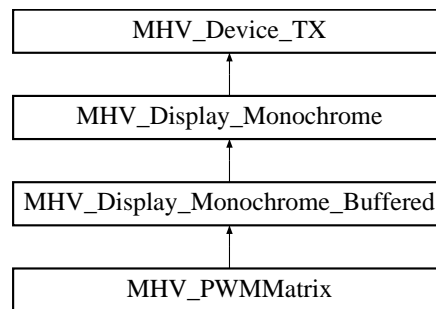
The documentation for this class was generated from the following file:

- A:/eclipse/mhplib/[MHV_PinChangeManager.h](#)

4.27 MHV_PWMMatrix Class Reference

```
#include <MHV_PWMMatrix.h>
```

Inheritance diagram for MHV_PWMMatrix:



Public Member Functions

- [MHV_PWMMatrix](#) (uint16_t rowCount, uint16_t colCount, uint8_t *frameBuffer, [MHV_RingBuffer](#) *txBuffers, void(*rowOn)(uint16_t row), void(*rowOff)(uint16_t row), void(*colOn)(uint16_t column), void(*colOff)(uint16_t column), [MHV_PWMATRIX_MODE](#) mode=MHV_PWMATRIX_MODE_AUTO)
- void [tick](#) ()

4.27.1 Detailed Description

Definition at line 41 of file MHV_PWMMatrix.h.

4.27.2 Constructor & Destructor Documentation

4.27.2.1 **MHV_PWMMatrix::MHV_PWMMatrix** (uint16_t *rowCount*, uint16_t *colCount*, uint8_t * *frameBuffer*, [MHV_RingBuffer](#) * *txBuffers*, void(*) (uint16_t row) *rowOn*, void(*) (uint16_t row) *rowOff*, void(*) (uint16_t column) *colOn*, void(*) (uint16_t column) *colOff*, [MHV_PWMATRIX_MODE](#) *mode* = MHV_PWMATRIX_MODE_AUTO)

Establish a new matrix

Parameters

<i>mode</i>	whether to scan rows, cols, individual pixels or auto
<i>rowCount</i>	the number of rows
<i>colCount</i>	the number of columns
<i>frameBuffer</i>	memory to use for the framebuffer, must be at least rows * cols * uint8_t
<i>txBuffers</i>	buffers used for text transmission
<i>rowOn</i>	callback to turn a row on
<i>rowOff</i>	callback to turn a row off
<i>colOn</i>	callback to turn a column on
<i>colOff</i>	callback to turn a column off

Definition at line 45 of file MHV_PWMMatrix.cpp.

4.27.3 Member Function Documentation

4.27.3.1 void MHV_PWMMatrix::tick (void)

Definition at line 176 of file MHV_PWMMatrix.cpp.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_PWMMatrix.h](#)
- [A:/eclipse/mhvlb/MHV_PWMMatrix.cpp](#)

4.28 MHV_RingBuffer Class Reference

```
#include <MHV_RingBuffer.h>
```

Public Member Functions

- [MHV_RingBuffer](#) (char *buffer, uint8_t length)
- bool [append](#) (char c)
- bool [append](#) (const void *p, uint8_t pLength)
- uint8_t [length](#) ()
- uint8_t [size](#) ()
- bool [full](#) ()
- bool [full](#) (uint8_t blockLength)
- void [flush](#) ()
- int [peekHead](#) ()
- int [consume](#) ()
- bool [consume](#) (void *p, uint8_t length)
- void [copyLine](#) (char *, uint8_t length)

4.28.1 Detailed Description

Definition at line 35 of file MHV_RingBuffer.h.

4.28.2 Constructor & Destructor Documentation

4.28.2.1 MHV_RingBuffer::MHV_RingBuffer (char * *buffer*, uint8_t *size*)

Create a new ringbuffer

Parameters

<i>buffer</i>	memory to use for the ringbuffer
<i>size</i>	the size of the available memory

Definition at line 36 of file MHV_RingBuffer.cpp.

4.28.3 Member Function Documentation

4.28.3.1 `bool MHV_RingBuffer::append (char c)`

Append a character to the buffer

Returns

false if we succeeded, true otherwise

Definition at line 61 of file MHV_RingBuffer.cpp.

4.28.3.2 `bool MHV_RingBuffer::append (const void * p, uint8_t pLength)`

Append a block of data to the buffer

Parameters

<i>p</i>	the pointer to append from
<i>pLength</i>	the number of bytes to append

Returns

false if we succeeded, true otherwise

Definition at line 81 of file MHV_RingBuffer.cpp.

4.28.3.3 `int MHV_RingBuffer::consume ()`

Pop a byte off the ringbuffer

Definition at line 99 of file MHV_RingBuffer.cpp.

4.28.3.4 `bool MHV_RingBuffer::consume (void * p, uint8_t pLength)`

Pop a block off the ringbuffer

Parameters

<i>p</i>	where to write the block
<i>pLength</i>	the length of the block

Returns

false if we succeeded, true otherwise

Definition at line 115 of file MHV_RingBuffer.cpp.

4.28.3.5 void MHV_RingBuffer::copyLine (char *, uint8_t *length*)

4.28.3.6 void MHV_RingBuffer::flush ()

Discard the contents of the ringbuffer

Definition at line 134 of file MHV_RingBuffer.cpp.

4.28.3.7 bool MHV_RingBuffer::full ()

Check if the ringbuffer is full

Returns

true if the ringbuffer is full

Definition at line 164 of file MHV_RingBuffer.cpp.

4.28.3.8 bool MHV_RingBuffer::full (uint8_t *blockLength*)

Check if an object can fit in the ringbuffer

Parameters

<i>blockLength</i>	the length of the object to fit in
--------------------	------------------------------------

Returns

true if the ringbuffer is full

Definition at line 173 of file MHV_RingBuffer.cpp.

4.28.3.9 uint8_t MHV_RingBuffer::length ()

Get the length of the contents of the ringbuffer Return the number of bytes in the ringbuffer

Definition at line 150 of file MHV_RingBuffer.cpp.

4.28.3.10 int MHV_RingBuffer::peekHead ()

Check the first character in the buffer

Returns

the character, or -1 if the buffer is empty

Definition at line 182 of file MHV_RingBuffer.cpp.

4.28.3.11 uint8_t MHV_RingBuffer::size ()

Get the size of the ringbuffer

Returns

the size of the ringbuffer

Definition at line 142 of file MHV_RingBuffer.cpp.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_RingBuffer.h](#)
- [A:/eclipse/mhvlb/MHV_RingBuffer.cpp](#)

4.29 MHV_RTC Class Reference

```
#include <MHV_RTC.h>
```

Public Member Functions

- [MHV_RTC](#) ([MHV_Timer8](#) *timer, [MHV_ALARM](#) *eventBuffer, uint8_t event-Count, int16_t timezone)
- void [synchronise](#) ()
- void [setTime](#) (uint32_t timestamp, uint16_t milliseconds)
- void [setTime](#) ([MHV_TIMESTAMP](#) *timestamp)
- void [tick](#) ()
- void [tick1ms](#) ()
- void [tickAndRunEvents](#) ()
- void [tick1msAndRunEvents](#) ()
- void [current](#) ([MHV_TIMESTAMP](#) *timestamp)
- void [elapsed](#) ([MHV_TIMESTAMP](#) *since, [MHV_TIMESTAMP](#) *elapsed)
- void [toTime](#) ([MHV_TIME](#) *to, [MHV_TIMESTAMP](#) *from)
- void [toTimestamp](#) ([MHV_TIMESTAMP](#) *to, [MHV_TIME](#) *from)
- bool [addAlarm](#) ([MHV_ALARM](#) *alarm)
- void [handleEvents](#) ()
- uint8_t [alarmsPending](#) ()
- void [removeAlarm](#) ([MHV_AlarmListener](#) *listener)

Protected Attributes

- [MHV_Timer8](#) * [_timer](#)
- [MHV_ALARM](#) * [_alarms](#)
- volatile uint8_t [_alarmCount](#)
- uint8_t [_alarmMax](#)
- uint32_t [_timestamp](#)
- uint16_t [_milliseconds](#)
- uint8_t [_ticks](#)
- uint8_t [_ticksPerMillisecond](#)
- int16_t [_tzOffset](#)

4.29.1 Detailed Description

Definition at line 102 of file MHV_RTC.h.

4.29.2 Constructor & Destructor Documentation

4.29.2.1 `MHV_RTC::MHV_RTC (MHV_Timer8 * timer, MHV_ALARM * eventBuffer, uint8_t eventCount, int16_t timezone)`

Create a new RTC

Parameters

<i>timer</i>	the timer this RTC is associated with
<i>eventBuffer</i>	A buffer to store events until they are executed
<i>eventCount</i>	The number of events that can be stored in the buffer
<i>timezone</i>	minutes offset from UTC

Definition at line 167 of file MHV_RTC.cpp.

4.29.3 Member Function Documentation

4.29.3.1 `bool MHV_RTC::addAlarm (MHV_ALARM * alarm)`

Insert an alarm, to be triggered at a later date

Parameters

<i>alarm</i>	the alarm, consisting of: when it should be triggered what should be called (it will be passed a pointer to the event) a pointer to user-defined data
--------------	-------------------------------------------------------------------------------------------------------------------------------------------------------

Returns

true if the event could not be added

Definition at line 443 of file MHV_RTC.cpp.

4.29.3.2 `uint8_t MHV_RTC::alarmsPending ()`

How many events are left in the queue

Returns

the number of events

Definition at line 508 of file MHV_RTC.cpp.

4.29.3.3 `void MHV_RTC::current (MHV_TIMESTAMP * timestamp)`

Get the current timestamp

Definition at line 276 of file MHV_RTC.cpp.

4.29.3.4 `void MHV_RTC::elapsed (MHV_TIMESTAMP * since, MHV_TIMESTAMP * elapsed)`

Determine how long has elapsed since the supplied timestamp

Parameters

<i>since</i>	the timestamp to measure against
<i>elapsed</i>	returns how much time has elapsed

Definition at line 289 of file MHV_RTC.cpp.

4.29.3.5 `void MHV_RTC::handleEvents (void)`

Check for events that are due, and run them Run this from your main loop if you have no blocking calls, otherwise, call tickAndRunEvents instead of tick from the timer (note that this will run your events in the interrupt handler, so keep them short!)

Definition at line 474 of file MHV_RTC.cpp.

4.29.3.6 `void MHV_RTC::removeAlarm (MHV_AlarmListener * listener)`

Remove all matching events from the list of pending events

Parameters

<i>listener</i>	the listener for the event to remove
-----------------	--------------------------------------

Definition at line 516 of file MHV_RTC.cpp.

4.29.3.7 void MHV_RTC::setTime (uint32_t *timestamp*, uint16_t *milliseconds*)

Set the current time

Parameters

<i>timestamp</i>	the current Unix timestamp
<i>milliseconds</i>	the current milliseconds offset

Definition at line 195 of file MHV_RTC.cpp.

4.29.3.8 void MHV_RTC::setTime (MHV_TIMESTAMP * *timestamp*)

Set the current time

Parameters

<i>timestamp</i>	the current Unix timestamp
------------------	----------------------------

Definition at line 206 of file MHV_RTC.cpp.

4.29.3.9 void MHV_RTC::synchronise (void)

Synchronise the ticksPerMillisecond with the timer (useful if you change the timer values)

Definition at line 185 of file MHV_RTC.cpp.

4.29.3.10 void MHV_RTC::tick (void)

Tick from the timer module

Definition at line 217 of file MHV_RTC.cpp.

4.29.3.11 void MHV_RTC::tick1ms (void)

Tick from the timer module that is exactly 1ms

Definition at line 234 of file MHV_RTC.cpp.

4.29.3.12 void MHV_RTC::tick1msAndRunEvents (void)

Tick from the timer module that is exactly 1ms, run any pending events

Definition at line 264 of file MHV_RTC.cpp.

4.29.3.13 void MHV_RTC::tickAndRunEvents (void)

Tick from the timer module, and run any pending events

Definition at line 245 of file MHV_RTC.cpp.

4.29.3.14 void MHV_RTC::toTime (MHV_TIME * *to*, MHV_TIMESTAMP * *from*)

Convert a timestamp into year,month,day,hours,minutes,seconds

Parameters

<i>to</i>	the MHV_TIME to store the results
<i>from</i>	the MHV_TIMESTAMP struct to convert from

Definition at line 327 of file MHV_RTC.cpp.

4.29.3.15 void MHV_RTC::toTimestamp (MHV_TIMESTAMP * *to*, MHV_TIME * *from*)

Convert year,month,day,hours,minutes,seconds into a timestamp

Parameters

<i>to</i>	the MHV_TIMESTAMP to store the results
<i>from</i>	the MHV_TIME struct to convert from

Definition at line 396 of file MHV_RTC.cpp.

4.29.4 Member Data Documentation**4.29.4.1 volatile uint8_t MHV_RTC::_alarmCount [protected]**

Definition at line 106 of file MHV_RTC.h.

4.29.4.2 uint8_t MHV_RTC::_alarmMax [protected]

Definition at line 107 of file MHV_RTC.h.

4.29.4.3 MHV_ALARM* MHV_RTC::_alarms [protected]

Definition at line 105 of file MHV_RTC.h.

4.29.4.4 uint16_t MHV_RTC::_milliseconds [protected]

Definition at line 109 of file MHV_RTC.h.

4.29.4.5 uint8_t MHV_RTC::_ticks [protected]

Definition at line 110 of file MHV_RTC.h.

4.29.4.6 uint8_t MHV_RTC::_ticksPerMillisecond [protected]

Definition at line 111 of file MHV_RTC.h.

4.29.4.7 MHV_Timer8* MHV_RTC::_timer [protected]

Definition at line 104 of file MHV_RTC.h.

4.29.4.8 uint32_t MHV_RTC::_timestamp [protected]

Definition at line 108 of file MHV_RTC.h.

4.29.4.9 int16_t MHV_RTC::_tzOffset [protected]

Definition at line 112 of file MHV_RTC.h.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_RTC.h](#)
- [A:/eclipse/mhvlb/MHV_RTC.cpp](#)

4.30 MHV_RXListener Class Reference

```
#include <MHV_Device_RX.h>
```

Public Member Functions

- virtual void [rxReady](#) ([MHV_Device_RX](#) *rx)=0

4.30.1 Detailed Description

Definition at line 43 of file MHV_Device_RX.h.

4.30.2 Member Function Documentation

4.30.2.1 `virtual void MHV_RXListener::rxReady (MHV_Device_RX * rx) [pure virtual]`

The documentation for this class was generated from the following file:

- [A:/eclipse/mhvlb/MHV_Device_RX.h](#)

4.31 mhv_time Struct Reference

```
#include <MHV_RTC.h>
```

Public Attributes

- `uint16_t milliseconds`
- `uint8_t seconds`
- `uint8_t minutes`
- `uint8_t hours`
- `uint8_t day`
- `MHV_MONTH month`
- `uint16_t year`
- `uint16_t yearday`
- `uint16_t timezone`

4.31.1 Detailed Description

Definition at line 67 of file MHV_RTC.h.

4.31.2 Member Data Documentation

4.31.2.1 `uint8_t mhv_time::day`

Definition at line 72 of file MHV_RTC.h.

4.31.2.2 `uint8_t mhv_time::hours`

Definition at line 71 of file MHV_RTC.h.

4.31.2.3 uint16_t mhv_time::milliseconds

Definition at line 68 of file MHV_RTC.h.

4.31.2.4 uint8_t mhv_time::minutes

Definition at line 70 of file MHV_RTC.h.

4.31.2.5 MHV_MONTH mhv_time::month

Definition at line 73 of file MHV_RTC.h.

4.31.2.6 uint8_t mhv_time::seconds

Definition at line 69 of file MHV_RTC.h.

4.31.2.7 uint16_t mhv_time::timezone

Definition at line 77 of file MHV_RTC.h.

4.31.2.8 uint16_t mhv_time::year

Definition at line 74 of file MHV_RTC.h.

4.31.2.9 uint16_t mhv_time::yearday

Definition at line 76 of file MHV_RTC.h.

The documentation for this struct was generated from the following file:

- [A:/eclipse/mhvlb/MHV_RTC.h](#)

4.32 MHV_Timer8 Class Reference

```
#include <MHV_Timer8.h>
```

Public Member Functions

- [MHV_Timer8](#) ([MHV_TIMER_TYPE](#) type, volatile uint8_t *controlRegA, volatile uint8_t *controlRegB, volatile uint8_t *overflowReg1, volatile uint8_t *overflowReg2, volatile uint8_t *counter, volatile uint8_t *interrupt, uint8_t interruptEnableA)
- bool [setPeriods](#) (uint32_t usec1, uint32_t usec2)

- uint8_t [current](#) ()
- void [setPeriods](#) (MHV_TIMER_PRESCALER prescaler, uint8_t time1, uint8_t time2)
- MHV_TIMER_PRESCALER [getPrescaler](#) ()
- uint16_t [getPrescalerMultiplier](#) ()
- void [setPrescaler](#) (MHV_TIMER_PRESCALER prescaler)
- uint8_t [getTop](#) ()
- void [setTop](#) (uint8_t value)
- void [setOutput](#) (uint8_t channel, uint8_t value)
- void [setOutput1](#) (uint8_t value)
- void [setOutput2](#) (uint8_t value)
- uint8_t [getOutput](#) (uint8_t channel)
- uint8_t [getOutput1](#) ()
- uint8_t [getOutput2](#) ()
- void [connectOutput1](#) (MHV_TIMER_CONNECT_TYPE type)
- void [connectOutput2](#) (MHV_TIMER_CONNECT_TYPE type)
- void [enable](#) ()
- void [disable](#) ()
- bool [enabled](#) ()
- void [trigger1](#) ()
- void [trigger2](#) ()
- void [setTriggers](#) (void(*triggerFunction1)(void *triggerData), void *triggerData1, void(*triggerFunction2)(void *triggerData), void *triggerData2)
- void [setMode](#) (MHV_TIMER_MODE mode)

Protected Member Functions

- uint8_t [calculatePrescaler](#) (uint32_t time, MHV_TIMER_PRESCALER *prescaler, uint16_t *factor)
- void [calculateTop](#) (uint32_t *time, uint16_t factor)
- void [setGenerationMode](#) ()
- MHV_Timer8 ()
- void [_setPrescaler](#) (MHV_TIMER_PRESCALER prescaler)

Protected Attributes

- volatile uint8_t * [_controlRegA](#)
- volatile uint8_t * [_controlRegB](#)
- volatile uint8_t * [_outputCompare1](#)
- volatile uint8_t * [_outputCompare2](#)
- volatile uint8_t * [_counter](#)
- volatile uint8_t * [_interrupt](#)
- uint8_t [_interruptEnableA](#)
- MHV_TIMER_PRESCALER [_prescaler](#)
- MHV_TIMER_MODE [_mode](#)
- MHV_TIMER_TYPE [_type](#)

- uint8_t [_counterSize](#)
- bool [_haveTime2](#)
- void(* [_triggerFunction1](#))(void *data)
- void * [_triggerData1](#)
- void(* [_triggerFunction2](#))(void *data)
- void * [_triggerData2](#)

4.32.1 Detailed Description

Definition at line 101 of file MHV_Timer8.h.

4.32.2 Constructor & Destructor Documentation

4.32.2.1 MHV_Timer8::MHV_Timer8 () [protected]

Definition at line 58 of file MHV_Timer8.cpp.

4.32.2.2 MHV_Timer8::MHV_Timer8 (MHV_TIMER_TYPE type, volatile uint8_t * controlRegA, volatile uint8_t * controlRegB, volatile uint8_t * overflowReg1, volatile uint8_t * overflowReg2, volatile uint8_t * counter, volatile uint8_t * interrupt, uint8_t interruptEnableA)

Definition at line 36 of file MHV_Timer8.cpp.

4.32.3 Member Function Documentation

4.32.3.1 void MHV_Timer8::setPrescaler (MHV_TIMER_PRESCALER prescaler) [protected]

Definition at line 182 of file MHV_Timer8.cpp.

4.32.3.2 uint8_t MHV_Timer8::calculatePrescaler (uint32_t time, MHV_TIMER_PRESCALER * prescaler, uint16_t * factor) [protected]

Definition at line 75 of file MHV_Timer8.cpp.

4.32.3.3 void MHV_Timer8::calculateTop (uint32_t * time, uint16_t * factor) [protected]

Definition at line 140 of file MHV_Timer8.cpp.

4.32.3.4 void MHV_Timer8::connectOutput1 (MHV_TIMER_CONNECT_TYPE type)

Definition at line 364 of file MHV_Timer8.cpp.

4.32.3.5 void MHV_Timer8::connectOutput2 (MHV_TIMER_CONNECT_TYPE *type*)

Definition at line 368 of file MHV_Timer8.cpp.

4.32.3.6 uint8_t MHV_Timer8::current (void)

Definition at line 60 of file MHV_Timer8.cpp.

4.32.3.7 void MHV_Timer8::disable (void)

Definition at line 393 of file MHV_Timer8.cpp.

4.32.3.8 void MHV_Timer8::enable (void)

Definition at line 375 of file MHV_Timer8.cpp.

4.32.3.9 bool MHV_Timer8::enabled (void)

Definition at line 403 of file MHV_Timer8.cpp.

4.32.3.10 uint8_t MHV_Timer8::getOutput (uint8_t *channel*)

Definition at line 342 of file MHV_Timer8.cpp.

4.32.3.11 uint8_t MHV_Timer8::getOutput1 (void)

Definition at line 356 of file MHV_Timer8.cpp.

4.32.3.12 uint8_t MHV_Timer8::getOutput2 (void)

Definition at line 360 of file MHV_Timer8.cpp.

4.32.3.13 MHV_TIMER_PRESCALER MHV_Timer8::getPrescaler (void)

Definition at line 189 of file MHV_Timer8.cpp.

4.32.3.14 uint16_t MHV_Timer8::getPrescalerMultiplier (void)

Definition at line 195 of file MHV_Timer8.cpp.

4.32.3.15 `uint8_t MHV_Timer8::getTop (void)`

Definition at line 293 of file MHV_Timer8.cpp.

4.32.3.16 `void MHV_Timer8::setGenerationMode () [protected]`

Definition at line 247 of file MHV_Timer8.cpp.

4.32.3.17 `void MHV_Timer8::setMode (MHV_TIMER_MODE mode)`

Definition at line 429 of file MHV_Timer8.cpp.

4.32.3.18 `void MHV_Timer8::setOutput (uint8_t channel, uint8_t value)`

Definition at line 323 of file MHV_Timer8.cpp.

4.32.3.19 `void MHV_Timer8::setOutput1 (uint8_t value)`

Definition at line 334 of file MHV_Timer8.cpp.

4.32.3.20 `void MHV_Timer8::setOutput2 (uint8_t value)`

Definition at line 338 of file MHV_Timer8.cpp.

4.32.3.21 `bool MHV_Timer8::setPeriods (uint32_t usec1, uint32_t usec2)`

Definition at line 151 of file MHV_Timer8.cpp.

4.32.3.22 `void MHV_Timer8::setPeriods (MHV_TIMER_PRESCALER prescaler, uint8_t time1, uint8_t time2)`

Definition at line 280 of file MHV_Timer8.cpp.

4.32.3.23 `void MHV_Timer8::setPrescaler (MHV_TIMER_PRESCALER prescaler)`

Definition at line 241 of file MHV_Timer8.cpp.

4.32.3.24 `void MHV_Timer8::setTop (uint8_t value)`

Definition at line 310 of file MHV_Timer8.cpp.

4.32.3.25 void MHV_Timer8::setTriggers (void(*) (void *triggerData) *triggerFunction1*, void * *triggerData1*, void(*) (void *triggerData) *triggerFunction2*, void * *triggerData2*)

Definition at line 420 of file MHV_Timer8.cpp.

4.32.3.26 void MHV_Timer8::trigger1 ()

Definition at line 407 of file MHV_Timer8.cpp.

4.32.3.27 void MHV_Timer8::trigger2 ()

Definition at line 414 of file MHV_Timer8.cpp.

4.32.4 Member Data Documentation

4.32.4.1 volatile uint8_t* MHV_Timer8::_controlRegA [protected]

Definition at line 103 of file MHV_Timer8.h.

4.32.4.2 volatile uint8_t* MHV_Timer8::_controlRegB [protected]

Definition at line 104 of file MHV_Timer8.h.

4.32.4.3 volatile uint8_t* MHV_Timer8::_counter [protected]

Definition at line 107 of file MHV_Timer8.h.

4.32.4.4 uint8_t MHV_Timer8::_counterSize [protected]

Definition at line 113 of file MHV_Timer8.h.

4.32.4.5 bool MHV_Timer8::_haveTime2 [protected]

Definition at line 114 of file MHV_Timer8.h.

4.32.4.6 volatile uint8_t* MHV_Timer8::_interrupt [protected]

Definition at line 108 of file MHV_Timer8.h.

4.32.4.7 uint8_t MHV_Timer8::_interruptEnableA [protected]

Definition at line 109 of file MHV_Timer8.h.

4.32.4.8 MHV_TIMER_MODE MHV_Timer8::_mode [protected]

Definition at line 111 of file MHV_Timer8.h.

4.32.4.9 volatile uint8_t* MHV_Timer8::_outputCompare1 [protected]

Definition at line 105 of file MHV_Timer8.h.

4.32.4.10 volatile uint8_t* MHV_Timer8::_outputCompare2 [protected]

Definition at line 106 of file MHV_Timer8.h.

4.32.4.11 MHV_TIMER_PRESCALER MHV_Timer8::_prescaler [protected]

Definition at line 110 of file MHV_Timer8.h.

4.32.4.12 void* MHV_Timer8::_triggerData1 [protected]

Definition at line 116 of file MHV_Timer8.h.

4.32.4.13 void* MHV_Timer8::_triggerData2 [protected]

Definition at line 118 of file MHV_Timer8.h.

4.32.4.14 void(* MHV_Timer8::_triggerFunction1)(void *data) [protected]

Definition at line 115 of file MHV_Timer8.h.

4.32.4.15 void(* MHV_Timer8::_triggerFunction2)(void *data) [protected]

Definition at line 117 of file MHV_Timer8.h.

4.32.4.16 MHV_TIMER_TYPE MHV_Timer8::_type [protected]

Definition at line 112 of file MHV_Timer8.h.

The documentation for this class was generated from the following files:

- [A:/eclipse/mhvlb/MHV_Timer8.h](#)
- [A:/eclipse/mhvlb/MHV_Timer8.cpp](#)

4.33 mhv_timestamp Struct Reference

```
#include <MHV_RTC.h>
```

Public Attributes

- uint32_t [timestamp](#)
- uint16_t [milliseconds](#)

4.33.1 Detailed Description

Definition at line 32 of file MHV_RTC.h.

4.33.2 Member Data Documentation

4.33.2.1 uint16_t mhv_timestamp::milliseconds

Definition at line 34 of file MHV_RTC.h.

4.33.2.2 uint32_t mhv_timestamp::timestamp

Definition at line 33 of file MHV_RTC.h.

The documentation for this struct was generated from the following file:

- A:/eclipse/mhvlb/MHV_RTC.h

4.34 mhv_tx_buffer Struct Reference

```
#include <MHV_Device_TX.h>
```

Public Attributes

- const char * [data](#)
- uint16_t [length](#)
- void(* [completeFunction](#))(const char *)
- bool [progmem](#)
- bool [isString](#)

4.34.1 Detailed Description

Definition at line 45 of file MHV_Device_TX.h.

4.34.2 Member Data Documentation

4.34.2.1 `void(* mhv_tx_buffer::completeFunction)(const char *)`

Definition at line 48 of file MHV_Device_TX.h.

4.34.2.2 `const char* mhv_tx_buffer::data`

Definition at line 46 of file MHV_Device_TX.h.

4.34.2.3 `bool mhv_tx_buffer::isString`

Definition at line 50 of file MHV_Device_TX.h.

4.34.2.4 `uint16_t mhv_tx_buffer::length`

Definition at line 47 of file MHV_Device_TX.h.

4.34.2.5 `bool mhv_tx_buffer::progmem`

Definition at line 49 of file MHV_Device_TX.h.

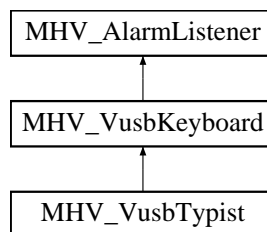
The documentation for this struct was generated from the following file:

- [A:/eclipse/mhvlb/MHV_Device_TX.h](#)

4.35 MHV_VusbKeyboard Class Reference

```
#include <MHV_VusbKeyboard.h>
```

Inheritance diagram for MHV_VusbKeyboard:



Public Member Functions

- [MHV_VusbKeyboard](#) ([MHV_RTC](#) *rtc)

- void [keyStroke](#) ([MHV_VUSB_KEYBOARD_KEY](#) key)
- void [keyStroke](#) ([MHV_VUSB_KEYBOARD_KEY](#) key, uint8_t modifiers)
- void [keyDown](#) ([MHV_VUSB_KEYBOARD_KEY](#) key, uint8_t modifiers)
- void [keysUp](#) (uint8_t modifiers)
- void [keysUp](#) ()
- void [alarm](#) ([MHV_ALARM](#) *alarm)

Protected Attributes

- [MHV_RTC](#) * [_rtc](#)

4.35.1 Detailed Description

Definition at line 178 of file [MHV_VusbKeyboard.h](#).

4.35.2 Constructor & Destructor Documentation

4.35.2.1 [MHV_VusbKeyboard::MHV_VusbKeyboard](#) ([MHV_RTC](#) * *rtc*)

Emulate a USB keyboard using V-USB Uses pins D4/D2 for ATmega (can be changed in [VUSBKeyboard/usbconfig.h](#)) Uses pins B0/B2 for ATtiny25/45/85

Parameters

<i>rtc</i>	an RTC to schedule jobs on
------------	----------------------------

Definition at line 158 of file [MHV_VusbKeyboard.cpp](#).

4.35.3 Member Function Documentation

4.35.3.1 void [MHV_VusbKeyboard::alarm](#) ([MHV_ALARM](#) * *alarm*) [virtual]

Periodically called to maintain USB comms

Implements [MHV_AlarmListener](#).

Reimplemented in [MHV_VusbTypist](#).

Definition at line 249 of file [MHV_VusbKeyboard.cpp](#).

4.35.3.2 void [MHV_VusbKeyboard::keyDown](#) ([MHV_VUSB_KEYBOARD_KEY](#) *key*, uint8_t *modifiers*)

Press a key

Parameters

<i>key</i>	the key to send
<i>modifiers</i>	the key modifiers

Definition at line 219 of file MHV_VusbKeyboard.cpp.

4.35.3.3 void MHV_VusbKeyboard::keyStroke (MHV_VUSB_KEYBOARD_KEY *key*)

Send a single keystroke

Parameters

<i>key</i>	the key to send
------------	-----------------

Returns

false if the keyStroke was not sent

Definition at line 210 of file MHV_VusbKeyboard.cpp.

4.35.3.4 void MHV_VusbKeyboard::keyStroke (MHV_VUSB_KEYBOARD_KEY *key*, uint8_t *modifiers*)

Send a single keystroke

Parameters

<i>key</i>	the key to send
<i>modifiers</i>	the key modifiers

Returns

false if the keyStroke was not sent

Definition at line 199 of file MHV_VusbKeyboard.cpp.

4.35.3.5 void MHV_VusbKeyboard::keysUp (uint8_t *modifiers*)

Release all keys

Parameters

<i>modifiers</i>	the key modifiers still held
------------------	------------------------------

Definition at line 231 of file MHV_VusbKeyboard.cpp.

4.35.3.6 void MHV_VusbKeyboard::keysUp ()

Release all keys

Definition at line 242 of file MHV_VusbKeyboard.cpp.

4.35.4 Member Data Documentation

4.35.4.1 MHV_RTC* MHV_VusbKeyboard::_rtc [protected]

Definition at line 180 of file MHV_VusbKeyboard.h.

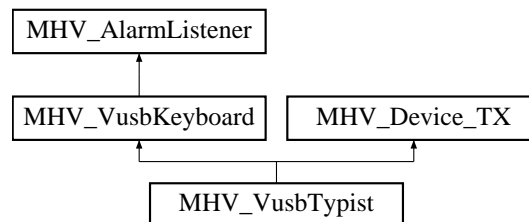
The documentation for this class was generated from the following files:

- A:/eclipse/mhvlb-Vusb-Keyboard/MHV_VusbKeyboard.h
- A:/eclipse/mhvlb-Vusb-Keyboard/MHV_VusbKeyboard.cpp

4.36 MHV_VusbTypist Class Reference

```
#include <MHV_VusbTypist.h>
```

Inheritance diagram for MHV_VusbTypist:



Public Member Functions

- [MHV_VusbTypist](#) ([MHV_RingBuffer](#) *txBuffer, [MHV_RTC](#) *rtc)
- void [alarm](#) ([MHV_ALARM](#) *alarm)

Protected Member Functions

- void [runTxBuffers](#) ()
- void [typeChar](#) (char c)

Protected Attributes

- bool [_isTyping](#)

4.36.1 Detailed Description

Definition at line 25 of file MHV_VusbTypist.h.

4.36.2 Constructor & Destructor Documentation

4.36.2.1 `MHV_VusbTypist::MHV_VusbTypist (MHV_RingBuffer * txBuffer, MHV_RTC * rtc)`

Emulate a USB keyboard using V-USB This class can also be passed strings, which it will type out on the keyboard

Parameters

<i>txBuffer</i>	a ringbuffer to store data in
<i>rtc</i>	an RTC to trigger events from

Definition at line 33 of file MHV_VusbTypist.cpp.

4.36.3 Member Function Documentation

4.36.3.1 `void MHV_VusbTypist::alarm (MHV_ALARM * alarm) [virtual]`

Periodically called to maintain USB comms

Reimplemented from [MHV_VusbKeyboard](#).

Definition at line 41 of file MHV_VusbTypist.cpp.

4.36.3.2 `void MHV_VusbTypist::runTxBuffers () [protected, virtual]`

Start transmitting a new string (does nothing, alarm will immediately pick up the next character)

Implements [MHV_Device_TX](#).

Definition at line 63 of file MHV_VusbTypist.cpp.

4.36.3.3 `void MHV_VusbTypist::typeChar (char c) [protected]`

Type a single character on the keyboard

Definition at line 69 of file MHV_VusbTypist.cpp.

4.36.4 Member Data Documentation

4.36.4.1 `bool MHV_VusbTypist::_isTyping` `[protected]`

Definition at line 27 of file MHV_VusbTypist.h.

The documentation for this class was generated from the following files:

- A:/eclipse/mhplib-Vusb-Keyboard/[MHV_VusbTypist.h](#)
- A:/eclipse/mhplib-Vusb-Keyboard/[MHV_VusbTypist.cpp](#)

Chapter 5

File Documentation

5.1 A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbKeyboard.cpp File Reference

```
#include <vusb/usbdrv.h> #include <MHV_VusbKeyboard.h> ×  
#include <avr/pgmspace.h> #include <util/delay.h>
```

Defines

- #define [MHV_OSCCAL_EEPROM_ADDRESS](#) 0

Functions

- unsigned char [usbFunctionSetup](#) (uchar data[8])

Variables

- [PROGMEM](#) const char [usbHidReportDescriptor](#) [USB_CFG_HID_REPORT_DESCRIPTOR_LENGTH]

5.1.1 Define Documentation

5.1.1.1 #define MHV_OSCCAL_EEPROM_ADDRESS 0

Definition at line 27 of file MHV_VusbKeyboard.cpp.

5.1.2 Function Documentation

5.1.2.1 unsigned char usbFunctionSetup (uchar *data*[8])

Definition at line 131 of file MHV_VusbKeyboard.cpp.

5.1.3 Variable Documentation

5.1.3.1 PROGRAMMEM const char usbHidReportDescriptor[USB_CFG_HID_REPORT_DESCRIPTOR_LENGTH]

Initial value:

```
{
    0x05, 0x01,
    0x09, 0x06,
    0xa1, 0x01,
    0x05, 0x07,
    0x19, 0xe0,
    0x29, 0xe7,
    0x15, 0x00,
    0x25, 0x01,
    0x75, 0x01,
    0x95, 0x08,
    0x81, 0x02,
    0x95, 0x01,
    0x75, 0x08,
    0x25, 0x65,
    0x19, 0x00,
    0x29, 0x65,
    0x81, 0x00,
    0xc0
}
```

Definition at line 37 of file MHV_VusbKeyboard.cpp.

5.2 A:/eclipse/mhvlb-Vusb-Keyboard/MHV_VusbKeyboard.h File - Reference

```
#include <MHV_io.h> #include <MHV_RTC.h>
```

Classes

- class [MHV_VusbKeyboard](#)

Defines

- #define [MHV_MOD_CONTROL_LEFT](#) (1<<0)
- #define [MHV_MOD_SHIFT_LEFT](#) (1<<1)
- #define [MHV_MOD_ALT_LEFT](#) (1<<2)
- #define [MHV_MOD_GUI_LEFT](#) (1<<3)

- #define `MHV_MOD_CONTROL_RIGHT` (1<<4)
- #define `MHV_MOD_SHIFT_RIGHT` (1<<5)
- #define `MHV_MOD_ALT_RIGHT` (1<<6)
- #define `MHV_MOD_GUI_RIGHT` (1<<7)

Typedefs

- typedef enum `mhv_vusb_keyboard_key` `MHV_VUSB_KEYBOARD_KEY`

Enumerations

- enum `mhv_vusb_keyboard_key` { `MHV_KEY_A` = 4, `MHV_KEY_B` = 5, `MHV_KEY_C` = 6, `MHV_KEY_D` = 7, `MHV_KEY_E` = 8, `MHV_KEY_F` = 9, `MHV_KEY_G` = 10, `MHV_KEY_H` = 11, `MHV_KEY_I` = 12, `MHV_KEY_J` = 13, `MHV_KEY_K` = 14, `MHV_KEY_L` = 15, `MHV_KEY_M` = 16, `MHV_KEY_N` = 17, `MHV_KEY_O` = 18, `MHV_KEY_P` = 19, `MHV_KEY_Q` = 20, `MHV_KEY_R` = 21, `MHV_KEY_S` = 22, `MHV_KEY_T` = 23, `MHV_KEY_U` = 24, `MHV_KEY_V` = 25, `MHV_KEY_W` = 26, `MHV_KEY_X` = 27, `MHV_KEY_Y` = 28, `MHV_KEY_Z` = 29, `MHV_KEY_1` = 30, `MHV_KEY_2` = 31, `MHV_KEY_3` = 32, `MHV_KEY_4` = 33, `MHV_KEY_5` = 34, `MHV_KEY_6` = 35, `MHV_KEY_7` = 36, `MHV_KEY_8` = 37, `MHV_KEY_9` = 38, `MHV_KEY_0` = 39, `MHV_KEY_ENTER` = 40, `MHV_KEY_ESCAPE` = 41, `MHV_KEY_BACKSPACE` = 42, `MHV_KEY_TAB` = 43, `MHV_KEY_SPACE` = 44, `MHV_KEY_MINUS` = 45, `MHV_KEY_EQUALS` = 46, `MHV_KEY_L_SQUARE` = 47, `MHV_KEY_R_SQUARE` = 48, `MHV_KEY_BACKSLASH` = 49, `MHV_KEY_NON_US_HASH` = 50, `MHV_KEY_SEMICOLON` = 51, `MHV_KEY_QUOTE` = 52, `MHV_KEY_GRAVE_ACCENT` = 53, `MHV_KEY_COMMA` = 54, `MHV_KEY_FULLSTOP` = 55, `MHV_KEY_SLASH` = 56, `MHV_KEY_CAPSLOCK` = 57, `MHV_KEY_F1` = 58, `MHV_KEY_F2` = 59, `MHV_KEY_F3` = 60, `MHV_KEY_F4` = 61, `MHV_KEY_F5` = 62, `MHV_KEY_F6` = 63, `MHV_KEY_F7` = 64, `MHV_KEY_F8` = 65, `MHV_KEY_F9` = 66, `MHV_KEY_F10` = 67, `MHV_KEY_F11` = 68, `MHV_KEY_F12` = 69, `MHV_KEY_PRINTSCREEN` = 70, `MHV_KEY_SCROLL_LOCK` = 71, `MHV_KEY_PAUSE` = 72, `MHV_KEY_INSERT` = 73, `MHV_KEY_HOME` = 74, `MHV_KEY_PAGE_UP` = 75, `MHV_KEY_DELETE` = 76, `MHV_KEY_END` = 77, `MHV_KEY_PAGE_DOWN` = 78, `MHV_KEY_ARROW_RIGHT` = 79, `MHV_KEY_ARROW_LEFT` = 80, `MHV_KEY_ARROW_DOWN` = 81, `MHV_KEY_ARROW_UP` = 82, `MHV_KEY_NUM_LOCK` = 83, `MHV_KEYPAD_SLASH` = 84, `MHV_KEYPAD_ASTERISK` = 85, `MHV_KEYPAD_MINUS` = 86, `MHV_KEYPAD_PLUS` = 87, `MHV_KEYPAD_ENTER` = 88, `MHV_KEYPAD_1` = 89, `MHV_KEYPAD_2` = 90, `MHV_KEYPAD_3` = 91, `MHV_KEYPAD_4` = 92, `MHV_KEYPAD_5` = 93, `MHV_KEYPAD_6` = 94, `MHV_KEYPAD_7` = 95, `MHV_KEYPAD_8` = 96, `MHV_KEYPAD_9` = 97, `MHV_KEYPAD_0` = 98, `MHV_KEYPAD_FULLSTOP` = 99, `MHV_KEY_NON_US_BACKSLASH` = 100, `MHV_KEY_APPLICATION` = 101, `MHV_KEYPAD_POWER` = 102, `MHV_KEYPAD_EQUALS` = 103, `MHV_KEY_F13` = 104, `MHV_KEY_F14` = 105, `MHV_KEY_F15` = 106, `MHV_KEY_F16` = 107, `MHV_KEY_F17` = 108, `MHV_KEY_F18` = 109, `MHV_KEY_F19` = 110, `MHV_KEY_F20` = 111, `MHV_KEY_F21` = 112, `MHV_KEY_F22` = 113, `MHV_KEY_F23` = 114, `MHV_KEY_F24` = 115, `MHV_KEY_EXECUTE` = 116, `MHV_KEY_HELP` = 117, `MHV_KEY_MENU` = 118, `MHV_KEY_SELECT` = 119,

```
    MHV_KEY_STOP = 120, MHV_KEY_AGAIN = 121, MHV_KEY_UNDO = 122,
    MHV_KEY_CUT = 123, MHV_KEY_COPY = 124, MHV_KEY_PASTE = 125, -
    MHV_KEY_FIND = 126, MHV_KEY_MUTE = 127, MHV_KEY_VOLUME_UP =
    128, MHV_KEY_VOLUME_DOWN = 129, MHV_KEY_LOCKING_CAPS_LOCK
    = 130, MHV_KEY_LOCKING_NUM_LOCK = 131, MHV_KEY_LOCKING_SC-
    ROLL_LOCK = 132, MHV_KEYPAD_COMMA = 133, MHV_KEYPAD_EQUAL
    = 134, MHV_KEY_CONTROL_LEFT = 224, MHV_KEY_SHIFT_LEFT = 225,
    MHV_KEY_ALT_LEFT = 226, MHV_KEY_GUI_LEFT = 227, MHV_KEY_CONT-
    ROL_RIGHT = 228, MHV_KEY_SHIFT_RIGHT = 229, MHV_KEY_ALT_RIGHT
    = 230, MHV_KEY_GUI_RIGHT = 231 }
```

5.2.1 Define Documentation

5.2.1.1 #define MHV_MOD_ALT_LEFT (1<<2)

Definition at line 27 of file MHV_VusbKeyboard.h.

5.2.1.2 #define MHV_MOD_ALT_RIGHT (1<<6)

Definition at line 31 of file MHV_VusbKeyboard.h.

5.2.1.3 #define MHV_MOD_CONTROL_LEFT (1<<0)

Definition at line 25 of file MHV_VusbKeyboard.h.

5.2.1.4 #define MHV_MOD_CONTROL_RIGHT (1<<4)

Definition at line 29 of file MHV_VusbKeyboard.h.

5.2.1.5 #define MHV_MOD_GUI_LEFT (1<<3)

Definition at line 28 of file MHV_VusbKeyboard.h.

5.2.1.6 #define MHV_MOD_GUI_RIGHT (1<<7)

Definition at line 32 of file MHV_VusbKeyboard.h.

5.2.1.7 #define MHV_MOD_SHIFT_LEFT (1<<1)

Definition at line 26 of file MHV_VusbKeyboard.h.

5.2.1.8 `#define MHV_MOD_SHIFT_RIGHT (1<<5)`

Definition at line 30 of file MHV_VusbKeyboard.h.

5.2.2 Typedef Documentation

5.2.2.1 `typedef enum mhv_vusb_keyboard_key MHV_VUSB_KEYBOARD_KEY`

Definition at line 176 of file MHV_VusbKeyboard.h.

5.2.3 Enumeration Type Documentation

5.2.3.1 `enum mhv_vusb_keyboard_key`

Enumerator:

MHV_KEY_A
MHV_KEY_B
MHV_KEY_C
MHV_KEY_D
MHV_KEY_E
MHV_KEY_F
MHV_KEY_G
MHV_KEY_H
MHV_KEY_I
MHV_KEY_J
MHV_KEY_K
MHV_KEY_L
MHV_KEY_M
MHV_KEY_N
MHV_KEY_O
MHV_KEY_P
MHV_KEY_Q
MHV_KEY_R
MHV_KEY_S
MHV_KEY_T
MHV_KEY_U
MHV_KEY_V
MHV_KEY_W
MHV_KEY_X
MHV_KEY_Y

MHV_KEY_Z
MHV_KEY_1
MHV_KEY_2
MHV_KEY_3
MHV_KEY_4
MHV_KEY_5
MHV_KEY_6
MHV_KEY_7
MHV_KEY_8
MHV_KEY_9
MHV_KEY_0
MHV_KEY_ENTER
MHV_KEY_ESCAPE
MHV_KEY_BACKSPACE
MHV_KEY_TAB
MHV_KEY_SPACE
MHV_KEY_MINUS
MHV_KEY_EQUALS
MHV_KEY_L_SQUARE
MHV_KEY_R_SQUARE
MHV_KEY_BACKSLASH
MHV_KEY_NON_US_HASH
MHV_KEY_SEMICOLON
MHV_KEY_QUOTE
MHV_KEY_GRAVE_ACCENT
MHV_KEY_COMMA
MHV_KEY_FULLSTOP
MHV_KEY_SLASH
MHV_KEY_CAPSLOCK
MHV_KEY_F1
MHV_KEY_F2
MHV_KEY_F3
MHV_KEY_F4
MHV_KEY_F5
MHV_KEY_F6
MHV_KEY_F7
MHV_KEY_F8
MHV_KEY_F9

MHV_KEY_F10
MHV_KEY_F11
MHV_KEY_F12
MHV_KEY_PRINTSCREEN
MHV_KEY_SCROLL_LOCK
MHV_KEY_PAUSE
MHV_KEY_INSERT
MHV_KEY_HOME
MHV_KEY_PAGE_UP
MHV_KEY_DELETE
MHV_KEY_END
MHV_KEY_PAGE_DOWN
MHV_KEY_ARROW_RIGHT
MHV_KEY_ARROW_LEFT
MHV_KEY_ARROW_DOWN
MHV_KEY_ARROW_UP
MHV_KEY_NUM_LOCK
MHV_KEYPAD_SLASH
MHV_KEYPAD_ASTERISK
MHV_KEYPAD_MINUS
MHV_KEYPAD_PLUS
MHV_KEYPAD_ENTER
MHV_KEYPAD_1
MHV_KEYPAD_2
MHV_KEYPAD_3
MHV_KEYPAD_4
MHV_KEYPAD_5
MHV_KEYPAD_6
MHV_KEYPAD_7
MHV_KEYPAD_8
MHV_KEYPAD_9
MHV_KEYPAD_0
MHV_KEYPAD_FULLSTOP
MHV_KEY_NON_US_BACKSLASH
MHV_KEY_APPLICATION
MHV_KEYPAD_POWER
MHV_KEYPAD_EQUALS
MHV_KEY_F13

MHV_KEY_F14
MHV_KEY_F15
MHV_KEY_F16
MHV_KEY_F17
MHV_KEY_F18
MHV_KEY_F19
MHV_KEY_F20
MHV_KEY_F21
MHV_KEY_F22
MHV_KEY_F23
MHV_KEY_F24
MHV_KEY_EXECUTE
MHV_KEY_HELP
MHV_KEY_MENU
MHV_KEY_SELECT
MHV_KEY_STOP
MHV_KEY_AGAIN
MHV_KEY_UNDO
MHV_KEY_CUT
MHV_KEY_COPY
MHV_KEY_PASTE
MHV_KEY_FIND
MHV_KEY_MUTE
MHV_KEY_VOLUME_UP
MHV_KEY_VOLUME_DOWN
MHV_KEY_LOCKING_CAPS_LOCK
MHV_KEY_LOCKING_NUM_LOCK
MHV_KEY_LOCKING_SCROLL_LOCK
MHV_KEYPAD_COMMA
MHV_KEYPAD_EQUAL
MHV_KEY_CONTROL_LEFT
MHV_KEY_SHIFT_LEFT
MHV_KEY_ALT_LEFT
MHV_KEY_GUI_LEFT
MHV_KEY_CONTROL_RIGHT
MHV_KEY_SHIFT_RIGHT
MHV_KEY_ALT_RIGHT
MHV_KEY_GUI_RIGHT

Definition at line 34 of file MHV_VusbKeyboard.h.

5.3 A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbTypist.cpp File - Reference

```
#include <MHV_VusbTypist.h>    #include <avr/pgmspace.h> ×  
#include <vusb/usbdrv.h>
```

5.4 A:/eclipse/mhvlb-Vusb-Keybaord/MHV_VusbTypist.h File - Reference

```
#include <MHV_VusbKeyboard.h>    #include <MHV_Device_TX.-  
h>
```

Classes

- class [MHV_VusbTypist](#)

5.5 A:/eclipse/mhvlb/MHV_AD.cpp File Reference

```
#include "MHV_AD.h"
```

Functions

- [uint16_t mhv_ad_busyRead](#) ([uint8_t channel](#), [uint8_t reference](#))
- [void mhv_ad_asyncRead](#) ([uint8_t channel](#), [uint8_t reference](#))
- [void mhv_ad_setPrescaler](#) ([MHV_AD_PRESCALER prescaler](#))

5.5.1 Function Documentation

5.5.1.1 void mhv_ad_asyncRead (uint8_t *channel*, uint8_t *reference*)

Definition at line 44 of file MHV_AD.cpp.

5.5.1.2 uint16_t mhv_ad_busyRead (uint8_t *channel*, uint8_t *reference*)

Definition at line 29 of file MHV_AD.cpp.

5.5.1.3 void mhv_ad_setPrescaler (MHV_AD_PRESCALER *prescaler*)

Definition at line 55 of file MHV_AD.cpp.

5.6 A:/eclipse/mhvlb/MHV_AD.h File Reference

```
#include <MHV_io.h>
```

Defines

- `#define MHV_AD_MAX 1024`
- `#define MHV_AD_REFERENCE (ADMUX & 0xF0)`
- `#define MHV_AD_CHANNEL (ADMUX & 0x0F)`
- `#define MHV_AD_ASSIGN_INTERRUPT(adcsTrigger)`
- `#define MHV_AD_ENABLE_INTERRUPT ADCSRA |= _BV(ADIE)`
- `#define MHV_AD_DISABLE_INTERRUPT ADCSRA &= ~_BV(ADIE)`
- `#define MHV_AD_ENABLE`
- `#define MHV_AD_DISABLE`

Typedefs

- `typedef enum mhv_ad_prescaler MHV_AD_PRESCALER`

Enumerations

- `enum mhv_ad_prescaler { MHV_AD_PRESCALER_2 = 1, MHV_AD_PRESCALER_4 = 2, MHV_AD_PRESCALER_8 = 3, MHV_AD_PRESCALER_16 = 4, MHV_AD_PRESCALER_32 = 5, MHV_AD_PRESCALER_64 = 6, MHV_AD_PRESCALER_128 = 7 }`

Functions

- `uint16_t mhv_ad_busyRead (uint8_t channel, uint8_t reference)`
- `void mhv_ad_asyncRead (uint8_t channel, uint8_t reference)`
- `void mhv_ad_setPrescaler (MHV_AD_PRESCALER prescaler)`

5.6.1 Define Documentation

5.6.1.1 `#define MHV_AD_ASSIGN_INTERRUPT(adcTrigger)`

Value:

```
ISR(ADC_vect) { \
    adcTrigger(ADC); \
}
```

Definition at line 55 of file MHV_AD.h.

5.6.1.2 #define MHV_AD_CHANNEL (ADMUX & 0x0F)

Definition at line 51 of file MHV_AD.h.

5.6.1.3 #define MHV_AD_DISABLE

Value:

```
do { \
    MHV_AD_PRR |= _BV(PRADC); \
    ADCSRA |= _BV(ADEN); \
} while (0)
```

Definition at line 71 of file MHV_AD.h.

5.6.1.4 #define MHV_AD_DISABLE_INTERRUPT ADCSRA &= ~_BV(ADIE)

Definition at line 62 of file MHV_AD.h.

5.6.1.5 #define MHV_AD_ENABLE

Value:

```
do { \
    MHV_AD_PRR &= ~_BV(PRADC); \
    ADCSRA |= _BV(ADEN); \
} while (0)
```

Definition at line 65 of file MHV_AD.h.

5.6.1.6 #define MHV_AD_ENABLE_INTERRUPT ADCSRA |= _BV(ADIE)

Definition at line 61 of file MHV_AD.h.

5.6.1.7 #define MHV_AD_MAX 1024

Definition at line 33 of file MHV_AD.h.

5.6.1.8 #define MHV_AD_REFERENCE (ADMUX & 0xF0)

Definition at line 47 of file MHV_AD.h.

5.6.2 Typedef Documentation

5.6.2.1 typedef enum mhv_ad_prescaler MHV_AD_PRESCALER

Definition at line 44 of file MHV_AD.h.

5.6.3 Enumeration Type Documentation

5.6.3.1 enum mhv_ad_prescaler

Enumerator:

MHV_AD_PRESCALER_2
MHV_AD_PRESCALER_4
MHV_AD_PRESCALER_8
MHV_AD_PRESCALER_16
MHV_AD_PRESCALER_32
MHV_AD_PRESCALER_64
MHV_AD_PRESCALER_128

Definition at line 35 of file MHV_AD.h.

5.6.4 Function Documentation

5.6.4.1 void mhv_ad_asyncRead (uint8_t channel, uint8_t reference)

Definition at line 44 of file MHV_AD.cpp.

5.6.4.2 uint16_t mhv_ad_busyRead (uint8_t channel, uint8_t reference)

Definition at line 29 of file MHV_AD.cpp.

5.6.4.3 void mhv_ad_setPrescaler (MHV_AD_PRESCALER prescaler)

Definition at line 55 of file MHV_AD.cpp.

5.7 A:/eclipse/mhvlb/MHV_ADC.cpp File Reference

```
#include <MHV_ADC.h>
```

5.8 A:/eclipse/mhvlb/MHV_ADC.h File Reference

```
#include <MHV_io.h> #include <MHV_AD.h>
```

Classes

- class [MHV_ADCListener](#)
- struct [mhv_eventADC](#)
- class [MHV_ADC](#)

Defines

- #define `MHV_ADC_ASSIGN_INTERRUPT`(__mhvADCManager)
- #define `MHV_ADC_BUFFER_CREATE`(_mhvADCBufferName, _mhvADCBufferCount) `MHV_EVENT_ADC` _mhvADCBufferName[_mhvADCBufferCount];

Typedefs

- typedef struct `mhv_eventADC` `MHV_EVENT_ADC`

5.8.1 Define Documentation

5.8.1.1 #define `MHV_ADC_ASSIGN_INTERRUPT`(__mhvADCManager)

Value:

```
ISR(ADC_vect) { \
    __mhvADCManager.adc(); \
}
```

Definition at line 35 of file `MHV_ADC.h`.

5.8.1.2 #define `MHV_ADC_BUFFER_CREATE`(_mhvADCBufferName, _mhvADCBufferCount) `MHV_EVENT_ADC` _mhvADCBufferName[_mhvADCBufferCount];

Allocate a buffer to contain event mappings

Definition at line 43 of file `MHV_ADC.h`.

5.8.2 Typedef Documentation

5.8.2.1 typedef struct `mhv_eventADC` `MHV_EVENT_ADC`

Definition at line 55 of file `MHV_ADC.h`.

5.9 A:/eclipse/mhvlb/MHV_Debounce.cpp File Reference

```
#include <MHV_Debounce.h>
```

5.10 A:/eclipse/mhvlb/MHV_Debounce.h File Reference

```
#include <string.h> #include <MHV_RTC.h> #include <MHV_
io.h> #include <MHV_PinChangeManager.h>
```

Classes

- class [MHV_DebounceListener](#)
- struct [mhv_debouncePin](#)
- class [MHV_Debounce](#)

Defines

- `#define MHV_DEBOUNCE_ASSIGN_PCINT(__mhvDebounce)`
- `#define MHV_DEBOUNCE_ASSIGN_PCINT0(__mhvDebounce)`
- `#define MHV_DEBOUNCE_ASSIGN_PCINT1(__mhvDebounce)`
- `#define MHV_DEBOUNCE_ASSIGN_PCINT2(__mhvDebounce)`
- `#define MHV_DEBOUNCE_ASSIGN_INTERRUPTS(mhvDebounce) MHV_DEBOUNCE_ASSIGN_PCINT(_mhvDebounce)`

Typedefs

- typedef struct [mhv_debouncePin](#) [MHV_DEBOUNCE_PIN](#)

5.10.1 Define Documentation

5.10.1.1 `#define MHV_DEBOUNCE_ASSIGN_INTERRUPTS(mhvDebounce) MHV_DEBOUNCE_ASSIGN_PCINT(_mhvDebounce)`

Definition at line 52 of file `MHV_Debounce.h`.

5.10.1.2 `#define MHV_DEBOUNCE_ASSIGN_PCINT(__mhvDebounce)`

Value:

```
ISR(PCINT_vect) { \
    __mhvDebounce.pinChange0(); \
}
```

Definition at line 18 of file `MHV_Debounce.h`.

5.10.1.3 `#define MHV_DEBOUNCE_ASSIGN_PCINT0(__mhvDebounce)`

Value:

```
ISR(PCINT0_vect) { \
    __mhvDebounce.pinChange0(); \
}
```

Definition at line 23 of file `MHV_Debounce.h`.

5.10.1.4 #define MHV_DEBOUNCE_ASSIGN_PCINT1(__mhvDebounce)

Value:

```
ISR(PCINT1_vect) { \
    __mhvDebounce.pinChange1(); \
}
```

Definition at line 28 of file MHV_Debounce.h.

5.10.1.5 #define MHV_DEBOUNCE_ASSIGN_PCINT2(__mhvDebounce)

Value:

```
ISR(PCINT2_vect) { \
    __mhvDebounce.pinChange2(); \
}
```

Definition at line 33 of file MHV_Debounce.h.

5.10.2 Typedef Documentation

5.10.2.1 typedef struct mhv_debouncePin MHV_DEBOUNCE_PIN

Definition at line 68 of file MHV_Debounce.h.

5.11 A:/eclipse/mhvlb/MHV_Device_RX.cpp File Reference

```
#include <stdio.h>    #include <avr/pgmspace.h>    #include
<stdlib.h>    #include <string.h>    #include <inttypes.h> ×
#include <MHV_Device_RX.h>
```

5.12 A:/eclipse/mhvlb/MHV_Device_RX.h File Reference

```
#include <inttypes.h> #include <avr/interrupt.h> #include
<MHV_io.h> #include <stdio.h> #include <MHV_RingBuffer.-
h>
```

Classes

- class [MHV_RXListener](#)
- class [MHV_Device_RX](#)

Defines

- `#define MHV_RX_BUFFER_CREATE(_mhvRxName, _mhvRxCharacterCount)`

5.12.1 Define Documentation

5.12.1.1 `#define MHV_RX_BUFFER_CREATE(_mhvRxName, _mhvRxCharacterCount)`

Value:

```
char _mhvRxName ## Buf[_mhvRxCharacterCount + 1]; \
    MHV_RingBuffer _mhvRxName(_mhvRxName ## Buf, _mhvRxCharacterCount + 1);
```

Definition at line 38 of file MHV_Device_RX.h.

5.13 A:/eclipse/mhvlb/MHV_Device_TX.cpp File Reference

```
#include <stdio.h>    #include <avr/pgmspace.h>    #include
<stdlib.h>    #include <string.h>    #include <inttypes.h> ×
#include "MHV_Device_TX.h"
```

5.14 A:/eclipse/mhvlb/MHV_Device_TX.h File Reference

```
#include <inttypes.h> #include <avr/interrupt.h> #include
<MHV_io.h> #include <stdio.h> #include <MHV_RingBuffer.-
h> #include <avr/pgmspace.h>
```

Classes

- struct [mhv_tx_buffer](#)
- class [MHV_Device_TX](#)

Defines

- `#define MHVLIB_NEED_PURE_VIRTUAL`
- `#define MHV_TX_BUFFER_CREATE(_mhvTxName, _mhvTxElementCount)`

Typedefs

- typedef struct [mhv_tx_buffer](#) [MHV_TX_BUFFER](#)

5.14.1 Define Documentation

5.14.1.1 #define MHV_TX_BUFFER_CREATE(_mhvTxName, _mhvTxElementCount)

Value:

```
char _mhvTxName ## Buf[_mhvTxElementCount * sizeof(MHV_TX_BUFFER) + 1]; \
    MHV_RingBuffer _mhvTxName(_mhvTxName ## Buf, _mhvTxElementCount *
    sizeof(MHV_TX_BUFFER) + 1);
```

Definition at line 41 of file MHV_Device_TX.h.

5.14.1.2 #define MHVLIB_NEED_PURE_VIRTUAL

Definition at line 32 of file MHV_Device_TX.h.

5.14.2 Typedef Documentation

5.14.2.1 typedef struct mhv_tx_buffer MHV_TX_BUFFER

Definition at line 52 of file MHV_Device_TX.h.

5.15 A:/eclipse/mhvlb/MHV_Display_Character.cpp File Reference

```
#include <MHV_Display_Character.h>
```

5.16 A:/eclipse/mhvlb/MHV_Display_Character.h File Reference

```
#include <MHV_Device_TX.h>
```

Classes

- class [MHV_Display_Character](#)

5.17 A:/eclipse/mhvlb/MHV_Display_HD44780.cpp File Reference

```
#include <util/delay.h>    #include <MHV_Display_HD44780.-
h>
```

Defines

- #define [HD44780_TINIT](#) 300

5.17.1 Define Documentation

5.17.1.1 #define HD44780_TINIT 300

Definition at line 30 of file MHV_Display_HD44780.cpp.

5.18 A:/eclipse/mhplib/MHV_Display_HD44780.h File Reference

```
#include <MHV_Display_Character.h>
```

Classes

- class [MHV_Display_HD44780](#)

Typedefs

- typedef enum [mhv_hd44780_command](#) [MHV_HD44780_COMMAND](#)

Enumerations

- enum [mhv_hd44780_command](#) { [MHV_44780_CMD_CLEAR](#) = 0x001, [MHV_44780_CMD_RETURN_HOME](#) = 0x002, [MHV_44780_CMD_SET_ENTRY_MODE](#) = 0x004, [MHV_44780_CMD_SET_DISPLAY_MODE](#) = 0x008, [MHV_44780_CMD_SET_CURSOR_MODE](#) = 0x010, [MHV_44780_CMD_SET_FUNCTION](#) = 0x020, [MHV_44780_CMD_SET_CG_ADDR](#) = 0x040, [MHV_44780_CMD_SET_DD_ADDR](#) = 0x080, [MHV_44780_WRITE_CHAR](#) = 0xff }

5.18.1 Typedef Documentation

5.18.1.1 typedef enum mhv_hd44780_command MHV_HD44780_COMMAND

Definition at line 43 of file MHV_Display_HD44780.h.

5.18.2 Enumeration Type Documentation

5.18.2.1 enum mhv_hd44780_command

Enumerator:

MHV_44780_CMD_CLEAR
MHV_44780_CMD_RETURN_HOME
MHV_44780_CMD_SET_ENTRY_MODE
MHV_44780_CMD_SET_DISPLAY_MODE

MHV_44780_CMD_SET_CURSOR_MODE
MHV_44780_CMD_SET_FUNCTION
MHV_44780_CMD_SET_CG_ADDR
MHV_44780_CMD_SET_DD_ADDR
MHV_44780_WRITE_CHAR

Definition at line 32 of file MHV_Display_HD44780.h.

5.19 A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.cpp File Reference

```
#include <util/delay.h>    #include <MHV_Display_HD44780_
Direct_Connect.h>
```

Defines

- #define [HD44780_DB4](#) (1 << _dataPin)
- #define [HD44780_DB5](#) (1 << (_dataPin + 1))
- #define [HD44780_DB6](#) (1 << (_dataPin + 2))
- #define [HD44780_DB7](#) (1 << (_dataPin + 3))
- #define [HD44780_RS](#) (1 << _controlPin)
- #define [HD44780_RW](#) (1 << (_controlPin + 1))
- #define [HD44780_E](#) (1 << (_controlPin + 2))
- #define [HD44780_CONTRAST](#) (1 << _visualPin)
- #define [HD44780_LED](#) (1 << (_visualPin + 1))
- #define [HD44780_TC](#) (1000 * 1000000 / F_CPU / 3 + 1)
- #define [HD44780_TSU1](#) (60 * 1000000 / F_CPU / 3 + 1)
- #define [HD44780_TSU2](#) (195 * 1000000 / F_CPU / 3 + 1)
- #define [HD44780_TDH](#) (5 * 1000000 / F_CPU / 3 + 1)
- #define [HD44780_TW](#) (450 * 1000000 / F_CPU / 3 + 1)
- #define [HD44780_TINIT](#) 300
- #define [HD44780_TCLEAR](#) 1530
- #define [HD44780_TINSTR](#) 39
- #define [HD44780_TRAM](#) 430

5.19.1 Define Documentation

5.19.1.1 #define HD44780_CONTRAST (1 << _visualPin)

Definition at line 37 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.2 #define HD44780_DB4 (1 << _dataPin)

Definition at line 30 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.3 #define HD44780_DB5 (1 << (_dataPin + 1))

Definition at line 31 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.4 #define HD44780_DB6 (1 << (_dataPin + 2))

Definition at line 32 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.5 #define HD44780_DB7 (1 << (_dataPin + 3))

Definition at line 33 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.6 #define HD44780_E (1 << (_controlPin + 2))

Definition at line 36 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.7 #define HD44780_LED (1 << (_visualPin + 1))

Definition at line 38 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.8 #define HD44780_RS (1 << _controlPin)

Definition at line 34 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.9 #define HD44780_RW (1 << (_controlPin + 1))

Definition at line 35 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.10 #define HD44780_TC (1000 * 1000000 / F_CPU / 3 + 1)

Definition at line 42 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.11 #define HD44780_TCLEAR 1530

Definition at line 48 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.20 A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.h File Reference 23

5.19.1.12 `#define HD44780_TDH (5 * 1000000 / F_CPU / 3 + 1)`

Definition at line 45 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.13 `#define HD44780_TINIT 300`

Definition at line 47 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.14 `#define HD44780_TINSTR 39`

Definition at line 49 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.15 `#define HD44780_TRAM 430`

Definition at line 50 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.16 `#define HD44780_TSU1 (60 * 1000000 / F_CPU / 3 + 1)`

Definition at line 43 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.17 `#define HD44780_TSU2 (195 * 1000000 / F_CPU / 3 + 1)`

Definition at line 44 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.19.1.18 `#define HD44780_TW (450 * 1000000 / F_CPU / 3 + 1)`

Definition at line 46 of file MHV_Display_HD44780_Direct_Connect.cpp.

5.20 A:/eclipse/mhvlb/MHV_Display_HD44780_Direct_Connect.h - File Reference

```
#include <MHV_Display_HD44780.h>
```

Classes

- class [MHV_Display_HD44780_Direct_Connect](#)

5.21 A:/eclipse/mhvlb/MHV_Display_HD44780_Shift_Register.cpp - File Reference

```
#include <util/delay.h>    #include <MHV_Display_HD44780_Shift_Register.h> #include <MHV_Shifter.h>
```

Defines

- `#define MHV_SHIFT_ORDER_MSB`
- `#define MHV_SHIFT_WRITECLOCK NULL,_clockOut,NULL,_clockPin,-1`
- `#define MHV_SHIFT_WRITEDATA NULL,_dataOut,NULL,_dataPin,-1`
- `#define HD44780_TSU1 (60 * 1000000 / F_CPU / 3 + 1)`
- `#define HD44780_TW (450 * 1000000 / F_CPU / 3 + 1)`

5.21.1 Define Documentation

5.21.1.1 `#define HD44780_TSU1 (60 * 1000000 / F_CPU / 3 + 1)`

Definition at line 38 of file MHV_Display_HD44780_Shift_Register.cpp.

5.21.1.2 `#define HD44780_TW (450 * 1000000 / F_CPU / 3 + 1)`

Definition at line 39 of file MHV_Display_HD44780_Shift_Register.cpp.

5.21.1.3 `#define MHV_SHIFT_ORDER_MSB`

Definition at line 33 of file MHV_Display_HD44780_Shift_Register.cpp.

5.21.1.4 `#define MHV_SHIFT_WRITECLOCK NULL,_clockOut,NULL,_clockPin,-1`

Definition at line 34 of file MHV_Display_HD44780_Shift_Register.cpp.

5.21.1.5 `#define MHV_SHIFT_WRITEDATA NULL,_dataOut,NULL,_dataPin,-1`

Definition at line 35 of file MHV_Display_HD44780_Shift_Register.cpp.

5.22 A:/eclipse/mhvlb/MHV_Display_HD44780_Shift_Register.h File Reference

```
#include <MHV_Display_HD44780.h>
```

Classes

- class [MHV_Display_HD44780_Shift_Register](#)

5.23 A:/eclipse/mhvlb/MHV_Display_Holtek_HT1632.cpp File Reference

```
#include <stdio.h> #include <MHV_Display_Holtek_HT1632.h> #include <MHV_Shifter.h>
```

Defines

- #define [MHV_SHIFT_WRITECLOCK](#) NULL, _port, NULL, _writePin, -1
- #define [MHV_SHIFT_WRITEDATA](#) NULL, _port, NULL, _dataPin, -1
- #define [MHV_SHIFT_ORDER_MSB](#)

5.23.1 Define Documentation

5.23.1.1 #define MHV_SHIFT_ORDER_MSB

Definition at line 37 of file MHV_Display_Holtek_HT1632.cpp.

5.23.1.2 #define MHV_SHIFT_WRITECLOCK NULL, _port, NULL, _writePin, -1

Todo: Use callbacks for writes to the display to allow the SHIFTER macros to be called directly on ports, instead of indirecting - this will save clocks

Definition at line 35 of file MHV_Display_Holtek_HT1632.cpp.

5.23.1.3 #define MHV_SHIFT_WRITEDATA NULL, _port, NULL, _dataPin, -1

Definition at line 36 of file MHV_Display_Holtek_HT1632.cpp.

5.24 A:/eclipse/mhvlb/MHV_Display_Holtek_HT1632.h File Reference

```
#include <MHV_Display_Monochrome.h>
```

Classes

- class [MHV_Display_Holtek_HT1632](#)

Defines

- `#define MHV_HT1632_BRIGHTNESS_MIN 0`
- `#define MHV_HT1632_BRIGHTNESS_MED 7`
- `#define MHV_HT1632_BRIGHTNESS_MAX 15`

Typedefs

- `typedef enum mhv_ht1632_command MHV_HT1632_COMMAND`
- `typedef mhv_ht1632_mode MHV_HT1632_MODE`

Enumerations

- `enum mhv_ht1632_command { MHV_HT1632_COMMAND_READ = 0b110, MHV_HT1632_COMMAND_WRITE = 0b101, MHV_HT1632_COMMAND_CMD = 0b100 }`
- `enum mhv_ht1632_mode { MHV_HT1632_NMOS_32x8 = 0b00, MHV_HT1632_NMOS_24x16 = 0b01, MHV_HT1632_PMOS_32x8 = 0b10, MHV_HT1632_PMOS_24x16 = 0b11 }`

5.24.1 Define Documentation

5.24.1.1 `#define MHV_HT1632_BRIGHTNESS_MAX 15`

Definition at line 33 of file MHV_Display_Holtek_HT1632.h.

5.24.1.2 `#define MHV_HT1632_BRIGHTNESS_MED 7`

Definition at line 32 of file MHV_Display_Holtek_HT1632.h.

5.24.1.3 `#define MHV_HT1632_BRIGHTNESS_MIN 0`

Definition at line 31 of file MHV_Display_Holtek_HT1632.h.

5.24.2 Typedef Documentation

5.24.2.1 `typedef enum mhv_ht1632_command MHV_HT1632_COMMAND`

Definition at line 40 of file MHV_Display_Holtek_HT1632.h.

5.24.2.2 `typedef mhv_ht1632_mode MHV_HT1632_MODE`

Definition at line 48 of file MHV_Display_Holtek_HT1632.h.

5.24.3 Enumeration Type Documentation

5.24.3.1 enum mhv_ht1632_command

Enumerator:

MHV_HT1632_COMMAND_READ
MHV_HT1632_COMMAND_WRITE
MHV_HT1632_COMMAND_CMD

Definition at line 35 of file MHV_Display_Holtek_HT1632.h.

5.24.3.2 enum mhv_ht1632_mode

Enumerator:

MHV_HT1632_NMOS_32x8
MHV_HT1632_NMOS_24x16
MHV_HT1632_PMOS_32x8
MHV_HT1632_PMOS_24x16

Definition at line 42 of file MHV_Display_Holtek_HT1632.h.

5.25 A:/eclipse/mhplib/MHV_Display_Monochrome.cpp File Reference

```
#include "MHV_Display_Monochrome.h"    #include <string.h>
#include <math.h>
```

5.26 A:/eclipse/mhplib/MHV_Display_Monochrome.h File Reference

```
#include <inttypes.h> #include <avr/pgmspace.h> #include
<MHV_Device_TX.h>    #include <MHV_io.h>    #include <MHV_-
Font.h>
```

Classes

- class [MHV_Display_Monochrome](#)

5.27 A:/eclipse/mhvlb/MHV_Display_Monochrome_Buffered.cpp - File Reference

```
#include "MHV_Display_Monochrome_Buffered.h"      #include
<string.h> #include <math.h>
```

Defines

- #define [pixel](#)(pixelRow, pixelCol) _frameBuffer[pixelRow * _colCount + pixelCol]

5.27.1 Define Documentation

5.27.1.1 #define [pixel](#)(*pixelRow*, *pixelCol*) _frameBuffer[pixelRow * _colCount + pixelCol]

Definition at line 31 of file MHV_Display_Monochrome_Buffered.cpp.

5.28 A:/eclipse/mhvlb/MHV_Display_Monochrome_Buffered.h File - Reference

```
#include <inttypes.h> #include <avr/pgmspace.h> #include
<MHV_Display_Monochrome.h>
```

Classes

- class [MHV_Display_Monochrome_Buffered](#)

5.29 A:/eclipse/mhvlb/MHV_EEPROM.cpp File Reference

```
#include <stddef.h> #include <MHV_EEPROM.h>
```

5.30 A:/eclipse/mhvlb/MHV_EEPROM.h File Reference

```
#include <MHV_io.h> #include <MHV_Lock.h> #include <inttypes.-
h>
```

Classes

- class [MHV_EEPROM](#)

Defines

- #define [MHV_EEPROM_ASSIGN_INTERRUPTS](#)(_mhvEeprom)
- #define [MHV_EEPROM_CREATE](#)(_mhvObjectName)
- #define [MHV_EEPROM_BUSY](#) -1

5.30.1 Define Documentation

5.30.1.1 #define MHV_EEPROM_ASSIGN_INTERRUPTS(*_mhvEeprom*)

Value:

```
ISR(MHV_EEPROM_VECT) { \
    _mhvEeprom.writeInterrupt(); \
}
```

Definition at line 35 of file MHV_EEPROM.h.

5.30.1.2 #define MHV_EEPROM_BUSY -1

Definition at line 44 of file MHV_EEPROM.h.

5.30.1.3 #define MHV_EEPROM_CREATE(*_mhvObjectName*)

Value:

```
MHV_EEPROM _mhvObjectName; \
    MHV_EEPROM_ASSIGN_INTERRUPTS(_mhvObjectName);
```

Definition at line 40 of file MHV_EEPROM.h.

5.31 A:/eclipse/mhvlb/MHV_EPP.cpp File Reference

5.32 A:/eclipse/mhvlb/MHV_EPP.h File Reference

5.33 A:/eclipse/mhvlb/MHV_Font.h File Reference

```
#include <inttypes.h> #include <avr/pgmspace.h>
```

Classes

- struct [mhv_font](#)

Typedefs

- typedef struct [mhv_font](#) MHV_FONT

5.33.1 Typedef Documentation

5.33.1.1 typedef struct mhv_font MHV_FONT

Definition at line 26 of file MHV_Font.h.

5.34 A:/eclipse/mhvlb/MHV_Font_SansSerif_10x8.h File Reference

```
#include <MHV_Font.h>    #include <inttypes.h>    #include  
<avr/pgmspace.h>
```

5.35 A:/eclipse/mhvlb/MHV_GammaCorrect.cpp File Reference

```
#include "MHV_GammaCorrect.h" #include <avr/pgmspace.h>
```

Functions

- uint8_t [mhv_calculatedGammaCorrect](#) (uint8_t value)
- uint8_t [mhv_precalculatedGammaCorrect](#) (uint8_t value)

Variables

- const uint8_t mhv_gammaValues[] [PROGMEM](#)

5.35.1 Function Documentation

5.35.1.1 uint8_t mhv_calculatedGammaCorrect (uint8_t value)

Definition at line 71 of file MHV_GammaCorrect.cpp.

5.35.1.2 uint8_t mhv_precalculatedGammaCorrect (uint8_t value)

Definition at line 80 of file MHV_GammaCorrect.cpp.

5.35.2 Variable Documentation

5.35.2.1 `const uint8_t mhv_gammaValues []` **PROGMEM**

Definition at line 31 of file MHV_GammaCorrect.cpp.

5.36 A:/eclipse/mhplib/MHV_GammaCorrect.h File Reference

```
#include <inttypes.h> #include <math.h> #include <avr/pgmspace.-  
h>
```

Defines

- `#define MHV_PRECALCULATED_GAMMA_CORRECT(gammaValue) pgm_read_byte(mhv_gammaValues + gammaValue)`

Functions

- `uint8_t mhv_calculatedGammaCorrect (uint8_t value)`
- `uint8_t mhv_precalculatedGammaCorrect (uint8_t value)`

5.36.1 Define Documentation

5.36.1.1 `#define MHV_PRECALCULATED_GAMMA_CORRECT(gammaValue) pgm_read_byte(mhv_gammaValues + gammaValue)`

Definition at line 37 of file MHV_GammaCorrect.h.

5.36.2 Function Documentation

5.36.2.1 `uint8_t mhv_calculatedGammaCorrect (uint8_t value)`

Definition at line 71 of file MHV_GammaCorrect.cpp.

5.36.2.2 `uint8_t mhv_precalculatedGammaCorrect (uint8_t value)`

Definition at line 80 of file MHV_GammaCorrect.cpp.

5.37 A:/eclipse/mhplib/MHV_HardwareSerial.cpp File Reference

```
#include <stdio.h> #include <avr/sfr_defs.h> #include  
<avr/pgmspace.h> #include <stdlib.h> #include <string.-
```

```
h> #include <inttypes.h> #include "MHV_HardwareSerial.h"
```

5.38 A:/eclipse/mhvlb/MHV_HardwareSerial.h File Reference

```
#include <inttypes.h> #include <avr/interrupt.h> #include
<MHV_io.h> #include <stdio.h> #include <MHV_RingBuffer.-
h> #include <avr/pgmspace.h> #include <MHV_Device_TX.h>
#include <MHV_Device_RX.h>
```

Classes

- class [MHV_HardwareSerial](#)

Defines

- #define [MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS](#)(mhvHardwareSerial, mhvHardwareSerialInterrupts) [_MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS](#)(mhvHardwareSerial, mhvHardwareSerialInterrupts)
- #define [_MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS](#)(mhvHardwareSerial, mhvRxVect, mhvTxVect)
- #define [MHV_HARDWARESERIAL_CREATE](#)(_mhvObjectName, _mhvRXBUFLLEN, _mhvTXBUFCOUNT, _mhvSERIAL, _mhvBAUD)
- #define [MHV_HARDWARESERIAL_DEBUG](#)(__dbg_serial, __dbg_format, __dbg_args...)

5.38.1 Define Documentation

5.38.1.1 #define [_MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS](#)(*mhvHardwareSerial*, *mhvRxVect*, *mhvTxVect*)

Value:

```
ISR(mhvRxVect) { \
    mhvHardwareSerial.rx(); \
} \
ISR(mhvTxVect) { \
    mhvHardwareSerial.tx(); \
}
```

Definition at line 44 of file MHV_HardwareSerial.h.

```
5.38.1.2 #define MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS(
        mhvHardwareSerial, mhvHardwareSerialInterrupts ) _MHV_H-
        ARDWARESERIAL_ASSIGN_INTERRUPTS(mhvHardwareSerial,
        mhvHardwareSerialInterrupts)
```

Definition at line 41 of file MHV_HardwareSerial.h.

```
5.38.1.3 #define MHV_HARDWARESERIAL_CREATE( _mhvObjectName, _mhvRXBUFLEN,
        _mhvTXBUFCOUNT, _mhvSERIAL, _mhvBAUD )
```

Value:

```
MHV_RX_BUFFER_CREATE(_mhvObjectName ## RX, _mhvRXBUFLEN); \
        MHV_TX_BUFFER_CREATE(_mhvObjectName ## TX, _mhvTXBUFCOUNT); \
        MHV_HardwareSerial _mhvObjectName(&_mhvObjectName ## RX, &
        _mhvObjectName ## RX, _mhvSERIAL, _mhvBAUD); \
        MHV_HARDWARESERIAL_ASSIGN_INTERRUPTS(_mhvObjectName, _mhvSERIAL
        ## _INTERRUPTS);
```

Create a new serial object

Parameters

<i>_mhvObject-Name</i>	the variable name of the object
<i>_mhvRXBU-FL</i> <i>EN</i>	the maximum length of the line to be received
<i>_mhvTXBU-FCOUNT</i>	the maximum number of tx buffers to send asynchronously
<i>_mhvSERI-AL</i>	serial port parameters
<i>_mhvBAUD</i>	the baud rate requested

Definition at line 60 of file MHV_HardwareSerial.h.

```
5.38.1.4 #define MHV_HARDWARESERIAL_DEBUG( __dbg_serial, __dbg_format, __dbg_args...
        )
```

Value:

```
do { \
        __dbg_serial.debug(__FILE__, __LINE__, __FUNCTION__, PSTR(__dbg_format)
        , ## __dbg_args); \
    } while (0)
```

Definition at line 66 of file MHV_HardwareSerial.h.

5.39 A:/eclipse/mhvlb/MHV_io.h File Reference

```
#include <avr/io.h> #include <inttypes.h> #include <stddef.-
h> #include <util/atomic.h>
```

Classes

- struct [mhv_pin](#)

Defines

- #define [MHV_IO_H](#)
- #define [NORETURN](#) __attribute__((noreturn))
- #define [PURE](#) __attribute__((pure))
- #define [GCC_VERSION](#) (__GNUC__ * 10000 + __GNUC_MINOR__ * 100 + __GNUC_PATCHLEVEL__)
- #define [mhv_make_pin](#)(_mhv_port, _mhv_bit) _mhv_make_pin(_mhv_port, _mhv_bit)
- #define [_mhv_make_pin](#)(_mhv_port, _mhv_bit) MHV_PIN_##_mhv_port##_mhv_bit
- #define [mhv_pin](#)(mhvParms) _mhv_pin(mhvParms)
- #define [_mhv_pin](#)(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) {mhvDir, mhvOutput, mhvInput, _BV(mhvBit), mhvPCInt}
- #define [mhv_out](#)(mhvParms) _mhv_out(mhvParms)
- #define [_mhv_out](#)(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvOutput
- #define [mhv_in](#)(mhvParms) _mhv_in(mhvParms)
- #define [_mhv_in](#)(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvInput
- #define [mhv_bit](#)(mhvParms) _mhv_bit(mhvParms)
- #define [_mhv_bit](#)(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvBit
- #define [mhv_dir](#)(mhvParms) _mhv_dir(mhvParms)
- #define [_mhv_dir](#)(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvDir
- #define [mhv_pcint](#)(mhvParms) _mhv_pcint(mhvParms)
- #define [_mhv_PCInt](#)(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvPCInt
- #define [mhv_declareExternalInterrupt](#)(mhvInterruptParms, mhvFunction) _mhv_declareExternalInterrupt(mhvInterruptParms, mhvFunction)
- #define [_mhv_declareExternalInterrupt](#)(mhvInterruptHandler, mhvModeRegister, mhvModeBitshift, mhvFunction) ISR(mhvInterruptHandler) mhvFunction
- #define [mhv_enableExternalInterrupt](#)(mhvInterruptParms, mhvInterruptMode)
- #define [_mhv_enableExternalInterrupt](#)(mhvInterruptHandler, mhvModeRegister, mhvModeBitshift, mhvInterruptMode) *mhvModeRegister = (*mhvModeRegister & ~(0x03 << mhvModeBitshift)) | (mhvInterruptMode << mhvModeBitshift)

Typedefs

- typedef struct [mhv_pin](#) [MHV_PIN](#)
- typedef enum [mhv_interruptMode](#) [MHV_INTERRUPTMODE](#)

Enumerations

- enum `mhv_interruptMode` { `MHV_INTERRUPT_LOW`, `MHV_INTERRUPT_CHANGE`, `MHV_INTERRUPT_FALLING`, `MHV_INTERRUPT_RISING` }

Functions

- void `mhv_pinOn` (`MHV_PIN` *pin)
- void `mhv_pinOnAtomic` (`MHV_PIN` *pin)
- void `mhv_pinOn` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_pinOnAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_pinOff` (`MHV_PIN` *pin)
- void `mhv_pinOffAtomic` (`MHV_PIN` *pin)
- void `mhv_pinOff` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_pinOffAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_pinSet` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt, bool state)
- void `mhv_pinSetAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt, bool state)
- void `mhv_setOutput` (`MHV_PIN` *pin)
- void `mhv_setOutputAtomic` (`MHV_PIN` *pin)
- void `mhv_setOutput` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_setOutputAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_setInput` (`MHV_PIN` *pin)
- void `mhv_setInputAtomic` (`MHV_PIN` *pin)
- void `mhv_setInput` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_setInputAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_setInputPullup` (`MHV_PIN` *pin)
- void `mhv_setInputPullupAtomic` (`MHV_PIN` *pin)
- void `mhv_setInputPullup` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_setInputPullupAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_pinToggle` (`MHV_PIN` *pin)
- void `mhv_pinToggleAtomic` (`MHV_PIN` *pin)
- void `mhv_pinToggle` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)
- void `mhv_pinToggleAtomic` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pclnt)

- bool `mhv_pinRead` (`MHV_PIN` *pin)
- bool `mhv_pinRead` (volatile uint8_t *dir, volatile uint8_t *out, volatile uint8_t *in, uint8_t bit, int8_t pcInt)
- void `mhv_memClear` (void *bufIn, uint8_t len, uint8_t count)
- void `mhv_memClear` (void *bufIn, uint8_t len)

5.39.1 Define Documentation

5.39.1.1 `#define _mhv_bit(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvBit`

Definition at line 527 of file `MHV_io.h`.

5.39.1.2 `#define _mhv_declareExternalInterrupt(mhvInterruptHandler, mhvModeRegister, mhvModeBitshift, mhvFunction) ISR(mhvInterruptHandler) mhvFunction`

Definition at line 559 of file `MHV_io.h`.

5.39.1.3 `#define _mhv_dir(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvDir`

Definition at line 537 of file `MHV_io.h`.

5.39.1.4 `#define _mhv_enableExternalInterrupt(mhvInterruptHandler, mhvModeRegister, mhvModeBitshift, mhvInterruptMode) *mhvModeRegister = (*mhvModeRegister & ~(0x03 << mhvModeBitshift)) | (mhvInterruptMode << mhvModeBitshift)`

Definition at line 584 of file `MHV_io.h`.

5.39.1.5 `#define _mhv_in(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvInput`

Definition at line 517 of file `MHV_io.h`.

5.39.1.6 `#define _mhv_make_pin(_mhv_port, _mhv_bit) MHV_PIN_ ## _mhv_port ## _mhv_bit`

Definition at line 94 of file `MHV_io.h`.

5.39.1.7 `#define _mhv_out(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvOutput`

Definition at line 507 of file `MHV_io.h`.

5.39.1.8 `#define _mhv_PCInt(mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt) mhvPCInt`

Definition at line 547 of file `MHV_io.h`.


```
5.39.1.9 #define _mhv_pin( mhvDir, mhvOutput, mhvInput, mhvBit, mhvPCInt ) {mhvDir,  
    mhvOutput, mhvInput, _BV(mhvBit), mhvPCInt}
```

Definition at line 104 of file MHV_io.h.

```
5.39.1.10 #define GCC_VERSION ( __GNUC__ * 10000 + __GNUC_MINOR__ * 100 +  
    __GNUC_PATCHLEVEL__ )
```

Definition at line 74 of file MHV_io.h.

```
5.39.1.11 #define mhv_bit( mhvParms ) _mhv_bit(mhvParms)
```

Grab the bit offset of a pin declaration

Parameters

<i>mhvParms</i>	A MHV_PIN_* Macro
-----------------	-------------------

Definition at line 524 of file MHV_io.h.

```
5.39.1.12 #define mhv_declareExternalInterrupt( mhvInterruptParms, mhvFunction  
    ) _mhv_declareExternalInterrupt(mhvInterruptParms, mhvFunction)
```

Assign a function to be triggered by an external interrupt

Parameters

<i>mhv- Interrupt- Parms</i>	A MHV_INTERRUPT_* Macro
<i>mhvFunction</i>	a block to execute when the interrupt occurs

Definition at line 556 of file MHV_io.h.

```
5.39.1.13 #define mhv_dir( mhvParms ) _mhv_dir(mhvParms)
```

Grab the direction register of a pin declaration

Parameters

<i>mhvParms</i>	A MHV_PIN_* Macro
-----------------	-------------------

Definition at line 534 of file MHV_io.h.

5.39.1.14 `#define mhv_enableExternalInterrupt(mhvInterruptParms, mhvInterruptMode)`

Value:

```
do { \
    _mhv_enableExternalInterrupt (mhvInterruptParms, mhvInterruptMode
    ); \
} while (0)
```

Enable an external interrupt

Parameters

<i><code>mhv-Interrupt-Parms</code></i>	A MHV_INTERRUPT_* Macro
<i><code>mhv-Interrupt-Mode</code></i>	When to raise the interrupt (see MHV_INTERRUPTMODE)

Definition at line 579 of file MHV_io.h.

5.39.1.15 `#define mhv_in(mhvParms) _mhv_in(mhvParms)`

Grab the input register of a pin declaration

Parameters

<i><code>mhvParms</code></i>	a MHV_PIN_* macro
------------------------------	-------------------

Definition at line 514 of file MHV_io.h.

5.39.1.16 `#define MHV_IO_H_`

Definition at line 44 of file MHV_io.h.

5.39.1.17 `#define mhv_make_pin(_mhv_port, _mhv_bit) _mhv_make_pin(_mhv_port, _mhv_bit)`

Convert a literal port and pin into a pin macro

Parameters

<i><code>_mhv_port</code></i>	the port (eg, B)
<i><code>_mhv_bit</code></i>	the bit (eg, 3)

Definition at line 91 of file MHV_io.h.

5.39.1.18 `#define mhv_out(mhvParms) _mhv_out(mhvParms)`

Grab the output register of a pin declaration

Parameters

<i>mhvParms</i>	a MHV_PIN_* macro
-----------------	-------------------

Definition at line 504 of file MHV_io.h.

5.39.1.19 `#define mhv_pcint(mhvParms) _mhv_pcint(mhvParms)`

Grab the pin change interrupt of a pin

Parameters

<i>mhvParms</i>	A MHV_PIN_* Macro
-----------------	-------------------

Definition at line 544 of file MHV_io.h.

5.39.1.20 `#define mhv_pin(mhvParms) _mhv_pin(mhvParms)`

Convert a pin declaration to a pin struct

Parameters

<i>mhvParms</i>	a MHV_PIN_* macro
-----------------	-------------------

Definition at line 101 of file MHV_io.h.

5.39.1.21 `#define NORETURN __attribute__((noreturn))`

Definition at line 54 of file MHV_io.h.

5.39.1.22 `#define PURE __attribute__((pure))`

Definition at line 57 of file MHV_io.h.

5.39.2 Typedef Documentation

5.39.2.1 `typedef enum mhv_interruptMode MHV_INTERRUPTMODE`

Definition at line 571 of file MHV_io.h.

5.39.2.2 typedef struct mhv_pin MHV_PIN

Definition at line 84 of file MHV_io.h.

5.39.3 Enumeration Type Documentation

5.39.3.1 enum mhv_interruptMode

Situations that interrupts can be triggered on

Enumerator:

MHV_INTERRUPT_LOW MHV_INTERRUPT_LOW to level trigger when low.

MHV_INTERRUPT_CHANGE MHV_INTERRUPT_CHANGE to edge trigger on change.

MHV_INTERRUPT_FALLING MHV_INTERRUPT_FALLING to edge trigger when falling.

MHV_INTERRUPT_RISING MHV_INTERRUPT_RISING to edge trigger when rising.

Definition at line 565 of file MHV_io.h.

5.39.4 Function Documentation

5.39.4.1 void mhv_memClear (void * *bufIn*, uint8_t *len*, uint8_t *count*) [inline]

Cheap memset to 0

Parameters

<i>bufIn</i>	a pointer to the buffer
<i>len</i>	the length of an element in the buffer
<i>count</i>	the number of elements in the buffer

Definition at line 466 of file MHV_io.h.

5.39.4.2 void mhv_memClear (void * *bufIn*, uint8_t *len*) [inline]

Cheap memset to 0

Parameters

<i>bufIn</i>	a pointer to the buffer
<i>len</i>	the length of an element in the buffer

Definition at line 481 of file MHV_io.h.

5.39.4.3 void mhv_pinOff (MHV_PIN * *pin*) [inline]

Set an output pin off

Parameters

<i>pin</i>	the pin to turn off
------------	---------------------

Definition at line 159 of file MHV_io.h.

5.39.4.4 void mhv_pinOff (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*,
uint8_t *bit*, int8_t *pcInt*) [inline]

Set an output pin off

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 182 of file MHV_io.h.

5.39.4.5 void mhv_pinOffAtomic (MHV_PIN * *pin*) [inline]

Set an output pin off (used if the state of a pin on the same port is altered in an interrupt handler)

Parameters

<i>pin</i>	the pin to turn off
------------	---------------------

Definition at line 167 of file MHV_io.h.

5.39.4.6 void mhv_pinOffAtomic (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t *
in, uint8_t *bit*, int8_t *pcInt*) [inline]

Set an output pin off (used if the state of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 195 of file MHV_io.h.

5.39.4.7 void mhv_pinOn (MHV_PIN * *pin*) [inline]

Set an output pin on

Parameters

<i>pin</i>	the pin to turn on
------------	--------------------

Definition at line 111 of file MHV_io.h.

5.39.4.8 void mhv_pinOn (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*,
uint8_t *bit*, int8_t *pclnt*) [inline]

Set an output pin on

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pclnt</i>	A member of the MHV_PIN_* macro

Definition at line 134 of file MHV_io.h.

5.39.4.9 void mhv_pinOnAtomic (MHV_PIN * *pin*) [inline]

Set an output pin on atomically (used if the state of a pin on the same port is altered in an interrupt handler)

Parameters

<i>pin</i>	the pin to turn on
------------	--------------------

Definition at line 119 of file MHV_io.h.

5.39.4.10 void mhv_pinOnAtomic (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t *
in, uint8_t *bit*, int8_t *pclnt*) [inline]

Set an output pin on (used if the state of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
------------	---------------------------------

<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 147 of file MHV_io.h.

5.39.4.11 `bool mhv_pinRead (MHV_PIN * pin) [inline]`

Read a pin

Parameters

<i>pin</i>	the pin to read
------------	-----------------

Definition at line 441 of file MHV_io.h.

5.39.4.12 `bool mhv_pinRead (volatile uint8_t * dir, volatile uint8_t * out, volatile uint8_t * in,
uint8_t bit, int8_t pcInt) [inline]`

Read a pin

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 454 of file MHV_io.h.

5.39.4.13 `void mhv_pinSet (volatile uint8_t * dir, volatile uint8_t * out, volatile uint8_t * in,
uint8_t bit, int8_t pcInt, bool state) [inline]`

Set an output pin on or off (state should really be constant for optimal performance)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro
<i>state</i>	true to turn the pin on

Definition at line 212 of file MHV_io.h.

5.39.4.14 `void mhv_pinSetAtomic (volatile uint8_t * dir, volatile uint8_t * out, volatile uint8_t * in, uint8_t bit, int8_t pclnt, bool state) [inline]`

Set an output pin on or off (state should really be constant for optimal performance)
(used if the state of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pclnt</i>	A member of the MHV_PIN_* macro
<i>state</i>	true to turn the pin on

Definition at line 231 of file MHV_io.h.

5.39.4.15 `void mhv_pinToggle (MHV_PIN * pin) [inline]`

Toggle a pin

Parameters

<i>pin</i>	the pin to toggle
------------	-------------------

Definition at line 393 of file MHV_io.h.

5.39.4.16 `void mhv_pinToggle (volatile uint8_t * dir, volatile uint8_t * out, volatile uint8_t * in, uint8_t bit, int8_t pclnt) [inline]`

Toggle a pin

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pclnt</i>	A member of the MHV_PIN_* macro

Definition at line 416 of file MHV_io.h.

5.39.4.17 `void mhv_pinToggleAtomic (MHV_PIN * pin) [inline]`

Toggle a pin (used if the state of a pin on the same port is altered in an interrupt handler)

Parameters

<i>pin</i>	the pin to toggle
------------	-------------------

Definition at line 401 of file MHV_io.h.

5.39.4.18 void mhv_pinToggleAtomic (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pcInt*) [inline]

Toggle a pin (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 429 of file MHV_io.h.

5.39.4.19 void mhv_setInput (MHV_PIN * *pin*) [inline]

Set a pin to be an input

Parameters

<i>pin</i>	the pin to become an output
------------	-----------------------------

Definition at line 292 of file MHV_io.h.

5.39.4.20 void mhv_setInput (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pcInt*) [inline]

Set a pin to be an input

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 316 of file MHV_io.h.

5.39.4.21 `void mhv_setInputAtomic (MHV_PIN * pin) [inline]`

Set a pin to be an input (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>pin</i>	the pin to become an output
------------	-----------------------------

Definition at line 301 of file MHV_io.h.

5.39.4.22 `void mhv_setInputAtomic (volatile uint8_t * dir, volatile uint8_t * out, volatile uint8_t * in, uint8_t bit, int8_t pclnt) [inline]`

Set a pin to be an input (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pclnt</i>	A member of the MHV_PIN_* macro

Definition at line 330 of file MHV_io.h.

5.39.4.23 `void mhv_setInputPullup (MHV_PIN * pin) [inline]`

Set a pin to be an input, with the internal pullup enabled

Parameters

<i>pin</i>	the pin to become an output
------------	-----------------------------

Definition at line 343 of file MHV_io.h.

5.39.4.24 `void mhv_setInputPullup (volatile uint8_t * dir, volatile uint8_t * out, volatile uint8_t * in, uint8_t bit, int8_t pclnt) [inline]`

Set a pin to be an input, with the internal pullup enabled

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pclnt</i>	A member of the MHV_PIN_* macro

Definition at line 368 of file MHV_io.h.

5.39.4.25 void mhv_setInputPullupAtomic (MHV_PIN * *pin*) [inline]

Set a pin to be an input, with the internal pullup enabled (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>pin</i>	the pin to become an output
------------	-----------------------------

Definition at line 352 of file MHV_io.h.

5.39.4.26 void mhv_setInputPullupAtomic (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pclnt*) [inline]

Set a pin to be an input, with the internal pullup enabled (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pclnt</i>	A member of the MHV_PIN_* macro

Definition at line 382 of file MHV_io.h.

5.39.4.27 void mhv_setOutput (MHV_PIN * *pin*) [inline]

Set a pin to be an output

Parameters

<i>pin</i>	the pin to become an output
------------	-----------------------------

Definition at line 245 of file MHV_io.h.

5.39.4.28 void mhv_setOutput (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pclnt*) [inline]

Set a pin to be an output

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
------------	---------------------------------

<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 268 of file MHV_io.h.

5.39.4.29 void mhv_setOutputAtomic (MHV_PIN * *pin*) [inline]

Set a pin to be an output (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>pin</i>	the pin to become an output
------------	-----------------------------

Definition at line 253 of file MHV_io.h.

5.39.4.30 void mhv_setOutputAtomic (volatile uint8_t * *dir*, volatile uint8_t * *out*, volatile uint8_t * *in*, uint8_t *bit*, int8_t *pcInt*) [inline]

Set a pin to be an output (used if the direction of a pin on the same port is altered in an interrupt handler)

Parameters

<i>dir</i>	A member of the MHV_PIN_* macro
<i>out</i>	A member of the MHV_PIN_* macro
<i>in</i>	A member of the MHV_PIN_* macro
<i>bit</i>	A member of the MHV_PIN_* macro
<i>pcInt</i>	A member of the MHV_PIN_* macro

Definition at line 281 of file MHV_io.h.

5.40 A:/eclipse/mhvlb/MHV_io_ArduinoDiecimilla.h File Reference

```
#include <avr/io.h>
```

Defines

- #define [MHV_ARDUINO_PIN_0](#) &DDRD, &PORTD, &PIND, 0, 16
- #define [MHV_ARDUINO_PIN_1](#) &DDRD, &PORTD, &PIND, 1, 17
- #define [MHV_ARDUINO_PIN_2](#) &DDRD, &PORTD, &PIND, 2, 18
- #define [MHV_ARDUINO_PIN_3](#) &DDRD, &PORTD, &PIND, 3, 19

- #define [MHV_ARDUINO_PIN_4](#) &DDRD, &PORTD, &PIND, 4, 20
- #define [MHV_ARDUINO_PIN_5](#) &DDRD, &PORTD, &PIND, 5, 21
- #define [MHV_ARDUINO_PIN_6](#) &DDRD, &PORTD, &PIND, 6, 22
- #define [MHV_ARDUINO_PIN_7](#) &DDRD, &PORTD, &PIND, 7, 23
- #define [MHV_ARDUINO_PIN_8](#) &DDRB, &PORTB, &PINB, 0, 0
- #define [MHV_ARDUINO_PIN_9](#) &DDRB, &PORTB, &PINB, 1, 1
- #define [MHV_ARDUINO_PIN_10](#) &DDRB, &PORTB, &PINB, 2, 2
- #define [MHV_ARDUINO_PIN_11](#) &DDRB, &PORTB, &PINB, 3, 3
- #define [MHV_ARDUINO_PIN_12](#) &DDRB, &PORTB, &PINB, 4, 4
- #define [MHV_ARDUINO_PIN_13](#) &DDRB, &PORTB, &PINB, 5, 5
- #define [MHV_ARDUINO_PIN_A0](#) &DDRC, &PORTC, &PINC, 0, 8
- #define [MHV_ARDUINO_PIN_A1](#) &DDRC, &PORTC, &PINC, 1, 9
- #define [MHV_ARDUINO_PIN_A2](#) &DDRC, &PORTC, &PINC, 2, 10
- #define [MHV_ARDUINO_PIN_A3](#) &DDRC, &PORTC, &PINC, 3, 11
- #define [MHV_ARDUINO_PIN_A4](#) &DDRC, &PORTC, &PINC, 4, 12
- #define [MHV_ARDUINO_PIN_A5](#) &DDRC, &PORTC, &PINC, 5, 13

5.40.1 Define Documentation

5.40.1.1 #define MHV_ARDUINO_PIN_0 &DDRD, &PORTD, &PIND, 0, 16

Definition at line 35 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.2 #define MHV_ARDUINO_PIN_1 &DDRD, &PORTD, &PIND, 1, 17

Definition at line 36 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.3 #define MHV_ARDUINO_PIN_10 &DDRB, &PORTB, &PINB, 2, 2

Definition at line 45 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.4 #define MHV_ARDUINO_PIN_11 &DDRB, &PORTB, &PINB, 3, 3

Definition at line 46 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.5 #define MHV_ARDUINO_PIN_12 &DDRB, &PORTB, &PINB, 4, 4

Definition at line 47 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.6 #define MHV_ARDUINO_PIN_13 &DDRB, &PORTB, &PINB, 5, 5

Definition at line 48 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.7 `#define MHV_ARDUINO_PIN_2 &DDRD, &PORTD, &PIND, 2, 18`

Definition at line 37 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.8 `#define MHV_ARDUINO_PIN_3 &DDRD, &PORTD, &PIND, 3, 19`

Definition at line 38 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.9 `#define MHV_ARDUINO_PIN_4 &DDRD, &PORTD, &PIND, 4, 20`

Definition at line 39 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.10 `#define MHV_ARDUINO_PIN_5 &DDRD, &PORTD, &PIND, 5, 21`

Definition at line 40 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.11 `#define MHV_ARDUINO_PIN_6 &DDRD, &PORTD, &PIND, 6, 22`

Definition at line 41 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.12 `#define MHV_ARDUINO_PIN_7 &DDRD, &PORTD, &PIND, 7, 23`

Definition at line 42 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.13 `#define MHV_ARDUINO_PIN_8 &DDRB, &PORTB, &PINB, 0, 0`

Definition at line 43 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.14 `#define MHV_ARDUINO_PIN_9 &DDRB, &PORTB, &PINB, 1, 1`

Definition at line 44 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.15 `#define MHV_ARDUINO_PIN_A0 &DDRC, &PORTC, &PINC, 0, 8`

Definition at line 49 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.16 `#define MHV_ARDUINO_PIN_A1 &DDRC, &PORTC, &PINC, 1, 9`

Definition at line 50 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.17 `#define MHV_ARDUINO_PIN_A2 &DDRC, &PORTC, &PINC, 2, 10`

Definition at line 51 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.18 `#define MHV_ARDUINO_PIN_A3 &DDRC, &PORTC, &PINC, 3, 11`

Definition at line 52 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.19 `#define MHV_ARDUINO_PIN_A4 &DDRC, &PORTC, &PINC, 4, 12`

Definition at line 53 of file MHV_io_ArduinoDiecimilla.h.

5.40.1.20 `#define MHV_ARDUINO_PIN_A5 &DDRC, &PORTC, &PINC, 5, 13`

Definition at line 54 of file MHV_io_ArduinoDiecimilla.h.

5.41 A:/eclipse/mhplib/MHV_io_ArduinoMega.h File Reference

```
#include <avr/io.h>
```

Defines

- `#define MHV_ARDUINO_PIN_0 &DDRE, &PORTE, &PINE, 0, 8`
- `#define MHV_ARDUINO_PIN_1 &DDRE, &PORTE, &PINE, 1, -1`
- `#define MHV_ARDUINO_PIN_2 &DDRE, &PORTE, &PINE, 4, -1`
- `#define MHV_ARDUINO_PIN_3 &DDRE, &PORTE, &PINE, 5, -1`
- `#define MHV_ARDUINO_PIN_4 &DDRG, &PORTG, &PING, 5, -1`
- `#define MHV_ARDUINO_PIN_5 &DDRE, &PORTE, &PINE, 3, -1`
- `#define MHV_ARDUINO_PIN_6 &DDRH, &PORTH, &PINH, 3, -1`
- `#define MHV_ARDUINO_PIN_7 &DDRH, &PORTH, &PINH, 4, -1`
- `#define MHV_ARDUINO_PIN_8 &DDRH, &PORTH, &PINH, 5, -1`
- `#define MHV_ARDUINO_PIN_9 &DDRH, &PORTH, &PINH, 6, -1`
- `#define MHV_ARDUINO_PIN_10 &DDRB, &PORTB, &PINB, 4, 4`
- `#define MHV_ARDUINO_PIN_11 &DDRB, &PORTB, &PINB, 5, 5`
- `#define MHV_ARDUINO_PIN_12 &DDRB, &PORTB, &PINB, 6, 6`
- `#define MHV_ARDUINO_PIN_13 &DDRB, &PORTB, &PINB, 7, 7`
- `#define MHV_ARDUINO_PIN_14 &DDRJ, &PORTJ, &PINJ, 1, 10`
- `#define MHV_ARDUINO_PIN_15 &DDRJ, &PORTJ, &PINJ, 0, 9`
- `#define MHV_ARDUINO_PIN_16 &DDRH, &PORTH, &PINH, 1, -1`
- `#define MHV_ARDUINO_PIN_17 &DDRH, &PORTH, &PINH, 0, -1`
- `#define MHV_ARDUINO_PIN_18 &DDRD, &PORTD, &PIND, 3, -1`
- `#define MHV_ARDUINO_PIN_19 &DDRD, &PORTD, &PIND, 2, -1`
- `#define MHV_ARDUINO_PIN_20 &DDRD, &PORTD, &PIND, 1, -1`

- #define MHV_ARDUINO_PIN_21 &DDRD, &PORTD, &PIND, 0, -1
- #define MHV_ARDUINO_PIN_22 &DDRA, &PORTA, &PINA, 0, -1
- #define MHV_ARDUINO_PIN_23 &DDRA, &PORTA, &PINA, 1, -1
- #define MHV_ARDUINO_PIN_24 &DDRA, &PORTA, &PINA, 2, -1
- #define MHV_ARDUINO_PIN_25 &DDRA, &PORTA, &PINA, 3, -1
- #define MHV_ARDUINO_PIN_26 &DDRA, &PORTA, &PINA, 4, -1
- #define MHV_ARDUINO_PIN_27 &DDRA, &PORTA, &PINA, 5, -1
- #define MHV_ARDUINO_PIN_28 &DDRA, &PORTA, &PINA, 6, -1
- #define MHV_ARDUINO_PIN_29 &DDRA, &PORTA, &PINA, 7, -1
- #define MHV_ARDUINO_PIN_30 &DDRC, &PORTC, &PINC, 7, -1
- #define MHV_ARDUINO_PIN_31 &DDRC, &PORTC, &PINC, 6, -1
- #define MHV_ARDUINO_PIN_32 &DDRC, &PORTC, &PINC, 5, -1
- #define MHV_ARDUINO_PIN_33 &DDRC, &PORTC, &PINC, 4, -1
- #define MHV_ARDUINO_PIN_34 &DDRC, &PORTC, &PINC, 3, -1
- #define MHV_ARDUINO_PIN_35 &DDRC, &PORTC, &PINC, 2, -1
- #define MHV_ARDUINO_PIN_36 &DDRC, &PORTC, &PINC, 1, -1
- #define MHV_ARDUINO_PIN_37 &DDRC, &PORTC, &PINC, 0, -1
- #define MHV_ARDUINO_PIN_38 &DDRD, &PORTD, &PIND, 7, -1
- #define MHV_ARDUINO_PIN_39 &DDRG, &PORTG, &PING, 2, -1
- #define MHV_ARDUINO_PIN_40 &DDRG, &PORTG, &PING, 1, -1
- #define MHV_ARDUINO_PIN_41 &DDRG, &PORTG, &PING, 0, -1
- #define MHV_ARDUINO_PIN_42 &DDRL, &PORTL, &PINL, 7, -1
- #define MHV_ARDUINO_PIN_43 &DDRL, &PORTL, &PINL, 6, -1
- #define MHV_ARDUINO_PIN_44 &DDRL, &PORTL, &PINL, 5, -1
- #define MHV_ARDUINO_PIN_45 &DDRL, &PORTL, &PINL, 4, -1
- #define MHV_ARDUINO_PIN_46 &DDRL, &PORTL, &PINL, 3, -1
- #define MHV_ARDUINO_PIN_47 &DDRL, &PORTL, &PINL, 2, -1
- #define MHV_ARDUINO_PIN_48 &DDRL, &PORTL, &PINL, 1, -1
- #define MHV_ARDUINO_PIN_49 &DDRL, &PORTL, &PINL, 0, -1
- #define MHV_ARDUINO_PIN_50 &DDRB, &PORTB, &PINB, 3, 3
- #define MHV_ARDUINO_PIN_51 &DDRB, &PORTB, &PINB, 2, 2
- #define MHV_ARDUINO_PIN_52 &DDRB, &PORTB, &PINB, 1, 1
- #define MHV_ARDUINO_PIN_53 &DDRB, &PORTB, &PINB, 0, 0
- #define MHV_ARDUINO_PIN_A0 &DDRF, &PORTF, &PINF, 0, -1
- #define MHV_ARDUINO_PIN_A1 &DDRF, &PORTF, &PINF, 1, -1
- #define MHV_ARDUINO_PIN_A2 &DDRF, &PORTF, &PINF, 2, -1
- #define MHV_ARDUINO_PIN_A3 &DDRF, &PORTF, &PINF, 3, -1
- #define MHV_ARDUINO_PIN_A4 &DDRF, &PORTF, &PINF, 4, -1
- #define MHV_ARDUINO_PIN_A5 &DDRF, &PORTF, &PINF, 5, -1
- #define MHV_ARDUINO_PIN_A6 &DDRF, &PORTF, &PINF, 6, -1
- #define MHV_ARDUINO_PIN_A7 &DDRF, &PORTF, &PINF, 7, -1
- #define MHV_ARDUINO_PIN_A8 &DDRK, &PORTK, &PINK, 0, 16
- #define MHV_ARDUINO_PIN_A9 &DDRK, &PORTK, &PINK, 1, 17
- #define MHV_ARDUINO_PIN_A10 &DDRK, &PORTK, &PINK, 2, 18
- #define MHV_ARDUINO_PIN_A11 &DDRK, &PORTK, &PINK, 3, 19
- #define MHV_ARDUINO_PIN_A12 &DDRK, &PORTK, &PINK, 4, 20
- #define MHV_ARDUINO_PIN_A13 &DDRK, &PORTK, &PINK, 5, 21
- #define MHV_ARDUINO_PIN_A14 &DDRK, &PORTK, &PINK, 6, 22
- #define MHV_ARDUINO_PIN_A15 &DDRK, &PORTK, &PINK, 7, 23

5.41.1 Define Documentation

5.41.1.1 `#define MHV_ARDUINO_PIN_0 &DDRE, &PORTE, &PINE, 0, 8`

Definition at line 35 of file MHV_io_ArduinoMega.h.

5.41.1.2 `#define MHV_ARDUINO_PIN_1 &DDRE, &PORTE, &PINE, 1, -1`

Definition at line 36 of file MHV_io_ArduinoMega.h.

5.41.1.3 `#define MHV_ARDUINO_PIN_10 &DDRB, &PORTB, &PINB, 4, 4`

Definition at line 45 of file MHV_io_ArduinoMega.h.

5.41.1.4 `#define MHV_ARDUINO_PIN_11 &DDRB, &PORTB, &PINB, 5, 5`

Definition at line 46 of file MHV_io_ArduinoMega.h.

5.41.1.5 `#define MHV_ARDUINO_PIN_12 &DDRB, &PORTB, &PINB, 6, 6`

Definition at line 47 of file MHV_io_ArduinoMega.h.

5.41.1.6 `#define MHV_ARDUINO_PIN_13 &DDRB, &PORTB, &PINB, 7, 7`

Definition at line 48 of file MHV_io_ArduinoMega.h.

5.41.1.7 `#define MHV_ARDUINO_PIN_14 &DDRJ, &PORTJ, &PINJ, 1, 10`

Definition at line 49 of file MHV_io_ArduinoMega.h.

5.41.1.8 `#define MHV_ARDUINO_PIN_15 &DDRJ, &PORTJ, &PINJ, 0, 9`

Definition at line 50 of file MHV_io_ArduinoMega.h.

5.41.1.9 `#define MHV_ARDUINO_PIN_16 &DDRH, &PORTH, &PINH, 1, -1`

Definition at line 51 of file MHV_io_ArduinoMega.h.

5.41.1.10 `#define MHV_ARDUINO_PIN_17 &DDRH, &PORTH, &PINH, 0, -1`

Definition at line 52 of file MHV_io_ArduinoMega.h.

5.41.1.11 `#define MHV_ARDUINO_PIN_18 &DDRD, &PORTD, &PIND, 3, -1`

Definition at line 53 of file MHV_io_ArduinoMega.h.

5.41.1.12 `#define MHV_ARDUINO_PIN_19 &DDRD, &PORTD, &PIND, 2, -1`

Definition at line 54 of file MHV_io_ArduinoMega.h.

5.41.1.13 `#define MHV_ARDUINO_PIN_2 &DDRE, &PORTE, &PINE, 4, -1`

Definition at line 37 of file MHV_io_ArduinoMega.h.

5.41.1.14 `#define MHV_ARDUINO_PIN_20 &DDRD, &PORTD, &PIND, 1, -1`

Definition at line 55 of file MHV_io_ArduinoMega.h.

5.41.1.15 `#define MHV_ARDUINO_PIN_21 &DDRD, &PORTD, &PIND, 0, -1`

Definition at line 56 of file MHV_io_ArduinoMega.h.

5.41.1.16 `#define MHV_ARDUINO_PIN_22 &DDRA, &PORTA, &PINA, 0, -1`

Definition at line 57 of file MHV_io_ArduinoMega.h.

5.41.1.17 `#define MHV_ARDUINO_PIN_23 &DDRA, &PORTA, &PINA, 1, -1`

Definition at line 58 of file MHV_io_ArduinoMega.h.

5.41.1.18 `#define MHV_ARDUINO_PIN_24 &DDRA, &PORTA, &PINA, 2, -1`

Definition at line 59 of file MHV_io_ArduinoMega.h.

5.41.1.19 `#define MHV_ARDUINO_PIN_25 &DDRA, &PORTA, &PINA, 3, -1`

Definition at line 60 of file MHV_io_ArduinoMega.h.

5.41.1.20 `#define MHV_ARDUINO_PIN_26 &DDRA, &PORTA, &PINA, 4, -1`

Definition at line 61 of file MHV_io_ArduinoMega.h.

5.41.1.21 `#define MHV_ARDUINO_PIN_27 &DDRA, &PORTA, &PINA, 5, -1`

Definition at line 62 of file MHV_io_ArduinoMega.h.

5.41.1.22 `#define MHV_ARDUINO_PIN_28 &DDRA, &PORTA, &PINA, 6, -1`

Definition at line 63 of file MHV_io_ArduinoMega.h.

5.41.1.23 `#define MHV_ARDUINO_PIN_29 &DDRA, &PORTA, &PINA, 7, -1`

Definition at line 64 of file MHV_io_ArduinoMega.h.

5.41.1.24 `#define MHV_ARDUINO_PIN_3 &DDRE, &PORTE, &PINE, 5, -1`

Definition at line 38 of file MHV_io_ArduinoMega.h.

5.41.1.25 `#define MHV_ARDUINO_PIN_30 &DDRC, &PORTC, &PINC, 7, -1`

Definition at line 65 of file MHV_io_ArduinoMega.h.

5.41.1.26 `#define MHV_ARDUINO_PIN_31 &DDRC, &PORTC, &PINC, 6, -1`

Definition at line 66 of file MHV_io_ArduinoMega.h.

5.41.1.27 `#define MHV_ARDUINO_PIN_32 &DDRC, &PORTC, &PINC, 5, -1`

Definition at line 67 of file MHV_io_ArduinoMega.h.

5.41.1.28 `#define MHV_ARDUINO_PIN_33 &DDRC, &PORTC, &PINC, 4, -1`

Definition at line 68 of file MHV_io_ArduinoMega.h.

5.41.1.29 `#define MHV_ARDUINO_PIN_34 &DDRC, &PORTC, &PINC, 3, -1`

Definition at line 69 of file MHV_io_ArduinoMega.h.

5.41.1.30 `#define MHV_ARDUINO_PIN_35 &DDRC, &PORTC, &PINC, 2, -1`

Definition at line 70 of file MHV_io_ArduinoMega.h.

5.41.1.31 `#define MHV_ARDUINO_PIN_36 &DDRC, &PORTC, &PINC, 1, -1`

Definition at line 71 of file MHV_io_ArduinoMega.h.

5.41.1.32 `#define MHV_ARDUINO_PIN_37 &DDRC, &PORTC, &PINC, 0, -1`

Definition at line 72 of file MHV_io_ArduinoMega.h.

5.41.1.33 `#define MHV_ARDUINO_PIN_38 &DDRD, &PORTD, &PIND, 7, -1`

Definition at line 73 of file MHV_io_ArduinoMega.h.

5.41.1.34 `#define MHV_ARDUINO_PIN_39 &DDRG, &PORTG, &PING, 2, -1`

Definition at line 74 of file MHV_io_ArduinoMega.h.

5.41.1.35 `#define MHV_ARDUINO_PIN_4 &DDRG, &PORTG, &PING, 5, -1`

Definition at line 39 of file MHV_io_ArduinoMega.h.

5.41.1.36 `#define MHV_ARDUINO_PIN_40 &DDRG, &PORTG, &PING, 1, -1`

Definition at line 75 of file MHV_io_ArduinoMega.h.

5.41.1.37 `#define MHV_ARDUINO_PIN_41 &DDRG, &PORTG, &PING, 0, -1`

Definition at line 76 of file MHV_io_ArduinoMega.h.

5.41.1.38 `#define MHV_ARDUINO_PIN_42 &DDRL, &PORTL, &PINL, 7, -1`

Definition at line 77 of file MHV_io_ArduinoMega.h.

5.41.1.39 `#define MHV_ARDUINO_PIN_43 &DDRL, &PORTL, &PINL, 6, -1`

Definition at line 78 of file MHV_io_ArduinoMega.h.

5.41.1.40 `#define MHV_ARDUINO_PIN_44 &DDRL, &PORTL, &PINL, 5, -1`

Definition at line 79 of file MHV_io_ArduinoMega.h.

5.41.1.41 #define MHV_ARDUINO_PIN_45 &DDRL, &PORTL, &PINL, 4, -1

Definition at line 80 of file MHV_io_ArduinoMega.h.

5.41.1.42 #define MHV_ARDUINO_PIN_46 &DDRL, &PORTL, &PINL, 3, -1

Definition at line 81 of file MHV_io_ArduinoMega.h.

5.41.1.43 #define MHV_ARDUINO_PIN_47 &DDRL, &PORTL, &PINL, 2, -1

Definition at line 82 of file MHV_io_ArduinoMega.h.

5.41.1.44 #define MHV_ARDUINO_PIN_48 &DDRL, &PORTL, &PINL, 1, -1

Definition at line 83 of file MHV_io_ArduinoMega.h.

5.41.1.45 #define MHV_ARDUINO_PIN_49 &DDRL, &PORTL, &PINL, 0, -1

Definition at line 84 of file MHV_io_ArduinoMega.h.

5.41.1.46 #define MHV_ARDUINO_PIN_5 &DDRE, &PORTE, &PINE, 3, -1

Definition at line 40 of file MHV_io_ArduinoMega.h.

5.41.1.47 #define MHV_ARDUINO_PIN_50 &DDRB, &PORTB, &PINB, 3, 3

Definition at line 85 of file MHV_io_ArduinoMega.h.

5.41.1.48 #define MHV_ARDUINO_PIN_51 &DDRB, &PORTB, &PINB, 2, 2

Definition at line 86 of file MHV_io_ArduinoMega.h.

5.41.1.49 #define MHV_ARDUINO_PIN_52 &DDRB, &PORTB, &PINB, 1, 1

Definition at line 87 of file MHV_io_ArduinoMega.h.

5.41.1.50 #define MHV_ARDUINO_PIN_53 &DDRB, &PORTB, &PINB, 0, 0

Definition at line 88 of file MHV_io_ArduinoMega.h.

5.41.1.51 `#define MHV_ARDUINO_PIN_6 &DDRH, &PORTH, &PINH, 3, -1`

Definition at line 41 of file MHV_io_ArduinoMega.h.

5.41.1.52 `#define MHV_ARDUINO_PIN_7 &DDRH, &PORTH, &PINH, 4, -1`

Definition at line 42 of file MHV_io_ArduinoMega.h.

5.41.1.53 `#define MHV_ARDUINO_PIN_8 &DDRH, &PORTH, &PINH, 5, -1`

Definition at line 43 of file MHV_io_ArduinoMega.h.

5.41.1.54 `#define MHV_ARDUINO_PIN_9 &DDRH, &PORTH, &PINH, 6, -1`

Definition at line 44 of file MHV_io_ArduinoMega.h.

5.41.1.55 `#define MHV_ARDUINO_PIN_A0 &DDRF, &PORTF, &PINF, 0, -1`

Definition at line 89 of file MHV_io_ArduinoMega.h.

5.41.1.56 `#define MHV_ARDUINO_PIN_A1 &DDRF, &PORTF, &PINF, 1, -1`

Definition at line 90 of file MHV_io_ArduinoMega.h.

5.41.1.57 `#define MHV_ARDUINO_PIN_A10 &DDRK, &PORTK, &PINK, 2, 18`

Definition at line 99 of file MHV_io_ArduinoMega.h.

5.41.1.58 `#define MHV_ARDUINO_PIN_A11 &DDRK, &PORTK, &PINK, 3, 19`

Definition at line 100 of file MHV_io_ArduinoMega.h.

5.41.1.59 `#define MHV_ARDUINO_PIN_A12 &DDRK, &PORTK, &PINK, 4, 20`

Definition at line 101 of file MHV_io_ArduinoMega.h.

5.41.1.60 `#define MHV_ARDUINO_PIN_A13 &DDRK, &PORTK, &PINK, 5, 21`

Definition at line 102 of file MHV_io_ArduinoMega.h.

5.41.1.61 `#define MHV_ARDUINO_PIN_A14 &DDRK, &PORTK, &PINK, 6, 22`

Definition at line 103 of file MHV_io_ArduinoMega.h.

5.41.1.62 `#define MHV_ARDUINO_PIN_A15 &DDRK, &PORTK, &PINK, 7, 23`

Definition at line 104 of file MHV_io_ArduinoMega.h.

5.41.1.63 `#define MHV_ARDUINO_PIN_A2 &DDRF, &PORTF, &PINF, 2, -1`

Definition at line 91 of file MHV_io_ArduinoMega.h.

5.41.1.64 `#define MHV_ARDUINO_PIN_A3 &DDRF, &PORTF, &PINF, 3, -1`

Definition at line 92 of file MHV_io_ArduinoMega.h.

5.41.1.65 `#define MHV_ARDUINO_PIN_A4 &DDRF, &PORTF, &PINF, 4, -1`

Definition at line 93 of file MHV_io_ArduinoMega.h.

5.41.1.66 `#define MHV_ARDUINO_PIN_A5 &DDRF, &PORTF, &PINF, 5, -1`

Definition at line 94 of file MHV_io_ArduinoMega.h.

5.41.1.67 `#define MHV_ARDUINO_PIN_A6 &DDRF, &PORTF, &PINF, 6, -1`

Definition at line 95 of file MHV_io_ArduinoMega.h.

5.41.1.68 `#define MHV_ARDUINO_PIN_A7 &DDRF, &PORTF, &PINF, 7, -1`

Definition at line 96 of file MHV_io_ArduinoMega.h.

5.41.1.69 `#define MHV_ARDUINO_PIN_A8 &DDRK, &PORTK, &PINK, 0, 16`

Definition at line 97 of file MHV_io_ArduinoMega.h.

5.41.1.70 `#define MHV_ARDUINO_PIN_A9 &DDRK, &PORTK, &PINK, 1, 17`

Definition at line 98 of file MHV_io_ArduinoMega.h.

5.42 A:/eclipse/mhvlb/MHV_io_ATmega1280.h File Reference

```
#include <avr/io.h>
```

Defines

- #define [MHV_TIMER8_0](#) MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B, &OCR0A, &OCR0B, &TCNT0, &TIMSK0, OCIE0A
- #define [MHV_TIMER8_2](#) MHV_TIMER_TYPE_7_PRESCALERS, &TCCR2A, &TCCR2B, &OCR2A, &OCR2B, &TCNT2, &TIMSK2, OCIE2A
- #define [MHV_TIMER0_INTERRUPTS](#) TIMER0_COMPA_vect, TIMER0_COMPB_vect, 0
- #define [MHV_TIMER2_INTERRUPTS](#) TIMER2_COMPA_vect, TIMER2_COMPB_vect, 0
- #define [MHV_TIMER16_1](#) &TCCR1A, &TCCR1B, &TCCR1C, &OCR1A, &OCR1B, &OCR1C, &TCNT1, &TIMSK1, &ICR1
- #define [MHV_TIMER16_3](#) &TCCR3A, &TCCR3B, &TCCR3C, &OCR3A, &OCR3B, &OCR3C, &TCNT3, &TIMSK3, &ICR3
- #define [MHV_TIMER16_4](#) &TCCR4A, &TCCR4B, &TCCR4C, &OCR4A, &OCR4B, &OCR4C, &TCNT4, &TIMSK4, &ICR4
- #define [MHV_TIMER16_5](#) &TCCR5A, &TCCR5B, &TCCR5C, &OCR5A, &OCR5B, &OCR5C, &TCNT5, &TIMSK5, &ICR5
- #define [MHV_TIMER1_INTERRUPTS](#) TIMER1_COMPA_vect, TIMER1_COMPB_vect, TIMER1_COMPC_vect
- #define [MHV_TIMER3_INTERRUPTS](#) TIMER3_COMPA_vect, TIMER3_COMPB_vect, TIMER3_COMPC_vect
- #define [MHV_TIMER4_INTERRUPTS](#) TIMER4_COMPA_vect, TIMER4_COMPB_vect, TIMER4_COMPC_vect
- #define [MHV_TIMER5_INTERRUPTS](#) TIMER5_COMPA_vect, TIMER5_COMPB_vect, TIMER5_COMPC_vect
- #define [MHV_USART0](#) &UBRR0, &UCSR0A, &UCSR0B, &UDR0, RXEN0, TXEN0, RXCIE0, TXCIE0, UDRE0, U2X0
- #define [MHV_USART1](#) &UBRR1, &UCSR1A, &UCSR1B, &UDR1, RXEN1, TXEN1, RXCIE1, TXCIE1, UDRE1, U2X1
- #define [MHV_USART2](#) &UBRR2, &UCSR2A, &UCSR2B, &UDR2, RXEN2, TXEN2, RXCIE2, TXCIE2, UDRE2, U2X2
- #define [MHV_USART3](#) &UBRR3, &UCSR3A, &UCSR3B, &UDR3, RXEN3, TXEN3, RXCIE3, TXCIE3, UDRE3, U2X3
- #define [MHV_USART0_INTERRUPTS](#) USART0_RX_vect, USART0_TX_vect
- #define [MHV_USART1_INTERRUPTS](#) USART1_RX_vect, USART1_TX_vect
- #define [MHV_USART2_INTERRUPTS](#) USART2_RX_vect, USART2_TX_vect
- #define [MHV_USART3_INTERRUPTS](#) USART3_RX_vect, USART3_TX_vect
- #define [MHV_AD_RESOLUTION](#) 1024
- #define [MHV_AD_REFERENCE_AREF](#) (uint8_t)(0x00 << 6)
- #define [MHV_AD_REFERENCE_AVCC](#) (uint8_t)(0x01 << 6)
- #define [MHV_AD_REFERENCE_1V1](#) (uint8_t)(0x02 << 6)
- #define [MHV_AD_REFERENCE_2V56](#) (uint8_t)(0x03 << 6)

- #define [MHV_AD_CHANNEL_0](#) 0x00
- #define [MHV_AD_CHANNEL_1](#) 0x01
- #define [MHV_AD_CHANNEL_2](#) 0x02
- #define [MHV_AD_CHANNEL_3](#) 0x03
- #define [MHV_AD_CHANNEL_4](#) 0x04
- #define [MHV_AD_CHANNEL_5](#) 0x05
- #define [MHV_AD_CHANNEL_6](#) 0x06
- #define [MHV_AD_CHANNEL_7](#) 0x07
- #define [MHV_AD_CHANNEL_0_X10_0](#) 0x08
- #define [MHV_AD_CHANNEL_1_X10_0](#) 0x09
- #define [MHV_AD_CHANNEL_0_X200_0](#) 0x0a
- #define [MHV_AD_CHANNEL_1_X200_0](#) 0x0b
- #define [MHV_AD_CHANNEL_2_X10_2](#) 0x0c
- #define [MHV_AD_CHANNEL_3_X10_2](#) 0x0d
- #define [MHV_AD_CHANNEL_2_X200_2](#) 0x0e
- #define [MHV_AD_CHANNEL_3_X200_2](#) 0x0f
- #define [MHV_AD_CHANNEL_0_X1_1](#) 0x10
- #define [MHV_AD_CHANNEL_1_X1_1](#) 0x11
- #define [MHV_AD_CHANNEL_2_X1_1](#) 0x12
- #define [MHV_AD_CHANNEL_3_X1_1](#) 0x13
- #define [MHV_AD_CHANNEL_4_X1_1](#) 0x14
- #define [MHV_AD_CHANNEL_5_X1_1](#) 0x15
- #define [MHV_AD_CHANNEL_6_X1_1](#) 0x16
- #define [MHV_AD_CHANNEL_7_X1_1](#) 0x17
- #define [MHV_AD_CHANNEL_0_X1_2](#) 0x18
- #define [MHV_AD_CHANNEL_1_X1_2](#) 0x19
- #define [MHV_AD_CHANNEL_2_X1_2](#) 0x1a
- #define [MHV_AD_CHANNEL_3_X1_2](#) 0x1b
- #define [MHV_AD_CHANNEL_4_X1_2](#) 0x1c
- #define [MHV_AD_CHANNEL_5_X1_2](#) 0x1d
- #define [MHV_AD_CHANNEL_1V1](#) 0x1e
- #define [MHV_AD_CHANNEL_0V](#) 0x1f
- #define [MHV_AD_CHANNEL_8](#) 0x20
- #define [MHV_AD_CHANNEL_9](#) 0x21
- #define [MHV_AD_CHANNEL_10](#) 0x22
- #define [MHV_AD_CHANNEL_11](#) 0x23
- #define [MHV_AD_CHANNEL_12](#) 0x24
- #define [MHV_AD_CHANNEL_13](#) 0x25
- #define [MHV_AD_CHANNEL_14](#) 0x26
- #define [MHV_AD_CHANNEL_15](#) 0x27
- #define [MHV_AD_CHANNEL_8_X10_8](#) 0x28
- #define [MHV_AD_CHANNEL_9_X10_8](#) 0x29
- #define [MHV_AD_CHANNEL_8_X200_8](#) 0x2a
- #define [MHV_AD_CHANNEL_9_X200_8](#) 0x2b
- #define [MHV_AD_CHANNEL_10_X10_10](#) 0x2c
- #define [MHV_AD_CHANNEL_11_X10_10](#) 0x2d

- #define [MHV_AD_CHANNEL_10_X200_10](#) 0x2e
- #define [MHV_AD_CHANNEL_11_X200_10](#) 0x2f
- #define [MHV_AD_CHANNEL_8_X1_9](#) 0x30
- #define [MHV_AD_CHANNEL_9_X1_9](#) 0x31
- #define [MHV_AD_CHANNEL_10_X1_9](#) 0x32
- #define [MHV_AD_CHANNEL_11_X1_9](#) 0x33
- #define [MHV_AD_CHANNEL_12_X1_9](#) 0x34
- #define [MHV_AD_CHANNEL_13_X1_9](#) 0x35
- #define [MHV_AD_CHANNEL_14_X1_9](#) 0x36
- #define [MHV_AD_CHANNEL_15_X1_9](#) 0x37
- #define [MHV_AD_CHANNEL_8_X1_10](#) 0x38
- #define [MHV_AD_CHANNEL_9_X1_10](#) 0x39
- #define [MHV_AD_CHANNEL_10_X1_10](#) 0x3a
- #define [MHV_AD_CHANNEL_11_X1_10](#) 0x3b
- #define [MHV_AD_CHANNEL_12_X1_10](#) 0x3c
- #define [MHV_AD_CHANNEL_13_X1_10](#) 0x3d
- #define [MHV_AD_PRR](#) PRR0
- #define [MHV_PIN_A0](#) &DDRA, &PORTA, &PINA, 0, -1
- #define [MHV_PIN_A1](#) &DDRA, &PORTA, &PINA, 1, -1
- #define [MHV_PIN_A2](#) &DDRA, &PORTA, &PINA, 2, -1
- #define [MHV_PIN_A3](#) &DDRA, &PORTA, &PINA, 3, -1
- #define [MHV_PIN_A4](#) &DDRA, &PORTA, &PINA, 4, -1
- #define [MHV_PIN_A5](#) &DDRA, &PORTA, &PINA, 5, -1
- #define [MHV_PIN_A6](#) &DDRA, &PORTA, &PINA, 6, -1
- #define [MHV_PIN_A7](#) &DDRA, &PORTA, &PINA, 7, -1
- #define [MHV_PIN_B0](#) &DDRB, &PORTB, &PINB, 0, 0
- #define [MHV_PIN_B1](#) &DDRB, &PORTB, &PINB, 1, 1
- #define [MHV_PIN_B2](#) &DDRB, &PORTB, &PINB, 2, 2
- #define [MHV_PIN_B3](#) &DDRB, &PORTB, &PINB, 3, 3
- #define [MHV_PIN_B4](#) &DDRB, &PORTB, &PINB, 4, 4
- #define [MHV_PIN_B5](#) &DDRB, &PORTB, &PINB, 5, 5
- #define [MHV_PIN_B6](#) &DDRB, &PORTB, &PINB, 6, 6
- #define [MHV_PIN_B7](#) &DDRB, &PORTB, &PINB, 7, 7
- #define [MHV_PIN_C0](#) &DDRC, &PORTC, &PINC, 0, -1
- #define [MHV_PIN_C1](#) &DDRC, &PORTC, &PINC, 1, -1
- #define [MHV_PIN_C2](#) &DDRC, &PORTC, &PINC, 2, -1
- #define [MHV_PIN_C3](#) &DDRC, &PORTC, &PINC, 3, -1
- #define [MHV_PIN_C4](#) &DDRC, &PORTC, &PINC, 4, -1
- #define [MHV_PIN_C5](#) &DDRC, &PORTC, &PINC, 5, -1
- #define [MHV_PIN_C6](#) &DDRC, &PORTC, &PINC, 6, -1
- #define [MHV_PIN_C7](#) &DDRC, &PORTC, &PINC, 7, -1
- #define [MHV_PIN_D0](#) &DDRD, &PORTD, &PIND, 0, -1
- #define [MHV_PIN_D1](#) &DDRD, &PORTD, &PIND, 1, -1
- #define [MHV_PIN_D2](#) &DDRD, &PORTD, &PIND, 2, -1
- #define [MHV_PIN_D3](#) &DDRD, &PORTD, &PIND, 3, -1
- #define [MHV_PIN_D4](#) &DDRD, &PORTD, &PIND, 4, -1

- #define [MHV_PIN_D5](#) &DDRD, &PORTD, &PIND, 5, -1
- #define [MHV_PIN_D6](#) &DDRD, &PORTD, &PIND, 6, -1
- #define [MHV_PIN_D7](#) &DDRD, &PORTD, &PIND, 7, -1
- #define [MHV_PIN_E0](#) &DDRE, &PORTE, &PINE, 0, 8
- #define [MHV_PIN_E1](#) &DDRE, &PORTE, &PINE, 1, -1
- #define [MHV_PIN_E2](#) &DDRE, &PORTE, &PINE, 2, -1
- #define [MHV_PIN_E3](#) &DDRE, &PORTE, &PINE, 3, -1
- #define [MHV_PIN_E4](#) &DDRE, &PORTE, &PINE, 4, -1
- #define [MHV_PIN_E5](#) &DDRE, &PORTE, &PINE, 5, -1
- #define [MHV_PIN_E6](#) &DDRE, &PORTE, &PINE, 6, -1
- #define [MHV_PIN_E7](#) &DDRE, &PORTE, &PINE, 7, -1
- #define [MHV_PIN_F0](#) &DDRF, &PORTF, &PINF, 0, -1
- #define [MHV_PIN_F1](#) &DDRF, &PORTF, &PINF, 1, -1
- #define [MHV_PIN_F2](#) &DDRF, &PORTF, &PINF, 2, -1
- #define [MHV_PIN_F3](#) &DDRF, &PORTF, &PINF, 3, -1
- #define [MHV_PIN_F4](#) &DDRF, &PORTF, &PINF, 4, -1
- #define [MHV_PIN_F5](#) &DDRF, &PORTF, &PINF, 5, -1
- #define [MHV_PIN_F6](#) &DDRF, &PORTF, &PINF, 6, -1
- #define [MHV_PIN_F7](#) &DDRF, &PORTF, &PINF, 7, -1
- #define [MHV_PIN_G0](#) &DDRG, &PORTG, &PING, 0, -1
- #define [MHV_PIN_G1](#) &DDRG, &PORTG, &PING, 1, -1
- #define [MHV_PIN_G2](#) &DDRG, &PORTG, &PING, 2, -1
- #define [MHV_PIN_G3](#) &DDRG, &PORTG, &PING, 3, -1
- #define [MHV_PIN_G4](#) &DDRG, &PORTG, &PING, 4, -1
- #define [MHV_PIN_G5](#) &DDRG, &PORTG, &PING, 5, -1
- #define [MHV_PIN_H0](#) &DDRH, &PORTH, &PINH, 0, -1
- #define [MHV_PIN_H1](#) &DDRH, &PORTH, &PINH, 1, -1
- #define [MHV_PIN_H2](#) &DDRH, &PORTH, &PINH, 2, -1
- #define [MHV_PIN_H3](#) &DDRH, &PORTH, &PINH, 3, -1
- #define [MHV_PIN_H4](#) &DDRH, &PORTH, &PINH, 4, -1
- #define [MHV_PIN_H5](#) &DDRH, &PORTH, &PINH, 5, -1
- #define [MHV_PIN_H6](#) &DDRH, &PORTH, &PINH, 6, -1
- #define [MHV_PIN_H7](#) &DDRH, &PORTH, &PINH, 7, -1
- #define [MHV_PIN_J0](#) &DDRJ, &PORTJ, &PINJ, 0, 9
- #define [MHV_PIN_J1](#) &DDRJ, &PORTJ, &PINJ, 1, 10
- #define [MHV_PIN_J2](#) &DDRJ, &PORTJ, &PINJ, 2, 11
- #define [MHV_PIN_J3](#) &DDRJ, &PORTJ, &PINJ, 3, 12
- #define [MHV_PIN_J4](#) &DDRJ, &PORTJ, &PINJ, 4, 13
- #define [MHV_PIN_J5](#) &DDRJ, &PORTJ, &PINJ, 5, 14
- #define [MHV_PIN_J6](#) &DDRJ, &PORTJ, &PINJ, 6, 15
- #define [MHV_PIN_J7](#) &DDRJ, &PORTJ, &PINJ, 7, -1
- #define [MHV_PIN_K0](#) &DDRK, &PORTK, &PINK, 0, 16
- #define [MHV_PIN_K1](#) &DDRK, &PORTK, &PINK, 1, 17
- #define [MHV_PIN_K2](#) &DDRK, &PORTK, &PINK, 2, 18
- #define [MHV_PIN_K3](#) &DDRK, &PORTK, &PINK, 3, 19
- #define [MHV_PIN_K4](#) &DDRK, &PORTK, &PINK, 4, 20

- #define [MHV_PIN_K5](#) &DDRK, &PORTK, &PINK, 5, 21
- #define [MHV_PIN_K6](#) &DDRK, &PORTK, &PINK, 6, 22
- #define [MHV_PIN_K7](#) &DDRK, &PORTK, &PINK, 7, 23
- #define [MHV_PIN_L0](#) &DDRL, &PORTL, &PINL, 0, -1
- #define [MHV_PIN_L1](#) &DDRL, &PORTL, &PINL, 1, -1
- #define [MHV_PIN_L2](#) &DDRL, &PORTL, &PINL, 2, -1
- #define [MHV_PIN_L3](#) &DDRL, &PORTL, &PINL, 3, -1
- #define [MHV_PIN_L4](#) &DDRL, &PORTL, &PINL, 4, -1
- #define [MHV_PIN_L5](#) &DDRL, &PORTL, &PINL, 5, -1
- #define [MHV_PIN_L6](#) &DDRL, &PORTL, &PINL, 6, -1
- #define [MHV_PIN_L7](#) &DDRL, &PORTL, &PINL, 7, -1
- #define [MHV_PIN_TIMER_0_A](#) MHV_PIN_B7
- #define [MHV_PIN_TIMER_0_B](#) MHV_PIN_G5
- #define [MHV_PIN_TIMER_1_A](#) MHV_PIN_B5
- #define [MHV_PIN_TIMER_1_B](#) MHV_PIN_B6
- #define [MHV_PIN_TIMER_1_C](#) MHV_PIN_B7
- #define [MHV_PIN_TIMER_2_A](#) MHV_PIN_B4
- #define [MHV_PIN_TIMER_2_B](#) MHV_PIN_H6
- #define [MHV_PIN_TIMER_3_A](#) MHV_PIN_E3
- #define [MHV_PIN_TIMER_3_B](#) MHV_PIN_E4
- #define [MHV_PIN_TIMER_3_C](#) MHV_PIN_E5
- #define [MHV_PIN_TIMER_4_A](#) MHV_PIN_H3
- #define [MHV_PIN_TIMER_4_B](#) MHV_PIN_H4
- #define [MHV_PIN_TIMER_4_C](#) MHV_PIN_H5
- #define [MHV_PIN_TIMER_5_A](#) MHV_PIN_L3
- #define [MHV_PIN_TIMER_5_B](#) MHV_PIN_L4
- #define [MHV_PIN_TIMER_5_C](#) MHV_PIN_L6
- #define [MHV_PC_INT_COUNT](#) 24
- #define [MHV_EEPROM_VECT](#) EE_READY_vect

5.42.1 Define Documentation

5.42.1.1 #define MHV_AD_CHANNEL_0 0x00

Definition at line 72 of file MHV_io_ATmega1280.h.

5.42.1.2 #define MHV_AD_CHANNEL_0_X10_0 0x08

Definition at line 80 of file MHV_io_ATmega1280.h.

5.42.1.3 #define MHV_AD_CHANNEL_0_X1_1 0x10

Definition at line 88 of file MHV_io_ATmega1280.h.

5.42.1.4 `#define MHV_AD_CHANNEL_0_X1_2 0x18`

Definition at line 96 of file MHV_io_ATmega1280.h.

5.42.1.5 `#define MHV_AD_CHANNEL_0_X200_0 0x0a`

Definition at line 82 of file MHV_io_ATmega1280.h.

5.42.1.6 `#define MHV_AD_CHANNEL_0V 0x1f`

Definition at line 103 of file MHV_io_ATmega1280.h.

5.42.1.7 `#define MHV_AD_CHANNEL_1 0x01`

Definition at line 73 of file MHV_io_ATmega1280.h.

5.42.1.8 `#define MHV_AD_CHANNEL_10 0x22`

Definition at line 106 of file MHV_io_ATmega1280.h.

5.42.1.9 `#define MHV_AD_CHANNEL_10_X10_10 0x2c`

Definition at line 116 of file MHV_io_ATmega1280.h.

5.42.1.10 `#define MHV_AD_CHANNEL_10_X1_10 0x3a`

Definition at line 130 of file MHV_io_ATmega1280.h.

5.42.1.11 `#define MHV_AD_CHANNEL_10_X1_9 0x32`

Definition at line 122 of file MHV_io_ATmega1280.h.

5.42.1.12 `#define MHV_AD_CHANNEL_10_X200_10 0x2e`

Definition at line 118 of file MHV_io_ATmega1280.h.

5.42.1.13 `#define MHV_AD_CHANNEL_11 0x23`

Definition at line 107 of file MHV_io_ATmega1280.h.

5.42.1.14 `#define MHV_AD_CHANNEL_11_X10_10 0x2d`

Definition at line 117 of file MHV_io_ATmega1280.h.

5.42.1.15 `#define MHV_AD_CHANNEL_11_X1_10 0x3b`

Definition at line 131 of file MHV_io_ATmega1280.h.

5.42.1.16 `#define MHV_AD_CHANNEL_11_X1_9 0x33`

Definition at line 123 of file MHV_io_ATmega1280.h.

5.42.1.17 `#define MHV_AD_CHANNEL_11_X200_10 0x2f`

Definition at line 119 of file MHV_io_ATmega1280.h.

5.42.1.18 `#define MHV_AD_CHANNEL_12 0x24`

Definition at line 108 of file MHV_io_ATmega1280.h.

5.42.1.19 `#define MHV_AD_CHANNEL_12_X1_10 0x3c`

Definition at line 132 of file MHV_io_ATmega1280.h.

5.42.1.20 `#define MHV_AD_CHANNEL_12_X1_9 0x34`

Definition at line 124 of file MHV_io_ATmega1280.h.

5.42.1.21 `#define MHV_AD_CHANNEL_13 0x25`

Definition at line 109 of file MHV_io_ATmega1280.h.

5.42.1.22 `#define MHV_AD_CHANNEL_13_X1_10 0x3d`

Definition at line 133 of file MHV_io_ATmega1280.h.

5.42.1.23 `#define MHV_AD_CHANNEL_13_X1_9 0x35`

Definition at line 125 of file MHV_io_ATmega1280.h.

5.42.1.24 `#define MHV_AD_CHANNEL_14 0x26`

Definition at line 110 of file MHV_io_ATmega1280.h.

5.42.1.25 `#define MHV_AD_CHANNEL_14_X1_9 0x36`

Definition at line 126 of file MHV_io_ATmega1280.h.

5.42.1.26 `#define MHV_AD_CHANNEL_15 0x27`

Definition at line 111 of file MHV_io_ATmega1280.h.

5.42.1.27 `#define MHV_AD_CHANNEL_15_X1_9 0x37`

Definition at line 127 of file MHV_io_ATmega1280.h.

5.42.1.28 `#define MHV_AD_CHANNEL_1_X10_0 0x09`

Definition at line 81 of file MHV_io_ATmega1280.h.

5.42.1.29 `#define MHV_AD_CHANNEL_1_X1_1 0x11`

Definition at line 89 of file MHV_io_ATmega1280.h.

5.42.1.30 `#define MHV_AD_CHANNEL_1_X1_2 0x19`

Definition at line 97 of file MHV_io_ATmega1280.h.

5.42.1.31 `#define MHV_AD_CHANNEL_1_X200_0 0x0b`

Definition at line 83 of file MHV_io_ATmega1280.h.

5.42.1.32 `#define MHV_AD_CHANNEL_1V1 0x1e`

Definition at line 102 of file MHV_io_ATmega1280.h.

5.42.1.33 `#define MHV_AD_CHANNEL_2 0x02`

Definition at line 74 of file MHV_io_ATmega1280.h.

5.42.1.34 `#define MHV_AD_CHANNEL_2_X10_2 0x0c`

Definition at line 84 of file MHV_io_ATmega1280.h.

5.42.1.35 `#define MHV_AD_CHANNEL_2_X1_1 0x12`

Definition at line 90 of file MHV_io_ATmega1280.h.

5.42.1.36 `#define MHV_AD_CHANNEL_2_X1_2 0x1a`

Definition at line 98 of file MHV_io_ATmega1280.h.

5.42.1.37 `#define MHV_AD_CHANNEL_2_X200_2 0x0e`

Definition at line 86 of file MHV_io_ATmega1280.h.

5.42.1.38 `#define MHV_AD_CHANNEL_3 0x03`

Definition at line 75 of file MHV_io_ATmega1280.h.

5.42.1.39 `#define MHV_AD_CHANNEL_3_X10_2 0x0d`

Definition at line 85 of file MHV_io_ATmega1280.h.

5.42.1.40 `#define MHV_AD_CHANNEL_3_X1_1 0x13`

Definition at line 91 of file MHV_io_ATmega1280.h.

5.42.1.41 `#define MHV_AD_CHANNEL_3_X1_2 0x1b`

Definition at line 99 of file MHV_io_ATmega1280.h.

5.42.1.42 `#define MHV_AD_CHANNEL_3_X200_2 0x0f`

Definition at line 87 of file MHV_io_ATmega1280.h.

5.42.1.43 `#define MHV_AD_CHANNEL_4 0x04`

Definition at line 76 of file MHV_io_ATmega1280.h.

5.42.1.44 `#define MHV_AD_CHANNEL_4_X1_1 0x14`

Definition at line 92 of file MHV_io_ATmega1280.h.

5.42.1.45 `#define MHV_AD_CHANNEL_4_X1_2 0x1c`

Definition at line 100 of file MHV_io_ATmega1280.h.

5.42.1.46 `#define MHV_AD_CHANNEL_5 0x05`

Definition at line 77 of file MHV_io_ATmega1280.h.

5.42.1.47 `#define MHV_AD_CHANNEL_5_X1_1 0x15`

Definition at line 93 of file MHV_io_ATmega1280.h.

5.42.1.48 `#define MHV_AD_CHANNEL_5_X1_2 0x1d`

Definition at line 101 of file MHV_io_ATmega1280.h.

5.42.1.49 `#define MHV_AD_CHANNEL_6 0x06`

Definition at line 78 of file MHV_io_ATmega1280.h.

5.42.1.50 `#define MHV_AD_CHANNEL_6_X1_1 0x16`

Definition at line 94 of file MHV_io_ATmega1280.h.

5.42.1.51 `#define MHV_AD_CHANNEL_7 0x07`

Definition at line 79 of file MHV_io_ATmega1280.h.

5.42.1.52 `#define MHV_AD_CHANNEL_7_X1_1 0x17`

Definition at line 95 of file MHV_io_ATmega1280.h.

5.42.1.53 `#define MHV_AD_CHANNEL_8 0x20`

Definition at line 104 of file MHV_io_ATmega1280.h.

5.42.1.54 `#define MHV_AD_CHANNEL_8_X10_8 0x28`

Definition at line 112 of file MHV_io_ATmega1280.h.

5.42.1.55 `#define MHV_AD_CHANNEL_8_X1_10 0x38`

Definition at line 128 of file MHV_io_ATmega1280.h.

5.42.1.56 `#define MHV_AD_CHANNEL_8_X1_9 0x30`

Definition at line 120 of file MHV_io_ATmega1280.h.

5.42.1.57 `#define MHV_AD_CHANNEL_8_X200_8 0x2a`

Definition at line 114 of file MHV_io_ATmega1280.h.

5.42.1.58 `#define MHV_AD_CHANNEL_9_0x21`

Definition at line 105 of file MHV_io_ATmega1280.h.

5.42.1.59 `#define MHV_AD_CHANNEL_9_X10_8 0x29`

Definition at line 113 of file MHV_io_ATmega1280.h.

5.42.1.60 `#define MHV_AD_CHANNEL_9_X1_10 0x39`

Definition at line 129 of file MHV_io_ATmega1280.h.

5.42.1.61 `#define MHV_AD_CHANNEL_9_X1_9 0x31`

Definition at line 121 of file MHV_io_ATmega1280.h.

5.42.1.62 `#define MHV_AD_CHANNEL_9_X200_8 0x2b`

Definition at line 115 of file MHV_io_ATmega1280.h.

5.42.1.63 `#define MHV_AD_PRR PRR0`

Definition at line 136 of file MHV_io_ATmega1280.h.

5.42.1.64 `#define MHV_AD_REFERENCE_1V1 (uint8_t)(0x02 << 6)`

Definition at line 69 of file MHV_io_ATmega1280.h.

5.42.1.65 `#define MHV_AD_REFERENCE_2V56 (uint8_t)(0x03 << 6)`

Definition at line 70 of file MHV_io_ATmega1280.h.

5.42.1.66 `#define MHV_AD_REFERENCE_AREF (uint8_t)(0x00 << 6)`

Definition at line 67 of file MHV_io_ATmega1280.h.

5.42.1.67 `#define MHV_AD_REFERENCE_AVCC (uint8_t)(0x01 << 6)`

Definition at line 68 of file MHV_io_ATmega1280.h.

5.42.1.68 `#define MHV_AD_RESOLUTION 1024`

Definition at line 64 of file MHV_io_ATmega1280.h.

5.42.1.69 `#define MHV_EEPROM_VECT EE_READY_vect`

Definition at line 246 of file MHV_io_ATmega1280.h.

5.42.1.70 `#define MHV_PC_INT_COUNT 24`

Definition at line 244 of file MHV_io_ATmega1280.h.

5.42.1.71 `#define MHV_PIN_A0 &DDRA, &PORTA, &PINA, 0, -1`

Definition at line 140 of file MHV_io_ATmega1280.h.

5.42.1.72 `#define MHV_PIN_A1 &DDRA, &PORTA, &PINA, 1, -1`

Definition at line 141 of file MHV_io_ATmega1280.h.

5.42.1.73 `#define MHV_PIN_A2 &DDRA, &PORTA, &PINA, 2, -1`

Definition at line 142 of file MHV_io_ATmega1280.h.

5.42.1.74 `#define MHV_PIN_A3 &DDRA, &PORTA, &PINA, 3, -1`

Definition at line 143 of file MHV_io_ATmega1280.h.

5.42.1.75 `#define MHV_PIN_A4 &DDRA, &PORTA, &PINA, 4, -1`

Definition at line 144 of file MHV_io_ATmega1280.h.

5.42.1.76 `#define MHV_PIN_A5 &DDRA, &PORTA, &PINA, 5, -1`

Definition at line 145 of file MHV_io_ATmega1280.h.

5.42.1.77 `#define MHV_PIN_A6 &DDRA, &PORTA, &PINA, 6, -1`

Definition at line 146 of file MHV_io_ATmega1280.h.

5.42.1.78 `#define MHV_PIN_A7 &DDRA, &PORTA, &PINA, 7, -1`

Definition at line 147 of file MHV_io_ATmega1280.h.

5.42.1.79 `#define MHV_PIN_B0 &DDRB, &PORTB, &PINB, 0, 0`

Definition at line 148 of file MHV_io_ATmega1280.h.

5.42.1.80 `#define MHV_PIN_B1 &DDRB, &PORTB, &PINB, 1, 1`

Definition at line 149 of file MHV_io_ATmega1280.h.

5.42.1.81 `#define MHV_PIN_B2 &DDRB, &PORTB, &PINB, 2, 2`

Definition at line 150 of file MHV_io_ATmega1280.h.

5.42.1.82 `#define MHV_PIN_B3 &DDRB, &PORTB, &PINB, 3, 3`

Definition at line 151 of file MHV_io_ATmega1280.h.

5.42.1.83 `#define MHV_PIN_B4 &DDRB, &PORTB, &PINB, 4, 4`

Definition at line 152 of file MHV_io_ATmega1280.h.

5.42.1.84 `#define MHV_PIN_B5 &DDRB, &PORTB, &PINB, 5, 5`

Definition at line 153 of file MHV_io_ATmega1280.h.

5.42.1.85 `#define MHV_PIN_B6 &DDRB, &PORTB, &PINB, 6, 6`

Definition at line 154 of file MHV_io_ATmega1280.h.

5.42.1.86 `#define MHV_PIN_B7 &DDRB, &PORTB, &PINB, 7, 7`

Definition at line 155 of file MHV_io_ATmega1280.h.

5.42.1.87 `#define MHV_PIN_C0 &DDRC, &PORTC, &PINC, 0, -1`

Definition at line 156 of file MHV_io_ATmega1280.h.

5.42.1.88 `#define MHV_PIN_C1 &DDRC, &PORTC, &PINC, 1, -1`

Definition at line 157 of file MHV_io_ATmega1280.h.

5.42.1.89 `#define MHV_PIN_C2 &DDRC, &PORTC, &PINC, 2, -1`

Definition at line 158 of file MHV_io_ATmega1280.h.

5.42.1.90 `#define MHV_PIN_C3 &DDRC, &PORTC, &PINC, 3, -1`

Definition at line 159 of file MHV_io_ATmega1280.h.

5.42.1.91 `#define MHV_PIN_C4 &DDRC, &PORTC, &PINC, 4, -1`

Definition at line 160 of file MHV_io_ATmega1280.h.

5.42.1.92 `#define MHV_PIN_C5 &DDRC, &PORTC, &PINC, 5, -1`

Definition at line 161 of file MHV_io_ATmega1280.h.

5.42.1.93 `#define MHV_PIN_C6 &DDRC, &PORTC, &PINC, 6, -1`

Definition at line 162 of file MHV_io_ATmega1280.h.

5.42.1.94 `#define MHV_PIN_C7 &DDRC, &PORTC, &PINC, 7, -1`

Definition at line 163 of file MHV_io_ATmega1280.h.

5.42.1.95 `#define MHV_PIN_D0 &DDRD, &PORTD, &PIND, 0, -1`

Definition at line 164 of file MHV_io_ATmega1280.h.

5.42.1.96 `#define MHV_PIN_D1 &DDRD, &PORTD, &PIND, 1, -1`

Definition at line 165 of file MHV_io_ATmega1280.h.

5.42.1.97 `#define MHV_PIN_D2 &DDRD, &PORTD, &PIND, 2, -1`

Definition at line 166 of file MHV_io_ATmega1280.h.

5.42.1.98 `#define MHV_PIN_D3 &DDRD, &PORTD, &PIND, 3, -1`

Definition at line 167 of file MHV_io_ATmega1280.h.

5.42.1.99 `#define MHV_PIN_D4 &DDRD, &PORTD, &PIND, 4, -1`

Definition at line 168 of file MHV_io_ATmega1280.h.

5.42.1.100 `#define MHV_PIN_D5 &DDRD, &PORTD, &PIND, 5, -1`

Definition at line 169 of file MHV_io_ATmega1280.h.

5.42.1.101 `#define MHV_PIN_D6 &DDRD, &PORTD, &PIND, 6, -1`

Definition at line 170 of file MHV_io_ATmega1280.h.

5.42.1.102 `#define MHV_PIN_D7 &DDRD, &PORTD, &PIND, 7, -1`

Definition at line 171 of file MHV_io_ATmega1280.h.

5.42.1.103 `#define MHV_PIN_E0 &DDRE, &PORTE, &PINE, 0, 8`

Definition at line 172 of file MHV_io_ATmega1280.h.

5.42.1.104 `#define MHV_PIN_E1 &DDRE, &PORTE, &PINE, 1, -1`

Definition at line 173 of file MHV_io_ATmega1280.h.

5.42.1.105 `#define MHV_PIN_E2 &DDRE, &PORTE, &PINE, 2, -1`

Definition at line 174 of file MHV_io_ATmega1280.h.

5.42.1.106 `#define MHV_PIN_E3 &DDRE, &PORTE, &PINE, 3, -1`

Definition at line 175 of file MHV_io_ATmega1280.h.

5.42.1.107 `#define MHV_PIN_E4 &DDRE, &PORTE, &PINE, 4, -1`

Definition at line 176 of file MHV_io_ATmega1280.h.

5.42.1.108 `#define MHV_PIN_E5 &DDRE, &PORTE, &PINE, 5, -1`

Definition at line 177 of file MHV_io_ATmega1280.h.

5.42.1.109 `#define MHV_PIN_E6 &DDRE, &PORTE, &PINE, 6, -1`

Definition at line 178 of file MHV_io_ATmega1280.h.

5.42.1.110 `#define MHV_PIN_E7 &DDRE, &PORTE, &PINE, 7, -1`

Definition at line 179 of file MHV_io_ATmega1280.h.

5.42.1.111 `#define MHV_PIN_F0 &DDRF, &PORTF, &PINF, 0, -1`

Definition at line 180 of file MHV_io_ATmega1280.h.

5.42.1.112 `#define MHV_PIN_F1 &DDRF, &PORTF, &PINF, 1, -1`

Definition at line 181 of file MHV_io_ATmega1280.h.

5.42.1.113 `#define MHV_PIN_F2 &DDRF, &PORTF, &PINF, 2, -1`

Definition at line 182 of file MHV_io_ATmega1280.h.

5.42.1.114 `#define MHV_PIN_F3 &DDRF, &PORTF, &PINF, 3, -1`

Definition at line 183 of file MHV_io_ATmega1280.h.

5.42.1.115 `#define MHV_PIN_F4 &DDRF, &PORTF, &PINF, 4, -1`

Definition at line 184 of file MHV_io_ATmega1280.h.

5.42.1.116 `#define MHV_PIN_F5 &DDRF, &PORTF, &PINF, 5, -1`

Definition at line 185 of file MHV_io_ATmega1280.h.

5.42.1.117 `#define MHV_PIN_F6 &DDRF, &PORTF, &PINF, 6, -1`

Definition at line 186 of file MHV_io_ATmega1280.h.

5.42.1.118 `#define MHV_PIN_F7 &DDRF, &PORTF, &PINF, 7, -1`

Definition at line 187 of file MHV_io_ATmega1280.h.

5.42.1.119 `#define MHV_PIN_G0 &DDRG, &PORTG, &PING, 0, -1`

Definition at line 188 of file MHV_io_ATmega1280.h.

5.42.1.120 `#define MHV_PIN_G1 &DDRG, &PORTG, &PING, 1, -1`

Definition at line 189 of file MHV_io_ATmega1280.h.

5.42.1.121 `#define MHV_PIN_G2 &DDRG, &PORTG, &PING, 2, -1`

Definition at line 190 of file MHV_io_ATmega1280.h.

5.42.1.122 `#define MHV_PIN_G3 &DDRG, &PORTG, &PING, 3, -1`

Definition at line 191 of file MHV_io_ATmega1280.h.

5.42.1.123 `#define MHV_PIN_G4 &DDRG, &PORTG, &PING, 4, -1`

Definition at line 192 of file MHV_io_ATmega1280.h.

5.42.1.124 `#define MHV_PIN_G5 &DDRG, &PORTG, &PING, 5, -1`

Definition at line 193 of file MHV_io_ATmega1280.h.

5.42.1.125 `#define MHV_PIN_H0 &DDRH, &PORTH, &PINH, 0, -1`

Definition at line 194 of file MHV_io_ATmega1280.h.

5.42.1.126 `#define MHV_PIN_H1 &DDRH, &PORTH, &PINH, 1, -1`

Definition at line 195 of file MHV_io_ATmega1280.h.

5.42.1.127 `#define MHV_PIN_H2 &DDRH, &PORTH, &PINH, 2, -1`

Definition at line 196 of file MHV_io_ATmega1280.h.

5.42.1.128 `#define MHV_PIN_H3 &DDRH, &PORTH, &PINH, 3, -1`

Definition at line 197 of file MHV_io_ATmega1280.h.

5.42.1.129 `#define MHV_PIN_H4 &DDRH, &PORTH, &PINH, 4, -1`

Definition at line 198 of file MHV_io_ATmega1280.h.

5.42.1.130 `#define MHV_PIN_H5 &DDRH, &PORTH, &PINH, 5, -1`

Definition at line 199 of file MHV_io_ATmega1280.h.

5.42.1.131 `#define MHV_PIN_H6 &DDRH, &PORTH, &PINH, 6, -1`

Definition at line 200 of file MHV_io_ATmega1280.h.

5.42.1.132 `#define MHV_PIN_H7 &DDRH, &PORTH, &PINH, 7, -1`

Definition at line 201 of file MHV_io_ATmega1280.h.

5.42.1.133 `#define MHV_PIN_J0 &DDRJ, &PORTJ, &PINJ, 0, 9`

Definition at line 202 of file MHV_io_ATmega1280.h.

5.42.1.134 `#define MHV_PIN_J1 &DDRJ, &PORTJ, &PINJ, 1, 10`

Definition at line 203 of file MHV_io_ATmega1280.h.

5.42.1.135 `#define MHV_PIN_J2 &DDRJ, &PORTJ, &PINJ, 2, 11`

Definition at line 204 of file MHV_io_ATmega1280.h.

5.42.1.136 `#define MHV_PIN_J3 &DDRJ, &PORTJ, &PINJ, 3, 12`

Definition at line 205 of file MHV_io_ATmega1280.h.

5.42.1.137 `#define MHV_PIN_J4 &DDRJ, &PORTJ, &PINJ, 4, 13`

Definition at line 206 of file MHV_io_ATmega1280.h.

5.42.1.138 `#define MHV_PIN_J5 &DDRJ, &PORTJ, &PINJ, 5, 14`

Definition at line 207 of file MHV_io_ATmega1280.h.

5.42.1.139 `#define MHV_PIN_J6 &DDRJ, &PORTJ, &PINJ, 6, 15`

Definition at line 208 of file MHV_io_ATmega1280.h.

5.42.1.140 `#define MHV_PIN_J7 &DDRJ, &PORTJ, &PINJ, 7, -1`

Definition at line 209 of file MHV_io_ATmega1280.h.

5.42.1.141 `#define MHV_PIN_K0 &DDRK, &PORTK, &PINK, 0, 16`

Definition at line 210 of file MHV_io_ATmega1280.h.

5.42.1.142 `#define MHV_PIN_K1 &DDRK, &PORTK, &PINK, 1, 17`

Definition at line 211 of file MHV_io_ATmega1280.h.

5.42.1.143 `#define MHV_PIN_K2 &DDRK, &PORTK, &PINK, 2, 18`

Definition at line 212 of file MHV_io_ATmega1280.h.

5.42.1.144 `#define MHV_PIN_K3 &DDRK, &PORTK, &PINK, 3, 19`

Definition at line 213 of file MHV_io_ATmega1280.h.

5.42.1.145 `#define MHV_PIN_K4 &DDRK, &PORTK, &PINK, 4, 20`

Definition at line 214 of file MHV_io_ATmega1280.h.

5.42.1.146 `#define MHV_PIN_K5 &DDRK, &PORTK, &PINK, 5, 21`

Definition at line 215 of file MHV_io_ATmega1280.h.

5.42.1.147 `#define MHV_PIN_K6 &DDRK, &PORTK, &PINK, 6, 22`

Definition at line 216 of file MHV_io_ATmega1280.h.

5.42.1.148 `#define MHV_PIN_K7 &DDRK, &PORTK, &PINK, 7, 23`

Definition at line 217 of file MHV_io_ATmega1280.h.

5.42.1.149 `#define MHV_PIN_L0 &DDRL, &PORTL, &PINL, 0, -1`

Definition at line 218 of file MHV_io_ATmega1280.h.

5.42.1.150 `#define MHV_PIN_L1 &DDRL, &PORTL, &PINL, 1, -1`

Definition at line 219 of file MHV_io_ATmega1280.h.

5.42.1.151 `#define MHV_PIN_L2 &DDRL, &PORTL, &PINL, 2, -1`

Definition at line 220 of file MHV_io_ATmega1280.h.

5.42.1.152 `#define MHV_PIN_L3 &DDRL, &PORTL, &PINL, 3, -1`

Definition at line 221 of file MHV_io_ATmega1280.h.

5.42.1.153 `#define MHV_PIN_L4 &DDRL, &PORTL, &PINL, 4, -1`

Definition at line 222 of file MHV_io_ATmega1280.h.

5.42.1.154 `#define MHV_PIN_L5 &DDRL, &PORTL, &PINL, 5, -1`

Definition at line 223 of file MHV_io_ATmega1280.h.

5.42.1.155 `#define MHV_PIN_L6 &DDRL, &PORTL, &PINL, 6, -1`

Definition at line 224 of file MHV_io_ATmega1280.h.

5.42.1.156 `#define MHV_PIN_L7 &DDRL, &PORTL, &PINL, 7, -1`

Definition at line 225 of file MHV_io_ATmega1280.h.

5.42.1.157 `#define MHV_PIN_TIMER_0_A MHV_PIN_B7`

Definition at line 227 of file MHV_io_ATmega1280.h.

5.42.1.158 `#define MHV_PIN_TIMER_0_B MHV_PIN_G5`

Definition at line 228 of file MHV_io_ATmega1280.h.

5.42.1.159 `#define MHV_PIN_TIMER_1_A MHV_PIN_B5`

Definition at line 229 of file MHV_io_ATmega1280.h.

5.42.1.160 `#define MHV_PIN_TIMER_1_B MHV_PIN_B6`

Definition at line 230 of file MHV_io_ATmega1280.h.

5.42.1.161 `#define MHV_PIN_TIMER_1_C MHV_PIN_B7`

Definition at line 231 of file MHV_io_ATmega1280.h.

5.42.1.162 `#define MHV_PIN_TIMER_2_A MHV_PIN_B4`

Definition at line 232 of file MHV_io_ATmega1280.h.

5.42.1.163 `#define MHV_PIN_TIMER_2_B MHV_PIN_H6`

Definition at line 233 of file MHV_io_ATmega1280.h.

5.42.1.164 `#define MHV_PIN_TIMER_3_A MHV_PIN_E3`

Definition at line 234 of file MHV_io_ATmega1280.h.

5.42.1.165 `#define MHV_PIN_TIMER_3_B MHV_PIN_E4`

Definition at line 235 of file MHV_io_ATmega1280.h.

5.42.1.166 `#define MHV_PIN_TIMER_3_C MHV_PIN_E5`

Definition at line 236 of file MHV_io_ATmega1280.h.

5.42.1.167 `#define MHV_PIN_TIMER_4_A MHV_PIN_H3`

Definition at line 237 of file MHV_io_ATmega1280.h.

5.42.1.168 `#define MHV_PIN_TIMER_4_B MHV_PIN_H4`

Definition at line 238 of file MHV_io_ATmega1280.h.

5.42.1.169 `#define MHV_PIN_TIMER_4_C MHV_PIN_H5`

Definition at line 239 of file MHV_io_ATmega1280.h.

5.42.1.170 `#define MHV_PIN_TIMER_5_A MHV_PIN_L3`

Definition at line 240 of file MHV_io_ATmega1280.h.

5.42.1.171 `#define MHV_PIN_TIMER_5_B MHV_PIN_L4`

Definition at line 241 of file MHV_io_ATmega1280.h.

5.42.1.172 `#define MHV_PIN_TIMER_5_C MHV_PIN_L6`

Definition at line 242 of file MHV_io_ATmega1280.h.

5.42.1.173 `#define MHV_TIMER0_INTERRUPTS TIMER0_COMPA_vect, TIMER0_COMPB_vect, 0`

Definition at line 39 of file MHV_io_ATmega1280.h.

5.42.1.174 `#define MHV_TIMER16_1 &TCCR1A, &TCCR1B, &TCCR1C, &OCR1A, &OCR1B,
&OCR1C, &TCNT1, &TIMSK1, &ICR1`

Definition at line 44 of file MHV_io_ATmega1280.h.

5.42.1.175 `#define MHV_TIMER16_3 &TCCR3A, &TCCR3B, &TCCR3C, &OCR3A, &OCR3B,
&OCR3C, &TCNT3, &TIMSK3, &ICR3`

Definition at line 45 of file MHV_io_ATmega1280.h.

5.42.1.176 `#define MHV_TIMER16_4 &TCCR4A, &TCCR4B, &TCCR4C, &OCR4A, &OCR4B,
&OCR4C, &TCNT4, &TIMSK4, &ICR4`

Definition at line 46 of file MHV_io_ATmega1280.h.

5.42.1.177 `#define MHV_TIMER16_5 &TCCR5A, &TCCR5B, &TCCR5C, &OCR5A, &OCR5B,
&OCR5C, &TCNT5, &TIMSK5, &ICR5`

Definition at line 47 of file MHV_io_ATmega1280.h.

5.42.1.178 `#define MHV_TIMER1_INTERRUPTS TIMER1_COMPA_vect, TIMER1_COMPB_vect,
TIMER1_COMPC_vect`

Definition at line 49 of file MHV_io_ATmega1280.h.

5.42.1.179 `#define MHV_TIMER2_INTERRUPTS TIMER2_COMPA_vect, TIMER2_COMPB_vect, 0`

Definition at line 40 of file MHV_io_ATmega1280.h.

5.42.1.180 `#define MHV_TIMER3_INTERRUPTS TIMER3_COMPA_vect, TIMER3_COMPB_vect,
TIMER3_COMPC_vect`

Definition at line 50 of file MHV_io_ATmega1280.h.

5.42.1.181 `#define MHV_TIMER4_INTERRUPTS TIMER4_COMPA_vect, TIMER4_COMPB_vect,
TIMER4_COMPC_vect`

Definition at line 51 of file MHV_io_ATmega1280.h.

5.42.1.182 `#define MHV_TIMER5_INTERRUPTS TIMER5_COMPA_vect, TIMER5_COMPB_vect,
TIMER5_COMPC_vect`

Definition at line 52 of file MHV_io_ATmega1280.h.

5.42.1.183 `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B,
&OCR0A, &OCR0B, &TCNT0, &TIMSK0, OCIE0A`

Definition at line 36 of file MHV_io_ATmega1280.h.

5.42.1.184 `#define MHV_TIMER8_2 MHV_TIMER_TYPE_7_PRESCALERS, &TCCR2A, &TCCR2B,
&OCR2A, &OCR2B, &TCNT2, &TIMSK2, OCIE2A`

Definition at line 37 of file MHV_io_ATmega1280.h.

5.42.1.185 `#define MHV_USART0 &UBRR0, &UCSR0A, &UCSR0B, &UDR0, RXEN0, TXEN0,
RXCIE0, TXCIE0, UDRE0, U2X0`

Definition at line 54 of file MHV_io_ATmega1280.h.

5.42.1.186 `#define MHV_USART0_INTERRUPTS USART0_RX_vect, USART0_TX_vect`

Definition at line 59 of file MHV_io_ATmega1280.h.

5.42.1.187 `#define MHV_USART1 &UBRR1, &UCSR1A, &UCSR1B, &UDR1, RXEN1, TXEN1,
RXCIE1, TXCIE1, UDRE1, U2X1`

Definition at line 55 of file MHV_io_ATmega1280.h.

5.42.1.188 `#define MHV_USART1_INTERRUPTS USART1_RX_vect, USART1_TX_vect`

Definition at line 60 of file MHV_io_ATmega1280.h.

5.42.1.189 `#define MHV_USART2 &UBRR2, &UCSR2A, &UCSR2B, &UDR2, RXEN2, TXEN2,
RXCIE2, TXCIE2, UDRE2, U2X2`

Definition at line 56 of file MHV_io_ATmega1280.h.

5.42.1.190 `#define MHV_USART2_INTERRUPTS USART2_RX_vect, USART2_TX_vect`

Definition at line 61 of file MHV_io_ATmega1280.h.

5.42.1.191 `#define MHV_USART3 &UBRR3, &UCSR3A, &UCSR3B, &UDR3, RXEN3, TXEN3,
RXCIE3, TXCIE3, UDRE3, U2X3`

Definition at line 57 of file MHV_io_ATmega1280.h.

5.42.1.192 `#define MHV_USART3_INTERRUPTS USART3_RX_vect, USART3_TX_vect`

Definition at line 62 of file MHV_io_ATmega1280.h.

5.43 A:/eclipse/mhvlb/MHV_io_ATmega168.h File Reference

```
#include <avr/io.h>
```

Defines

- `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B, &OCR0A, &OCR0B, &TCNT0, &TIMSK0, OCIE0A`
- `#define MHV_TIMER8_2 MHV_TIMER_TYPE_7_PRESCALERS, &TCCR2A, &TCCR2B, &OCR2A, &OCR2B, &TCNT2, &TIMSK2, OCIE2A`
- `#define MHV_TIMER0_INTERRUPTS TIMER0_COMPA_vect, TIMER0_COMPB_vect, 0`
- `#define MHV_TIMER2_INTERRUPTS TIMER2_COMPA_vect, TIMER2_COMPB_vect, 0`
- `#define MHV_TIMER16_1 &TCCR1A, &TCCR1B, 0, &OCR1A, &OCR1B, 0, &TCNT1, &TIMSK1, &ICR1`
- `#define MHV_TIMER1_INTERRUPTS TIMER1_COMPA_vect, TIMER1_COMPB_vect, 0`
- `#define MHV_USART0 &UBRR0, &UCSR0A, &UCSR0B, &UDR0, RXEN0, TXEN0, RXCIE0, TXCIE0, UDRE0, U2X0`
- `#define MHV_USART0_INTERRUPTS USART_RX_vect, USART_TX_vect`
- `#define MHV_AD_RESOLUTION 1024`
- `#define MHV_AD_REFERENCE_AREF (uint8_t)(0x00 << 6)`
- `#define MHV_AD_REFERENCE_AVCC (uint8_t)(0x01 << 6)`
- `#define MHV_AD_REFERENCE_1V1 (uint8_t)(0x03 << 6)`
- `#define MHV_AD_CHANNEL_0 0x00`
- `#define MHV_AD_CHANNEL_1 0x01`
- `#define MHV_AD_CHANNEL_2 0x02`
- `#define MHV_AD_CHANNEL_3 0x03`
- `#define MHV_AD_CHANNEL_4 0x04`
- `#define MHV_AD_CHANNEL_5 0x05`
- `#define MHV_AD_CHANNEL_6 0x06`
- `#define MHV_AD_CHANNEL_7 0x07`
- `#define MHV_AD_CHANNEL_8 0x08`
- `#define MHV_AD_TEMPERATURE 0x08`
- `#define MHV_AD_CHANNEL_1V1 0xfe`
- `#define MHV_AD_CHANNEL_0V 0xff`
- `#define MHV_AD_PRR PRR`
- `#define MHV_PIN_B0 &DDRB, &PORTB, &PINB, 0, 0`
- `#define MHV_PIN_B1 &DDRB, &PORTB, &PINB, 1, 1`
- `#define MHV_PIN_B2 &DDRB, &PORTB, &PINB, 2, 2`

- #define [MHV_PIN_B3](#) &DDRB, &PORTB, &PINB, 3, 3
- #define [MHV_PIN_B4](#) &DDRB, &PORTB, &PINB, 4, 4
- #define [MHV_PIN_B5](#) &DDRB, &PORTB, &PINB, 5, 5
- #define [MHV_PIN_B6](#) &DDRB, &PORTB, &PINB, 6, 6
- #define [MHV_PIN_B7](#) &DDRB, &PORTB, &PINB, 7, 7
- #define [MHV_PIN_C0](#) &DDRC, &PORTC, &PINC, 0, 8
- #define [MHV_PIN_C1](#) &DDRC, &PORTC, &PINC, 1, 9
- #define [MHV_PIN_C2](#) &DDRC, &PORTC, &PINC, 2, 10
- #define [MHV_PIN_C3](#) &DDRC, &PORTC, &PINC, 3, 11
- #define [MHV_PIN_C4](#) &DDRC, &PORTC, &PINC, 4, 12
- #define [MHV_PIN_C5](#) &DDRC, &PORTC, &PINC, 5, 13
- #define [MHV_PIN_C6](#) &DDRC, &PORTC, &PINC, 6, 14
- #define [MHV_PIN_D0](#) &DDRD, &PORTD, &PIND, 0, 16
- #define [MHV_PIN_D1](#) &DDRD, &PORTD, &PIND, 1, 17
- #define [MHV_PIN_D2](#) &DDRD, &PORTD, &PIND, 2, 18
- #define [MHV_PIN_D3](#) &DDRD, &PORTD, &PIND, 3, 19
- #define [MHV_PIN_D4](#) &DDRD, &PORTD, &PIND, 4, 20
- #define [MHV_PIN_D5](#) &DDRD, &PORTD, &PIND, 5, 21
- #define [MHV_PIN_D6](#) &DDRD, &PORTD, &PIND, 6, 22
- #define [MHV_PIN_D7](#) &DDRD, &PORTD, &PIND, 7, 23
- #define [MHV_PIN_TIMER_0_A](#) MHV_PIN_D6
- #define [MHV_PIN_TIMER_0_B](#) MHV_PIN_D5
- #define [MHV_PIN_TIMER_1_A](#) MHV_PIN_B1
- #define [MHV_PIN_TIMER_1_B](#) MHV_PIN_B2
- #define [MHV_PIN_TIMER_2_A](#) MHV_PIN_B3
- #define [MHV_PIN_TIMER_2_B](#) MHV_PIN_D3
- #define [MHV_INTERRUPT_INT0](#) INT0_vect, &MCUCR, ISC00
- #define [MHV_INTERRUPT_INT1](#) INT1_vect, &MCUCR, ISC10
- #define [MHV_PC_INT_COUNT](#) 24
- #define [MHV_EEPROM_VECT](#) EE_READY_vect

5.43.1 Define Documentation

5.43.1.1 #define MHV_AD_CHANNEL_0 0x00

Definition at line 60 of file MHV_io_ATmega168.h.

5.43.1.2 #define MHV_AD_CHANNEL_0V 0xff

Definition at line 71 of file MHV_io_ATmega168.h.

5.43.1.3 #define MHV_AD_CHANNEL_1 0x01

Definition at line 61 of file MHV_io_ATmega168.h.

5.43.1.4 `#define MHV_AD_CHANNEL_1V1 0xfe`

Definition at line 70 of file MHV_io_ATmega168.h.

5.43.1.5 `#define MHV_AD_CHANNEL_2 0x02`

Definition at line 62 of file MHV_io_ATmega168.h.

5.43.1.6 `#define MHV_AD_CHANNEL_3 0x03`

Definition at line 63 of file MHV_io_ATmega168.h.

5.43.1.7 `#define MHV_AD_CHANNEL_4 0x04`

Definition at line 64 of file MHV_io_ATmega168.h.

5.43.1.8 `#define MHV_AD_CHANNEL_5 0x05`

Definition at line 65 of file MHV_io_ATmega168.h.

5.43.1.9 `#define MHV_AD_CHANNEL_6 0x06`

Definition at line 66 of file MHV_io_ATmega168.h.

5.43.1.10 `#define MHV_AD_CHANNEL_7 0x07`

Definition at line 67 of file MHV_io_ATmega168.h.

5.43.1.11 `#define MHV_AD_CHANNEL_8 0x08`

Definition at line 68 of file MHV_io_ATmega168.h.

5.43.1.12 `#define MHV_AD_PRR PRR`

Definition at line 74 of file MHV_io_ATmega168.h.

5.43.1.13 `#define MHV_AD_REFERENCE_1V1 (uint8_t)(0x03 << 6)`

Definition at line 58 of file MHV_io_ATmega168.h.

5.43.1.14 `#define MHV_AD_REFERENCE_AREF (uint8_t)(0x00 << 6)`

Definition at line 56 of file MHV_io_ATmega168.h.

5.43.1.15 `#define MHV_AD_REFERENCE_AVCC (uint8_t)(0x01 << 6)`

Definition at line 57 of file MHV_io_ATmega168.h.

5.43.1.16 `#define MHV_AD_RESOLUTION 1024`

Definition at line 54 of file MHV_io_ATmega168.h.

5.43.1.17 `#define MHV_AD_TEMPERATURE 0x08`

Definition at line 69 of file MHV_io_ATmega168.h.

5.43.1.18 `#define MHV_EEPROM_VECT EE_READY_vect`

Definition at line 113 of file MHV_io_ATmega168.h.

5.43.1.19 `#define MHV_INTERRUPT_INT0 INT0_vect, &MCUCR, ISC00`

Definition at line 108 of file MHV_io_ATmega168.h.

5.43.1.20 `#define MHV_INTERRUPT_INT1 INT1_vect, &MCUCR, ISC10`

Definition at line 109 of file MHV_io_ATmega168.h.

5.43.1.21 `#define MHV_PC_INT_COUNT 24`

Definition at line 111 of file MHV_io_ATmega168.h.

5.43.1.22 `#define MHV_PIN_B0 &DDRB, &PORTB, &PINB, 0, 0`

Definition at line 77 of file MHV_io_ATmega168.h.

5.43.1.23 `#define MHV_PIN_B1 &DDRB, &PORTB, &PINB, 1, 1`

Definition at line 78 of file MHV_io_ATmega168.h.

5.43.1.24 `#define MHV_PIN_B2 &DDRB, &PORTB, &PINB, 2, 2`

Definition at line 79 of file MHV_io_ATmega168.h.

5.43.1.25 `#define MHV_PIN_B3 &DDRB, &PORTB, &PINB, 3, 3`

Definition at line 80 of file MHV_io_ATmega168.h.

5.43.1.26 `#define MHV_PIN_B4 &DDRB, &PORTB, &PINB, 4, 4`

Definition at line 81 of file MHV_io_ATmega168.h.

5.43.1.27 `#define MHV_PIN_B5 &DDRB, &PORTB, &PINB, 5, 5`

Definition at line 82 of file MHV_io_ATmega168.h.

5.43.1.28 `#define MHV_PIN_B6 &DDRB, &PORTB, &PINB, 6, 6`

Definition at line 83 of file MHV_io_ATmega168.h.

5.43.1.29 `#define MHV_PIN_B7 &DDRB, &PORTB, &PINB, 7, 7`

Definition at line 84 of file MHV_io_ATmega168.h.

5.43.1.30 `#define MHV_PIN_C0 &DDRC, &PORTC, &PINC, 0, 8`

Definition at line 85 of file MHV_io_ATmega168.h.

5.43.1.31 `#define MHV_PIN_C1 &DDRC, &PORTC, &PINC, 1, 9`

Definition at line 86 of file MHV_io_ATmega168.h.

5.43.1.32 `#define MHV_PIN_C2 &DDRC, &PORTC, &PINC, 2, 10`

Definition at line 87 of file MHV_io_ATmega168.h.

5.43.1.33 `#define MHV_PIN_C3 &DDRC, &PORTC, &PINC, 3, 11`

Definition at line 88 of file MHV_io_ATmega168.h.

5.43.1.34 `#define MHV_PIN_C4 &DDRC, &PORTC, &PINC, 4, 12`

Definition at line 89 of file MHV_io_ATmega168.h.

5.43.1.35 `#define MHV_PIN_C5 &DDRC, &PORTC, &PINC, 5, 13`

Definition at line 90 of file MHV_io_ATmega168.h.

5.43.1.36 `#define MHV_PIN_C6 &DDRC, &PORTC, &PINC, 6, 14`

Definition at line 91 of file MHV_io_ATmega168.h.

5.43.1.37 `#define MHV_PIN_D0 &DDRD, &PORTD, &PIND, 0, 16`

Definition at line 92 of file MHV_io_ATmega168.h.

5.43.1.38 `#define MHV_PIN_D1 &DDRD, &PORTD, &PIND, 1, 17`

Definition at line 93 of file MHV_io_ATmega168.h.

5.43.1.39 `#define MHV_PIN_D2 &DDRD, &PORTD, &PIND, 2, 18`

Definition at line 94 of file MHV_io_ATmega168.h.

5.43.1.40 `#define MHV_PIN_D3 &DDRD, &PORTD, &PIND, 3, 19`

Definition at line 95 of file MHV_io_ATmega168.h.

5.43.1.41 `#define MHV_PIN_D4 &DDRD, &PORTD, &PIND, 4, 20`

Definition at line 96 of file MHV_io_ATmega168.h.

5.43.1.42 `#define MHV_PIN_D5 &DDRD, &PORTD, &PIND, 5, 21`

Definition at line 97 of file MHV_io_ATmega168.h.

5.43.1.43 `#define MHV_PIN_D6 &DDRD, &PORTD, &PIND, 6, 22`

Definition at line 98 of file MHV_io_ATmega168.h.

5.43.1.44 `#define MHV_PIN_D7 &DDRD, &PORTD, &PIND, 7, 23`

Definition at line 99 of file MHV_io_ATmega168.h.

5.43.1.45 `#define MHV_PIN_TIMER_0_A MHV_PIN_D6`

Definition at line 101 of file MHV_io_ATmega168.h.

5.43.1.46 `#define MHV_PIN_TIMER_0_B MHV_PIN_D5`

Definition at line 102 of file MHV_io_ATmega168.h.

5.43.1.47 `#define MHV_PIN_TIMER_1_A MHV_PIN_B1`

Definition at line 103 of file MHV_io_ATmega168.h.

5.43.1.48 `#define MHV_PIN_TIMER_1_B MHV_PIN_B2`

Definition at line 104 of file MHV_io_ATmega168.h.

5.43.1.49 `#define MHV_PIN_TIMER_2_A MHV_PIN_B3`

Definition at line 105 of file MHV_io_ATmega168.h.

5.43.1.50 `#define MHV_PIN_TIMER_2_B MHV_PIN_D3`

Definition at line 106 of file MHV_io_ATmega168.h.

5.43.1.51 `#define MHV_TIMER0_INTERRUPTS TIMER0_COMPA_vect, TIMER0_COMPB_vect, 0`

Definition at line 39 of file MHV_io_ATmega168.h.

5.43.1.52 `#define MHV_TIMER16_1 &TCCR1A, &TCCR1B, 0, &OCR1A, &OCR1B, 0, &TCNT1,
&TIMSK1, &ICR1`

Definition at line 44 of file MHV_io_ATmega168.h.

5.43.1.53 `#define MHV_TIMER1_INTERRUPTS TIMER1_COMPA_vect, TIMER1_COMPB_vect, 0`

Definition at line 46 of file MHV_io_ATmega168.h.

5.43.1.54 `#define MHV_TIMER2_INTERRUPTS TIMER2_COMPA_vect, TIMER2_COMPB_vect, 0`

Definition at line 40 of file MHV_io_ATmega168.h.

5.43.1.55 `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B, &OCR0A, &OCR0B, &TCNT0, &TIMSK0, OCIE0A`

Definition at line 36 of file MHV_io_ATmega168.h.

5.43.1.56 `#define MHV_TIMER8_2 MHV_TIMER_TYPE_7_PRESCALERS, &TCCR2A, &TCCR2B, &OCR2A, &OCR2B, &TCNT2, &TIMSK2, OCIE2A`

Definition at line 37 of file MHV_io_ATmega168.h.

5.43.1.57 `#define MHV_USART0 &UBRR0, &UCSR0A, &UCSR0B, &UDR0, RXEN0, TXEN0, RXCIE0, TXCIE0, UDRE0, U2X0`

Definition at line 51 of file MHV_io_ATmega168.h.

5.43.1.58 `#define MHV_USART0_INTERRUPTS USART_RX_vect, USART_TX_vect`

Definition at line 52 of file MHV_io_ATmega168.h.

5.44 A:/eclipse/mhplib/MHV_io_ATtiny2313.h File Reference

```
#include <avr/io.h>
```

Defines

- `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B, &OCR0A, &OCR0B, &TCNT0, &TIMSK, OCIE0A`
- `#define MHV_TIMER0_INTERRUPTS SIG_OUTPUT_COMPARE0A, SIG_OUTPUT_COMPARE0B`
- `#define MHV_TIMER16_1 &TCCR1A, &TCCR1B, 0, &OCR1A, &OCR1B, 0, &TCNT1, &TIMSK, &ICR1`
- `#define MHV_TIMER1_INTERRUPTS SIG_OUTPUT_COMPARE1A, SIG_OUTPUT_COMPARE1B`
- `#define MHV_USART0 &UBRRH, &UBRRL, &UCSRA, &UCSRB, &UDR, RXEN, TXEN, RXCIE, TXCIE, UDRE, U2X`
- `#define MHV_USART0_INTERRUPTS USART_RX_vect, USART_TX_vect`
- `#define MHV_PIN_A0 &DDRA, &PORTA, &PINA, 0, -1`
- `#define MHV_PIN_A1 &DDRA, &PORTA, &PINA, 1, -1`
- `#define MHV_PIN_A2 &DDRA, &PORTA, &PINA, 2, -1`

- #define [MHV_PIN_B0](#) &DDRB, &PORTB, &PINB, 0, 0
- #define [MHV_PIN_B1](#) &DDRB, &PORTB, &PINB, 1, 1
- #define [MHV_PIN_B2](#) &DDRB, &PORTB, &PINB, 2, 2
- #define [MHV_PIN_B3](#) &DDRB, &PORTB, &PINB, 3, 3
- #define [MHV_PIN_B4](#) &DDRB, &PORTB, &PINB, 4, 4
- #define [MHV_PIN_B5](#) &DDRB, &PORTB, &PINB, 5, 5
- #define [MHV_PIN_B6](#) &DDRB, &PORTB, &PINB, 6, 6
- #define [MHV_PIN_B7](#) &DDRB, &PORTB, &PINB, 7, 7
- #define [MHV_PIN_D0](#) &DDRD, &PORTD, &PIND, 0, -1
- #define [MHV_PIN_D1](#) &DDRD, &PORTD, &PIND, 1, -1
- #define [MHV_PIN_D2](#) &DDRD, &PORTD, &PIND, 2, -1
- #define [MHV_PIN_D3](#) &DDRD, &PORTD, &PIND, 3, -1
- #define [MHV_PIN_D4](#) &DDRD, &PORTD, &PIND, 4, -1
- #define [MHV_PIN_D5](#) &DDRD, &PORTD, &PIND, 5, -1
- #define [MHV_PIN_D6](#) &DDRD, &PORTD, &PIND, 6, -1
- #define [MHV_PIN_TIMER_0_A](#) MHV_PIN_B2
- #define [MHV_PIN_TIMER_0_B](#) MHV_PIN_D5
- #define [MHV_PIN_TIMER_1_A](#) MHV_PIN_B3
- #define [MHV_PIN_TIMER_1_B](#) MHV_PIN_B4
- #define [MHV_INTERRUPT_INT0](#) INT0_vect, &MCUCR, ISC00
- #define [MHV_INTERRUPT_INT1](#) INT1_vect, &MCUCR, ISC10
- #define [MHV_PC_INT_COUNT](#) 8
- #define [MHV_EEPROM_VECT](#) EEPROM_READY_vect

5.44.1 Define Documentation

5.44.1.1 #define MHV_EEPROM_VECT EEPROM_READY_vect

Definition at line 78 of file MHV_io_ATtiny2313.h.

5.44.1.2 #define MHV_INTERRUPT_INT0 INT0_vect, &MCUCR, ISC00

Definition at line 73 of file MHV_io_ATtiny2313.h.

5.44.1.3 #define MHV_INTERRUPT_INT1 INT1_vect, &MCUCR, ISC10

Definition at line 74 of file MHV_io_ATtiny2313.h.

5.44.1.4 #define MHV_PC_INT_COUNT 8

Definition at line 76 of file MHV_io_ATtiny2313.h.

5.44.1.5 #define MHV_PIN_A0 &DDRA, &PORTA, &PINA, 0, -1

Definition at line 49 of file MHV_io_ATtiny2313.h.

5.44.1.6 #define MHV_PIN_A1 &DDRA, &PORTA, &PINA, 1, -1

Definition at line 50 of file MHV_io_ATtiny2313.h.

5.44.1.7 #define MHV_PIN_A2 &DDRA, &PORTA, &PINA, 2, -1

Definition at line 51 of file MHV_io_ATtiny2313.h.

5.44.1.8 #define MHV_PIN_B0 &DDRB, &PORTB, &PINB, 0, 0

Definition at line 52 of file MHV_io_ATtiny2313.h.

5.44.1.9 #define MHV_PIN_B1 &DDRB, &PORTB, &PINB, 1, 1

Definition at line 53 of file MHV_io_ATtiny2313.h.

5.44.1.10 #define MHV_PIN_B2 &DDRB, &PORTB, &PINB, 2, 2

Definition at line 54 of file MHV_io_ATtiny2313.h.

5.44.1.11 #define MHV_PIN_B3 &DDRB, &PORTB, &PINB, 3, 3

Definition at line 55 of file MHV_io_ATtiny2313.h.

5.44.1.12 #define MHV_PIN_B4 &DDRB, &PORTB, &PINB, 4, 4

Definition at line 56 of file MHV_io_ATtiny2313.h.

5.44.1.13 #define MHV_PIN_B5 &DDRB, &PORTB, &PINB, 5, 5

Definition at line 57 of file MHV_io_ATtiny2313.h.

5.44.1.14 #define MHV_PIN_B6 &DDRB, &PORTB, &PINB, 6, 6

Definition at line 58 of file MHV_io_ATtiny2313.h.

5.44.1.15 #define MHV_PIN_B7 &DDRB, &PORTB, &PINB, 7, 7

Definition at line 59 of file MHV_io_ATtiny2313.h.

5.44.1.16 `#define MHV_PIN_D0 &DDRD, &PORTD, &PIND, 0, -1`

Definition at line 60 of file MHV_io_ATtiny2313.h.

5.44.1.17 `#define MHV_PIN_D1 &DDRD, &PORTD, &PIND, 1, -1`

Definition at line 61 of file MHV_io_ATtiny2313.h.

5.44.1.18 `#define MHV_PIN_D2 &DDRD, &PORTD, &PIND, 2, -1`

Definition at line 62 of file MHV_io_ATtiny2313.h.

5.44.1.19 `#define MHV_PIN_D3 &DDRD, &PORTD, &PIND, 3, -1`

Definition at line 63 of file MHV_io_ATtiny2313.h.

5.44.1.20 `#define MHV_PIN_D4 &DDRD, &PORTD, &PIND, 4, -1`

Definition at line 64 of file MHV_io_ATtiny2313.h.

5.44.1.21 `#define MHV_PIN_D5 &DDRD, &PORTD, &PIND, 5, -1`

Definition at line 65 of file MHV_io_ATtiny2313.h.

5.44.1.22 `#define MHV_PIN_D6 &DDRD, &PORTD, &PIND, 6, -1`

Definition at line 66 of file MHV_io_ATtiny2313.h.

5.44.1.23 `#define MHV_PIN_TIMER_0_A MHV_PIN_B2`

Definition at line 68 of file MHV_io_ATtiny2313.h.

5.44.1.24 `#define MHV_PIN_TIMER_0_B MHV_PIN_D5`

Definition at line 69 of file MHV_io_ATtiny2313.h.

5.44.1.25 `#define MHV_PIN_TIMER_1_A MHV_PIN_B3`

Definition at line 70 of file MHV_io_ATtiny2313.h.

5.44.1.26 `#define MHV_PIN_TIMER_1_B MHV_PIN_B4`

Definition at line 71 of file MHV_io_ATtiny2313.h.

5.44.1.27 `#define MHV_TIMER0_INTERRUPTS SIG_OUTPUT_COMPARE0A,
SIG_OUTPUT_COMPARE0B`

Definition at line 38 of file MHV_io_ATtiny2313.h.

5.44.1.28 `#define MHV_TIMER16_1 &TCCR1A, &TCCR1B, 0, &OCR1A, &OCR1B, 0, &TCNT1,
&TIMSK, &ICR1`

Definition at line 42 of file MHV_io_ATtiny2313.h.

5.44.1.29 `#define MHV_TIMER1_INTERRUPTS SIG_OUTPUT_COMPARE1A,
SIG_OUTPUT_COMPARE1B`

Definition at line 44 of file MHV_io_ATtiny2313.h.

5.44.1.30 `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B,
&OCR0A, &OCR0B, &TCNT0, &TIMSK, OCIE0A`

Definition at line 36 of file MHV_io_ATtiny2313.h.

5.44.1.31 `#define MHV_USART0 &UBRRH, &UBRRL, &UCSRA, &UCSRB, &UDR, RXEN, TXEN,
RXCIE, TXCIE, UDRE, U2X`

Definition at line 45 of file MHV_io_ATtiny2313.h.

5.44.1.32 `#define MHV_USART0_INTERRUPTS USART_RX_vect, USART_TX_vect`

Definition at line 46 of file MHV_io_ATtiny2313.h.

5.45 A:/eclipse/mhplib/MHV_io_ATtiny85.h File Reference

```
#include <avr/io.h>
```

Defines

- `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &-
TCCR0B, &OCR0A, &OCR0B, &TCNT0, &TIMSK, OCIE0A`

- #define [MHV_TIMER8_1](#) MHV_TIMER_TYPE_5_PRESCALERS, &TCCR1A, &TCCR1B, &OCR1A, &OCR1B, &TCNT1, &TIMSK, OCIE1A
- #define [MHV_TIMER0_INTERRUPTS](#) SIG_OUTPUT_COMPARE0A, SIG_OUTPUT_COMPARE0B, 0
- #define [MHV_TIMER1_INTERRUPTS](#) SIG_OUTPUT_COMPARE1A, SIG_OUTPUT_COMPARE1B, 0
- #define [MHV_AD_RESOLUTION](#) 1024
- #define [MHV_AD_REFERENCE_VCC](#) (uint8_t)(0x00 << 4)
- #define [MHV_AD_REFERENCE_AREF](#) (uint8_t)(0x04 << 4)
- #define [MHV_AD_REFERENCE_1V1](#) (uint8_t)(0x08 << 6)
- #define [MHV_AD_REFERENCE_2V56](#) (uint8_t)(0x09 << 6)
- #define [MHV_AD_REFERENCE_2V56_AREF](#) (uint8_t)(0x0d << 6)
- #define [MHV_AD_CHANNEL_0](#) 0x00
- #define [MHV_AD_CHANNEL_1](#) 0x01
- #define [MHV_AD_CHANNEL_2](#) 0x02
- #define [MHV_AD_CHANNEL_3](#) 0x03
- #define [MHV_AD_CHANNEL_2_X1_2](#) 0x04
- #define [MHV_AD_CHANNEL_2_X20_2](#) 0x05
- #define [MHV_AD_CHANNEL_2_X1_3](#) 0x06
- #define [MHV_AD_CHANNEL_2_X20_3](#) 0x07
- #define [MHV_AD_CHANNEL_0_X1_0](#) 0x08
- #define [MHV_AD_CHANNEL_0_X20_0](#) 0x09
- #define [MHV_AD_CHANNEL_0_X1_1](#) 0x0a
- #define [MHV_AD_CHANNEL_0_X20_1](#) 0x0b
- #define [MHV_AD_V_BANDGAP](#) 0x0c
- #define [MHV_AD_0V](#) 0x0d
- #define [MHV_AD_TEMPERATURE](#) 0x0f
- #define [MHV_AD_PRR](#) PRR
- #define [MHV_PIN_B0](#) &DDRB, &PORTB, &PINB, 0, 0
- #define [MHV_PIN_B1](#) &DDRB, &PORTB, &PINB, 1, 1
- #define [MHV_PIN_B2](#) &DDRB, &PORTB, &PINB, 2, 2
- #define [MHV_PIN_B3](#) &DDRB, &PORTB, &PINB, 3, 3
- #define [MHV_PIN_B4](#) &DDRB, &PORTB, &PINB, 4, 4
- #define [MHV_PIN_B5](#) &DDRB, &PORTB, &PINB, 5, 5
- #define [MHV_PIN_TIMER_0_A](#) MHV_PIN_B0
- #define [MHV_PIN_TIMER_0_B](#) MHV_PIN_B1
- #define [MHV_PIN_TIMER_1_A](#) MHV_PIN_B1
- #define [MHV_PIN_TIMER_1_B](#) MHV_PIN_B4
- #define [MHV_INTERRUPT_INT0](#) INT0_vect, &MCUCR, ISC00
- #define [MHV_PC_INT_COUNT](#) 6
- #define [MHV_EEPROM_VECT](#) EE_RDY_vect

5.45.1 Define Documentation

5.45.1.1 #define MHV_AD_0V 0x0d

Definition at line 63 of file MHV_io_ATtiny85.h.

5.45.1.2 `#define MHV_AD_CHANNEL_0 0x00`

Definition at line 50 of file MHV_io_ATtiny85.h.

5.45.1.3 `#define MHV_AD_CHANNEL_0_X1_0 0x08`

Definition at line 58 of file MHV_io_ATtiny85.h.

5.45.1.4 `#define MHV_AD_CHANNEL_0_X1_1 0x0a`

Definition at line 60 of file MHV_io_ATtiny85.h.

5.45.1.5 `#define MHV_AD_CHANNEL_0_X20_0 0x09`

Definition at line 59 of file MHV_io_ATtiny85.h.

5.45.1.6 `#define MHV_AD_CHANNEL_0_X20_1 0x0b`

Definition at line 61 of file MHV_io_ATtiny85.h.

5.45.1.7 `#define MHV_AD_CHANNEL_1 0x01`

Definition at line 51 of file MHV_io_ATtiny85.h.

5.45.1.8 `#define MHV_AD_CHANNEL_2 0x02`

Definition at line 52 of file MHV_io_ATtiny85.h.

5.45.1.9 `#define MHV_AD_CHANNEL_2_X1_2 0x04`

Definition at line 54 of file MHV_io_ATtiny85.h.

5.45.1.10 `#define MHV_AD_CHANNEL_2_X1_3 0x06`

Definition at line 56 of file MHV_io_ATtiny85.h.

5.45.1.11 `#define MHV_AD_CHANNEL_2_X20_2 0x05`

Definition at line 55 of file MHV_io_ATtiny85.h.

5.45.1.12 `#define MHV_AD_CHANNEL_2_X20_3 0x07`

Definition at line 57 of file MHV_io_ATtiny85.h.

5.45.1.13 `#define MHV_AD_CHANNEL_3 0x03`

Definition at line 53 of file MHV_io_ATtiny85.h.

5.45.1.14 `#define MHV_AD_PRR PRR`

Definition at line 67 of file MHV_io_ATtiny85.h.

5.45.1.15 `#define MHV_AD_REFERENCE_1V1 (uint8_t)(0x08 << 6)`

Definition at line 46 of file MHV_io_ATtiny85.h.

5.45.1.16 `#define MHV_AD_REFERENCE_2V56 (uint8_t)(0x09 << 6)`

Definition at line 47 of file MHV_io_ATtiny85.h.

5.45.1.17 `#define MHV_AD_REFERENCE_2V56_AREF (uint8_t)(0x0d << 6)`

Definition at line 48 of file MHV_io_ATtiny85.h.

5.45.1.18 `#define MHV_AD_REFERENCE_AREF (uint8_t)(0x04 << 4)`

Definition at line 45 of file MHV_io_ATtiny85.h.

5.45.1.19 `#define MHV_AD_REFERENCE_VCC (uint8_t)(0x00 << 4)`

Definition at line 44 of file MHV_io_ATtiny85.h.

5.45.1.20 `#define MHV_AD_RESOLUTION 1024`

Definition at line 42 of file MHV_io_ATtiny85.h.

5.45.1.21 `#define MHV_AD_TEMPERATURE 0x0f`

Definition at line 64 of file MHV_io_ATtiny85.h.

5.45.1.22 `#define MHV_AD_V_BANDGAP 0x0c`

Definition at line 62 of file MHV_io_ATtiny85.h.

5.45.1.23 `#define MHV_EEPROM_VECT EE_RDY_vect`

Definition at line 87 of file MHV_io_ATtiny85.h.

5.45.1.24 `#define MHV_INTERRUPT_INT0 INT0_vect, &MCUCR, ISC00`

Definition at line 83 of file MHV_io_ATtiny85.h.

5.45.1.25 `#define MHV_PC_INT_COUNT 6`

Definition at line 85 of file MHV_io_ATtiny85.h.

5.45.1.26 `#define MHV_PIN_B0 &DDRB, &PORTB, &PINB, 0, 0`

Definition at line 71 of file MHV_io_ATtiny85.h.

5.45.1.27 `#define MHV_PIN_B1 &DDRB, &PORTB, &PINB, 1, 1`

Definition at line 72 of file MHV_io_ATtiny85.h.

5.45.1.28 `#define MHV_PIN_B2 &DDRB, &PORTB, &PINB, 2, 2`

Definition at line 73 of file MHV_io_ATtiny85.h.

5.45.1.29 `#define MHV_PIN_B3 &DDRB, &PORTB, &PINB, 3, 3`

Definition at line 74 of file MHV_io_ATtiny85.h.

5.45.1.30 `#define MHV_PIN_B4 &DDRB, &PORTB, &PINB, 4, 4`

Definition at line 75 of file MHV_io_ATtiny85.h.

5.45.1.31 `#define MHV_PIN_B5 &DDRB, &PORTB, &PINB, 5, 5`

Definition at line 76 of file MHV_io_ATtiny85.h.

5.45.1.32 `#define MHV_PIN_TIMER_0_A MHV_PIN_B0`

Definition at line 78 of file MHV_io_ATtiny85.h.

5.45.1.33 `#define MHV_PIN_TIMER_0_B MHV_PIN_B1`

Definition at line 79 of file MHV_io_ATtiny85.h.

5.45.1.34 `#define MHV_PIN_TIMER_1_A MHV_PIN_B1`

Definition at line 80 of file MHV_io_ATtiny85.h.

5.45.1.35 `#define MHV_PIN_TIMER_1_B MHV_PIN_B4`

Definition at line 81 of file MHV_io_ATtiny85.h.

5.45.1.36 `#define MHV_TIMER0_INTERRUPTS SIG_OUTPUT_COMPARE0A,
SIG_OUTPUT_COMPARE0B, 0`

Definition at line 39 of file MHV_io_ATtiny85.h.

5.45.1.37 `#define MHV_TIMER1_INTERRUPTS SIG_OUTPUT_COMPARE1A,
SIG_OUTPUT_COMPARE1B, 0`

Definition at line 40 of file MHV_io_ATtiny85.h.

5.45.1.38 `#define MHV_TIMER8_0 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR0A, &TCCR0B,
&OCR0A, &OCR0B, &TCNT0, &TIMSK, OCIE0A`

Definition at line 36 of file MHV_io_ATtiny85.h.

5.45.1.39 `#define MHV_TIMER8_1 MHV_TIMER_TYPE_5_PRESCALERS, &TCCR1A, &TCCR1B,
&OCR1A, &OCR1B, &TCNT1, &TIMSK, OCIE1A`

Definition at line 37 of file MHV_io_ATtiny85.h.

5.46 A:/eclipse/mhvlb/MHV_Lock.cpp File Reference

```
#include "MHV_Lock.h"
```


5.47 A:/eclipse/mhvlb/MHV_Lock.h File Reference

```
#include <util/atomic.h>
```

Classes

- class [MHV_Lock](#)

5.48 A:/eclipse/mhvlb/MHV_PID.cpp File Reference

```
#include <MHV_PID.h>
```

5.49 A:/eclipse/mhvlb/MHV_PID.h File Reference

```
#include <MHV_io.h>
```

Classes

- class [MHV_PID](#)

5.50 A:/eclipse/mhvlb/MHV_PinChangeManager.cpp File Reference

```
#include <MHV_PinChangeManager.h>
```

5.51 A:/eclipse/mhvlb/MHV_PinChangeManager.h File Reference

```
#include <MHV_io.h> #include <MHV_Device_RX.h>
```

Classes

- class [MHV_PinEventListener](#)
- struct [mhv_eventPin](#)
- class [MHV_PinChangeManager](#)

Defines

- #define [MHV_PINCHANGE_MANAGER_ASSIGN_PCINT](#)(__mhvEventManager)

- `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT0(__mhvEventManager)`
- `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT1(__mhvEventManager)`
- `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT2(__mhvEventManager)`
- `#define MHV_PINCHANGE_MANAGER_ASSIGN_INTERRUPTS(__mhvEventManager) MHV_PINCHANGE_MANAGER_ASSIGN_PCINT(__mhvEventManager)`

Typedefs

- typedef struct `mhv_eventPin` `MHV_EVENT_PIN`

5.51.1 Define Documentation

5.51.1.1 `#define MHV_PINCHANGE_MANAGER_ASSIGN_INTERRUPTS(__mhvEventManager) MHV_PINCHANGE_MANAGER_ASSIGN_PCINT(__mhvEventManager)`

Definition at line 67 of file `MHV_PinChangeManager.h`.

5.51.1.2 `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT(__mhvEventManager)`

Value:

```
ISR(PCINT_vect) { \
    __mhvEventManager.pinChange0(); \
}
```

Definition at line 34 of file `MHV_PinChangeManager.h`.

5.51.1.3 `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT0(__mhvEventManager)`

Value:

```
ISR(PCINT0_vect) { \
    __mhvEventManager.pinChange0(); \
}
```

Definition at line 39 of file `MHV_PinChangeManager.h`.

5.51.1.4 `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT1(__mhvEventManager)`

Value:

```
ISR(PCINT1_vect) { \
    __mhvEventManager.pinChange1(); \
}
```

Definition at line 44 of file MHV_PinChangeManager.h.

5.51.1.5 `#define MHV_PINCHANGE_MANAGER_ASSIGN_PCINT2(__mhvEventManager)`

Value:

```
ISR(PCINT2_vect) { \
    __mhvEventManager.pinChange2(); \
}
```

Definition at line 49 of file MHV_PinChangeManager.h.

5.51.2 Typedef Documentation

5.51.2.1 `typedef struct mhv_eventPin MHV_EVENT_PIN`

Definition at line 87 of file MHV_PinChangeManager.h.

5.52 A:/eclipse/mhplib/MHV_PWMMatrix.cpp File Reference

```
#include "MHV_PWMMatrix.h" #include <string.h> #include
<math.h>
```

Defines

- `#define pixel(pixelCol, pixelRow) _frameBuffer[pixelRow * _colCount + pixelCol]`

5.52.1 Define Documentation

5.52.1.1 `#define pixel(pixelCol, pixelRow) _frameBuffer[pixelRow * _colCount + pixelCol]`

Definition at line 31 of file MHV_PWMMatrix.cpp.

5.53 A:/eclipse/mhplib/MHV_PWMMatrix.h File Reference

```
#include <inttypes.h> #include <MHV_Display_Monochrome_-
Buffered.h>
```

Classes

- class [MHV_PWMMatrix](#)

Typedefs

- typedef enum [MHV_PWMMatrix_Mode](#) [MHV_PWMMATRIX_MODE](#)

Enumerations

- enum [MHV_PWMMatrix_Mode](#) { [MHV_PWMMATRIX_MODE_AUTO](#), [MHV_PWMMATRIX_MODE_ROWS](#), [MHV_PWMMATRIX_MODE_COLS](#), [MHV_PWMMATRIX_MODE_INDIVIDUAL](#) }

5.53.1 Typedef Documentation

5.53.1.1 typedef enum [MHV_PWMMatrix_Mode](#) [MHV_PWMMATRIX_MODE](#)

Definition at line 39 of file [MHV_PWMMatrix.h](#).

5.53.2 Enumeration Type Documentation

5.53.2.1 enum [MHV_PWMMatrix_Mode](#)

Enumerator:

[MHV_PWMMATRIX_MODE_AUTO](#)
[MHV_PWMMATRIX_MODE_ROWS](#)
[MHV_PWMMATRIX_MODE_COLS](#)
[MHV_PWMMATRIX_MODE_INDIVIDUAL](#)

Definition at line 33 of file [MHV_PWMMatrix.h](#).

5.54 A:/eclipse/mhvlb/MHV_RingBuffer.cpp File Reference

```
#include "MHV_RingBuffer.h"
```

5.55 A:/eclipse/mhvlb/MHV_RingBuffer.h File Reference

```
#include <inttypes.h> #include <MHV_io.h>
```

Classes

- class [MHV_RingBuffer](#)

5.56 A:/eclipse/mhvlb/MHV_RTC.cpp File Reference

```
#include <MHV_RTC.h>  #include <avr/pgmspace.h>  #include
<string.h>
```

Functions

- bool [mhv_timestampLessThan](#) (MHV_TIMESTAMP *first, MHV_TIMESTAMP *second)
- bool [mhv_timestampGreaterThanOrEqual](#) (MHV_TIMESTAMP *first, MHV_TIMESTAMP *second)
- bool [mhv_isLeapYear](#) (uint16_t year)
- void [mhv_timestampIncrement](#) (MHV_TIMESTAMP *timestamp, uint32_t seconds, uint16_t milliseconds)
- void [mhv_timestampIncrement](#) (MHV_TIMESTAMP *timestamp, MHV_TIMESTAMP *timestamp2)
- uint8_t [mhv_daysInMonth](#) (MHV_MONTH month, uint16_t year)

Variables

- const uint8_t [mhv_daysInMonthArray](#)[] [PROGMEM](#)

5.56.1 Function Documentation

5.56.1.1 uint8_t [mhv_daysInMonth](#) (MHV_MONTH *month*, uint16_t *year*)

Get the number of days in a month

Parameters

<i>month</i>	the month
<i>year</i>	the year

Returns

the number of days

Definition at line 151 of file MHV_RTC.cpp.

5.56.1.2 bool [mhv_isLeapYear](#) (uint16_t *year*)

Determine if a year is a leap year

Definition at line 89 of file MHV_RTC.cpp.

5.56.1.3 **bool** mhv_timestampGreaterThanOrEqual (**MHV_TIMESTAMP** * *first*,
MHV_TIMESTAMP * *second*)

Compare 2 timestamps

Parameters

<i>first</i>	the first timestamp
<i>second</i>	the second timestamp

Returns

true if the first timestamp is greater than or equal to the second

Definition at line 72 of file MHV_RTC.cpp.

5.56.1.4 **void** mhv_timestampIncrement (**MHV_TIMESTAMP** * *timestamp*, **uint32_t** *seconds*,
uint16_t *milliseconds*)

Increment a timestamp

Parameters

<i>timestamp</i>	the timestamp to increment
<i>seconds</i>	the number of seconds to increment by
<i>milliseconds</i>	the number of milliseconds to increment by

Definition at line 102 of file MHV_RTC.cpp.

5.56.1.5 **void** mhv_timestampIncrement (**MHV_TIMESTAMP** * *timestamp*,
MHV_TIMESTAMP * *timestamp2*)

Increment a timestamp

Parameters

<i>timestamp</i>	the timestamp to increment
<i>timestamp2</i>	the timestamp to increment by

Definition at line 118 of file MHV_RTC.cpp.

5.56.1.6 **bool** mhv_timestampLessThan (**MHV_TIMESTAMP** * *first*, **MHV_TIMESTAMP** *
second)

A Realtime clock

Takes a trigger from a timer, and keeps time.

Recommended values: Clock Prescaler Top Ticks/ms 16MHz 64 249 1 Compare 2 timestamps

Parameters

<i>first</i>	the first timestamp
<i>second</i>	the second timestamp

Returns

true if the first timestamp is less than the second

Definition at line 52 of file MHV_RTC.cpp.

5.56.2 Variable Documentation

5.56.2.1 `const uint32_t mhv_secondsFromYearStart []` **PROGMEM**

Initial value:

```
{
    31,
    28,
    31,
    30,
    31,
    30,
    31,
    30,
    30,
    30,
    31,
    30,
    31
}
```

Definition at line 130 of file MHV_RTC.cpp.

5.57 A:/eclipse/mhvlb/MHV_RTC.h File Reference

```
#include <MHV_Timer8.h>
```

Classes

- struct [mhv_timestamp](#)
- struct [mhv_time](#)
- struct [mhv_alarm](#)
- class [MHV_AlarmListener](#)
- class [MHV_RTC](#)

Typedefs

- typedef struct [mhv_timestamp](#) MHV_TIMESTAMP
- typedef enum [mhv_weekday](#) MHV_WEEKDAY
- typedef enum [mhv_month](#) MHV_MONTH
- typedef struct [mhv_time](#) MHV_TIME
- typedef struct [mhv_alarm](#) MHV_ALARM

Enumerations

- enum [mhv_weekday](#) { [MHV_SUNDAY](#), [MHV_MONDAY](#), [MHV_TUESDAY](#), [MHV_WEDNESDAY](#), [MHV_THURSDAY](#), [MHV_FRIDAY](#), [MHV_SATURDAY](#) }
- enum [mhv_month](#) { [MHV_JANUARY](#) = 1, [MHV_FEBRUARY](#) = 2, [MHV_MARCH](#) = 3, [MHV_APRIL](#) = 4, [MHV_MAY](#) = 5, [MHV_JUNE](#) = 6, [MHV_JULY](#) = 7, [MHV_AUGUST](#) = 8, [MHV_SEPTEMBER](#) = 9, [MHV_OCTOBER](#) = 10, [MHV_NOVEMBER](#) = 11, [MHV_DECEMBER](#) = 12 }

Functions

- void [mhv_timestampIncrement](#) (MHV_TIMESTAMP *timestamp, uint32_t seconds, uint16_t milliseconds)
- void [mhv_timestampIncrement](#) (MHV_TIMESTAMP *timestamp, MHV_TIMESTAMP *timestamp2)
- bool [mhv_isLeapYear](#) (uint16_t year)
- bool [mhv_timestampGreaterThanOrEqual](#) (MHV_TIMESTAMP *first, MHV_TIMESTAMP *second)
- bool [mhv_timestampLessThan](#) (MHV_TIMESTAMP *first, MHV_TIMESTAMP *second)
- uint8_t [mhv_daysInMonth](#) (MHV_MONTH month, uint16_t year)

5.57.1 Typedef Documentation

5.57.1.1 typedef struct [mhv_alarm](#) MHV_ALARM

Definition at line 87 of file MHV_RTC.h.

5.57.1.2 typedef enum [mhv_month](#) MHV_MONTH

Definition at line 64 of file MHV_RTC.h.

5.57.1.3 typedef struct [mhv_time](#) MHV_TIME

Definition at line 79 of file MHV_RTC.h.

5.57.1.4 typedef struct mhv_timestamp MHV_TIMESTAMP

Definition at line 36 of file MHV_RTC.h.

5.57.1.5 typedef enum mhv_weekday MHV_WEEKDAY

Definition at line 48 of file MHV_RTC.h.

5.57.2 Enumeration Type Documentation

5.57.2.1 enum mhv_month

Enumerator:

MHV_JANUARY
MHV_FEBRUARY
MHV_MARCH
MHV_APRIL
MHV_MAY
MHV_JUNE
MHV_JULY
MHV_AUGUST
MHV_SEPTEMBER
MHV_OCTOBER
MHV_NOVEMBER
MHV_DECEMBER

Definition at line 50 of file MHV_RTC.h.

5.57.2.2 enum mhv_weekday

Enumerator:

MHV_SUNDAY
MHV_MONDAY
MHV_TUESDAY
MHV_WEDNESDAY
MHV_THURSDAY
MHV_FRIDAY
MHV_SATURDAY

Definition at line 39 of file MHV_RTC.h.

5.57.3 Function Documentation

5.57.3.1 `uint8_t mhv_daysInMonth (MHV_MONTH month, uint16_t year)`

Get the number of days in a month

Parameters

<i>month</i>	the month
<i>year</i>	the year

Returns

the number of days

Definition at line 151 of file MHV_RTC.cpp.

5.57.3.2 `bool mhv_isLeapYear (uint16_t year)`

Determine if a year is a leap year

Definition at line 89 of file MHV_RTC.cpp.

5.57.3.3 `bool mhv_timestampGreaterThanOrEqual (MHV_TIMESTAMP * first, MHV_TIMESTAMP * second)`

Compare 2 timestamps

Parameters

<i>first</i>	the first timestamp
<i>second</i>	the second timestamp

Returns

true if the first timestamp is greater than or equal to the second

Definition at line 72 of file MHV_RTC.cpp.

5.57.3.4 `void mhv_timestampIncrement (MHV_TIMESTAMP * timestamp, uint32_t seconds, uint16_t milliseconds)`

Increment a timestamp

Parameters

<i>timestamp</i>	the timestamp to increment
<i>seconds</i>	the number of seconds to increment by
<i>milliseconds</i>	the number of milliseconds to increment by

Definition at line 102 of file MHV_RTC.cpp.

5.57.3.5 void mhv_timestampIncrement (MHV_TIMESTAMP * *timestamp*,
MHV_TIMESTAMP * *timestamp2*)

Increment a timestamp

Parameters

<i>timestamp</i>	the timestamp to increment
<i>timestamp2</i>	the timestamp to increment by

Definition at line 118 of file MHV_RTC.cpp.

5.57.3.6 bool mhv_timestampLessThan (MHV_TIMESTAMP * *first*, MHV_TIMESTAMP *
second)

A Realtime clock

Takes a trigger from a timer, and keeps time.

Recommended values: Clock Prescaler Top Ticks/ms 16MHz 64 249 1 Compare 2 timestamps

Parameters

<i>first</i>	the first timestamp
<i>second</i>	the second timestamp

Returns

true if the first timestamp is less than the second

Definition at line 52 of file MHV_RTC.cpp.

5.58 A:/eclipse/mhvlb/MHV_ServoControl.cpp File Reference

```
#include <MHV_ServoControl.h>
```

5.59 A:/eclipse/mhvlb/MHV_ServoControl.h File Reference

```
#include <MHV_Timer16.h>
```

5.60 A:/eclipse/mhvlb/MHV_Shifter.cpp File Reference

```
#include "MHV_Shifter.h"
```

Functions

- void [mhv_shiftout_byte_lsb](#) (MHV_PIN *data, MHV_PIN *clock, uint8_t byte)
- void [mhv_shiftout_byte_msb](#) (MHV_PIN *data, MHV_PIN *clock, uint8_t byte)

5.60.1 Function Documentation

5.60.1.1 void [mhv_shiftout_byte_lsb](#) (MHV_PIN * *data*, MHV_PIN * *clock*, uint8_t *byte*)

Definition at line 29 of file MHV_Shifter.cpp.

5.60.1.2 void [mhv_shiftout_byte_msb](#) (MHV_PIN * *data*, MHV_PIN * *clock*, uint8_t *byte*)

Definition at line 43 of file MHV_Shifter.cpp.

5.61 A:/eclipse/mhvlb/MHV_Shifter.h File Reference

```
#include <MHV_io.h>
```

Defines

- #define [SHIFTOUT_BYTE_LOOP](#) (mhv_shift_i = 0; mhv_shift_i < 8; mhv_shift_i++)
- #define [MHV_BIT_1_BV](#)(0)
- #define [MHV_BIT_2_BV](#)(1)
- #define [MHV_BIT_3_BV](#)(2)
- #define [MHV_BIT_4_BV](#)(3)
- #define [MHV_BIT_5_BV](#)(4)
- #define [MHV_BIT_6_BV](#)(5)
- #define [MHV_BIT_7_BV](#)(6)
- #define [MHV_BIT_8_BV](#)(7)
- #define [MHV_SHIFTOUT_BYTE](#)(mhv_data)
- #define [MHV_SHIFTOUT_ARRAY_CLOCKED_RISING](#)(mhv_data, mhv_data-
Length)
- #define [MHV_SHIFTOUT_ARRAY_CLOCKED_FALLING](#)(mhv_data, mhv_data-
Length)

Functions

- void `mhv_shiftout_byte_lsb` (`MHV_PIN` *data, `MHV_PIN` *clock, uint8_t byte)
- void `mhv_shiftout_byte_msb` (`MHV_PIN` *data, `MHV_PIN` *clock, uint8_t byte)

5.61.1 Define Documentation

5.61.1.1 `#define MHV_BIT_1 _BV(0)`

Definition at line 54 of file `MHV_Shifter.h`.

5.61.1.2 `#define MHV_BIT_2 _BV(1)`

Definition at line 55 of file `MHV_Shifter.h`.

5.61.1.3 `#define MHV_BIT_3 _BV(2)`

Definition at line 56 of file `MHV_Shifter.h`.

5.61.1.4 `#define MHV_BIT_4 _BV(3)`

Definition at line 57 of file `MHV_Shifter.h`.

5.61.1.5 `#define MHV_BIT_5 _BV(4)`

Definition at line 58 of file `MHV_Shifter.h`.

5.61.1.6 `#define MHV_BIT_6 _BV(5)`

Definition at line 59 of file `MHV_Shifter.h`.

5.61.1.7 `#define MHV_BIT_7 _BV(6)`

Definition at line 60 of file `MHV_Shifter.h`.

5.61.1.8 `#define MHV_BIT_8 _BV(7)`

Definition at line 61 of file `MHV_Shifter.h`.

5.61.1.9 `#define MHV_SHIFTOUT_ARRAY_CLOCKED_FALLING(mhv_data, mhv_dataLength)`

Definition at line 180 of file `MHV_Shifter.h`.

5.61.1.10 `#define MHV_SHIFTOUT_ARRAY_CLOCKED_RISING(mhv_data, mhv_dataLength)`

Definition at line 94 of file MHV_Shifter.h.

5.61.1.11 `#define MHV_SHIFTOUT_BYTE(mhv_data)`

Value:

```
do { \
    int8_t          mhv_shift_i; \
    \
    for SHIFTOUT_BYTE_LOOP { \
        if ((mhv_data >> mhv_shift_i) & 0x01) { \
            mhv_pinOn(MHV_SHIFT_WRITEDATA); \
        } else { \
            mhv_pinOff(MHV_SHIFT_WRITEDATA); \
        } \
        mhv_pinOn(MHV_SHIFT_WRITECLOCK); \
        mhv_pinOff(MHV_SHIFT_WRITECLOCK); \
    } \
} while (0)
```

Definition at line 67 of file MHV_Shifter.h.

5.61.1.12 `#define SHIFTOUT_BYTE_LOOP (mhv_shift_i = 0; mhv_shift_i < 8; mhv_shift_i++)`

Definition at line 53 of file MHV_Shifter.h.

5.61.2 Function Documentation

5.61.2.1 `void mhv_shiftout_byte_lsb(MHV_PIN * data, MHV_PIN * clock, uint8_t byte)`

Definition at line 29 of file MHV_Shifter.cpp.

5.61.2.2 `void mhv_shiftout_byte_msb(MHV_PIN * data, MHV_PIN * clock, uint8_t byte)`

Definition at line 43 of file MHV_Shifter.cpp.

5.62 [A:/eclipse/mhvlb/MHV_SoftwareHBridge.cpp File Reference](#)

5.63 [A:/eclipse/mhvlb/MHV_SoftwareHBridge.h File Reference](#)

5.64 [A:/eclipse/mhvlb/MHV_Timer16.cpp File Reference](#)

```
#include <MHV_Timer16.h>          #include <avr/interrupt.h> ×
#include <util/atomic.h>
```

5.65 A:/eclipse/mhvlb/MHV_Timer16.h File Reference

```
#include <MHV_Timer8.h>
```

5.66 A:/eclipse/mhvlb/MHV_Timer8.cpp File Reference

```
#include "MHV_Timer8.h" #include "MHV_io.h" #include <avr/io.-  
h> #include <avr/interrupt.h>
```

5.67 A:/eclipse/mhvlb/MHV_Timer8.h File Reference

```
#include <inttypes.h> #include <avr/interrupt.h> #include  
<avr/io.h> #include <stdio.h> #include <MHV_io.h>
```

Classes

- class [MHV_Timer8](#)

Defines

- #define [MHV_TIMER_ASSIGN_1INTERRUPT](#)(mhvTimer, mhvTimerVectors) _
MHV_TIMER_ASSIGN_1INTERRUPT(mhvTimer, mhvTimerVectors)
- #define [_MHV_TIMER_ASSIGN_1INTERRUPT](#)(mhvTimer, mhvTimerVect1,
mhvTimerVect2, mhvTimerVect3)
- #define [MHV_TIMER_ASSIGN_2INTERRUPTS](#)(mhvTimer, mhvTimerVectors) -
_MHV_TIMER_ASSIGN_2INTERRUPTS(mhvTimer, mhvTimerVectors)
- #define [_MHV_TIMER_ASSIGN_2INTERRUPTS](#)(mhvTimer, mhvTimerVect1,
mhvTimerVect2, mhvTimerVect3)

Typedefs

- typedef enum [mhv_timer_mode](#) [MHV_TIMER_MODE](#)
- typedef enum [mhv_timer_type](#) [MHV_TIMER_TYPE](#)
- typedef enum [mhv_timer_prescaler](#) [MHV_TIMER_PRESCALER](#)
- typedef enum [mhv_timer_connect_type](#) [MHV_TIMER_CONNECT_TYPE](#)

Enumerations

- enum [mhv_timer_mode](#) { [MHV_TIMER_ONE_SHOT](#), [MHV_TIMER_REPETITIVE](#), [MHV_TIMER_8_PWM_PHASE_CORRECT_VAR_FREQ](#), [MHV_TIMER_8_PWM_PHASE_CORRECT_2_OUTPUT](#), [MHV_TIMER_8_PWM_FAST_VAR_FREQ](#), [MHV_TIMER_8_PWM_FAST_2_OUTPUT](#), [MHV_TIMER_16_PWM_FA](#)

- ```
ST, MHV_TIMER_16_PWM_PHASE_CORRECT, MHV_TIMER_16_PWM_PHASE_FREQ_CORRECT }
```
- enum `mhv_timer_type` { `MHV_TIMER_TYPE_5_PRESCALERS`, `MHV_TIMER_TYPE_7_PRESCALERS` }
  - enum `mhv_timer_prescaler` { `MHV_TIMER_PRESCALER_DISABLED` = 0, `MHV_TIMER_PRESCALER_5_1` = 1, `MHV_TIMER_PRESCALER_5_8` = 2, `MHV_TIMER_PRESCALER_5_64` = 3, `MHV_TIMER_PRESCALER_5_256` = 4, `MHV_TIMER_PRESCALER_5_1024` = 5, `MHV_TIMER_PRESCALER_5_EXT_RISE` = 6, `MHV_TIMER_PRESCALER_5_EXT_FALL` = 7, `MHV_TIMER_PRESCALER_7_1` = 1, `MHV_TIMER_PRESCALER_7_8` = 2, `MHV_TIMER_PRESCALER_7_32` = 3, `MHV_TIMER_PRESCALER_7_64` = 4, `MHV_TIMER_PRESCALER_7_128` = 5, `MHV_TIMER_PRESCALER_7_256` = 6, `MHV_TIMER_PRESCALER_7_1024` = 7 }
  - enum `mhv_timer_connect_type` { `MHV_TIMER_CONNECT_DISCONNECTED` = 0, `MHV_TIMER_CONNECT_TOGGLE` = 1, `MHV_TIMER_CONNECT_CLEAR` = 2, `MHV_TIMER_CONNECT_SET` = 3 }

### 5.67.1 Define Documentation

5.67.1.1 `#define _MHV_TIMER_ASSIGN_1INTERRUPT( mhvTimer, mhvTimerVect1, mhvTimerVect2, mhvTimerVect3 )`

#### Value:

```
ISR(mhvTimerVect1) { \
 mhvTimer.trigger1(); \
}
```

Definition at line 86 of file `MHV_Timer8.h`.

5.67.1.2 `#define _MHV_TIMER_ASSIGN_2INTERRUPTS( mhvTimer, mhvTimerVect1, mhvTimerVect2, mhvTimerVect3 )`

#### Value:

```
ISR(mhvTimerVect1) { \
 mhvTimer.trigger1(); \
} \
ISR(mhvTimerVect2) { \
 mhvTimer.trigger2(); \
}
```

Definition at line 93 of file `MHV_Timer8.h`.

5.67.1.3 `#define MHV_TIMER_ASSIGN_1INTERRUPT( mhvTimer, mhvTimerVectors ) _MHV_TIMER_ASSIGN_1INTERRUPT(mhvTimer, mhvTimerVectors)`

Definition at line 84 of file `MHV_Timer8.h`.



5.67.1.4 `#define MHV_TIMER_ASSIGN_2INTERRUPTS( mhvTimer, mhvTimerVectors  
 ) _MHV_TIMER_ASSIGN_2INTERRUPTS(mhvTimer, mhvTimerVectors)`

Definition at line 91 of file MHV\_Timer8.h.

## 5.67.2 Typedef Documentation

5.67.2.1 `typedef enum mhv_timer_connect_type MHV_TIMER_CONNECT_TYPE`

Definition at line 82 of file MHV\_Timer8.h.

5.67.2.2 `typedef enum mhv_timer_mode MHV_TIMER_MODE`

Definition at line 49 of file MHV\_Timer8.h.

5.67.2.3 `typedef enum mhv_timer_prescaler MHV_TIMER_PRESCALER`

Definition at line 74 of file MHV\_Timer8.h.

5.67.2.4 `typedef enum mhv_timer_type MHV_TIMER_TYPE`

Definition at line 55 of file MHV\_Timer8.h.

## 5.67.3 Enumeration Type Documentation

5.67.3.1 `enum mhv_timer_connect_type`

Enumerator:

***MHV\_TIMER\_CONNECT\_DISCONNECTED***  
***MHV\_TIMER\_CONNECT\_TOGGLE***  
***MHV\_TIMER\_CONNECT\_CLEAR***  
***MHV\_TIMER\_CONNECT\_SET***

Definition at line 76 of file MHV\_Timer8.h.

5.67.3.2 `enum mhv_timer_mode`

Enumerator:

***MHV\_TIMER\_ONE\_SHOT***  
***MHV\_TIMER\_REPETITIVE***  
***MHV\_TIMER\_8\_PWM\_PHASE\_CORRECT\_VAR\_FREQ***  
***MHV\_TIMER\_8\_PWM\_PHASE\_CORRECT\_2\_OUTPUT***

```

MHV_TIMER_8_PWM_FAST_VAR_FREQ
MHV_TIMER_8_PWM_FAST_2_OUTPUT
MHV_TIMER_16_PWM_FAST
MHV_TIMER_16_PWM_PHASE_CORRECT
MHV_TIMER_16_PWM_PHASE_FREQ_CORRECT

```

Definition at line 38 of file MHV\_Timer8.h.

#### 5.67.3.3 enum mhv\_timer\_prescaler

Enumerator:

```

MHV_TIMER_PRESCALER_DISABLED
MHV_TIMER_PRESCALER_5_1
MHV_TIMER_PRESCALER_5_8
MHV_TIMER_PRESCALER_5_64
MHV_TIMER_PRESCALER_5_256
MHV_TIMER_PRESCALER_5_1024
MHV_TIMER_PRESCALER_5_EXT_RISE
MHV_TIMER_PRESCALER_5_EXT_FALL
MHV_TIMER_PRESCALER_7_1
MHV_TIMER_PRESCALER_7_8
MHV_TIMER_PRESCALER_7_32
MHV_TIMER_PRESCALER_7_64
MHV_TIMER_PRESCALER_7_128
MHV_TIMER_PRESCALER_7_256
MHV_TIMER_PRESCALER_7_1024

```

Definition at line 57 of file MHV\_Timer8.h.

#### 5.67.3.4 enum mhv\_timer\_type

Enumerator:

```

MHV_TIMER_TYPE_5_PRESCALERS
MHV_TIMER_TYPE_7_PRESCALERS

```

Definition at line 51 of file MHV\_Timer8.h.

## 5.68 A:/eclipse/mhvlb/MHV\_VoltageRegulator.cpp File Reference

```

#include <MHV_io.h> #include <MHV_VoltageRegulator.h> ×
#include <MHV_AD.h>

```

## **5.69 A:/eclipse/mhplib/MHV\_VoltageRegulator.h File Reference**