



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



STLUX and STNRG digital controllers designed for lighting and power conversion

Introduction

This reference manual provides complete information for application developers on how to use the STLUX and STNRG family of digital controllers.

The STLUX™ family of controllers is a part of the STMicroelectronics® digital devices tailored for lighting applications. The STLUX controllers have been successfully integrated in a wide range of architectures and applications, starting from simple buck converters for driving multiple LED strings, boost for power factor corrections, half-bridge resonant converters for high power dimmable LED strings and up to full bridge controllers for HID lamp ballasts. STLUX natively supports the DALI via the internal DALI communication module (DCM). DALI is a serial communication standard used in the lighting industry.

STNRG devices are a part of the STNRG family of STMicroelectronics digital devices designed for advanced power conversion applications. The STNRG improves the design of the STLUX™ family to support industrial power conversion applications such as PFC+LLC, interleaved LC DC-DC, interleaved PFC for smart power supplies as well as the full bridge for pilot line drivers for electric vehicles.

The heart of the STLUX (and consequently STNRG where not differently specified) is the SMED (“State Machine, Event Driven”) technology which allows the device to operate several independently configurable PWM clocks with an up to 1.3 ns resolution. An SMED is a powerful autonomous state machine which is programmed to react to both external and internal events and may evolve without any software intervention. The SMED even reaction time can be as low as 10 ns, giving the STLUX the ability of operating in time critical applications.

The SMEDs are configured and programmed via the STLUX internal low-power microcontroller (STM8). The STM8 controller extends the STLUX reliability and guarantees more than 15 years of both operating lifetime and memory data retention to the program and data memory after cycling. The STM8 device also provides powerful processing while maintaining the advantages of the CISC architecture, a 24-bit linear addressing mode, an 8-bit data bus and an optimized architecture for low power applications.

The STLUX controller 1.8 V core voltage is provided by an internal power supply DC regulator in order to simplify and reduce the application design and cost.

The STLUX device has an embedded low power, low voltage, single voltage Flash program memory designed for reliability. The same technology provides an embedded true RWW data EEPROM dedicated area and dedicated 128 byte data used for the IC configuration.

This manual describes how to program the SMED and the STM8 companion controller.

Refer to the product datasheets for ordering information, pin description, mechanical and electrical device characteristics, and for specific peripherals and implementation details.

Contents

1	Reference documents	21
2	Acronyms	21
3	Register model	23
4	Description	24
4.1	STLUX/STNRG feature comparison	25
5	Central processing unit (CPU)	26
5.1	CPU introduction	26
5.2	CPU architecture overview	26
5.2.1	Architecture and registers	26
5.2.2	Addressing mode	26
5.2.3	Instruction set	26
5.3	CPU registers description	26
5.3.1	Accumulator (A)	27
5.3.2	Index registers (X and Y)	27
5.3.3	Program counter (PC)	27
5.3.4	Stack pointer (SP)	27
5.3.5	Condition code register (CC)	29
5.4	Core registers description	30
5.4.1	A (accumulator)	30
5.4.2	PCE (program counter)	31
5.4.3	PCH (program counter)	31
5.4.4	PCL (program counter)	31
5.4.5	XH (X-index)	31
5.4.6	XL (X-index)	32
5.4.7	YH (Y-index)	32
5.4.8	YL (Y-index)	32
5.4.9	SPH (stack pointer)	33
5.4.10	SPL (stack pointer)	33
5.4.11	CC (code condition)	33
5.5	CPU register map	34

5.6	Global configuration register description (CFG_GCR)	34
5.6.1	SWIM disable	34
5.6.2	Global configuration descriptions	34
6	Debug architecture	36
6.1	Introduction	36
6.2	Single wire interface module (SWIM)	36
6.3	SWIM main features	37
6.4	SWIM register description	37
	SWIM_CSR (SWIM control register)	37
6.5	SWIM register map	38
6.6	Debug module	38
6.7	DM main features	38
6.8	Internal registers description	39
6.8.1	DM_BK1E (DM breakpoint 1 ext. byte)	39
6.8.2	DM_BK1H (DM breakpoint 1 high byte)	39
6.8.3	DM_BK1L (DM breakpoint 1 low byte)	39
6.8.4	DM_BK2E (DM breakpoint 2 ext. byte)	39
6.8.5	DM_BK2H (DM breakpoint 2 high byte)	40
6.8.6	DM_BK2L (DM breakpoint 2 low byte)	40
6.8.7	DM_CR (DM control register)	40
6.8.8	DM_CR2 (DM control register 2)	41
6.8.9	DM_CSR (DM status/control register 1)	41
6.8.10	DM_CSR2 (DM status/control register 2)	43
6.8.11	DM_ENFCT (DM enable function register)	44
6.8.12	DM_VER (DM version register)	45
6.9	Internal registers overview	45
7	Boot ROM	46
8	Memory and register map	47
8.1	Memory layout	47
8.1.1	Memory map	47
8.1.2	Stack handling	48
9	Flash program memory and data EEPROM	50

9.1	Main Flash memory features	50
9.2	Flash memory organization	51
9.2.1	Memory access / wait state configuration	52
9.2.2	Program memory	52
9.2.3	Data memory	53
9.2.4	Option byte memory	53
9.3	Memory protection mechanism	53
9.3.1	Readout protection (ROP)	53
9.3.2	Memory access security system (MASS)	53
9.4	Memory programming	55
9.4.1	Read-while-write (RWW)	55
9.4.2	Programming modes	55
9.5	ICP and IAP	58
9.6	Flash registers description	60
9.6.1	FLASH_CR1 (Flash control register1)	60
9.6.2	FLASH_CR2 (Flash control register2)	60
9.6.3	FLASH_nCR2 (Flash control register2 protection)	61
9.6.4	FLASH_FPR (Flash write page protection)	61
9.6.5	FLASH_nFPRP (FLASH_FPR protection)	62
9.6.6	FLASH_IAPSR (Flash status register)	62
9.6.7	FLASH_PUKR (Flash program memory unprotection)	63
9.6.8	FLASH_DUKR (Flash data memory unprotection)	63
9.6.9	FLASH_WAIT (Flash wait state register)	64
9.7	Flash registers overview	65
10	Power supply	66
11	Reset control unit (RST)	67
11.1	Reset state and reset in progress definition	67
11.2	Reset circuit description	67
11.3	Internal reset sources	68
11.3.1	Power-on reset (POR) and brownout reset (BOR)	68
11.3.2	Watchdog reset	69
11.3.3	Software reset	69
11.3.4	SWIM reset	69
11.3.5	Illegal opcode reset	69

11.3.6	EMC reset	69
11.4	RST register description	69
	RST_SR (status register)	69
11.5	RST register overview	70
12	Clock control unit (CKC)	71
12.1	EMS - hardened clock configuration registers (optional)	71
12.2	Features overview	71
12.3	Master clock sources	71
12.3.1	HSI	72
12.3.2	LSI	72
12.3.3	PLL	73
12.3.4	HSE	74
12.3.5	Internal clock scheme	75
12.4	Oscillators control	77
12.4.1	Oscillator enable control logic	77
12.4.2	Oscillator startup	77
12.5	Master clock switch	77
12.5.1	Switch trigger	77
12.5.2	CPU clock during switch	78
12.5.3	Glitch filter	78
12.5.4	SWBSY utility	78
12.5.5	Switch enable	78
12.5.6	Automatic vs. manual switch	78
12.5.7	Interrupts/flags for application	81
12.5.8	Old CKM after switch	81
12.5.9	Wait mode synchronization	81
12.6	Peripheral clock gating	81
12.7	Clock dividers	82
12.7.1	PLL clock divider	82
12.7.2	ADC real-time clock divider	82
12.7.3	SMED real-time clock dividers	83
12.8	Clock security system (CSS)	84
12.9	Configurable clock output	85
12.10	CLK interrupts	87
12.11	Clock registers description	87

12.11.1	CLK_SMD0 (SMED0 clock configuration)	87
12.11.2	CLK_SMD1 (SMED1 clock configuration)	88
12.11.3	CLK_SMD2 (SMED2 clock configuration)	88
12.11.4	CLK_SMD3 (SMED3 clock configuration)	89
12.11.5	CLK_SMD4 (SMED4 clock configuration)	90
12.11.6	CLK_SMD5 (SMED5 clock configuration)	90
12.11.7	CLK_PLLDIVR (PLL prescaler configuration)	91
12.11.8	CLK_AWUDIVR (AWU clock prescaler configuration)	92
12.11.9	CLK_IICKR (internal clocks control)	92
12.11.10	CLK_EICKR (external clocks control)	94
12.11.11	CLK_PLLR (PLL configuration)	94
12.11.12	CLK_CMSR (clock master)	95
12.11.13	CLK_SWR (clock switch)	96
12.11.14	CLK_SWCR (switch control)	97
12.11.15	CLK_CKDIVR (clock dividers)	98
12.11.16	CLK_PCKENR1 (peripherals clock enable)	99
12.11.17	CLK_CSSR (clock security system)	100
12.11.18	CLK_CCOR (configurable clock output)	101
12.11.19	CLK_PCKENR2 (peripherals clock enable)	103
12.11.20	CLK_HSTRIMR (HSI calibration trimmer)	104
12.11.21	CLK_SWIMCCR (SWIM clock division)	104
12.11.22	CLK_CCODIVR (CCO divider)	104
12.11.23	CLK_ADCR (ADC clock configuration)	105
12.12	Clock registers overview	106
13	Power management (PM)	107
13.1	Overview	107
13.2	Clock switching reduction	107
13.2.1	Peripheral clock gating	107
13.2.2	Slow down clock	107
13.2.3	Analog peripherals power-down	108
13.3	Operating modes	108
13.3.1	RUN mode	110
13.3.2	WAIT mode	110
13.3.3	Active-halt mode	111
13.3.4	Halt mode	112

14	Interrupt controller (ITC)	113
14.1	Functional overview	113
14.2	Interrupt masking and processing flow	113
14.3	Interrupt priority	115
14.4	Servicing pending interrupts	116
14.5	Interrupt sources	116
14.5.1	Not-maskable interrupt sources	116
14.5.2	Maskable interrupt sources	117
14.6	Interrupts and low power modes	117
14.7	Activation level/low power mode control	118
14.8	Concurrent and nested interrupt management	118
14.8.1	Concurrent interrupt management mode	118
14.8.2	Nested interrupt management mode	119
14.9	External interrupts	120
14.9.1	Interrupt on port 3 (comparators)	121
14.9.2	Interrupt polling mode	121
14.9.3	Auxiliary/basic timers interrupt functionality	121
14.9.4	Edge detector configuration	122
14.10	Interrupt instructions	124
14.11	Interrupt controller registers description	124
14.11.1	ITC_SPR0 (interrupt SW priority 0)	124
14.11.2	ITC_SPR1 (interrupt SW priority 1)	125
14.11.3	ITC_SPR2 (interrupt SW priority 2)	125
14.11.4	ITC_SPR3 (interrupt SW priority 3)	125
14.11.5	ITC_SPR4 (interrupt SW priority 4)	126
14.11.6	ITC_SPR5 (interrupt SW priority 5)	126
14.11.7	ITC_SPR6 (interrupt SW priority 6)	126
14.11.8	ITC_SPR7 (interrupt SW priority 7)	127
14.12	Miscellaneous interrupt registers	127
14.12.1	MSC_CFGP<xy> (port P<x>[<y>] input control register)	127
14.12.2	MSC_STSP<x> (port P<x> interrupt status reg.; <x> = 0, 1, 2, 3)	128
14.12.3	MSC_INPP2AUX1 (INPP2 aux. register1)	129
14.12.4	MSC_INPP2AUX2 (INPP2 aux. register2)	130
14.12.5	MSC_INPP2 (Port2 input data register)	130
14.13	Interrupt controller registers overview	131

14.14	Miscellaneous interrupt registers overview	131
14.15	Interrupt exception vector table	132
15	Independent watchdog (IWDG)	133
15.1	IWDG introduction	133
15.2	IWDG functional description	133
15.2.1	IWDG HW feature	133
15.2.2	IWDG timeout period	134
15.3	IWDG program sequence	134
15.4	IWDG and AWU concurrent activation	135
15.5	Independent watchdog registers description	135
15.5.1	IWDG_KR (key register)	135
15.5.2	IWDG_PR (prescaler register)	136
15.5.3	IWDG_RLR (reload counter)	136
15.6	Watchdog registers overview	137
16	Window watchdog (WWDG)	138
16.1	WWDG introduction	138
16.2	WWDG main features	138
16.3	WWDG functional description	138
16.4	Watchdog reset on Halt option	139
16.5	How to program the watchdog timeout	140
16.6	WWDG low power modes	141
16.6.1	Hardware watchdog option	141
16.6.2	Using Halt mode with the WWDG (WWDG_HALT option byte)	141
16.7	Window watchdog registers description	141
16.7.1	WWDG_CR (control register)	141
16.7.2	WWDG_WR (window register)	142
16.8	Window watchdog registers overview	142
17	Auto-wakeup unit (AWU)	143
17.1	AWU functional description	143
17.2	Time base selection	143
17.3	Interval time formulas	144
17.4	AWU program sequence	145

17.5	Auto wake unit registers description	145
17.5.1	AWU_CSR (control status register)	145
17.5.2	AWU_APR (prescaler divider register)	146
17.5.3	AWU_TBR (time base register)	147
17.6	Auto wake unit registers overview	147
18	System timer (STMR)	148
18.1	STMR introduction	148
18.2	STMR main features	148
18.3	STMR timer overview	148
18.4	Configuration of the 16-bit counter	149
18.5	Write sequence for 16-bit STMR_ARR register	150
18.6	Prescaler	150
18.7	STMR counting mode	150
18.8	STMR program sequence	152
18.9	STMR interrupts	152
18.10	System timer as ADC trigger source	153
18.11	System timer unit registers description	153
18.11.1	STMR_CR1 (control register)	153
18.11.2	STMR_IER (interrupt enable register)	154
18.11.3	STMR_SR1 (status register 1)	154
18.11.4	STMR_EGR (event generation register)	155
18.11.5	STMR_CNTH (counter high register)	155
18.11.6	STMR_CNTL (counter low register)	155
18.11.7	STMR_PSCL (prescaler register)	155
18.11.8	STMR_ARRH (auto-reload high register)	156
18.11.9	STMR_ARRL (auto-reload low register)	156
18.12	System timer unit registers overview	157
19	Auxiliary timer	158
19.1	Auxiliary timer introduction	158
19.2	Auxiliary timer main features	158
19.3	Auxiliary timer functional description	158
19.4	Program sequence	159
19.5	Auxiliary timer unit register overview	159

20	Basic timer	160
20.1	Basic timer introduction	160
20.2	Basic timer main features	160
20.3	Basic timer functional description	160
20.4	Programming sequence	161
20.5	Basic timer control registers	162
20.5.1	MSC_FTM0CKSEL (BscTim10 clock selection)	162
20.5.2	MSC_FTM0CKDIV (BscTim0 clock division factor)	162
20.5.3	MSC_FTM0CONF (BscTim0 counter)	163
20.5.4	MSC_FTM1CKDIV (BscTim1 clock division factor)	163
20.5.5	MSC_FTM1CONF (BscTim0 counter)	163
20.6	Additional basic timer register overview	164
21	State machine, event driven (SMED)	165
21.1	SMED architecture overview	165
21.2	SMED main features	166
21.3	Block diagram	167
21.3.1	Subblocks descriptions	167
21.3.2	Clock domains	167
21.4	Processor interface	168
21.5	FSM controller	168
21.6	External events	169
21.6.1	Level/edge detection logic	169
21.6.2	External event handling	169
21.6.3	External input multiplexing	170
21.7	Internal events	170
21.8	Events priority	171
21.9	State transition	171
21.9.1	IDLE state	174
21.9.2	HOLD state	174
21.9.3	Sequential state transitions (S0→S3)	174
21.9.4	Non-sequential state transitions	175
21.9.5	State evolution equations	176
21.10	State transition occurrence	178
21.10.1	Output PWM level	179

21.10.2	Counter reset	179
21.10.3	Hold jump	180
21.10.4	Hold exit	181
21.11	Time control	182
21.11.1	State timer registers	182
21.11.2	Dithering	183
21.11.3	Time stamp functionality	185
21.12	Interrupt functionality	186
21.13	Status information	187
21.13.1	GSTS (global status register)	187
21.13.2	FSM_STS (finite state machine status register)	187
21.14	PWM reset output value	188
21.15	PWM pad (pseudo) open drain configuration	188
21.16	SMED as ADC trigger source	188
21.17	SMED coupled expansion interface	189
21.17.1	SMED coupled interface configuration scheme	189
21.17.2	Chapter assumptions:	190
21.17.3	SMED subsystem configuration	190
21.17.4	Unlock feature	196
21.17.5	Connection box	197
21.17.6	FSM diagnostic trace	199
21.18	SMED registers description	199
21.18.1	SMED core registers	199
21.18.2	SMED environment configuration registers	215
22	General purpose I/O port (GPIO)	221
22.1	GPIO main features	221
22.2	Port configuration and usage	221
22.2.1	Input modes	222
22.2.2	Output modes	222
22.2.3	Slope control	222
22.2.4	Reset configuration	223
22.2.5	Unused I/O pins	223
22.2.6	Low power modes	223
22.3	Alternate output function	223
22.4	Interrupt functionality	223

22.5	Interrupt masking	223
22.6	GPIO registers description	224
22.6.1	P<n>_ODR (port output data register)	224
22.6.2	P<n>_IDR (port input data register)	224
22.6.3	P<n>_DDR (port data direction register)	225
22.6.4	P<n>_CR1 (port control register1)	225
22.6.5	P<n>_CR2 (port control register2)	226
22.7	GPIO registers overview	227
23	Inter-integrated circuit interface (I²C)	228
23.1	I ² C introduction	228
23.2	I ² C main features	228
23.3	I ² C signal interface	229
23.4	I ² C general description	229
23.5	Communication flow	229
23.6	I ² C functional description	230
23.6.1	I ² C slave mode	230
23.6.2	I ² C master mode	233
23.6.3	Error conditions	241
23.6.4	SDA/SCL line control	242
23.7	Limitations on I ² C peripheral usage	242
23.7.1	I ² C event management	242
23.7.2	Corrupted last received data in master receiver mode	243
23.7.3	Wrong behavior of I ² C peripheral in master mode after misplaced STOP	243
23.7.4	Violation of I ² C “setup time for repeated START condition” parameter	244
23.7.5	In I ² C slave “NOSTRETCH” mode, underrun errors may not be detected and may generate bus errors	244
23.7.6	I ² C pulse missed	245
23.8	I ² C low power modes	246
23.9	I ² C interrupts	246
23.10	I ² C registers description	247
23.10.1	I2C_CR1 (control register1)	247
23.10.2	I2C_CR2 (control register2)	248
23.10.3	I2C_FREQR (frequency register)	249
23.10.4	I2C_OARL (own add. LSB register)	250

23.10.5	I2C_OARH (own add. MSB register)	250
23.10.6	I2C_DR (data register)	251
23.10.7	I2C_SR1 (status register1)	251
23.10.8	I2C_SR2 (status register2)	254
23.10.9	I2C_SR3 (status register3)	255
23.10.10	I2C_ITR (interrupt register)	256
23.10.11	I2C_CCRL (clock control low)	257
23.10.12	I2C_CCRH (clock control high)	258
23.10.13	I2C_TRISER (time rise register)	259
23.11	I ² C registers overview	260
24	Universal asynchronous receiver transmitter (UART)	261
24.1	UART main features	261
24.2	UART functional description	262
24.2.1	UART character description	262
24.2.2	UART transmitter	263
24.2.3	UART receiver	265
24.2.4	Parity control	270
24.2.5	Multi-processor communication	271
24.3	UART low power mode	273
24.4	UART interrupts	273
24.5	UART registers description	274
24.5.1	UART_SR (status register)	274
24.5.2	UART_DR (data register)	276
24.5.3	UART_BRR1 (baud rate 1 register)	276
24.5.4	UART_BRR2 (baud rate 2 register)	277
24.5.5	UART_CR1 (control register 1)	278
24.5.6	UART_CR2 (control register 2)	279
24.5.7	UART_CR3 (control register 3)	280
24.5.8	UART_CR4 (control register 4)	281
24.6	UART registers overview	281
25	Digital addressable lighting interface (DALI)	282
25.1	DALI main features	282
25.2	DALI block diagram	283
25.3	DALI functional overview	283

25.4	Monitor receiver line	285
25.5	DALI signal interface	285
25.5.1	Polarity configurable signal interface	285
25.5.2	Start bit handling	285
25.5.3	Digital noise rejection filter	285
25.6	DALI data rate	288
25.7	DALI low power mode	288
25.8	DALI interrupts	289
25.9	DALI programming sequences	289
25.9.1	DALI initialization procedure	289
25.9.2	Data interrupt handling	289
25.10	DALI registers description	291
25.10.1	DALI_CLK_L (clock prescaler LSB)	291
25.10.2	DALI_CLK_H (clock prescaler MSB)	291
25.10.3	DALI_FB0 [forward message register (7:0)]	291
25.10.4	DALI_FB1 [forward message register(15:8)]	292
25.10.5	DALI_FB2 [forward message register(23:16)]	292
25.10.6	DALI_BD [backward message register(7:0)]	292
25.10.7	DALI_CR (control register)	293
25.10.8	DALI_CSR (control and status register)	295
25.10.9	DALI_CSR1 (control and status register1)	296
25.10.10	DALI_REVLN (control reverse signal line)	296
25.11	DALI filter control registers	297
25.11.1	MSC_DALICKSEL (DALI filter clock selection)	297
25.11.2	MSC_DALICKDIV (DALI filter clock division factor)	297
25.11.3	MSC_DALICONF (DALI filter mode configuration)	298
25.12	DALI registers overview	299
26	Analog comparator unit (ACU)	300
26.1	Overview description	300
26.2	Comparator logic	302
26.2.1	Main features	302
26.2.2	DAC and comparator programming	302
26.2.3	Interrupt and wake up capability	303
26.2.4	Hysteresis programming	303
26.2.5	Processor interface to the comparator outputs	304

26.3	DAC converter logic	304
26.4	DAC main features	304
26.5	ACU programming sequence	305
26.6	ACU low power modes	306
26.7	ACU registers description	306
26.7.1	MSC_DACCTR (DAC comparator control register)	307
26.7.2	MSC_DACIN<n> (Dac<n> input data register)	309
26.7.3	MSC_INPP3 (port P3 input data register)	309
26.7.4	MSC_DAC<n>HYST (DAC <n>hysteresis selection)	310
26.8	ACU registers overview	311
27	Analog-to-digital converter (ADC)	312
27.1	ADC main features	312
27.2	ADC unit block diagram	312
27.3	ADC analog unit	313
27.3.1	ADC analog main features	314
27.4	ADC digital unit	314
27.4.1	ADC digital main features	315
27.4.2	The ADC sequencer	315
27.4.3	ADC conversion modes	315
27.4.4	Start the ADC conversion	318
27.4.5	Reading the conversion result	319
27.4.6	DELAY register	320
27.4.7	ADC clock selection	320
27.4.8	ADC timing	320
27.4.9	ADC HW trigger latency	322
27.5	Interrupts	322
27.6	ADC programming sequences	322
27.6.1	Single mode programming sequence	323
27.6.2	Continuous (circular buffer) programming sequence	324
27.6.3	Abort sequence	324
27.6.4	ADC power-down sequence	325
27.7	ADC low power modes	325
27.8	ADC register description	326
27.8.1	ADC_CFG (configuration register)	326
27.8.2	ADC_SOC (start of conversion)	327

27.8.3	ADC_IER (interrupt enable register)	327
27.8.4	ADC_SEQ (sequencer register)	328
27.8.5	ADC_DATL_<n> (data low register right alignment)	328
27.8.6	ADC_DATH_<n> (data high register right alignment)	329
27.8.7	ADC_DATL_<n> (data low register left alignment)	329
27.8.8	ADC_DATH_<n> (data high register left alignment)	329
27.8.9	ADC_SR (status register)	330
27.8.10	ADC_DLYCNT (SOC delay counter register)	330
27.9	ADC HW trigger configuration registers description	331
27.9.1	MSC_FTM0CONF (AddTimer0 mode configuration)	331
27.9.2	MSC_DALICKSEL (DALI filter clock selection)	331
27.9.3	MSC_INPP2AUX1 (INPP2 aux register 1)	332
27.10	ADC registers overview	333
28	Revision history	334

List of tables

Table 1.	Acronym descriptions	21
Table 2.	Register abbreviations	23
Table 3.	STLUX/STNRG feature comparison	25
Table 4.	Interrupt priority level	29
Table 5.	CPU register map	34
Table 6.	SWIM internal register overview	38
Table 7.	DM internal registers overview	45
Table 8.	Memory access versus programming method	59
Table 9.	Flash registers overview	65
Table 10.	Reset controller register overview	70
Table 11.	RC16 MHz oscillator user trimming	72
Table 12.	Oscillators stabilization cycles	77
Table 13.	PLL output frequency divisor	82
Table 14.	Clock interrupt event	87
Table 15.	Clock controller registers overview	106
Table 16.	Power management scheme	109
Table 17.	Interrupt disable / enable inside the ISR	115
Table 18.	Interrupt priority encoding value	115
Table 19.	Dedicated interrupt instruction set	124
Table 20.	ITC interrupt control registers	131
Table 21.	MISC interrupt control registers	131
Table 22.	Watchdog timeout period (with LSI/2 counter clock)	134
Table 23.	IWDG internal registers overview	137
Table 24.	Low power-on WWDG	141
Table 25.	WWDG internal registers overview	142
Table 26.	AWU timer interval range	144
Table 27.	AWU internal registers overview	147
Table 28.	System timer internal registers overview	157
Table 29.	Auxiliary timer register overview	159
Table 30.	BASIC timer register overview	164
Table 31.	FSM states transition overview	178
Table 32.	FSM states coding	188
Table 33.	SMED low power mode	189
Table 34.	SMED coupled interface configuration scheme	190
Table 35.	Tmr(S) NextHold event assignment	195
Table 36.	Connection box interconnection matrix	198
Table 37.	SMED core register overview	214
Table 38.	SMED environment configuration register overview	220
Table 39.	GPIO port signal configuration	222
Table 40.	Effect of low power modes on GPIO ports	223
Table 41.	GPIO internal registers overview	227
Table 42.	I ² C low power mode	246
Table 43.	I ² C interrupt requests	246
Table 44.	I ² C internal registers overview	260
Table 45.	Noise detection from sampled data	268
Table 46.	Baud rate programming and error calculation	270
Table 47.	Frame format	270
Table 48.	UART low power mode	273

Table 49.	UART interrupt request	273
Table 50.	UART internal registers overview	281
Table 51.	DALI low power mode	288
Table 52.	DALI internal registers overview	299
Table 53.	DALI filter internal registers overview	299
Table 54.	DAC and comparator selection	303
Table 55.	DAC conversion voltage scale	305
Table 56.	ACU low power mode	306
Table 57.	ACU register control unit overview	311
Table 58.	ACU MISC indirect register overview	311
Table 59.	ADC operating mode	316
Table 60.	Conversion data alignment format	319
Table 61.	ADC low power mode	325
Table 62.	ADC digital interface internal registers overview	333
Table 63.	ADC MISC register overview	334
Table 64.	Document revision history	334

List of figures

Figure 1.	STNRG high level view	24
Figure 2.	CPU register model	27
Figure 3.	Stacking order.	28
Figure 4.	Addition example	30
Figure 5.	In-circuit debug block diagram	36
Figure 6.	Memory map overview	47
Figure 7.	Default stack model	48
Figure 8.	Customized stack model.	49
Figure 9.	Flash memory program and data EEPROM memory organization.	51
Figure 10.	UBC memory area definition.	52
Figure 11.	Power supply internal scheme	66
Figure 12.	Reset circuit	67
Figure 13.	V _{DD} voltage detection: POR/BOR threshold.	68
Figure 14.	PLL output clock in the two different spread modes	73
Figure 15.	HSE clock source	74
Figure 16.	Main clock tree processor interfaces	76
Figure 17.	Automatic clock switch scheme	79
Figure 18.	Manual clock switch scheme	80
Figure 19.	ADC real-time clock scheme	83
Figure 20.	SMED real-time clock scheme	84
Figure 21.	CCO clock scheme	86
Figure 22.	Power management transition state	110
Figure 23.	Interrupt processing flowchart.	114
Figure 24.	Concurrent interrupt management	119
Figure 25.	Nested interrupt management	120
Figure 26.	Independent watchdog (IWDG) block diagram.	133
Figure 27.	Window watchdog block diagram	139
Figure 28.	Approximate timeout (ms) at 16 MHz f _{CPU}	140
Figure 29.	Windows watchdog timing diagram	140
Figure 30.	AWU block diagram	143
Figure 31.	STMR block diagram	149
Figure 32.	STMR counting mode	150
Figure 33.	Counter update when ARPE = 0 (ARR not preloaded) with prescaler = 2	151
Figure 34.	Counter update when ARPE = 1 (ARR preloaded).	152
Figure 35.	SMED architecture	167
Figure 36.	FSM state transition graph	173
Figure 37.	Sequential state transition	174
Figure 38.	Event controlled state transitions	175
Figure 39.	OR state transition graph	176
Figure 40.	AND state transition graph with Event_NoSeq(I)	177
Figure 41.	AND transition graph without Event_NoSeq(I)	177
Figure 42.	Counter reset example	180
Figure 43.	Dithering example.	185
Figure 44.	Single couple SMEDs interconnection scheme	191
Figure 45.	Single couple SMEDs with SMD<n,n+1>_DRVOUT bit cleared.	191
Figure 46.	Single coupled SMED with SMD<n>_DRVOUT bit set.	192
Figure 47.	Two couple SMEDs interconnection scheme	193
Figure 48.	Two couple SMEDs with SMD<0,1,2,3>_DRVOUT bits cleared	194

Figure 49.	Two couple SMEDs with SMD<x>_DRVOUT bits set	195
Figure 50.	SMED unlock sequence	197
Figure 51.	Interconnection matrix for one SMED	198
Figure 52.	I ² C bus protocol	230
Figure 53.	Transfer sequence diagram for slave transmitter	231
Figure 54.	Transfer sequence diagram for slave receiver	232
Figure 55.	Transfer sequence diagram for master transmitter	235
Figure 56.	Transfer sequence diagram for master receiver	236
Figure 57.	Method 2 - transfer sequence diagram for master receiver when N > 2	238
Figure 58.	Method 2 - transfer sequence diagram for master receiver when N = 2	239
Figure 59.	Method 2- transfer sequence diagram for master receiver when N = 1	240
Figure 60.	I ² C interrupt mapping diagram	247
Figure 61.	Word length programming	263
Figure 62.	TC/TXE behavior when transmitting	265
Figure 63.	Start bit detection	266
Figure 64.	Data sampling for noise detection	268
Figure 65.	UART baud rate generator	269
Figure 66.	Mute mode using idle line detection	272
Figure 67.	Mute mode using address mark detection	272
Figure 68.	UART Interrupt mapping diagram	274
Figure 69.	DALI block diagram	283
Figure 70.	DALI transfer sequences	284
Figure 71.	Digital input filter interconnection scheme	286
Figure 72.	Filtering modes	287
Figure 73.	Comparator unit native logic	300
Figure 74.	Comparator unit with external reference voltage native logic	301
Figure 75.	Comparator unit with external reference voltage optimized logic	301
Figure 76.	ADC block diagram overview	313
Figure 77.	Single mode buffer management (w/o FIFO flush or auto-flush)	317
Figure 78.	Continuous mode buffer management	317
Figure 79.	ADC single conversion cycle timing diagram	320
Figure 80.	ADC continue conversion cycle timing diagram	321
Figure 81.	ADC delayed start of conversion timing diagram	321

1 Reference documents

- For hardware information on the STLUX controller and product specific SMED configuration, please refer to the STLUX product datasheets for STLUX385A, STLUX383A, STLUX325A and STLUX285A.
- For hardware information of the STNRG controller and product specific SMED configuration, please refer to the STNRG product datasheets for STNRG388A, STNRG328A and STNRG288A.
- For information on programming, erasing and protection of the internal Flash memory please refer to the programming manual PM0051 - “How to program STM8S and STM8A Flash program memory and data EEPROM”.
- For information about the debug module and SWIM (single wire interface module) refer to the STM8 SWIM communication protocol and debug module user manual (UM0470).
- For information on the STM8 core and assembler instruction please refer to the STM8 CPU programming manual (PM0044).
- For information on the development tools please refer to the UM1792.

2 Acronyms

The following is a list of acronyms used in this document:

Table 1. Acronym descriptions

Acronym	Description
ACU	Analog comparator unit
ADC	Analog-to-digital converter
AWU	Auto-wakeup unit
BL	Bootloader - used to load the user program without the emulator
CKC	Clock control unit
CPU	Central processing unit
CSS	Clock security system
DAC	Digital-to-analog converter
DALI	Digital addressable lighting interface
ECC	Error Correction Code
FSM	Finite state machine
FW	Firmware loaded and running on the CPU
GPIO	General purpose input/output
HSE	High-speed external crystal - ceramic resonator
HSI	High-speed internal RC oscillator

Table 1. Acronym descriptions (continued)

Acronym	Description
I ² C	Inter-integrated circuit interface
IAP	In-application programming
ICP	In-circuit programming
ITC	Interrupt controller
IWDG	Independent watchdog
LSI	Low-speed internal RC oscillator
MCU	Microprocessor central unit
MSC	Miscellaneous
PM	Power management
RFU	Reserved for future use
ROP	Read-out protection
RST	Reset control unit
RTC	Real-time clock
SMED	State machine, event driven
STMR	System timer
SW	Software is the firmware loaded and running on the CPU (synonymous of FW)
SWIM	Single wire interface module
UART	Universal asynchronous receiver transmitter
WWDG	Window watchdog

3 Register model

Table 2 shows an outline picture of the register abbreviations used in this document.

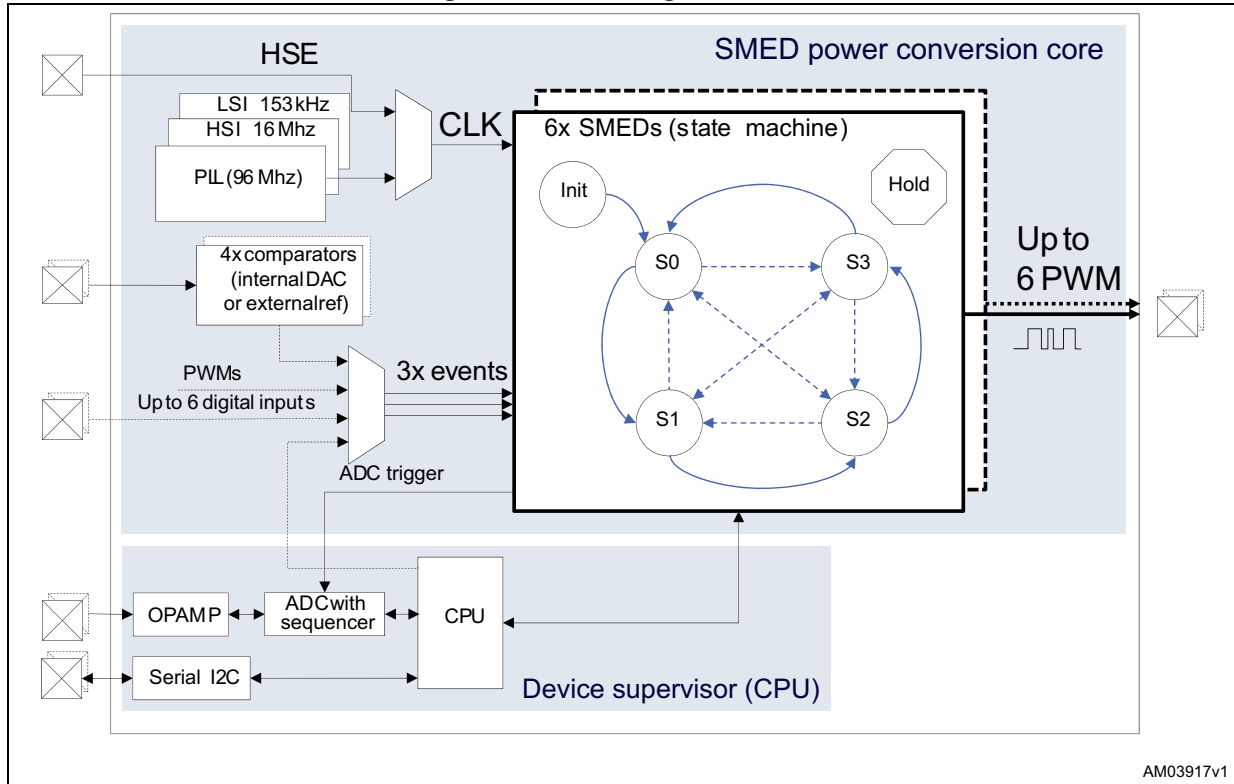
Table 2. Register abbreviations

Abbreviation	Description
read/write (rw) or (r/w)	Software can read and write to these bits.
read only (r, ro)	Software can only read these bits.
write only (w, wo)	Software can only write to this bit. Reading the bit returns a meaningless value.
read/write clear (r/wc)	SW reads and writes clear.
read/write clear (r/w0)	SW reads and writes clear when writing-bit 0.
read/write clear (r/w1)	SW reads and writes clear when writing-bit 1.
read/write once (rwo)	Software can only write once to this bit and can also read it at any time. Only a reset can return the bit to its reset value.
read/clear (rc/w1)	Software can read as well as clear this bit by writing 1. Writing '0' has no effect on the bit value.
read/clear (rc/w0)	Software can read as well as clear this bit by writing 0. Writing '1' has no effect on the bit value.
read/set (rs)	Software can read as well as set this bit. Writing '0' has no effect on the bit value.
read/write0 (r/w0)	Software can read and writing '0'. Writing '1' has to be avoiding for future product compatibility.

4 Description

The STLUX and STNRG device families generate and control PWM signals by means of a state machine, called SMED (state machine event driven). *Figure 1* gives an overview of the internal architecture.

Figure 1. STNRG high level view



The core of the device is the SMED unit: a hardware state machine driven by system events. The SMED includes 4 states (S0, S1, S2 and S3) available during running operations. A special HOLD state is provided as well.

The SMED allows the user to configure, for every state, which system events will trigger a transaction to a new state.

During a transaction from one state to the other, the PWM output signal level can be updated.

Once a SMED is configured and running, it becomes an autonomous unit, so no interaction is required since the SMED automatically reacts to system events.

Thanks to the SMED's 96 MHz operating frequency and their automatic dithering function, the PWM maximum resolution is 1.3 ns.

The device has 6 SMEDs available. Multiple SMEDs can operate independently from each other or they can be grouped together to form a more powerful state machine.

4.1 STLUX/STNRG feature comparison

The following table summarizes the differences between the STLUX and STNRG family of products

Table 3. STLUX/STNRG feature comparison

Feature		Device	
		STLUX	STNRG
Max. pin count		38	
SMED available		6	
Max. SMED PWM output pins		6	
Max. fast digital inputs pins		6	
Max. positive comparator input pin		4	
Max. negative comparator input pins		1	3
Comparator hysteresis		No	Yes
Internal DACs		4	
Max. ADC input pins		8	
Max. ADC gain		x1 – x4	
ADC hardware trigger		No	Yes
Max. GPIO port 0 pins		6	
Communication	UART peripheral	Yes	
	DALI peripheral	Yes	No
	I2C peripheral	Yes	
HSE function		Yes	
Timers	System timer	1	
	Auxiliary timer	1	
	Basic timer	No	2
Auto-wakeup timer		1	
Watchdog	Window watchdog timer	1	
	Independent watchdog timer	1	
Flash program memory		32 Kb	
EEPROM data memory		1 Kb	
RAM		2 Kb	6 Kb