RuGBy Datasheet Rev 0.4

1 REVISION HISTORY

Table 1 Revision History

Rev #	Date	Action	By
0.1	12/12/2018	Generated from Orion PRD V0.8: 1. 24 line LED driver -> 18 line LED driver 2. remove RS485, keep UART. 3. add 2 GPIO	AlexHuang
0.2	09/30/2019	1. DCDC requirements updated: a) Output voltage range:3.5~6V b) Overtemp/Short/Overvoltage protection 2. LED Driver drain voltage < = 1V. 3. Enhanced VFW measurement through a dedicated current source + real differential measurement.	JackZhu/JackPan
0.3	11/11/2019	1. VBAT range: 6V~19V 2. Package changed to QFN48	JackZhu/JackPan/Jian
0.4	04/23/2019	Updated some electrical characteristics after characterization: 1. Buck	DavidDong/JackZhu

2 TABLE OF CONTENTS

3 List of Tables

4 List of Figures

5 SYSTEM OVERVIEW

"RuGBy" IC is an automotive LED lighting integrated device that combines together a 32bit MCU (Cortex M0, indie 'Verne' core) with a flexible power management unit including a step down buck converter and associated regulators, 18x high voltage programmable current open drain IO with PWM along with specific monitoring features, 2x LIN slave transceivers and their controllers, 1x UART and an integrated 10 bit ADC for monitoring purpose. The IC can withstand 45V load dump from the car battery on its VBAT pin while has the option to operate with a 12V and a 5V input supplies saving the buck passive components entirely.

- Full automotive qualification AEC-Q100 Grade1
- CPU architecture:
  - ARM Cortex M0 processor
  - System Tick Timer (Systick, 24bits, interruptible)
  - Serial Wire Debugger (ARM)
  - Built-in Nested Vectored Interrupt Controller (NVIC)
  - 32bits multiplier
  - Programmable Watch-Dog Timer
  - 3 programmable timers
- Memory:
  - 64kBytes of Flash Program Memory, 10 years retention in automotive environment
  - 16kBytes of SRAM
- Peripherals/Digital Features
  - Clock and Reset Manager
    - RCO: system and always on (wake up support)
    - Reset POR and BOR (no external reset)
  - One SAE J2602 LIN Slave Controller and Transceiver
    - Supports LIN auto-addressing through an internal LIN switch.
  - One SAE J2602 LIN Master Controller and Transceiver
    - Only available when the internal LIN switch is not used.

- Watch dog timer (ASIC side)
- 18x16bits PWM required to control LED current driver:
  - Common prescaler and 16bit timer
  - Support power balance with independent rise/fall timing configuration
- Serial Communication interfaces:
  - Integrated UART Controller with two independent fifos(TX:32 bytes;RX:32 bytes)
- Up to 5 GPIOs with flexible configurations:
  - General GPIO with pullup & pulldown
  - Analog Input mode
  - Support Edge detects
  - Wakeup sources in hibernate mode

- Peripherals/Analog Features:
  - 18 Programmable 45mA max constant current Low voltage(VDD5P0) IO open drain
    - Drain voltage < 1V
    - Integrated pullup capability
  - PN voltage measurement for temp compensation:
    - Integrated a dedicated 2.5mA current source
    - Fully differential measurement
  - Temperature Sensor/Monitor with ADC
  - Battery voltage detection and monitoring
  - Hardware over temperature protection
  - 10 bits SAR ADC with 28 channels
    - Buffered bandgap voltage
    - Junction Temperature
    - Forward voltages of 18 external LEDs
    - GPIO1~5
    - One accurate VBAT channel
    - MCU Core/IO Voltage

        - Buck Output voltage
    - Fully integrated step-down buck converter
        - Operation Freq adjustable: 100KHz~500KHz
        - Output voltage adjustable: 3.5~6V
        - Overcurrent/Dead short protection with hiccup support
    - Integrated voltage regulators
        - LDO 3.3Vout (ASIC Core and IO supply + MCU I/O)
        - LDO 1.5VOut (MCU Core/Flash)

5.1 APPLICATION BLOCK DIAGRAM

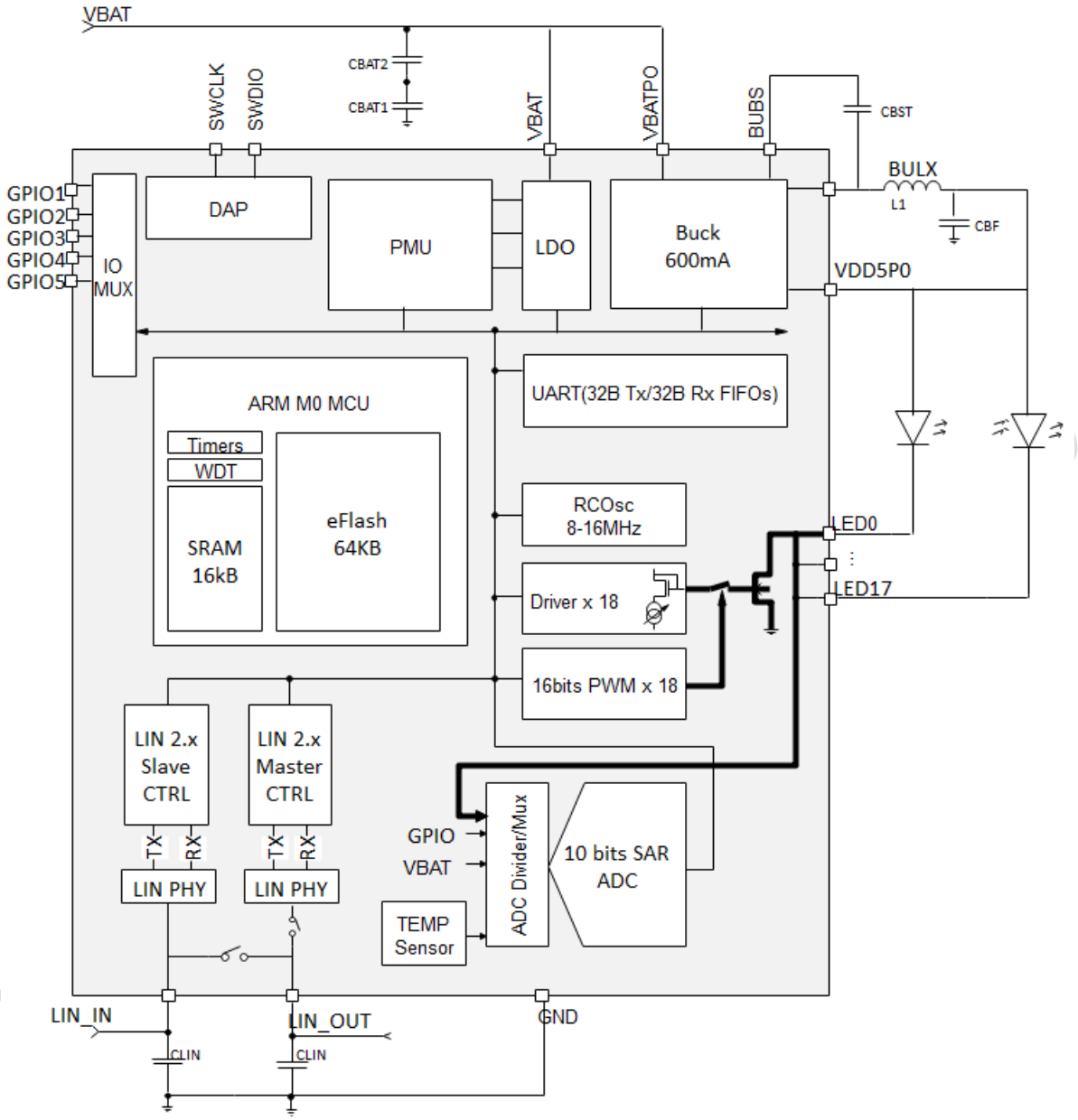


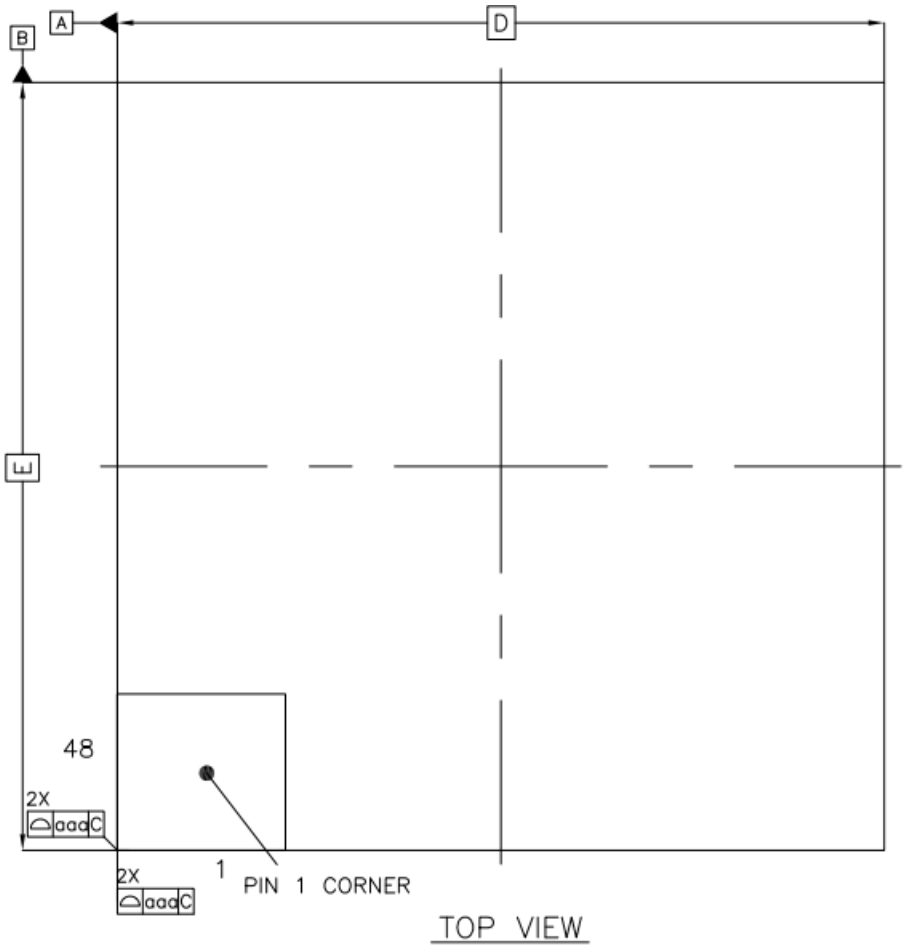
Figure 1 IC block diagram with external components

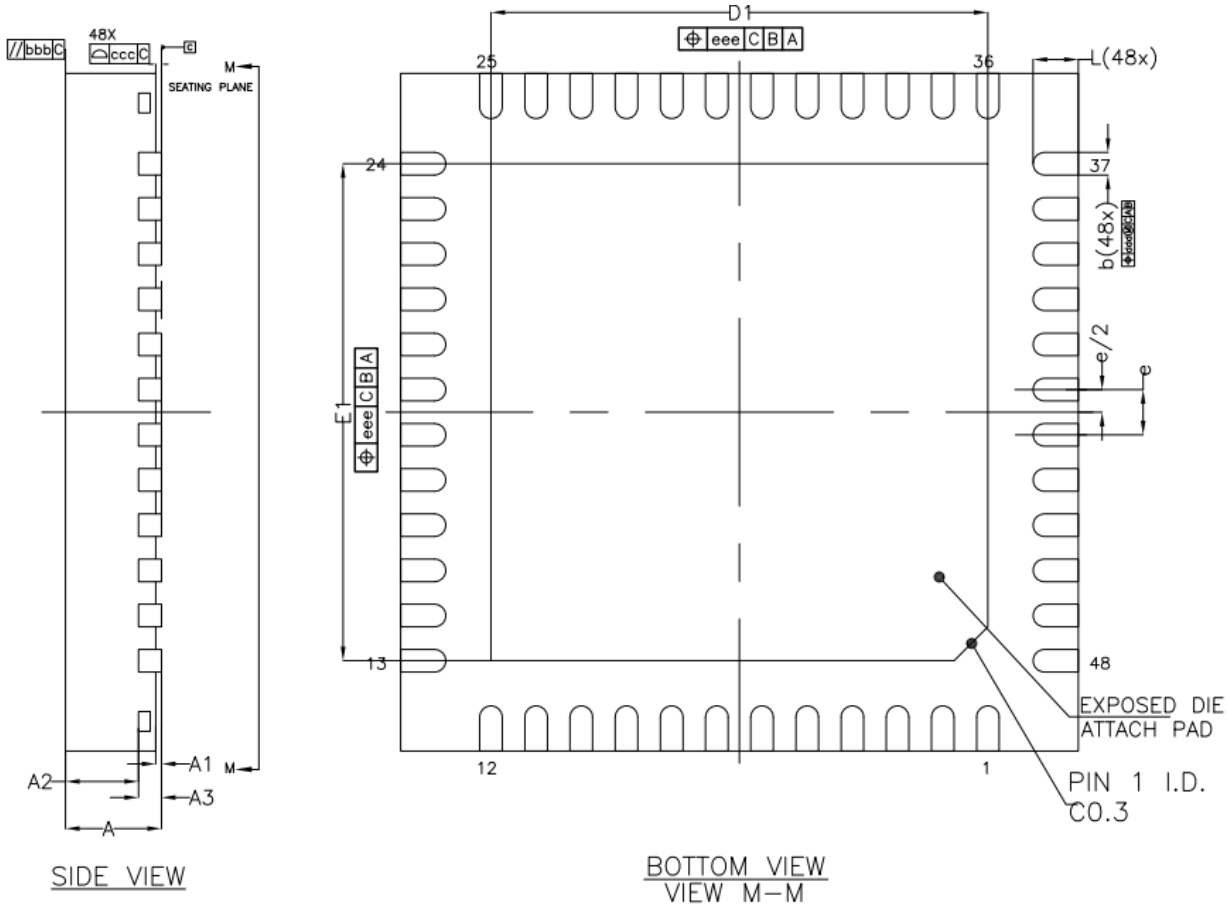
Note: The application block diagram does not include components used to qualify the system against ISO7637-2/-3.

5.2 PACKAGE OVERVIEW AND PIN DESCRIPTION

5.2.1 QFN48 Package Outline

QFN 6x6 48 pins 0.4mm pitch



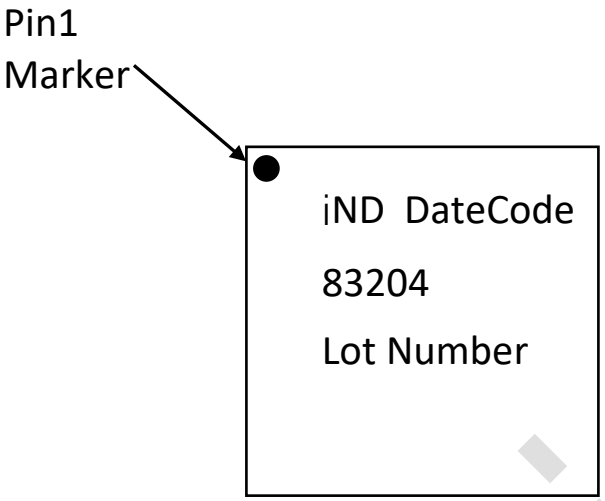


DESCRIPTION		SYMBOL	MIN	NOM	MAX
TOTAL THICKNESS		A	0.80	0.85	0.90
STAND OFF		A1	0.00	---	0.05
MOLD THICKNESS		A2	0.60	0.65	0.70
L/F THICKNESS		A3	0.203 REF		
LEAD WIDTH		b	0.14	0.20	0.26
BODY SIZE	X	D	5.90	6.00	6.10
	Y	E	5.90	6.00	6.10
LEAD PITCH		e	0.40 BSC		
EP SIZE	X	D1	4.35	4.40	4.45
	Y	E1	4.35	4.40	4.45
LEAD LENGTH		L	0.30	0.40	0.50
Tolerance of form and position					
PACKAGE EDGE TOLERANCE		aaa	0.1		
MOLD FLATNESS		bbb	0.1		
COPLANARITY		ccc	0.08		
LEAD OFFSET		ddd	0.1		
EXPOSED PAD OFFSET		eee	0.1		

5.2.2 Part Number

Part Number: iND83204

Package Branding:



5.3 IO PIN DESCRIPTIONS

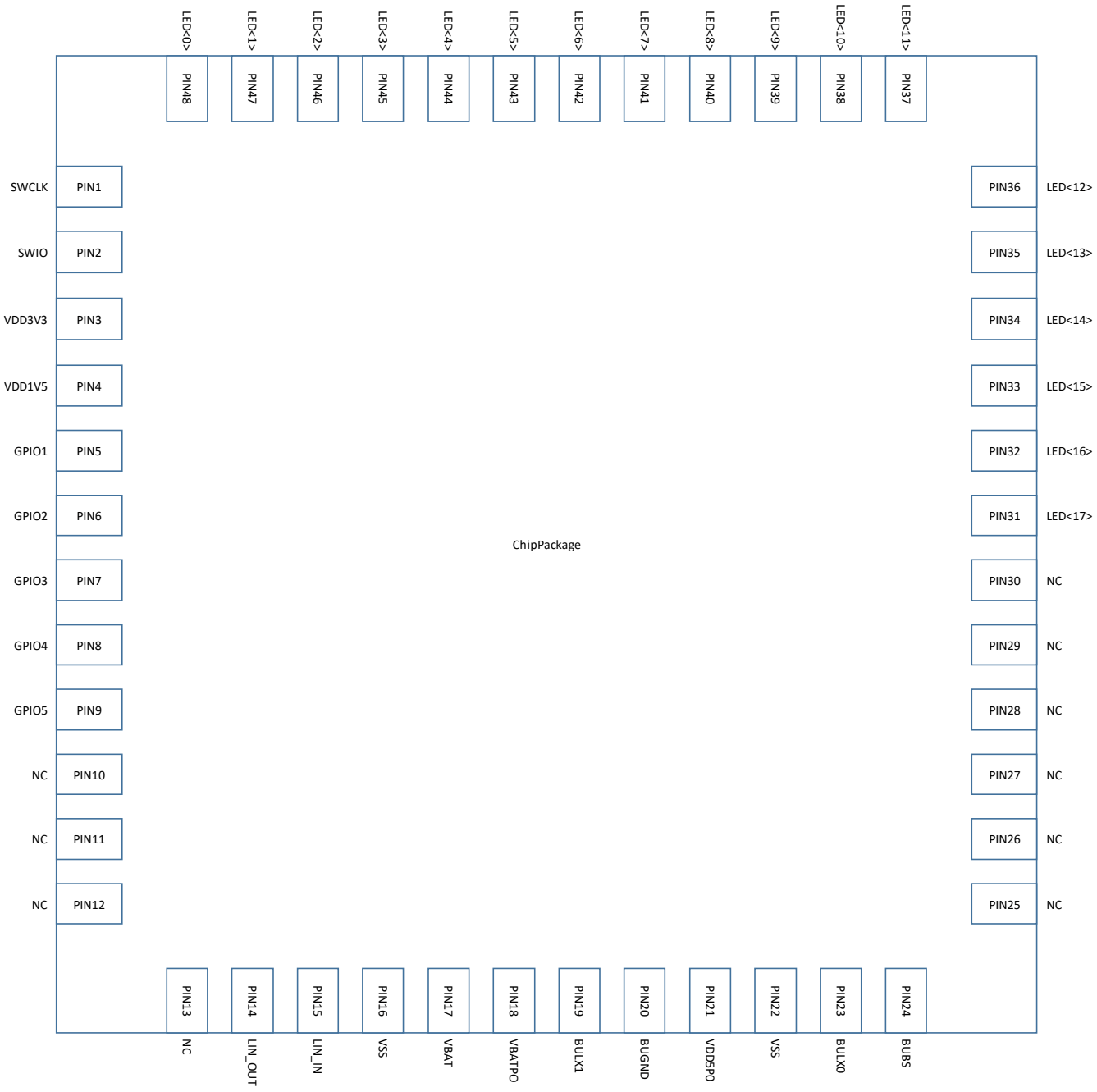


Figure 2 Pin Configuration

Table 2 pin list

#	Pin Name	Type	Voltage	Direction	Description
1	SWDCLK	GPIO	VDD3V3	Input	SWD clock
2	SWDIO	GPIO	VDD3V3	I/O	SWD data
3	VDD3V3	Output	VDD3V3	Analog	Voltage regulator decoupling cap
4	VDD1V5	Output	VDD1V5	Analog	Voltage regulator decoupling cap
5	GPIO1	GPIO	VDD3V3	I/O	PWM0/General purpose IO, debug
6	GPIO2	GPIO	VDD3V3	I/O	PWM1 /General purpose IO, debug
7	GPIO3	GPIO	VDD3V3	I/O	PWM2 /General purpose IO, debug
8	GPIO4	GPIO	VDD3V3	I/O	PWM3 /UART RX input
9	GPIO5	GPIO	VDD3V3	I/O	PWM4 /UART TX output
10	NC	-	-	-	-
11	NC	-	-	-	-
12	NC	-	-	-	-
13	NC	-	-	-	-
14	LIN_OUT	IO	VBAT	I/O	LIN Master pin/LIN Switch Out
15	LIN_IN	IO	VBAT	I/O	LIN slave pin

16	VSS	Supply	Ground	Analog	Bonding with ground plane
17	VBAT	Supply	VBAT	n/a	Input VBAT
18	VBATPO	Supply	VBAT	n/a	Input VBAT for buck (power)
19	BULX1	Analog	VBAT	n/a	connect to external inductor
20	BUGND	Ground	Ground	Analog	Buck power gound
21	VDD5P0	Analog	VDD5P0	Analog	buck 5V output(inner), also is feedback input(external)
22	VSS	Supply	Ground	Analog	Bonding with ground plane
23	BULX0	Analog	VBAT	n/a	inner connect to BULX1, external connect to bootstrap cap(100nF)
24	BUBS	Analog	BULX2+3.3V	Analog	Buck Bootstrap cap (100nF) bootstrap 3.3V w.r.t BULX2
25	NC	-	-	-	-
26	NC	-	-	-	-
27	NC	-	-	-	-
28	NC	-	-	-	-
29	NC	-	-	-	-
30	NC	-	-	-	-
31	LED17	Analog	VDD5P0	Analog	Current Regulated Sink
32	LED16	Analog	VDD5P0	Analog	Current Regulated Sink
33	LED15	Analog	VDD5P0	Analog	Current Regulated Sink
34	LED14	Analog	VDD5P0	Analog	Current Regulated Sink

35	LED13	Analog	VDD5P0	Analog	Current Regulated Sink
36	LED12	Analog	VDD5P0	Analog	Current Regulated Sink
37	LED11	Analog	VDD5P0	Analog	Current Regulated Sink
38	LED10	Analog	VDD5P0	Analog	Current Regulated Sink
39	LED9	Analog	VDD5P0	Analog	Current Regulated Sink
40	LED8	Analog	VDD5P0	Analog	Current Regulated Sink
41	LED7	Analog	VDD5P0	Analog	Current Regulated Sink
42	LED6	Analog	VDD5P0	Analog	Current Regulated Sink
43	LED5	Analog	VDD5P0	Analog	Current Regulated Sink
44	LED4	Analog	VDD5P0	Analog	Current Regulated Sink
45	LED3	Analog	VDD5P0	Analog	Current Regulated Sink
46	LED2	Analog	VDD5P0	Analog	Current Regulated Sink
47	LED1	Analog	VDD5P0	Analog	Current Regulated Sink
48	LED0	Analog	VDD5P0	Analog	Current Regulated Sink
*	EPAD	Supply	Ground	n/a	Ground

Note: the above pinout is not representing the final pin allocation for the IC. The exact pin allocation will be defined during the design phase including floorplan studies and PCB constraints provided by the customer.

5.3.1 Pin state upon Power-on Reset

- Unless otherwise noted, all pins default to tristate/Isolation mode (Hi-Z) upon power-on reset.

6 ELECTRICAL CHARACTERISTICS

6.1 ABSOLUTE MAXIMUM RATINGS

Table 3 Absolute Maximum Ratings, Voltages Referenced to ground

Names	Conditions	Min.	Max.	Unit
VBAT/VBATPO	No damage, t<500ms	-0.3	+45	V
VBAT/VBATPO	No damage, t<5min	-0.3	+28	V
VBAT/VBATPO	No damage, t<5ms	-1.1		V
VBAT/VBATPO	No damage, t<20ns	-4.0		V
VBAT/VBATPO	No damage, ISO 7637-2 pulse 1, VBAT=13.5V, TA=23°+/-5C, test pulse applied to VBAT via reverse polarity diode and more than 4.7uF capacitor	-100		V
VBAT/VBATPO	No damage, ISO 7637-2 pulse 2 VBAT=13.5V, TA=23°+/-5C, test pulse applied to VBAT via reverse polarity diode and more than 4.7uF capacitor		+50	V
VBAT/VBATPO	No damage, accept ISO 7637-2 pulses 3A, 3B, VBAT=13.5V, TA=(23+/-5)°C, test pulse applied to VBAT via reverse polarity diode and more than 4.7uF capacitor	-150	+100	V
VBAT/VBATPO	No damage, ISO 7637-2 pulses 5b VBAT=13.5V, TA=(23+/-5)°C, test pulse applied to VBAT via reverse polarity diode and more than 4.7uF capacitor		+45	V
LIN	No damage, t<500ms	-40	+40	V
LIN	No damage, ISO 7637-3 pulse 1 VBAT=13.5V, TA=23°+/-5C, test pulse applied to LIN via 1nF capacitor	-100		V

LIN	No damage, ISO 7637-2 pulse 2 VBAT=13.5V, TA=23°+/-5C, test pulse applied via 1nF capacitor		+50	V
LIN	No damage, ISO 7637-2 pulses 3A, 3B VBAT=13.5V, TA= (23+/-5) °C, test pulse applied via 1nF capacitor	-150	+100	V
LEDx	No damage	-0.3	6.0	V
VDD5P0	No Damage	-0.3	6.0	V
BULX, BUBS	No Damage	-0.3	45	V
GPIO1~5, SWDCLK, SWDIO	No Damage	-0.3	3.6	V
VBAT/VBATPO/LIN_IN/ LIN_OUT/GND	ESD HBM	-4	+4	kV
All pins except VBAT/VBATPO and LIN	ESD HBM	-2	+2	kV
All pins	ESD CDM	-750	+750	V
Storage Temp		-55	+150	°C

Note: Stresses beyond those listed under absolute maximum ratings may cause permanent damage to the device. The absolute maximum ratings provided in the table above are limiting values that do not lead to a permanent damage of the part. But functional operation of the device at these or any other conditions beyond those indicated under recommended operating conditions is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

6.2 ELECTRICAL CHARACTERISTICS

Table 4 Electrical Characteristics

Parameter	Conditions	Min.	Typ.	Max.	Unit
Operating ambient Temperature		-40	25	125	°C
Operating Junction Temperature		-40	25	125	°C

Parameter	Conditions	Min.	Typ.	Max.	Unit
Package Thermal Resistance	Junction to Board (ThetaJB)		30		K/W
VBAT		6	13.5	19	V
IO Supply (VDD3P3)		3.0	3.3	3.6	V
ASIC Core Supply (VDD3P3)		3.0	3.3	3.6	V
MCU Core Supply	MCU Core Supply including SRAM and Flash	1.35	1.5	1.65	V
Flash Memory					
Sector Endurance		20k			cycles
Data Retention	@25degC	100			Years
Data Retention	@85degC	25			Years
5V Buck Converter					
Buck Input Supply	VBAT input supply (Maximum Range)	6	13.5	19	V
	Recommended VBAT input supply for maximum output capability	8	13.5	19	V
Buck Output Voltage VDD5P0	V_{BAT}=13.5V, configurable range of VDD5P0.	3.5	5	6	V
Buck Load current	8V<=V_{BAT}<=19V, V_{VDD5P0}=5.0V.			600	mA
	V_{BAT}=6V, V_{VDD5P0}=5.0V.			500	mA
Buck Peak Efficiency	At I_{LOAD}=500mA, VBAT from 8V to 19V, V_{VDD5P0}=5V, Fsw=500kHz.	85	88	91	%
Buck Quiescent current	Normal Switching mode		1	5	mA
Buck Switching Frequency(Fsw)	Forced PWM mode with integrated FETs		500		kHz
Buck Ripple	V_{BAT}=13.5V, V_{VDD5P0}=5.0V, I_{Load}=300mA			60	mV
Buck Load regulation	V_{BAT}=13.5V, V_{VDD5P0}=5.0V, load current from 1uA to 500mA			20	mV

Parameter	Conditions	Min.	Typ.	Max.	Unit
Buck Line regulation	I_{Load}=500mA, Input voltage VBAT from 8V to 19V.			20	mV
Buck Soft Start	VBAT=13.5V, I_{load}<200mA, VDD5P0 from 0 to 5V			10	ms
Buck Inductor	L1 nominal value		10		uH
Buck Output Capacitor	C_{OUT} nominal value		44		uF
Buck Input Capacitor	C_{IN} nominal value, place near VBATPO as close as possible.		10		uF
Buck Monitor Over Voltage Threshold				6.78	V
Buck Monitor Under Voltage Threshold		1.47		4.43	V
Clocks					
System RC Oscillator Frequency		8		16	MHz
System RC Oscillator Accuracy	16MHz	-5		5	%
System RC Oscillator start up time			10		us
Auxiliary clock	Used in SLEEP mode		16		kHz
POR/BOR					
POR (VDD3P3)		1.8		2.1	V
BOR VDD3P3	200mV window, steps every 100mV (2.3 to 3.2V)	2.3		3.2	V
BOR VDD1P5	Max Value at which system resume operation	1.31			V
Battery Monitor					
Under Voltage Threshold	Analog Comparator Generates interrupt to MCU except in SLEEP mode (disabled feature)	4.5	5.0	5.5	V
		5.5	6.0	6.5	V
		6.5	7.0	7.5	V
		7.5	8.0	8.5	V

Parameter	Conditions	Min.	Typ.	Max.	Unit
		8.5	9.0	9.5	V
Under Voltage hysteresis	Includes digital debounce ~50ms	0.1	0.3	0.4	V
Under Voltage Digital debounce time	Programmable, 10ms steps	10	30	60	ms
Over Voltage Threshold	Analog Comparator, generates interrupt to MCU except in SLEEP mode (disabled feature)	17	19	20	V
Over Voltage Hysteresis		0.5	0.6	0.8	V
Current Source LED(Low Voltage Pins)					
LED voltage	minimum voltage to ensure current regulation			1.0	V
Sink Current	VBAT>6V	0.1		45	mA
Sink Current step size			100		uA
Sink Current Error	Ta=25degC	-7		+7	%
Temperature Drift			-0.025		%/K
Over Temperature Monitor					
Overtemp Threshold	Analog Comparator, generates interrupt or reset to MCU.	100		150	degC
Overtemp hysteresis		-10			degC
Temperature Sensor					
Temperature range	The MCU is in charge to pull the ADC related data from temperature sensor.	-40		150	degC
Temperature Accuracy		-10		+10	degC
Active current			20		uA
SAR ADC					
Resolution			10		bits
Conversion Speed	17 cycles per conversion (4 cycles for S/H and 13 cycles for conversion)			200	ks/s

Parameter	Conditions	Min.	Typ.	Max.	Unit
ADC Clock	16MHz RC clock divided by 4			4	MHz
INL	Guaranteed by design	-2		2	LSB
DNL	Guaranteed by design	-1		1	LSB
TCURR	Minimum time to wait after the positive edge of the sync input before starting the 1st conversion of the sequence to cover ADC input buffer transient time. Programmable Value (TCURR+1)x250ns, TCURR=7 to 15	2		4	us
TGUARD	Minimum guard time during which there is no channel input selected. (TGUARD+1)x250ns Programmable Value, default 1us	0.25	1	4	ns
TCHNL	Minimum time to wait after TGUARD time to start a new ADC conversion. Programmable Value (TCHNL+1)x250ns, TCHNL=7 to 15	2		4	us
Reference voltage	Post Calibration	1.19	1.20	1.21	V
LIN EC specified with VBAT=8V to 16V – refer to LIN 2.x specification, VBUS=LIN pin/line					
Supply Voltage	supply voltage range	6	13.5	19	V
IBUS_LIM	Current Limitation for Driver dominant state driver on VBUS = VBAT=16V	40		200	mA
Rslave	Lin Slave Pullup	20	30	60	kΩ
Rmaster	Lin Master Pullup	900	1000	1100	Ω
IBUS_PAS_dom	Input Leakage Current at the Receiver including Pull-Up Resistor driver off VBUS = 0V	-1			mA

Parameter	Conditions	Min.	Typ.	Max.	Unit
	VBAT= 12V				
IBUS_PAS_rec	Driver off, VBUS>VBAT 8V<VBAT<16V 8V<VBUS<16V			20	uA
IBUS_no_GND	Control unit disconnected from ground GND Device = VSUP 0V<VBUS<16V VBAT = 12V Loss of local ground must not affect communication in the residual network. **LIN 2.2A**	-1		+1	mA
Device Bus Leakage Current Ground Disconnected	VBAT= VGND=12V, 0V<VBUS<18V **J2602**	-100		100	uA
IBUS_no_BAT	VBAT disconnected 0<VBUS<16V, VBAT=0V **LIN 2.2A** Node has to sustain the current that can flow under this condition. Bus must remain operational under this condition.			100	uA
Device Bus Leakage current VBAT disconnected	0V<VBUS<18V, VBAT=VGND=0V **J2602**	-23		23	uA
BUS_VOL Transmitter dominant voltage	Load 500Ohms, driver open drain active	0.0		0.2	VSUP
BUS_VOH Transmitter recessive voltage	Driver open drain high impedance	0.8		1.0	VSUP
CSLAVE	LIN pin input capacitance Note that LIN 2.2A spec 220pF typ, 250pF max as total node capacitance at the connector including the physical bus driver and all other components including C_{LIN}			35	pF

Parameter	Conditions	Min.	Typ.	Max.	Unit
VBUSdom	Receiver dominant state			0.4	VSUP
VBUSrec	Receiver recessive state	0.6			VSUP
VBUS_CNT	Center point Receiver VBUS_CNT =(Vth_dom+ Vth_rec)/2	0.475	0.5	0.525	VSUP
Vhys	Receiver hysteresis VHYS = Vth_rec -Vth_dom			0.175	VSUP
Trx_pd	propagation delay of receiver C_{RXD} load 20pF (RX output of transceiver, internal node, access in test mode) minimum slew rate for the LIN rising and falling edges is 50V/us			6	us
Trx_sym	symmetry of receiver propagation delay rising edge w.r.t. falling edge C_{RXD} load 20pF C_{RXD} load 20pF (RX output of transceiver, internal node, access in test mode)	-2		+2	us
LIN Timing parameters (CBUS ; RBUS): (1nF; 1kΩ) / (6.8nF;660Ω) / (10nF;500Ω)					
D1 Duty Cycle (20kbits/s)	THRec(max) = 0.744 x VSUP; THDom(max) = 0.581 x VSUP; VSUP = 7.0V...16V; tBit = 50µs; D1 = tBus_rec(min) / (2 x tBit)	0.396			-
D2 Duty Cycle (20kbits/s)	THRec(min) = 0.422 x VSUP; THDom(min) = 0.284 x VSUP; VSUP = 7.6V...16V; tBit = 50µs; D2 = tBus_rec(max) / (2 x tBit)			0.581	-
D3 Duty Cycle (10.4kbits/s)	THRec(max) = 0.778 x VSUP; THDom(max) = 0.616 x VSUP; VSUP = 7.0V...16V; tBit = 96µs; D3 = tBus_rec(min) / (2 x tBit)	0.417			-

Parameter	Conditions	Min.	Typ.	Max.	Unit
D4 Duty Cycle (10.4kbits/s)	THRec(min) = 0.389 x VSUP; THDom(min) = 0.251 x VSUP; VSUP = 7.6V...16V; tBit = 96µs; D4 = tBus_rec(max) / (2 x tBit)			0.590	-
tBus_rec(max)-tBus_dom(min)	Δt3, 10.4kbs operation, low speed mode, **J2602**			15.9	us
tBus_dom(max)-tBus_rec(min)	Δt4, 10.4kbs operation, low speed mode, **J2602**			17.28	us
GPIOs					
GPIOVIL	Input Low Voltage			0.3* VDD3P3	V
GPIOVIH	Input High Voltage	0.7* VDDD 3P3			V
GPIOIOL	Max load current with output voltage=VOL			10	mA
GPIOIOH	Max load current with output voltage=VOLH			10	mA
GPIOVOL	Output Low Voltage			0.4	V
GPIOVOH	Output High Voltage	2.4			V
GPIOPU	Pull Up Resistance			110	kOhm
GPIOPD	Pull Down Resistance			110	kOhm
SWDCLK, SWDIO					
SWDVIL				0.8	V
SWDVIH		2			V
SWDCLKIOL	SWCLK, Max load current with output voltage=VOL			4	mA
SWDCLKIOH	SWCLK, Max load current with output voltage=VOH			4	mA

Parameter	Conditions	Min.	Typ.	Max.	Unit
SWDIOIOL	SWDIO, Max load current with output voltage=VOL			8	mA
SWDIOIOH	SWDIO, Max load current with output voltage=VOH			8	mA
SWDVOL				0.4	V
SWDVOH		2.4			V
SWDPU (SWDIO IO)	Pull Up Resistance	22		110	kOhm
SWDPD (SWDCLK IO)	Pull Down Resistance	22		110	kOhm
SWDVIL				0.8	V

Electrical Characteristics are valid over the full temperature range of Tj = -40°C to +125°C and a supply range of 6V <= VBAT <= 19V unless otherwise noted.

The figure below shows the relation between the propagation delay, the TX thresholds and associated receiver duty cycles. Refer to D1 to D4 duty cycles in the table above for THRec and THDom threshold levels.

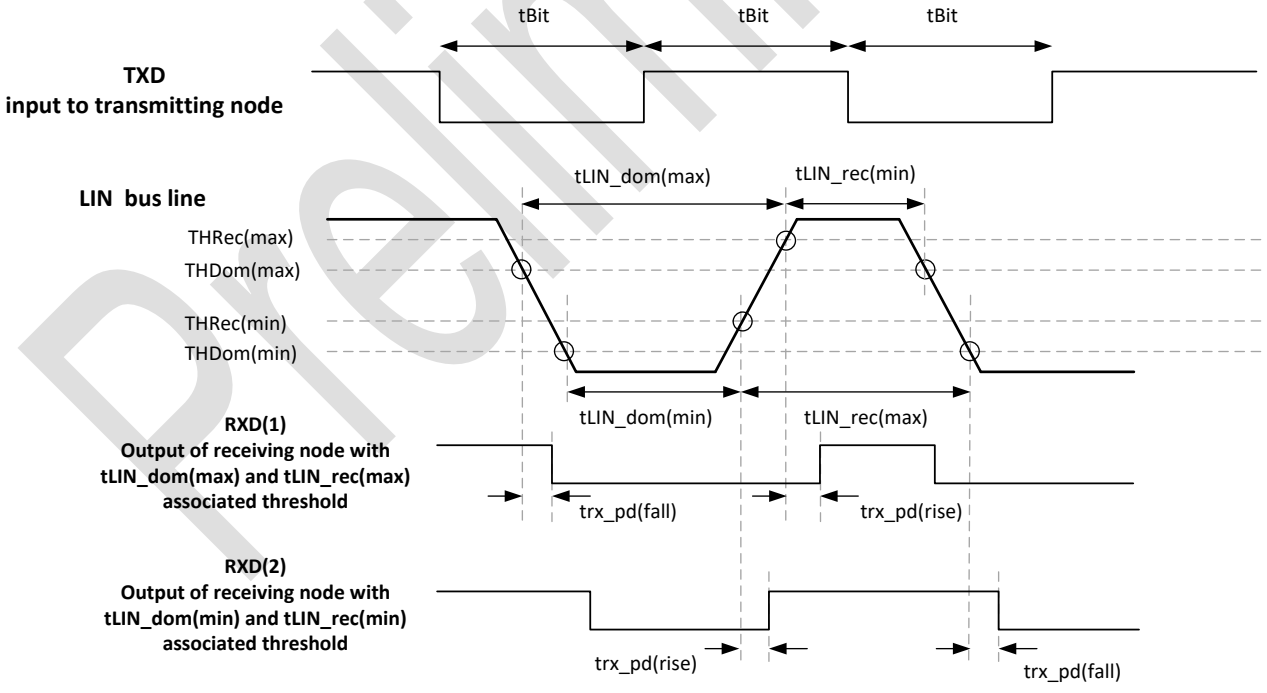


Figure 3 LIN timing Diagram

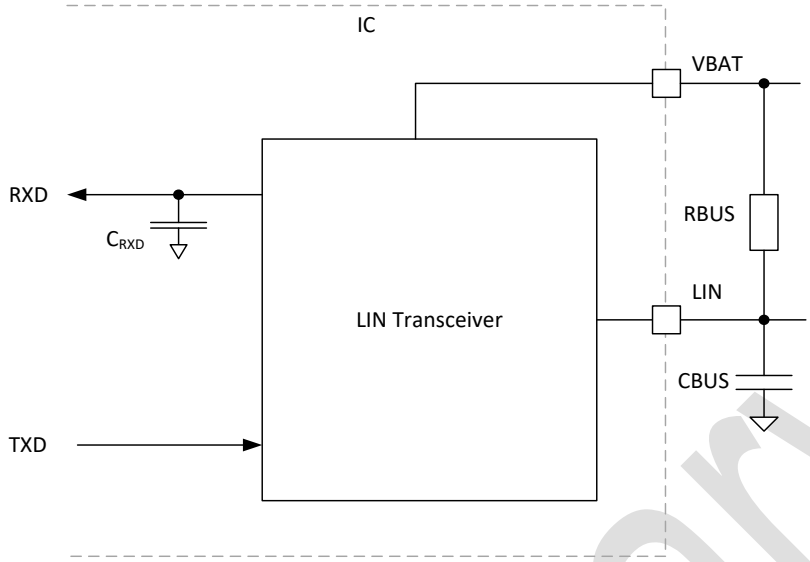


Figure 4 LIN AC Test Circuit

6.3 CURRENT CONSUMPTION

Table 5 : Current Consumption					
Mode	**Conditions**	**Min.**	**Typ.**	**Max.**	**Unit**
Normal	Ta=85C, VBAT=18V, RCO=16MHz, full functionality: MCU running, no flash write, LED OFF, ADC ON, VBAT and TEMP monitor ON, WDT ON.			10	mA
Sleep Mode1	Main regulator (3.3V) ON, Load dump protection active. Ta=85degC max, VBAT=13.5V Overvoltage/Undervoltage detection, PWM, LED driver, Tempsensor and ADC are OFF except LIN Switch on and GPIO toggling and wake up timer.	70	85	100	uA
Sleep Mode2	Main regulator (3.3V) ON, Load dump protection active. Ta=85degC max, VBAT=13.5V Overvoltage/Undervoltage detection, PWM, LED driver, Tempsensor and ADC are OFF except one LIN RX on and GPIO toggling and wake up timer.	60	75	90	uA

7 DEVICE FUNCTIONAL DESCRIPTION

7.1 MCU FEATURES

7.1.1 MCU Core

The chip implements one ARM Cortex M0 core.

Additional documentation on ARM Cortex-M0 32 bits microcontroller can be found at

http://www.arm.com/products/processors/cortex-m/cortex-m0.php

7.1.2 System Memory (SRAM)

MCU core implements 16kbytes of SRAM. MCU can execute codes from the SRAM memories.

7.1.3 Flash Non Volatile Memory

MCU implements a Programmable Flash Memory with x32 configuration, sector and chip erase and byte program capability. It integrates five 512bytes nonvolatile registers (NVR) sectors.

In normal operation the ARM core fetches instructions (or data permanently stored) from the Flash memory but it is also possible for a program to alter the content of the flash memory. The following operations can be performed in the Flash Memory:

- Byte Write
- Sector Erase
- Block Erase
- Code Protect

For a description of the flash memory registers, please refer to the product register map. Here is a simple description of the basic features supported:

- Registers support to write/erase data to a byte, sector address
- Support programmable read wait states*
- Support system clock divider for write/erase functions
- Protection mechanism to unlock flash memory write and start flash memory byte-write
- Protection mechanism to unlock flash memory sector erase

* The design is implemented such that the timings associated with the flash macro meet the maximum speed of the system clock requirements.

7.1.4 Interrupt vectors

The first 148 bytes of Flash Memory are organized following the standard created by ARM. In this standard the Address 0x00000 contains the top-of-stack address (four bytes). The following addresses contain interrupt vectors used by the microcontroller:

Table 6 Interrupt Vector

Vector Name	Address	Comments
STACK_VALUE	0x00000	Typically set to 0x20000FFFF (Top of SRAM)
Reset_Handler	0x00004	Reset routine entry
Reserved	0x00008	Reserved. No NMI implemented.
HardFault Handler	0x0000C	
Reserved	0x00010 to 0x00028	
SVC_Handler	0x0002C	
Reserved	0x00030-0x00034	

PendSV_Handler	0x00038	
SysTick_Handler	0x0003C	
WULIN_Handler,	0x00040	Wake Up LIN Slave
Reserved,	0x00044	
WUTIMER_Handler,	0x00048	Wake Up Timer
BOR_Handler,	0x0004C	Brown out event
VBAT_Handler,	0x00050	Over/Under Voltage event
OVTEMP_Handler,	0x00054	Over Temperature event
WatchdogA_Handler,	0x00058	ASIC watchdog timeout
GPIO_Handler,	0x0005C	GPIO Interrupts
LINS_Handler,	0x00060	LIN Slave bus event
LINM_Handler,	0x00064	LIN Master bus event
UART_Handler,	0x00068	UART event
ADC_Handler,	0x0006C	ADC data ready
PWM_Handler,	0x00070	PWM event
BUCK_Handler	0x00074	Buck event
Reserved	0x00078	
Lullaby_Handler,	0x0007C	Software Interrupt
Timer0_Handler	0x00080	
Timer1_Handler	0x00084	
Timer2_Handler	0x00088	
Watchdog_Handler	0x0008C	

BTE_Handler	0x00090	Block Transfer – Contact indie to get more information.
Reserved	0x00094	

All other addresses in the flash memory can be used for the user's program.

The meanings of the standard interrupt vectors (Provided with the ARM Cortex M0 core) are defined in ARM's documentation. One of the sources of information is:

http://infocenter.arm.com/help/topic/com.arm.doc.dui0497a/DUI0497A_cortex_m0_r0p0_generic_ug.pdf

7.1.5 Interrupt Enabling/Disabling Process

Cortex-M0 implements a NVIC (**N**ested **V**ector **I**nterrupt **C**ontroller) peripheral capable of handling up to 16 peripheral's interrupts. Upon reset the microcontroller can answer only to Reset, NMI (Non-Maskable Interrupt) and Hard-Fault interrupts/exceptions. All other interrupts must be enabled. To enable and disable the interrupts the user must use access the ISER (Interrupt Set Enable Register) and ICER (Interrupt Clear Enable Interrupt) registers associated with the desired interrupt.

NOTE: Both inline functions and all parameters are defined in the product_file.h file, which must be included in the source files. Besides that the product_file.h file contains a list of available interrupts. The format of this list is as follows:

```
typedef enum IRQn
{
 //******  Cortex-M0 Processor Exceptions Numbers ***************************
 NonMaskableInt_IRQn       = -14,    // Non Maskable Interrupt
 HardFault_IRQn            = -13,    // Hard Fault Interrupt
 SVCall_IRQn               = -5,     // SV Call Interrupt
 PendSV_IRQn               = -2,     // Pend SV Interrupt
 SysTick_IRQn              = -1,     // System Tick Interrupt
```

```
  //*****  CM0IKMCU Cortex-M0 specific Interrupt Numbers **********************
  IRQ04_IRQn              = 0,     // Product specific
  IRQ04_IRQn              = 1,     // Product specific
  IRQ04_IRQn              = 2,     // Product specific
  IRQ04_IRQn              = 3,     // Product specific
  IRQ04_IRQn              = 4,     // Product specific
  IRQ05_IRQn              = 5,     // Product Specific
  IRQ06_IRQn              = 6,     // Product Specific
  IRQ07_IRQn              = 7,     // Product Specific
  IRQ08_IRQn              = 8,     // Product Specific
  IRQ09_IRQn              = 9,     // Product Specific
  IRQ10_IRQn              = 10,    // Product Specific
  IRQ11_IRQn              = 11,    // Product Specific
  IRQ12_IRQn              = 12,    // Product Specific
  IRQ13_IRQn              = 13,    // Product Specific
  IRQ14_IRQn              = 14,    // Product Specific
  IRQ15_IRQn              = 15,    // Product Specific
  TIMER0_IRQn              = 16,    // Timer 0
  TIMER1_IRQn              = 17,    // Timer 1
  TIMER2_IRQn             = 18,    // Timer 2
  WATCHDOG_IRQn             = 19     // Watchdog timer
} IRQn_Type;
```

7.1.6 Flash Code protection

The controlled access to the flash content is based on disabling all communications with the debug interface, therefore preventing any external attack. Hence, the application code is still able to modify the Flash content.

Upon Power-On Reset or Normal Reset, MCU core disables the communication with the debug interface for a small time interval (8192 system clock cycles). If the application needs to be protected it is mandatory to set the protection register with the appropriate code in the beginning of the initialization process and before the internal hardware enable the debug communication. In other words, if during this time interval the protection register is loaded with a specific pattern, then the communication remains disabled after the end of this interval and stays disabled until this register is loaded with a different pattern. To allow for debug communication the application has only to write a different value in the lock register.

If a part is protected, the emulator can still erase and program the part, but first it will be required to erase the Flash content, therefore protecting it.

7.1.7 Systick Timer

This timer is an optional peripheral created by ARM and implemented in the Cortex M0 160/8. It is fully described in the Cortex-M0 Devices Generic User Guide (Chapter 4.4 Optional System Timer, Systick) found at:

http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.dui0497a/Babieigh.html

7.1.8 Timers (0, 1 and 2)

The MCU implements three identical timers: Timer0, Timer1 and Timer2. All three timers operate using the system clock as clock source. They increment at the system clock rate starting from the loaded value in the counter until they roll over from 0xFFFFFFFF to 0x00000000. At this point an interrupt is generated if enabled. The interrupt routine is responsible for reloading the value if needed as this timer does not auto-reload the original content.

7.1.9 Watch Dog Timer

The MCU implements a WDT (**W**atch **D**og **T**imer) that can operate in one of two ways:

- Interrupt Mode: In the event of a WDT rollover an interrupt will be generated.
- Reset Mode: In the event of a WDT rollover the microcontroller will reset.

The WDT supports Reset, Enable, status/flag and clear functions. It integrates a pre-scaler that can divide the system clock by 2^{13}, 2^{19}, 2^{22} or 2^{32}. It means that the WDT internal counter will count from 0 to the pre-scaler value at the system clock speed and trigger if not cleared.

For instance, a system running from a 30MHz system clock and 2^{22} pre-scaler value will trigger the WDT after approximately 0.14 seconds if not cleared properly and in time by the application.

7.1.10 MCU Core to ASIC interface

The ASIC die will be communicating to the indie Cortex M0 through a proprietary interface. The interface used with Clough should be fully kept as is to enable any swap between ASIC die and MCU die.

7.2 CLOCK AND RESET SYSTEM

7.2.1 Clock Generation

Two clock sources are integrated in the device. The system clock is based on a RC network and will be trimmed to meet the accuracy requirements specified in the EC Table. Additionally an auxiliary clock will be used during sleep.

7.2.2 Reset

Both ASIC and MCU integrates Power on Reset (POR) circuitry: MCU POR monitors its input supply and generate a reset every time the MCU supply is recycled. ASIC POR monitors the main 3V3 LDO supply and generate a reset every time the LDO supply is recycled. Both POR maintain their output reset active as long as the monitored supply voltage is not above the minimum supply level to ensure safe logic operation of the Power Management Unit including the necessary analog features such as clock generator, bandgap, etc…This level is hardcoded and process technology dependent.

Additionally, the ASIC integrates Brown Out (BOR) circuit detectors that are configurable by SW (enable/disable as well as threshold programmable for the main ASIC core supply) and are actively monitored by the ASIC power management unit (PMU). In case any BOR is triggered, the PMU can be configured to either do the following (per BOR blocks).

- trigger a system reset
- generate an interrupt to the MCU for further actions.
- do nothing

The table below show the BOR levels settings for both VDD3P3 (ASIC Core supply) and VDD1P5 (MCU Core Supply). The * are the default settings values after reset and do not need to be changed as the thresholds are guaranteed to provide safe margin for full operation across PVT.

BOR Setting <3:0>	**BOR Level VDD3P3**		**BOR Level VDD1P5**	
	Threshold [V]	VBAT [V]	Threshold [V]	VBAT [V]
0	2.10	4.40	1.376*	5.1
1	2.10	4.40	1.344	5.03
2	2.10	4.40	1.315	4.95
3	2.32	4.47	1.285	4.88
4	2.39	4.55	1.413	5.19
5	2.46	4.63	1.449	5.28
6	2.56*	4.73	1.484	5.37
7	2.65	4.83	1.523	5.47
8	2.74	4.94	1.568	5.59
9	2.84	5.05	1.608	5.71
A	2.96	5.18		
B	3.08	5.32		
C	3.21	5.48		
D				
E				
F				

Table 7 BOR Trigger Level [* reset default]

Finally, the MCU watchdog timer (if configured to do so) or a reset instruction can reset the MCU logic while on the ASIC side the watchdog timer (if configured to do so) or a reset register write can reset the ASIC system.

The block diagram below illustrates the possible triggers of a reset on both side of the design: ASIC and MCU.

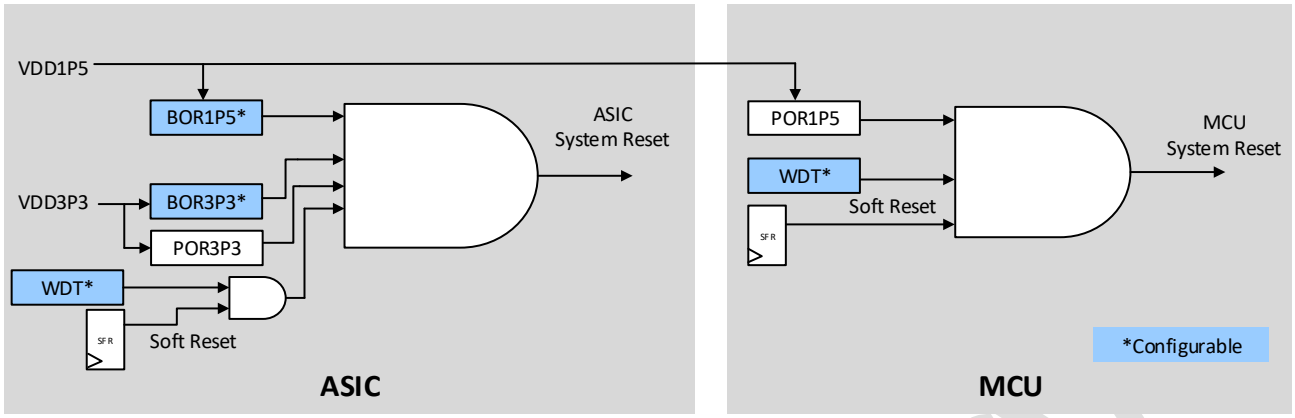


Figure 5 System Reset sources (Reset active LOW)

7.3 ANALOG FEATURES

7.3.1 PMU and Load Dump Protect circuits

ASIC integrates battery monitoring functionality that will detect load dump events occurring at the battery supply pin. An analog comparator will detect over voltage transients after going through a load dump limitation circuit. The protection circuit can handle DC voltage up to 45V and will limit these to maximum of *Over Voltage Threshold* nominal, therefore the comparator will detect any over voltage transients at the battery pin beyond that threshold. The circuit will generate a digital output signal to be filtered and sent to the interrupt controller. The circuit will be active during normal operation mode – not during sleep condition.

The voltage regulators themselves are supplied by VBAT directly and sustain DC level of load dump voltages.

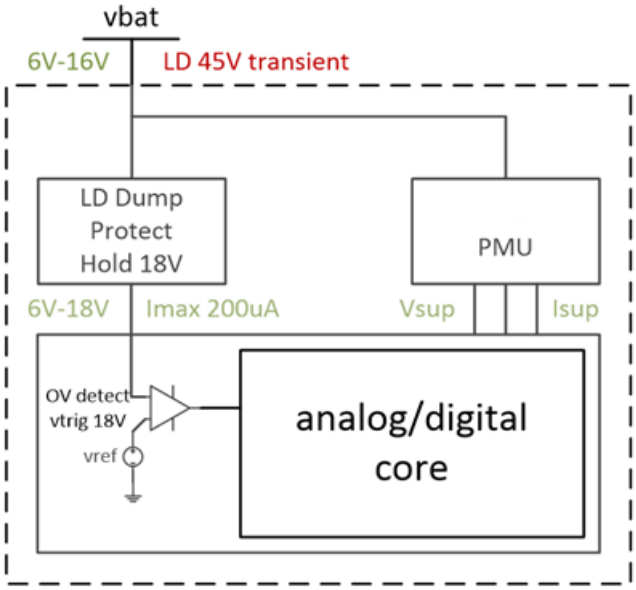


Figure 6 Load Dump Protect

7.3.2 LIN Transceiver

Supports LIN Protocol Controller according to LIN 2.x and SAE J2602 (rev J2602-1_201211). The IC contains an integrated PHY for low speed vehicle serial data network communication using the LIN protocol. It includes a wake up function using a dominant (low) bus pin message.

7.3.3 Buck Regulator

The IC integrates a switching synchronous buck converter with integrated high and low side switches to allow for small external components and reduced output ripple. The buck operates in forced PWM mode at a fixed frequency. Additionally, a bit can be set to operate the buck with a clock with spread spectrum frequency content that can reduce the peak radiated and conducting noise on both the input and output supplies, making it easier to comply with electromagnetic interference (EMI) standards. The buck should also support soft start feature to prevent high in rush current after enabling the regulator. The maximum in-rush current shall not exceed 10 times the nominal buck load (so about 3A).

7.3.4 LED Driver Stage

ASIC integrates a high precision open drain LED Driver Stage that allows for LED currents in the range of 100uA to 45mA in 100uA increments. The LED bias circuit uses a precise bandgap referenced

current(**CurrentV2I**) which is multiplied over stages to sink current via the LED which is connected to the HVIO(LED) pin. After factory test, the trim value for **CurrentV2I** = 25uA will be recorded and boot program is in charge of initializing the V2I trim bits correctly(refer to register description). The mirror stages consist of 100uA unit cells which are weighted linear to provide 100uA ~ 45mA current （CurrentLED）. The desired current is provided according the formula below:

$$\mathrm{CurrentLED(mA)} = \mathrm{TRIM}[8:0] * CurrentV2I * 4 = TRIM[8:0] * 100uA$$

Notice that if LED current is set to > 45mA, the accuracy is not guaranteed.

After delicate calibration by LED trim bits(refer to register description), the combination of stages allows for high accurate LED current in 100uA steps that are combined at the HVIO(LED) pad on chip (refer to Figure 7).

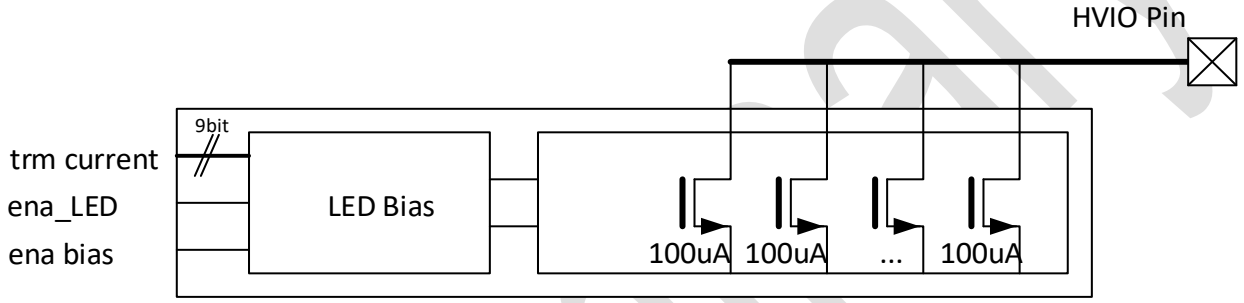


Figure 7 LED Driver Concept

The operation of the LED Driver IP requires two actions. At first the LED bias circuit needs to be enabled (ena_bias) (refer to), followed by enabling the LED (ena_LED). Latter control signal is driven by the pulse width modulator that will drive the current from the battery via the LED. The PWM signal utilizes an input signal at maximum of 250Hz and modulates its pulse width (refer to section 7.4.4 LED PWM).

7.3.5 House Keeping SAR ADC

- 10-bit resolution, single ended input
- Bandgap Voltage reference
- ADC is used for monitoring:
  - CH0: Band gap reference
  - CH1/CH3:
    - Accurate VBAT(1/32): Supply Voltage (limited to max voltage limited by the load dump protection circuit)
    - VDD5V(1/8)

        - Internal Temperature Sensor
    - CH2: Forward voltage of the external LEDs
    - CH4: GPIO analog inputs
    - CH5: VDD1V5
- ADC system capable of being configured for single or automatic multiple channel conversions (VBAT, HVIO and Temperature Sensor).
- Interrupt on conversion complete regardless of digital comparator configuration

ADC automatic sequencer:

The following diagram shows how the VBAT (CH1), LED Forward voltage-VFW (CH2) voltages are measured along with the junction temperature monitor (CH3). VFW voltages are converted from differential voltage to single-end voltage and shifted to ADC range(1/4 attenuation). This topology guarantees excellent (VBAT-HVIO) differential voltage measurement. The DC Voltage of VBAT can also be taken but with accuracy. While both PWM and ADC sequencer run from the same system clock (System RC Oscillator), the PWM signal is further downscaled while the ADC sampling clock can be adjusted to meet sample rate requirements. In order to reduce SW overhead, the PWM block provides a SYNC signal to the ADC sequencer which can then use this information to start the programmed conversions. In order to optimize the time taken to convert all channels (CH1,2,3) sequentially, the ADC sequencer can be configured to automatically start CHx conversion after TCURR and follow by the other channels after a duration set by TGUARD+TCHNL. TGUARD is the guard time where there is no channel selected, while switching from one channel in the sequence to the other to avoid any overlap. TCHNL is the time to wait after the guard time TGUARD for the conversions of 2nd or 3rd channel in the sequence, to allow settling of the channel before start of the conversion. The sequencer also provides a register that defines which channel need to be converted (1 to 3) and in which sequence. The sync feature is enabled by setting the SYNCENA bit and SYNCEDGE bit in ADC CNTRL Register. This enables the ADC conversion to be synchronized with the **positive edge or negative edge** of Sync input which is coming from PWM output. In short, ADC conversion is synchronized with the edge of the PWM output, if SYNCENA is set, and ADC is asked for conversion.

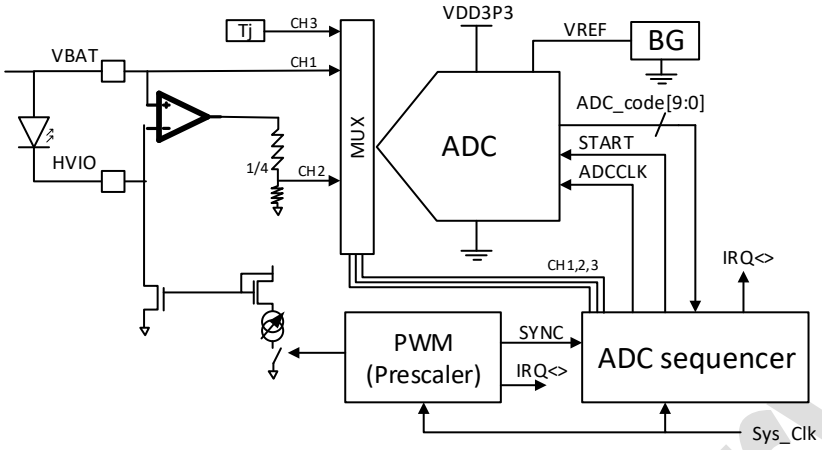


Figure 8 ADC synchronization with PWM

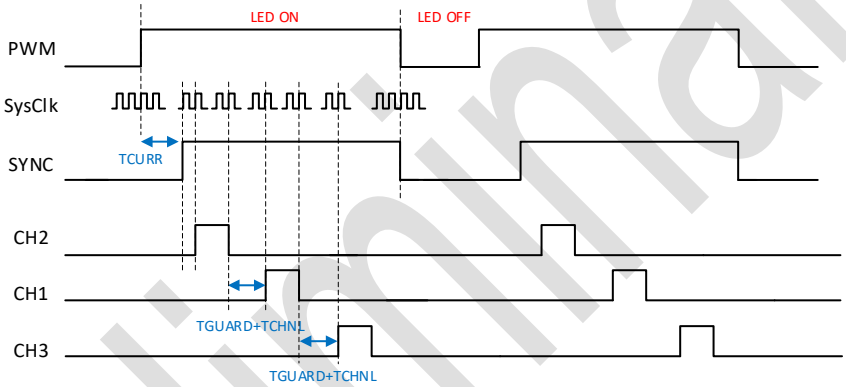


Figure 9 ADC read channels sequence triggered by PWM posedge

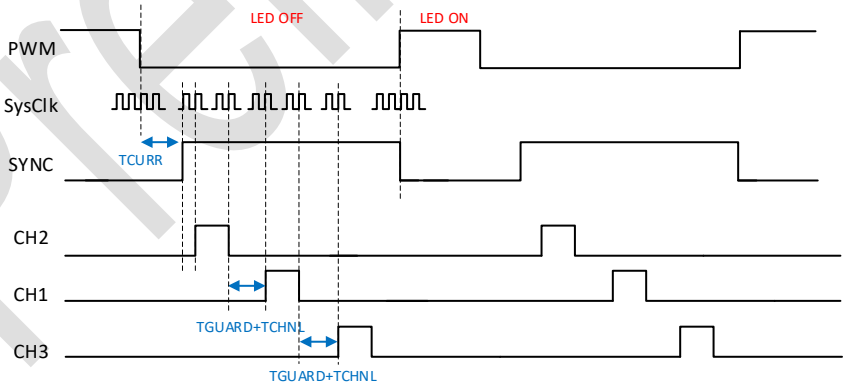


Figure 10 ADC read channels sequence triggered by PWM negedge

7.3.6 Over and Under Voltage detection (VBAT)

The over and under voltage comparators are based on comparing a divided voltage from the Load Dump limiter output feeding an analog comparator with hysteresis (refer to EC Table for electrical parameters and PMU description). The Over and Under voltage events generate Interrupts to the interrupt controller.

7.3.7 Temperature monitor

The MCU is in charge to pull the ADC related data from temperature sensor and in case the measured data is too high, MCU will reduce power profile to reduce heat. Table 8 shows the tempsensor output voltage corresponding to Tj from analog simulation for reference. Calibration is required due to different offset per chip.

Table 8 Tempsensor Output voltage vs Junction Temp

Junction Temp (°C)	Tempsensor Output Voltage(V) @VBAT=13.5V
-40	0.560568295
-30	0.585118532
-20	0.609746794
-10	0.634447014
0	0.659214959
10	0.68404771
20	0.70894409
30	0.733905116
40	0.758934482
50	0.784039045
60	0.809229285
70	0.834519924
80	0.859930773
90	0.885488251
100	0.911227796
110	0.937197714
120	0.963465673
130	0.990130866
140	1.017345206
150	1.045353086

7.3.8 Over Temperature detection

The over temperature comparator monitors the junction temperature with hysteresis. The Over temperature event generates reset or interrupt to the interrupt controller.

7.4 DIGITAL FEATURES

7.4.1 LIN controller

The LIN core is a communication controller that performs serial communication. It implements the datalink layer of the LIN Protocol Specification. LIN uses a single master / multiple slave concept for the message transfer between nodes of the LIN network. The LIN controller core comprises an interface to connect a micro controller that accesses the LIN core registers to control the transmission and reception of message frames.

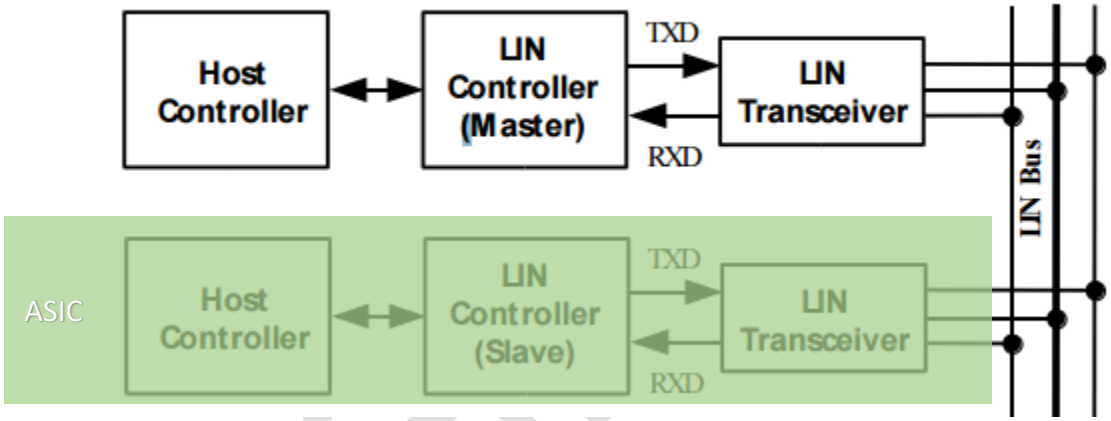


Figure 11 Lin System

Features:

- Support of LIN specification 2.2A
- Backward compatibility to LIN 1.3
- Programmable data rate between 1 Kbit/s and 20 Kbit/s (for master)
- Automatic bit rate detection (for slave)
- 8-byte data buffer
- 8-bit host controller interface
- Fully synchronous design, available in VHDL or Verilog, completely synthesizable
- Support auto addressing
- Note: Node configuration and diagnostics implemented by the host controller

7.4.1.1 LIN Usage Description

7.4.1.1.1 Data Length Register and Enhanced Checksum

The host controller has to define the length of the data field of the current LIN frame by adjusting the DATA LENGTH register. If the data length bits[3:0] are loaded with the value “1111b” the length of the data field is decoded from Bit 5 and 4 of the identifier register (ID) according to table below (e.g. compatibility to LIN specification 1.1). Otherwise the amount of data bytes can be written directly to the DATA LENGTH register (supported values are 0...8).

Table 9 ID bits and number of bits

ID (Bit 5)	ID (Bit 4)	Number of Bytes in the data field
0	0	2
0	1	2
1	0	4
1	1	8

The LIN core supports classic checksum (Spec 1.3, inverted eight bit sum with carry over all data bytes) and enhanced checksum (Spec. 2.0, inverted eight bit sum with carry over all data bytes and protected identifier). The host controller has to set the checksum type used in the current frame by adjusting Bit ENCHK in the data length register (‘1’ for enhanced checksum, ‘0’ for classic checksum).

7.4.1.1.2 Timing Settings for "Wake up Repeat Time" and "Bus Inactivity Time"

The time for repeating of wake up because of no reaction on the bus and for go to sleep because of inactivity on the bus can be optionally written by the host controller to registers WUPREPEAT and BUSINACTIVE (address 0x0F).

BUSINACTIVE [1:0]	LIN Inactivity Time (sec.)
00	4*
01	6
10	8
11	10

Table 10 LIN Inactivity Time

WUPREPEAT [1:0]	LIN Wake-Up Repeat Time (msec.)
00	180*
01	200
10	220
11	240

Table 11 LIN Wake-Up Repeat Time

7.4.1.1.3 Bit Time Settings

The Bit rate of the LIN system has to be defined in the bit timing registers.

Name	Description	Width(bits)
BTDIV	Bit time divider integer value	9
PRESCL	Clock Prescaler	2

Table 12 Bit Timing Related Registers

The LIN bit rate *f*bit can be calculated from system clock *f*clk and bit timing registers according to the following general equation:

$$f_{bit} = \frac{f_{clock}}{2^{prescl} * bt\_div * (bt\_mul + 1)}$$

Note that the procedure of adjusting the bit timing registers is different between master and slave. For the slave controller, the Bit timing register adjustment of slave is the following:

The steps for adjusting the bit timing registers of the LIN slave are explained below.

Note: Register bt_mul does not exist in the slave. The LIN core slave synchronizes to any bit rate between 1 Kbit/s and 20 Kbit/s. Nevertheless, the bit timing registers have to be adjusted to adapt the LIN core to the used system clock frequency.

- Setting up the pre-scaler register depending on system clock according to the following equation; the value has to be rounded down to the next integer value:

$$prescl = \ln\left(\frac{f_{clock}}{20KHz * 200}\right) * \frac{1}{\ln 2}$$

- Adjusting the bit time divider depending on system clock and pre-scaler according to the following equation; the value has to be rounded down to the next integer value:

$$bt\_div = \frac{f_{clock}}{2^{prescl} * 20KHz}$$

System Clock	PRESCL	BTDIV
8MHz	1	200
12MHz	1	300
16MHz	2	200

Table 13 Sample value for setting up bit timing registers

7.4.1.1.4 Controlling the LIN core (slave) by a host controller

The first step before transmitting messages with the LIN core is setting up the bit rate of the LIN system. For that, the host controller has to load the bit time registers. After that, the message transfer can be started. The LIN core slave detects the header of the message frame sent by the LIN master and synchronizes its internal bit time to the master bit time. An interrupt to the host controller is requested after the reception of the IDENTIFIER FIELD, after the reception of a wakeup signal (if the slave is in sleep mode), when an error is detected or when the message transfer is completed.

The following steps have to be done by the host controller when an interrupt is requested.

1) Check bit DATAREQ in the status register (it is 1 when the IDENTIFIER FIELD has been received). Proceed with the following if DATAREQ is set else proceed with step 2.

    a. Load the identifier from the ID register and process it.
    b. Adjust the bit TRANSMIT in the control register ("1" - if the current frame is a transmit operation for the slave, "0" – if the current frame is a receive operation for the slave).
    c. Load the data length in the data LENGHT register (number of data bytes or value "1111b" if the data length should be decoded from the identifier) and set the checksum type (enhanced checksum (Bit ENHCHK = '1') or classic checksum (Bit ENHCHK = '0')).
    d. Load the data to transmit into the data buffer (for transmit operation only).
    e. Set the bit DATAACK in the control register.

Note 1: Steps a..e have to be done during the IN-FRAME RESPONSE SPACE, if the current frame is a transmit operation for the slave; otherwise a timeout will be detected by the master. If the current frame is a receive operation for the slave, steps a..e have to be finished until the reception of the first byte after the IDENTIFIER FIELD. Otherwise, the internal receive buffer of the slave core will be overwritten and a timeout error will be detected in the slave core.

Note 2: If the host controller of the slave detects an unknown identifier (e.g. extended identifier) it has to write a '1' to bit "stop" in control register instead of setting bit DATAACK (steps b .. e can be skipped). In that case the LIN core slave stops the processing of the LIN communication until the next SYNC BREAK is received.

2) Check bit ERROR in the status register. Perform error handling and proceed with step 6 if bit ERROR is set else proceed with step 3.

Note 3: Bit TIMEOUT in the error register and bit WAKEUP in the status register are set if the slave has sent a wakeup signal but the master did not respond within 150ms.

3) Check bit BUSIDLETIMEOUT in the status register and activate the sleep mode by setting bit SLEEP in the control register if BUSIDLETIMEOUT is set.
4) Check bit WAKEUP in the status register (it is set if the slave has received a wakeup signal). If WAKEUP is set proceed with step 6 else proceed with step 4.

Note 4: Bit COMPLETE in the status register is not changed when a wakeup signal is transmitted or received. Therefore, bit WAKEUP has to be checked before bit COMPLETE

5) Check bit COMPLETE in the status register (it is set if the transmission was successful). If COMPLETE is set and the current frame was a receive operation for the slave, load the received data bytes from the data buffer.
6) Set the bits RSTINT (reset interrupt) and RSTERR (reset error) in the control register to reset the interrupt request and the error flags.

7.4.1.1.5 Sleep Mode and Wakeup

To reduce the systems power consumption the LIN Protocol Specification defines a Sleep Mode. The message used to broadcast a Sleep Mode request has to be started by the host controller of the LIN core master in the same way as a normal transmit message. The host controller of the LIN core slave has to decode the Sleep Mode Frame from Identifier and data bytes. After that, it has to put the LIN slave node into the Sleep Mode by setting bit SLEEP in the control register. If bit SLEEP in the control register of the LIN core slave is not set and there is no bus activity for 4 s to 10 s (specified bus idle timeout) bit BUSIDLETIMOUT is set and an interrupt request is generated. After that application may assume that the LIN bus is in Sleep Mode and set bit SLEEP in the control register of the LIN core slave. The bus inactivity time which should be defined as bus idle timeout for the slave can optionally set to values 4s, 6s, 8s or 10s as possible accordingly with specification 2.2A.

After receiving a Wakeup signal from the master or any slave node an a wakeup request is generated, the host controller terminates the Sleep Mode of the LIN bus by clearing bit SLEEP in the control register.

To send a Wakeup signal, the host controller of the LIN core has to set the bit WAKEUPREQ in the control register. After successful transmission of the wakeup signal with the LIN core master the WAKEUP bit in the status register of the sending LIN core master is set and an interrupt request is generated. The LIN core slave does not generate an interrupt request after successful transmission of

the Wakeup signal but it generates an interrupt request if the master does not respond to the Wakeup signal within 150ms to 250ms. This value can be set optionally to 180ms, 200ms, 220ms or 240ms as it is possible accordingly with specification 2.2A. In that case, bit ERROR and bit TIMOUT are set**.** The host controller has to decide whether to transmit another Wakeup signal or not.

7.4.1.1.6 Error Detection and Handling

The LIN core generates an interrupt request and stops the processing of the current frame if it detects an error. The application has to check the type of error by processing the ERROR register. After that, it has to reset the ERROR register and the ERROR bit in status register by writing a 1 to bit RSTERR in control register. Starting a new message with the LIN core master or sending a Wakeup signal with master or slave is possible only if bit ERROR in status register is 0.

7.4.1.2 Auto Addressing(Lin Switch Mode):

While RuGBy integrates a lin switch and two LIN interface pins, LIN_IN and LIN_OUT, connecting to LIN transceiver. Software is in charge to control the lin switch through IOCTRLA LIN control registers. When SWON is set to 1, the lin switch will connect LIN_IN and LIN_OUT. At the same time, the lin master controller will lose the connection with the lin master transceiver.

With this lin switch, RuGBy can support auto addressing function. BCM connects RuGBy in a chain by connecting LIN_IN to upstream LIN bus, and connecting LIN_OUT to downstream LIN bus, as Figure 10. At the first boot of the system, every switch is on. The following is a SNPD sequence for instance:

1. Lin Master sends a diagnostic frame (ID=3C, 8 bytes data frame) to inform the slaves that SNPD sequence is started. Then the slaves disconnect downstream LIN bus by opening the internal switch. Thus, only the first one, RuGBy 1, can receive message from BCM via LIN_IN pin.
2. Lin Master sends 1st NAD configure frame with NAD=”01”. Thus “01” address is assigned to RuGBy 1 at first. And then RuGBy 1 close its internal LIN switch, thus RuGBy 2 can receive message from LIN bus.
3. Then system can assign the second address for RuGBy 2 accordingly. By analogy, all RuGBys on the chain can be assigned address.

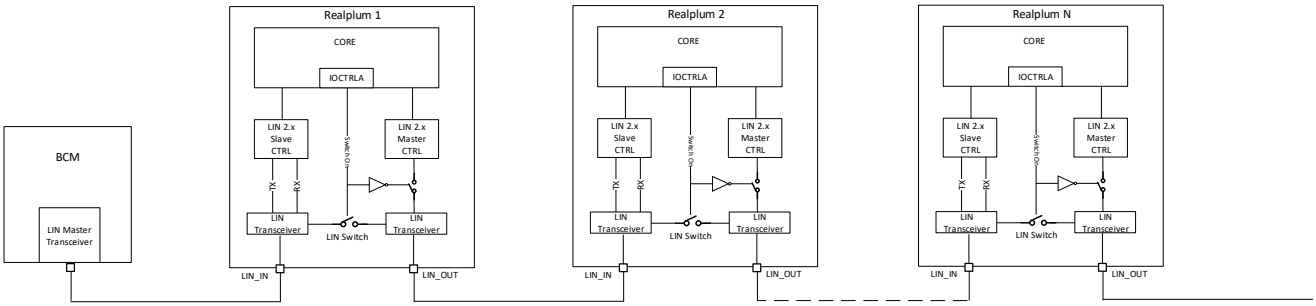


Figure 10 LIN auto addressing

Cautions:

1. LIN protocol requires inter-frame space not negative. That means any space > = 0 is acceptable. Since the internal switch is controlled by software, there are a big latency from frame complete interrupt to control the switch by interrupt routine. So a big inter-frame space is required during SNPD sequence, while this is determined by LIN master.

7.4.2 UART

The UART circuit includes a 'baud' rate generator with software-programmable divider ratios for rates from 100kHz to 2MHz. The UART features a 32-bytes-deep receive/transmit FIFO that minimizes the microcontroller overhead command and provides a set of maskable interrupt support to monitor and manage TX and RX data paths. The UART does not include any HW to filter out received messages that are not matching the LIN ID address. In other words, the address associated with the master command will have to be decoded and executed if the ID matches. It is expected that given the clock source, the embedded FW may have to compensate for clock frequency drift over temperature to keep a reliable communication path between nodes.

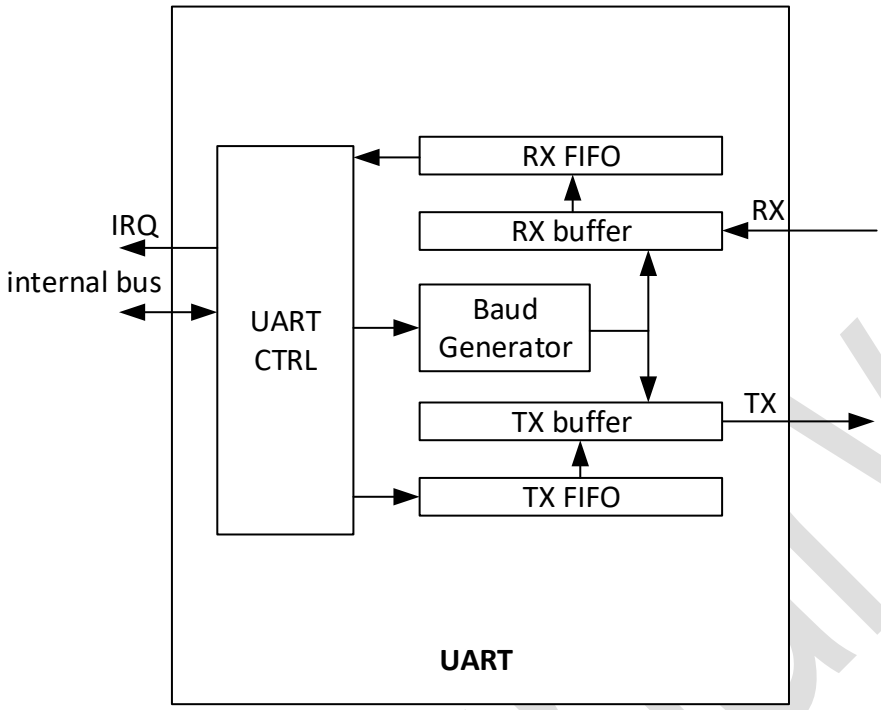


Figure 15 UART Interface

7.4.3 ASIC Watchdog Timer

While it exists a watchdog timer in the MCU, the ASIC integrates another watch-dog timer that is intended to be used to recover from a situation where there may have been a software fault or other system failure where the software has ceased to operate correctly. If the timer is not reset by software periodically, it will time out, and this event can be used to reset the system.

Features:

- Programmable timeout period (Refer to EC table) with instantaneous access to the value of watchdog timeout counter
- Status flag, stop and clear/reset control registers
- The Watchdog timer is by default active after power up and set to its maximum duration setting (EC table)

7.4.4 LED PWM

LED PWMs are used to control accurately the light intensity.

Features:

- 18x16bit PWM channels with shared period length, independent pulse rise and independent pulse fall timestamps.
- Frequency and duty cycle of PWM waveform support up to a maximum Resolution of 16 bits. The 16-bit resolution is only achievable if the pre-scaler from system clock is correctly chosen to have enough clock cycles depth to count up to 2^16.
- Programmable pre-scaler: system clock division (PWM_DIV)
- Programmable PWM Period (PWM_PER)

$$\text{Period} = \frac{1 + (\text{PWM\_PER} \times \text{PWM\_DIV})}{SystemClock}$$

- Programmable duty cycle 0 to 100% (PWM_PW)

$$\text{PulseWidth} = \frac{1 + ((\text{PWM}_{\text{PFALL}} - \text{PWM}_{\text{PRISE}}) \times \text{PWM\_DIV}))}{SystemClock}$$

- Pre-scaler, period, pulse rise & pulse fall configurations will be updated at the end of the current output period.
- Support interrupt generation when new programmed PWM control data become active. After new pulse parameters (Period, pulse rise & pulse fall) have been loaded into their respective registers, an UPDATE bit can be set to 1 that will trigger the activation of the new parameters at the end of the current pulse as not to affect the pulse shape. Basically the UPDATE bit clear is the interrupt.
- PWM Frequency range 80 - 250 Hz
- Pulse rise -> fall cases, listed in priority:
    - PRISE = 0, PFALL = PERIOD: 100% On
    - PRISE = 0, PFALL = 0: equivalent to PRISE = 0 & PFALL = 1
    - PRISE > PFALL: 100% off
    - PRISE = PFALL: 100% off
    - PRISE < PFALL: Normal case. On at PRISE, Off at PFALL.

7.4.5 Sleep Mode

IC must be able to enter SLEEP mode through SW request. The device should be able to come out of SLEEP with either the slow auxiliary or the system clock. It's up to the SW to select which clock shall be used after a wake event is detected. The SW should not have to request to go to sleep with the same clock selected for wake up.

7.4.6 Wake up Mode

Coming out from SLEEP mode can happen through the following events:

- After a low pulse on the LIN pin such that a dominant (low) voltage level is applied for longer than TWAKEUP time.
- GPIOs pin toggling either from high to low or low to high levels (VIL/VIH) and maintaining the active state for more than TWAKEUP time.
- Wake up timer. Programmable range, refer the EC Table. Wake up timer should have the option to be disabled.

MCU should be able to check which wake up events triggered the system through a status register read. GPIOs trigger should be consolidated into 1 status register. MCU to clear the register after status check.

8 BOM

#	Description	Quantity	Comment
CBATx	2.2uF 50V X7R	1x	
CV1p5	4.7uF 10V	1x	
CV3p3	10uF 10V	1x	
CLIN	220pF 0603 X7R	2x	one CLIN in master and one CLIN in slave. Only one CLIN in slave if no auto addressing is required.
C1	20uF	2x	Buck converter output capacitor
CBST	100nF	1x	Boostrap cap
L1	Size (AxBxHeight?) L=10uH DCR<100mOhm	1x	Buck power inductor
C3		1x	Buck input capacitor

9 ERRATA

9.1 ADC CONTINUOUS MODE IRQ HANDLING

The hardware implementation of continuous mode can't be trusted as it's depending on the irq subroutine execution time vs the time required to do a set sequence of conversion. For a typical interrupt handler, clear interrupt flag is required in order to allow new interrupt to be fired. But for ADC in continuous mode, the interrupts occur very intensively, new interrupt can be fired while current interrupt handling is still not yet completed, in such use cases the IRQCLR is ineffective if an new interrupt event occurs at the same time, potentially it could block any new ADC interrupt event to be fired. To avoid of such conflicts, firmware needs do additional check in the interrupt handler to make sure ADC interrupt flag is cleared successfully.

Here is an example implementation of ISR where after IRQCLR, do a check to see if there is new ADC has been done, if yes, set IRQCLR to again.

```
void ADC_Handler(void)
{
   ADC_SFRS->CNTRL.IRQCLR = 1
   while(ADC_SFRS->STATUS.CONVDONE)
      ADC_SFRS->CNTRL.IRQCLR = 1;
}
```

Alternatively, user can choose not to use ADC in continuous mode and use single conversion and setting the CONVERT every time before leaving the IRQ handler.