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

ATPL230A

ATPL Series Power Line Communications Device

DATASHEET

Description

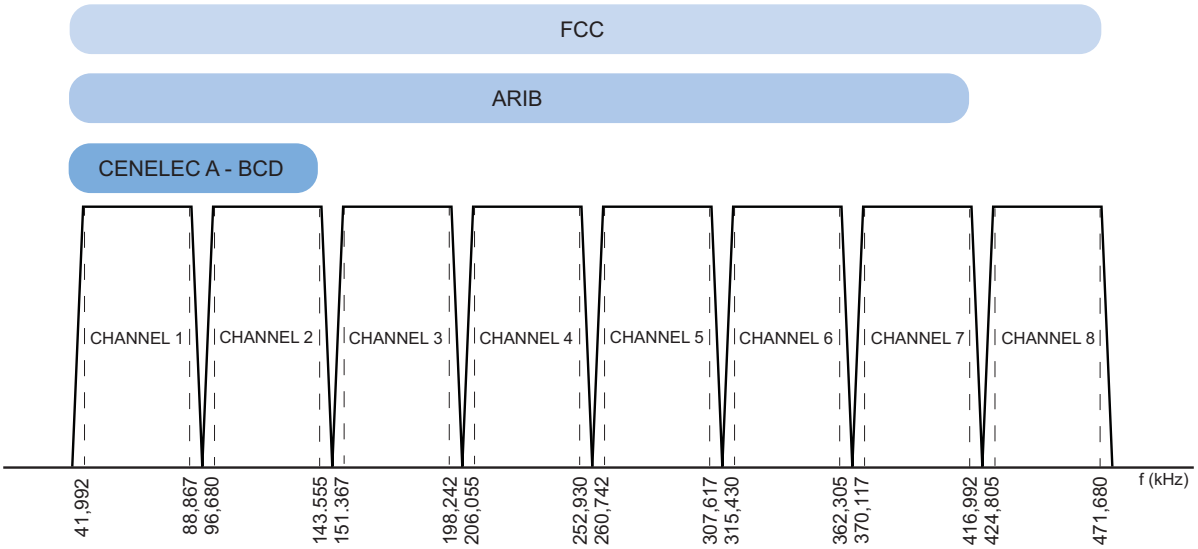
ATPL230A is a power line communications base band modem, compliant with the PHY layer of PRIME (Power Line Intelligent Metering Evolution) specification. PRIME is an open standard technology used for Smart Grid applications like Smart Metering, Industrial Lighting and Automation, Home Automation, Street Lighting, Solar Energy and PHEV Charging Stations.

ATPL230A PRIME device includes enhanced features such as additional robust modes and frequency band extension. ATPL230A is able to operate in independently selectable transmission bands up to 472 kHz, achieving baud rates ranging from 5.4 kbps up to 128.6 kbps.

ATPL230A has been conceived to be bundled with an external Atmel® MCU or MPU. Atmel provides a PRIME PHY layer library which is used by the external MCU/MPU to take control of ATPL230A PHY layer device.

1. Features

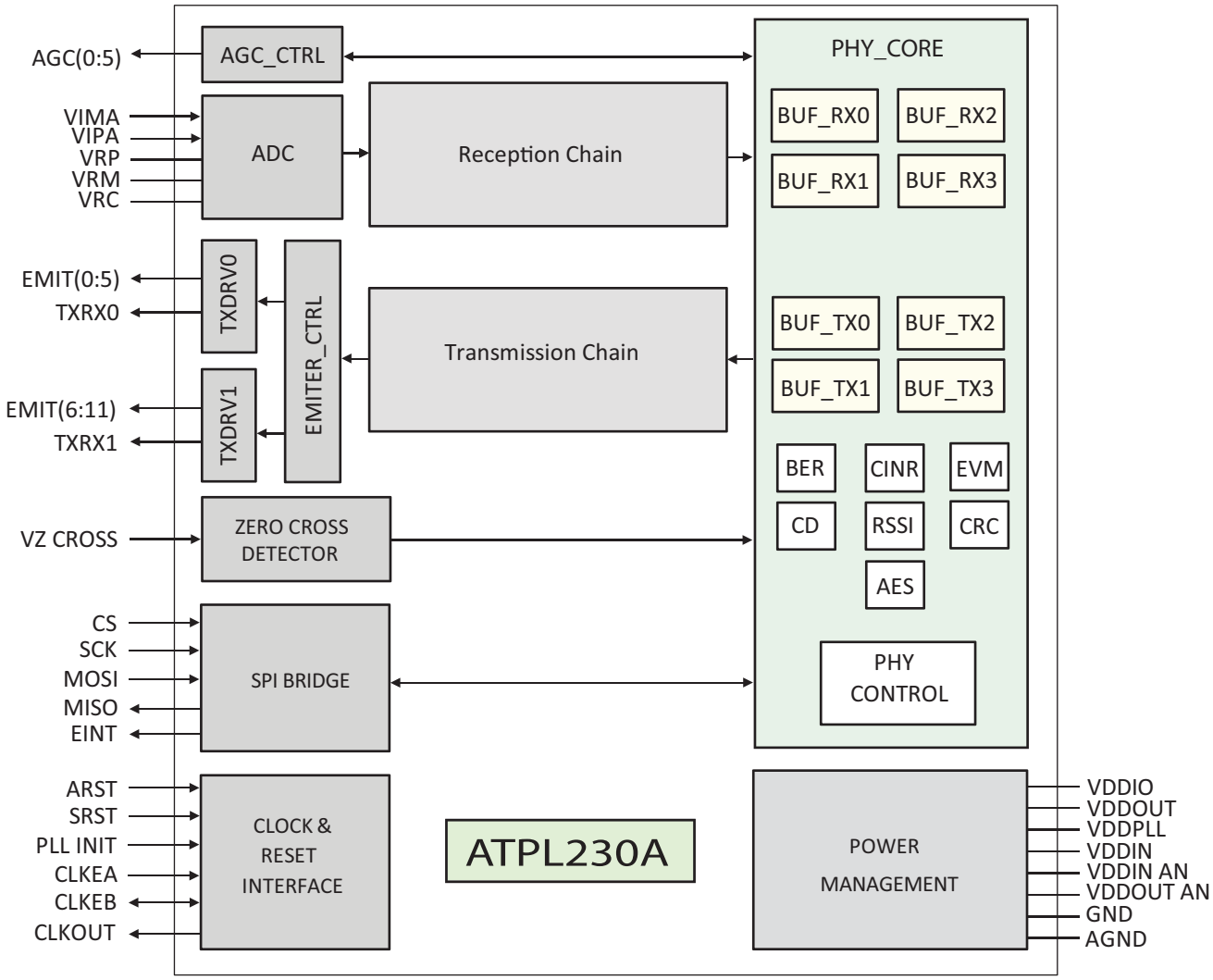
- Modem
  - Power Line Carrier Modem for 50 Hz and 60 Hz mains
  - 97-carriers OFDM PRIME compliant
  - DBPSK, DQPSK, D8PSK modulation schemes available
  - Additional enhanced modes available: DBPSK Robust and DQPSK Robust
  - Eight selectable channels between 42kHz and 472kHz available. Only one channel can be active at a time



  - Baud rate Selectable: 5.4 to 128.6 kbps
  - Four dedicated buffers for transmission/reception
  - Up to 124.6 dBµVrms injected signal against PRIME load
  - Up to 79.6 dB of dynamic range in PRIME networks
- Automatic Gain Control and continuous amplitude tracking in signal reception
- Class D switching power amplifier control
- Integrated 1.2V LDO regulator to supply analog functions
- Medium Access Control co-processor features
  - Viterbi soft decoding and PRIME CRC calculation
  - 128-bit AES encryption
  - Channel sensing and collision pre-detection

2. Block Diagram

Figure 2-1. ATPL230A Functional Block Diagram



3. Signal Description

Table 3-1. Signal Description List

Signal Name	Function	Type	Active Level	Voltage reference	Comments
Power Supplies					
VDDIO	3.3V digital supply. Digital power supply must be decoupled by external capacitors	Power			3.0V to 3.6V
VDDIN	3.3V Digital LDO input supply	Power			3.0V to 3.6V
VDDIN AN	3.3V Analog LDO input supply	Power			3.0V to 3.6V
VDDOUT AN	1.2V Analog LDO output. A capacitor in the range 0.1µF - 10µF must be connected to each pin	Power			1.2V
VDDOUT	1.2V Digital LDO output. A capacitor in the range 0.1µF - 10µF must be connected to each pin	Power			1.2V
VDDPLL	1.2V PLL supply. It must be decoupled by a 100nF external capacitor, and connected to VDDOUT through a filter (Cut off frequency: 25kHz)	Power			1.2V
GND(1)	Digital Ground	Power			
AGND(1)	Analog Ground	Power			
Clocks, Oscillators and PLLs					
CLKEA(2)	External Clock Oscillator • CLKEA must be connected to one terminal of a crystal (when a crystal is being used) or used as input for external clock signal	Input		VDDIO	
CLKEB(2)	External Clock Oscillator • CLKEB must be connected to one terminal of a crystal (when a crystal is being used) or must be floating when an external clock signal is connected through CLKEA	I/O		VDDIO	
CLKOUT	10MHz External Clock Output	Output		VDDIO	
Reset/Test					
ARST	Asynchronous Reset	Input	Low	VDDIO	Internal pull up(3)
SRST	Synchronous Reset	Input	Low	VDDIO	Internal pull up(3)
PLL INIT	PLL Initialization Signal	Input	Low	VDDIO	Internal pull up(3)
PPLC (PRIME Power Line Communications) Transceiver					
EMIT [0:11](4)	PLC Tri-state Transmission ports	Output		VDDIO	
AGC [0:5]	Automatic Gain Control: • These digital tri-state outputs are managed by AGC hardware logic to drive external circuitry when input signal attenuation is needed	Output		VDDIO	

Table 3-1. Signal Description List

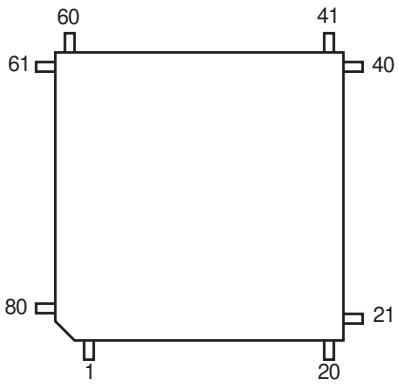
Signal Name	Function	Type	Active Level	Voltage reference	Comments
TXRX0	Analog Front-End Transmission/Reception for TXDRV0 • This digital output is used to modify external coupling behavior in Transmission/Reception. The suitable value depends on the external circuitry configuration. The polarity of this pin can be inverted by software.	Output		VDDIO	
TXRX1	Analog Front-End Transmission/Reception for TXDRV1 • This digital output is used to modify external coupling behavior in Transmission/Reception. The suitable value depends on the external circuitry configuration. The polarity of this pin can be inverted by software.	Output		VDDIO	
VZ CROSS(5)	Mains Zero-Cross Detection Signal: • This input detects the zero-crossing of the mains voltage	Input		VDDIO	Internal pull down(3)
VIMA	Negative Differential Voltage Input	Input		VDDOUT AN	
VIPA	Positive Differential Voltage Input	Input		VDDOUT AN	
VRP	Internal Reference “Plus” Voltage. Connect an external decoupling capacitor between VRP and VRM (1nF - 100nF)	Output		VDDOUT AN	
VRM	Internal Reference “Minus” Voltage. Connect an external decoupling capacitor between VRP and VRM (1nF - 100nF)	Output		VDDOUT AN	
VRC	Common-mode Voltage. Bypass to analog ground with an external decoupling capacitor (100pF - 1nF)	Output		VDDOUT AN	
Serial Peripheral Interface - SPI					
CS	SPI CS • SPI bridge Slave Select	Input	Low	VDDIO	Internal pull up(3)
SCK	SPI SCK • SPI bridge Clock signal	Input		VDDIO	Internal pull up(3)
MOSI	SPI MOSI • SPI bridge Master Out Slave In	Input		VDDIO	Internal pull up(3)
MISO	SPI MISO • SPI bridge Master In Slave Out	Output		VDDIO	
EINT	PHY Layer External Interrupt	Output	Low	VDDIO	

Notes:
1. Separate pins are provided for GND and AGND grounds. Layout considerations should be taken into account to reduce interference. Ground pins should be connected as shortly as possible to the system ground plane. For more details about EMC Considerations, please refer to AVR040 application note.
2. The crystal should be located as close as possible to CLKEA and CLKEB pins. See Table 10-7 on page 112.
3. See Table 10-5 on page 109.
4. Different configurations allowed depending on external topology and net behavior.
5. Depending on whether an isolated or a non-isolated power supply is being used, isolation of this pin should be taken into account in the circuitry design. Please refer to the Reference Design for further information.

4. Package and Pinout

4.1 80-Lead LQFP Package Outline

Figure 4-1. Orientation of the 80-Lead LQFP Package



4.2 80-Lead LQFP Pinout

Table 4-1. 80 - Lead LQFP Pinout

1	NC	21	VDDIO	41	GND	61	GND
2	NC	22	NC	42	EMIT8	62	AGND
3	NC	23	CLKOUT	43	EMIT9	63	VDDOUT AN
4	ARST	24	CS	44	EMIT10	64	VIMA
5	PLL INIT	25	SCK	45	EMIT11	65	VIPA
6	GND	26	MOSI	46	VDDIO	66	VDDOUT AN
7	CLKEA	27	MISO	47	GND	67	AGND
8	GND	28	VDDIO	48	VDDOUT	68	VRP
9	CLKEB	29	GND	49	TXRX0	69	VRM
10	VDDIO	30	EMIT0	50	TXRX1	70	VRC
11	GND	31	EMIT1	51	GND	71	VDDIN AN
12	VDDPLL	32	EMIT2	52	AGC2	72	AGND
13	GND	33	EMIT3	53	AGC5	73	AGND
14	VDDIN	34	VDDIO	54	AGC1	74	VDDIN AN
15	VDDIN	35	GND	55	AGC4	75	GND
16	GND	36	EMIT4	56	AGC0	76	VDDIO
17	VDDOUT	37	EMIT5	57	AGC3	77	VZ CROSS
18	GND	38	EMIT6	58	VDDIO	78	NC
19	NC	39	EMIT7	59	GND	79	NC
20	SRST	40	VDDIO	60	EINT	80	NC

5. Analog Front-End

5.1 PLC coupling circuitry description

Atmel PLC coupling reference designs have been designed to achieve high performance, low cost and simplicity.

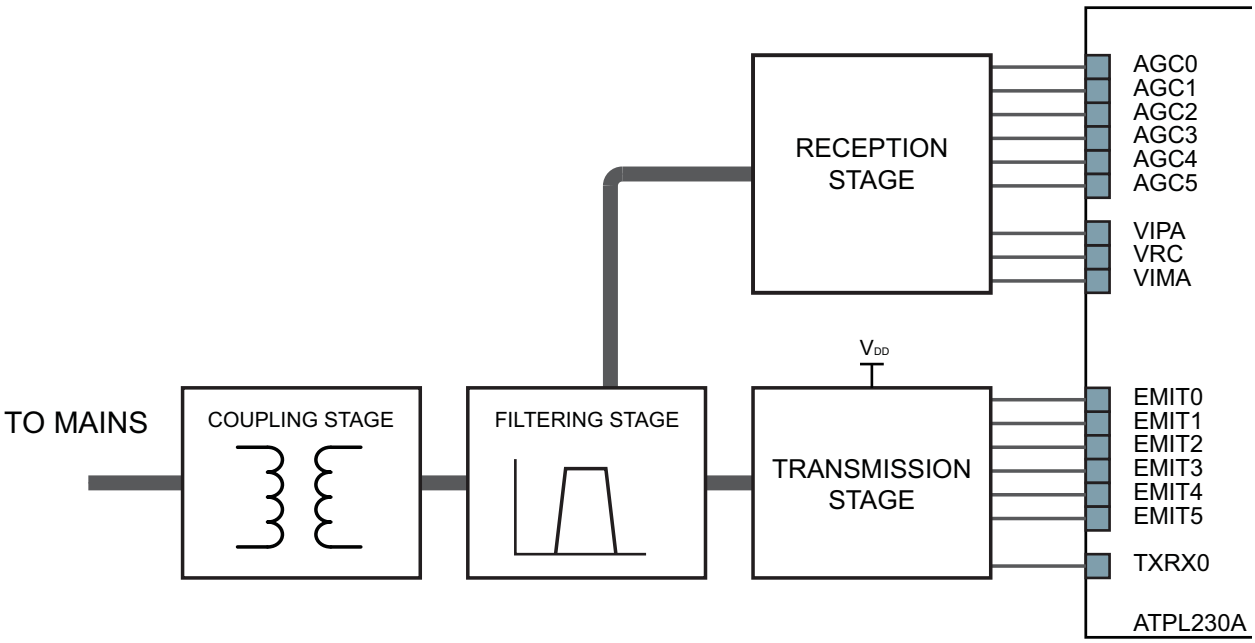
With these values on mind, Atmel has developed a set of PLC couplings covering frequencies up to 472 kHz compliant with different applicable regulations.

Atmel PLC technology is purely digital and does not require external DAC/ADC, thus simplifying the external required circuitry. Generally Atmel PLC coupling reference designs make use of few passive components plus a Class D amplification stage for transmission.

All PLC coupling reference designs are generally composed by the same sub-circuits:

- Transmission Stage
- Reception Stage
- Filtering Stage
- Coupling Stage

Figure 5-1. PLC coupling block diagram



A particular reference design can contain more than one sub-circuit of the same kind (i.e.: two transmission stages).

5.1.1 Transmission Stage

The transmission stage adapts the EMIT signals and amplifies them if required. It can be composed by:

- Driver: A group of resistors which adapt the EMIT signals to either control the Class-D amplifier or to be filtered by the next stage.
- Amplifier: If required, a Class-D amplifier which generates a square waveform from 0 to V_{DD} is included.
- Bias and protection: A couple of resistors and a couple of Schottky barrier diodes provide a DC component and provide protection from received disturbances.

Transmission stage shall be always followed by a filtering stage.

5.1.2 Filtering Stage

The filtering stage is composed by band-pass filters which have been designed to achieve high performance in field deployments complying at the same time with the proper normative and standards.

The in-band flat response filtering stage does not distort the injected signal, reduces spurious emission to the limits set by the corresponding regulation and blocks potential interferences from other transmission channels.

The filtering stage has three aims:

- Band-pass filtering of high frequency components of the square waveform generated by the transmission stage
- Adapt Input/Output impedances for optimal reception/transmission. This is controlled by TXRX signals
- In some cases, Band-pass filtering for received signals

When the system is intended to be connected to a physical channel with high voltage or which is not electrically referenced to the same point then the filtering stage must be always followed by a coupling stage.

5.1.3 Coupling Stage

The coupling stage blocks the DC component of the line to/from which the signal is injected/received (i.e.: 50/60 Hz of the mains). This is carried out by a high voltage capacitor.

Coupling stage could also electrically isolate the coupling circuitry from the external world by means of a 1:1 transformer.

5.1.4 Reception Stage

The reception stage adapts the received analog signal to be properly captured by the ATPL230A internal reception chain. Reception circuit is independent of the PLC channel which is being used. It basically consists of:

- Anti aliasing filter (RC Filter)
- Automatic Gain Control (AGC) circuit
- Driver of the internal ADC

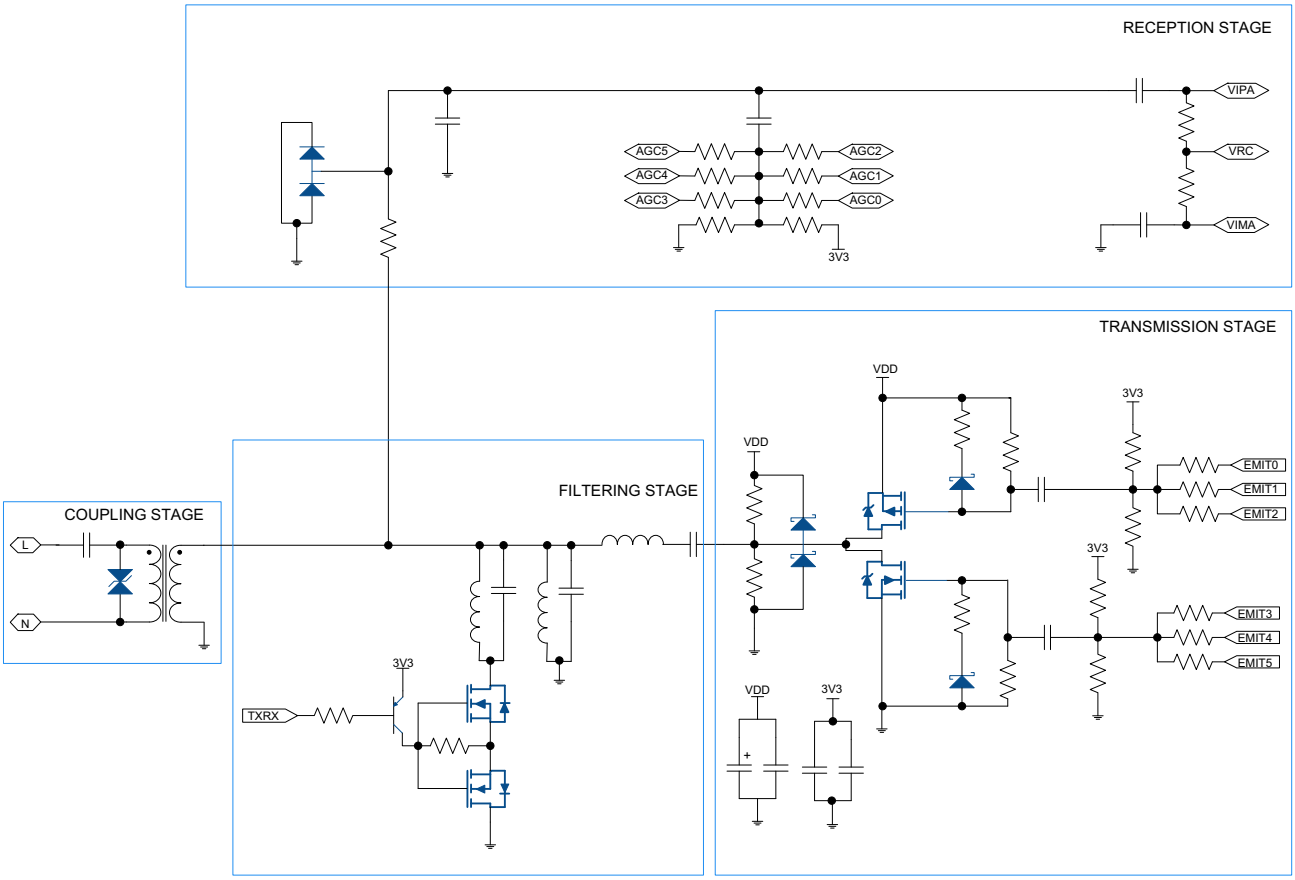
The AGC circuit avoids distortion on the received signal that may arise when the input signal is high enough to polarize the protective diodes in direct region.

The driver to the internal ADC comprises a couple of resistors and a couple of capacitors. This driver provides a DC component and adapts the received signal to be properly converted by the internal reception chain.

5.1.5 Generic PLC Coupling

Please consider that this is a generic PLC Coupling design for a particular application please refer to Atmel doc43052 “PLC Coupling Reference Designs”.

Figure 5-2. PLC Coupling block diagram detailed



5.2 ATPLCOUP reference designs

Atmel provides PLC coupling reference designs for different applications and frequency bands up to 500 kHz. Please refer to Atmel doc43052 “PLC Coupling Reference Designs” for a detailed description.

5.3 Zero-crossing detection

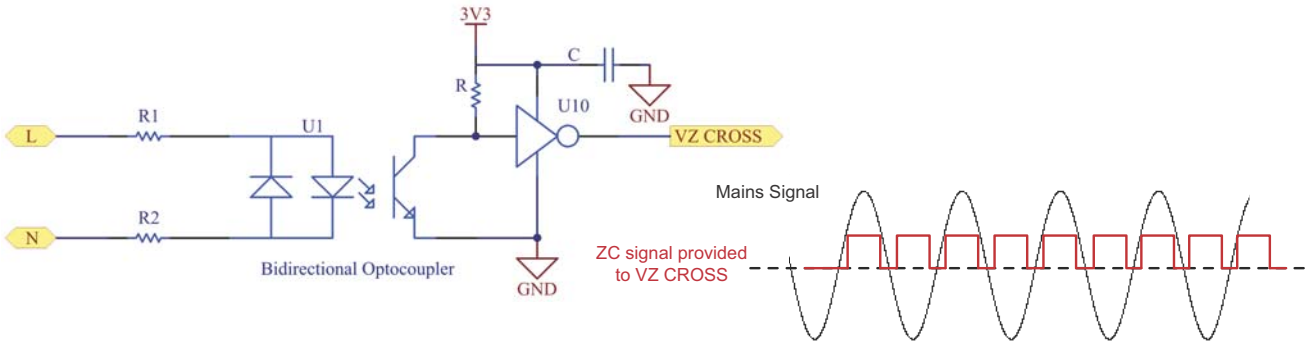
5.3.1 Overview

Zero Crossing Detector block works predicting future zero crossings of the Mains signal in function of its past zero crossings. To achieve this, the system embeds a configurable Input Signal Management (ISM) block and a PLL, both of which manage Zero Crossing Detector Input Signal to calculate Zero Crossing Output Flag. The zero-cross detection of waves of 50 Hz and 60 Hz with ±10% of error is supported.

The PLL block interprets its input signal such a way that it indicates a zero cross in the middle of a positive pulse. It is important to note that depending on the external circuit implementing the Zero Crossing Detector Input Signal this interpretation is not always correct. Thus, for some cases it is required to transform the Input Signal in a signal where the middle of a positive pulse corresponds to a truly zero cross. This transformation is implemented through the Input Signal Management (ISM) block, configured by MODE_INV and MODE_REP fields in ZC_CONFIG register.

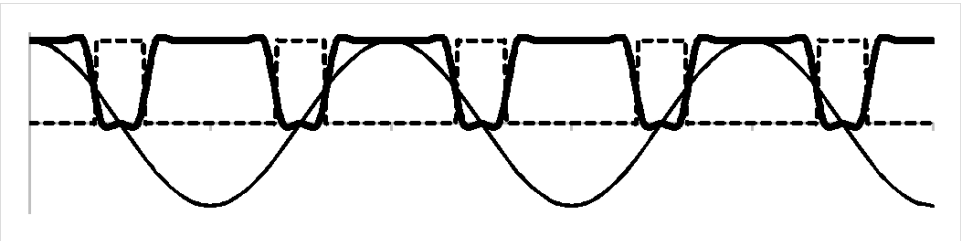
Zero Crossing Detector Input Signal (VZ CROSS) must fulfil some requirements. The first requirement is that VZ CROSS signal must be a pulse train with a duty cycle being >60% or <40% (polarity is configurable). In addition, if we have to detect ascent or descent zero-crossing, the Zero Crossing Detector Input Signal period must be equal than the period of the wave we need to obtain zero-crossing. Ascent and descent Zero Crossing Detection are configured by setting MODE_MUX and MODE_ASC fields in ZC_CONFIG register.

Figure 5-3. Typical circuit, using a bidirectional optocoupler and a Schmitt trigger



The input signal “VZ CROSS” (wider line) generated by this circuit for Zero Cross Detection of the wave “L”-“N” (finer line) is plotted in next figure. The digital signal at output of Input Signal Management (ISM) is plotted in Figure 5-4.

Figure 5-4. Digital signal (dashed line) at output of Input Signal Management (ISM) internal block

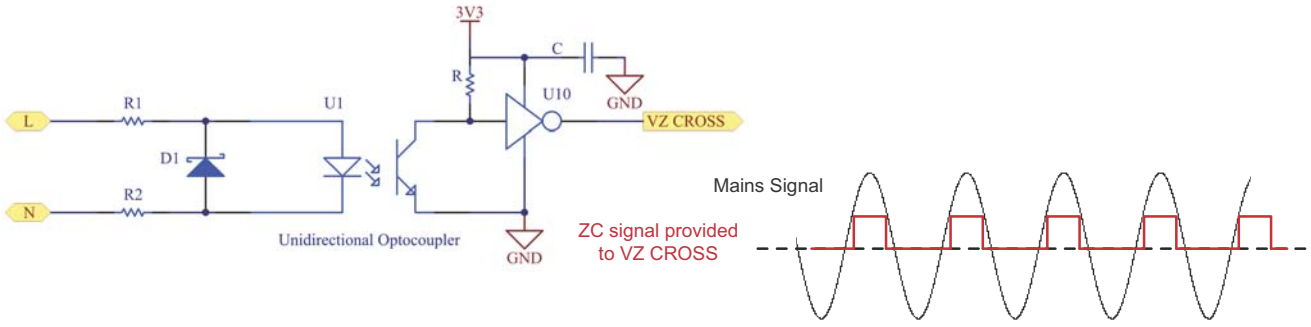


For this circuit, Zero Cross Internal registers should be configured this way:

ZC_CONFIG.MODE_MUX = ‘0’
ZC_CONFIG.MODE_ASC = ‘0’
ZC_CONFIG.MODE_INV = ‘1’
ZC_CONFIG.MODE_REP = ‘0’
ZC_FILTER.ZC_FILTER_BP = ‘0’

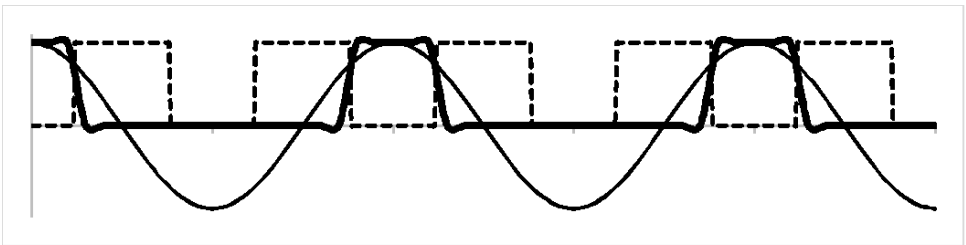
Some situations (for example in some protocols needing to differentiate rising/falling edges in mains signal) could require only ascent (or descent) mains signal zero-crossings to be detected. When we have to detect ascent or descent Zero-Cross of the wave (finer line), the circuit should generate an input signal “VZ CROSS” (wider line) with the same period, as specified in next figure. This could be easily implemented by using an unidirectional optocoupler or a Zener diode topology in the external circuitry.

Figure 5-5. Typical circuit, using a unidirectional optocoupler and a Schmitt trigger



The digital signal at output of Input Signal Management (ISM) is plotted in Figure 5-6.

Figure 5-6. Digital signal (dashed line) at output of Input Signal Management (ISM) internal block



For this case, Zero Cross Internal registers should be configured this way:

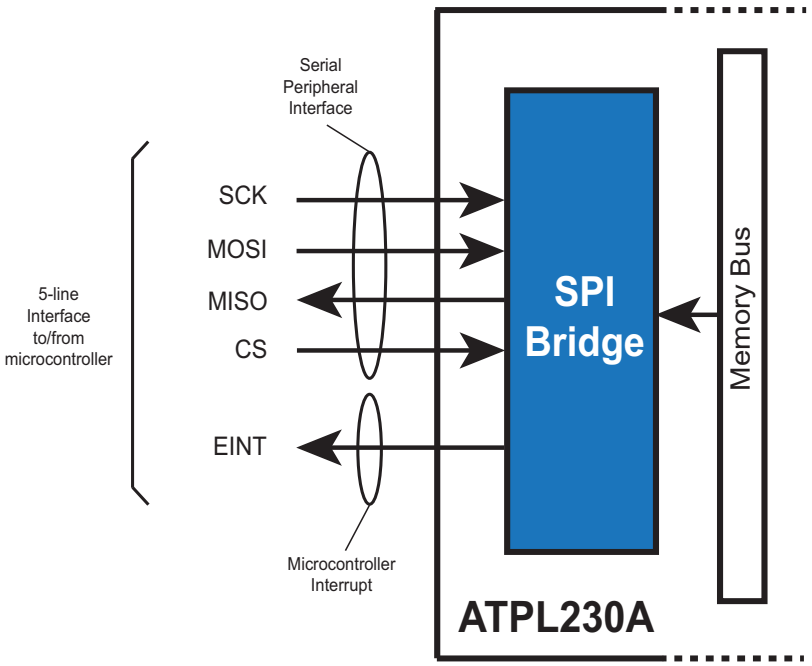
ZC_CONFIG.MODE_MUX = ‘1’
ZC_CONFIG.MODE_ASC = ‘0’(ascent) or ‘1’(descent)
ZC_CONFIG.MODE_INV = ‘1’
ZC_CONFIG.MODE_REP = ‘1’
ZC_FILTER.ZC_FILTER_BP = ‘0’

See register description in Section 9.3.7.2 ”Zero Crossing Configuration Register” and Section 9.3.7.3 ”Zero Crossing Filter Register”.

6. SPI Controller

ATPL230A has been conceived to be easily managed by an external microcontroller through a 5-line interface. This interface is comprised of a 4-line standard Serial Peripheral Interface (SPI) and an additional line used as interrupt from the ATPL230A to the external microcontroller. A diagram is shown below.

Figure 6-1. SPI Controller Block Diagram

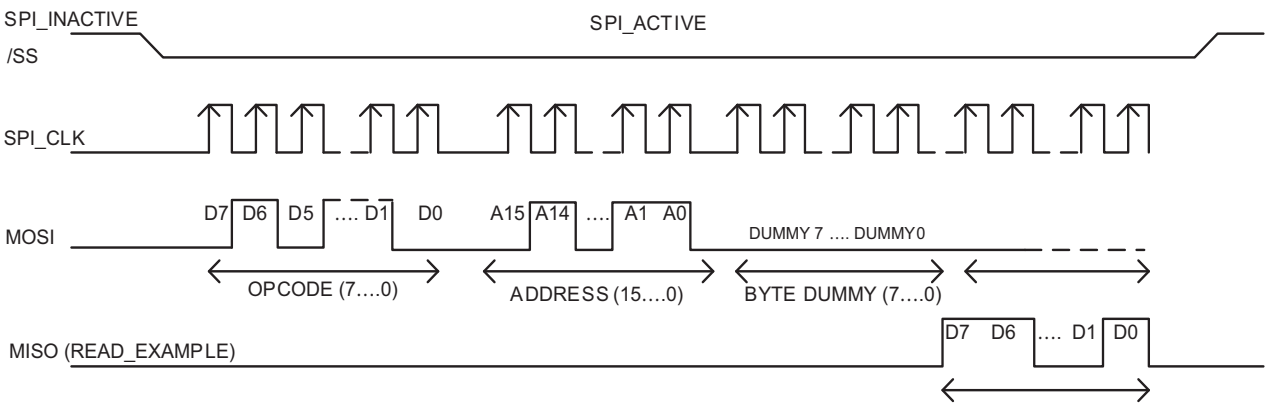


By means of this SPI interface, an external microcontroller can access the ATPL230A and can carry out “write”, “write_rep”, “read” and “mask” operations. All the “Peripheral Registers” in ATPL230A are reachable via the SPI interface, thus the microcontroller can fully manage and control the ATPL230A (PHY layer, MAC co-processing, etc).

6.1 Serial Peripheral Interface

The Serial Peripheral Interface Bus or SPI bus is a synchronous serial data link standard. Devices communicate in master/slave mode where the master device initiates the data frame. Multiple slave devices are allowed with individual slave select (chip select) lines.

Figure 6-2. SPI Communication Example in ATPL230A



The ATPL230A SPI allows an external device (working as a master), to communicate with the ATPL230A (working as a slave). Below is a brief description of the SPI signals:

- **CS, Chip Select (pin no.24):** This input enables/disables the slave SPI. The ATPL230A is configured to work always as a slave. When disabled (CS pin is tied high), the other SPI signals (SCK, MOSI and MISO) are not taken into account.

  CS = '0': SPI enabled.

  CS = '1': SPI disabled.

- **SCK, Serial Peripheral Interface Clock (pin no.25):** In reception (master slave), data is read from MOSI line in the rising edge of the SPI clock. In transmission (slave master), data is released to MISO in the falling edge of the SPI clock.

  It is recommended not to work with clock frequencies above 10MHz.

  This input only will be taken into account when CS='0'.

- **MOSI, Master Out Slave In (pin no.26):** MOSI is the slave's data input line. Data is read from MOSI line in the rising edge of SCK.

  This input only will be taken into account when CS='0'.

- **MISO, Master In Slave Out (pin no.27):** MISO is the slave's data output line. Data is released to MISO in the falling edge of SCK.

Furthermore, ATPL230A SPI bridge uses an additional line to send interrupts to the host CPU:

- **EINT (pin no.60):** This signal is an interrupt from ATPL230A PHY layer to the microcontroller.

  In reception, every time a PLC message is received, the PHY Layer generates two interrupts. One of them when the physical header is correctly received (two first symbols), and the other one when the message is completely received.

  In transmission, an interrupt will be generated every time a complete message has been sent.

  This signal is low level active.

6.2 SPI Operation

When establishing a SPI communication (CS line is set to '0' by the master), the first byte sent through MOSI line corresponds to the operation code. Four different operation types are defined over ATPL230A SPI. The operation codes are shown in the following table:

Table 6-1. Operation Codes

Operation	Mask type	OpCode
Read	---	0x63
Write	---	0x2A
Mask	AND	0x4C
	OR	0x71
	XOR	0x6D
Write_rep	---	0x1E

Following the operation code, the second and third bytes correspond to the SRAM address (16-bit address). Depending on the operation code, the master will "read data from"/"write data to"/"mask data in"/"write some data to" that address.

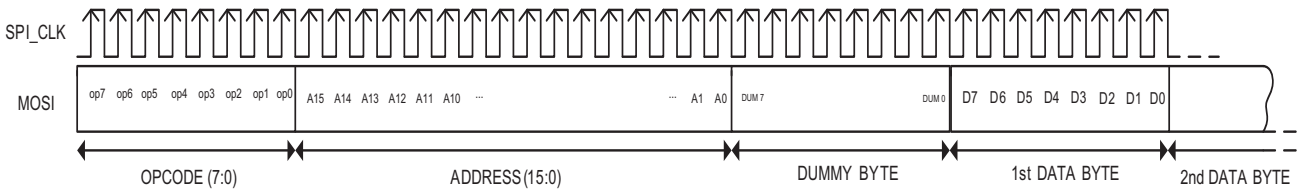
After the address, a dummy byte is sent.

Following the dummy byte, n data bytes (where n ≥ 1) are sent/received:

- If the operation code corresponds to a write operation in memory, the first data byte will be written in the specified address. If more data bytes are sent, they will be written in subsequent memory positions.
- If the operation code corresponds to a read operation from memory, the ATPL230A will output the data byte in MISO line. If the master continues sending SCK cycles, subsequent memory positions will be written in MISO line by the slave.
- If the operation code corresponds to a mask operation (AND, OR, XOR), the master will send the byte mask that have to be applied to the byte located at the specified address memory. If the master continues sending bytes, they will be applied as masks to the bytes stored in subsequent memory positions.
- If the operation code corresponds to a write_rep operation in memory, the dummy byte is changed by a number between 0x00 and 0xFF, called OFFSET_ADDRESS. Data bytes will be written from ADDRESS to ADDRESS+OFFSET_ADDRESS. For example, if OFFSET_ADDRESS = 0x04, the five first data bytes will be written between ADDRESS and ADDRESS+4, and then, the sixth data byte, will be written in ADDRESS, the seventh in ADDRESS+1, and so on. It is used to fill the some memories in PHY Layer (Chirp, Angle and IIR).

Bytes will be always sent with the most significant bit first.

Figure 6-3. SPI Frame Example



7. Peripheral Registers

A total of 768 bytes are reserved on-chip to allocate the system peripheral registers.

A detailed description of each peripheral register can be found in its corresponding section. On the next pages, there is a list of all of them.

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFD00 - 0xFD03	TX Time Registers	TXRXBUF_EMITIME_TX0	Read/Write	0x00..00
0xFD04 - 0xFD07		TXRXBUF_EMITIME_TX1	Read/Write	0x00..00
0xFD08 - 0xFD0B		TXRXBUF_EMITIME_TX2	Read/Write	0x00..00
0xFD0C - 0xFD0F		TXRXBUF_EMITIME_TX3	Read/Write	0x00..00
0xFD10 - 0xFD11	TX Post-activation Time TxRx Registers	TXRXBUF_TXRX_TA_TX0	Read/Write	0x0000
0xFD12 - 0xFD13		TXRXBUF_TXRX_TA_TX1	Read/Write	0x0000
0xFD14 - 0xFD15		TXRXBUF_TXRX_TA_TX2	Read/Write	0x0000
0xFD16 - 0xFD17		TXRXBUF_TXRX_TA_TX3	Read/Write	0x0000
0xFD18 - 0xFD19	TX Pre-activation Time TxRx Registers	TXRXBUF_TXRX_TB_TX0	Read/Write	0x0000
0xFD1A - 0xFD1B		TXRXBUF_TXRX_TB_TX1	Read/Write	0x0000
0xFD1C - 0xFD1D		TXRXBUF_TXRX_TB_TX2	Read/Write	0x0000
0xFD1E - 0xFD1F		TXRXBUF_TXRX_TB_TX3	Read/Write	0x0000
0xFD20	Global Amplitude Registers	TXRXBUF_GLBL_AMP_TX0	Read/Write	0xFF
0xFD21		TXRXBUF_GLBL_AMP_TX1	Read/Write	0xFF
0xFD22		TXRXBUF_GLBL_AMP_TX2	Read/Write	0xFF
0xFD23		TXRXBUF_GLBL_AMP_TX3	Read/Write	0xFF
0xFD24	Signal Amplitude Registers	TXRXBUF_SGNL_AMP_TX0	Read/Write	0x60
0xFD25		TXRXBUF_SGNL_AMP_TX1	Read/Write	0x60
0xFD26		TXRXBUF_SGNL_AMP_TX2	Read/Write	0x60
0xFD27		TXRXBUF_SGNL_AMP_TX3	Read/Write	0x60
0xFD28	Chirp Amplitude Registers	TXRXBUF_CHIRP_AMP_TX0	Read/Write	0x60
0xFD29		TXRXBUF_CHIRP_AMP_TX1	Read/Write	0x60
0xFD2A		TXRXBUF_CHIRP_AMP_TX2	Read/Write	0x60
0xFD2B		TXRXBUF_CHIRP_AMP_TX3	Read/Write	0x60
0xFD2C - 0xFD2F	TX Timeout Registers	TXRXBUF_TIMEOUT_TX0	Read/Write	0x000124F8
0xFD30 - 0xFD33		TXRXBUF_TIMEOUT_TX1	Read/Write	0x000124F8
0xFD34 - 0xFD37		TXRXBUF_TIMEOUT_TX2	Read/Write	0x000124F8
0xFD38 - 0xFD3B		TXRXBUF_TIMEOUT_TX3	Read/Write	0x000124F8
0xFD3C	TX Configuration Registers	TXRXBUF_TXCONF_TX0	Read/Write	0xA0
0xFD3D		TXRXBUF_TXCONF_TX1	Read/Write	0xA0
0xFD3E		TXRXBUF_TXCONF_TX2	Read/Write	0xA0
0xFD3F		TXRXBUF_TXCONF_TX3	Read/Write	0xA0
0xFD40 - 0xFD41	TX Initial Address Registers	TXRXBUF_INITAD_TX0	Read/Write	0x0000
0xFD42 - 0xFD43		TXRXBUF_INITAD_TX1	Read/Write	0x0000
0xFD44 - 0xFD45		TXRXBUF_INITAD_TX2	Read/Write	0x0000
0xFD46 - 0xFD47		TXRXBUF_INITAD_TX3	Read/Write	0x0000

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFD48 - 0xFD49	Reserved	-	-	0x0000
0xFD4A - 0xFD4B		-	-	0x0000
0xFD4C - 0xFD4D		-	-	0x0000
0xFD4E - 0xFD4F		-	-	0x0000
0xFD50 - 0xFD51	TX Result Register	TXRXBUF_RESULT_TX	Read-only	0x1111
0xFD52	TX Interrupts Register	TXRXBUF_TX_INT	Read-only	0x00
0xFD53	Reserved	-	-	0x00
0xFD54		-	-	0x00
0xFD55		-	-	0x00
0xFD56		-	-	0x00
0xFD57	BER SOFT Average Error Registers	TXRXBUF_BERSOFT_AVG_RX0	Read-only	0x00
0xFD58		TXRXBUF_BERSOFT_AVG_RX1	Read-only	0x00
0xFD59		TXRXBUF_BERSOFT_AVG_RX2	Read-only	0x00
0xFD5A		TXRXBUF_BERSOFT_AVG_RX3	Read-only	0x00
0xFD5B	BER SOFT Maximum Error Registers	TXRXBUF_BERSOFT_MAX_RX0	Read-only	0x00
0xFD5C		TXRXBUF_BERSOFT_MAX_RX1	Read-only	0x00
0xFD5D		TXRXBUF_BERSOFT_MAX_RX2	Read-only	0x00
0xFD5E		TXRXBUF_BERSOFT_MAX_RX3	Read-only	0x00
0xFD5F	Reserved	-	-	0x00
0xFD60		-	-	0x00
0xFD61		-	-	0x00
0xFD62		-	-	0x00
0xFD63	BER HARD Average Error Registers	TXRXBUF_BERHARD_AVG_RX0	Read-only	0x00
0xFD64		TXRXBUF_BERHARD_AVG_RX1	Read-only	0x00
0xFD65		TXRXBUF_BERHARD_AVG_RX2	Read-only	0x00
0xFD66		TXRXBUF_BERHARD_AVG_RX3	Read-only	0x00
0xFD67	BER HARD Maximum Error Registers	TXRXBUF_BERHARD_MAX_RX0	Read-only	0x00
0xFD68		TXRXBUF_BERHARD_MAX_RX1	Read-only	0x00
0xFD69		TXRXBUF_BERHARD_MAX_RX2	Read-only	0x00
0xFD6A		TXRXBUF_BERHARD_MAX_RX3	Read-only	0x00
0xFD6B	Minimum RSSI Registers	TXRXBUF_RSSIMIN_RX0	Read-only	0x00
0xFD6C		TXRXBUF_RSSIMIN_RX1	Read-only	0x00
0xFD6D		TXRXBUF_RSSIMIN_RX2	Read-only	0x00
0xFD6E		TXRXBUF_RSSIMIN_RX3	Read-only	0x00
0xFD6F	Average RSSI Registers	TXRXBUF_RSSIAVG_RX0	Read-only	0x00
0xFD70		TXRXBUF_RSSIAVG_RX1	Read-only	0x00
0xFD71		TXRXBUF_RSSIAVG_RX2	Read-only	0x00
0xFD72		TXRXBUF_RSSIAVG_RX3	Read-only	0x00
0xFD73	Maximum RSSI Registers	TXRXBUF_RSSIMAX_RX0	Read-only	0x00
0xFD74		TXRXBUF_RSSIMAX_RX1	Read-only	0x00
0xFD75		TXRXBUF_RSSIMAX_RX2	Read-only	0x00
0xFD76		TXRXBUF_RSSIMAX_RX3	Read-only	0x00

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFD77	Minimum CINR Registers	TXRXBUF_CINRMIN_RX0	Read-only	0x00
0xFD78		TXRXBUF_CINRMIN_RX1	Read-only	0x00
0xFD79		TXRXBUF_CINRMIN_RX2	Read-only	0x00
0xFD7A		TXRXBUF_CINRMIN_RX3	Read-only	0x00
0xFD7B	Average CINR Registers	TXRXBUF_CINRAVG_RX0	Read-only	0x00
0xFD7C		TXRXBUF_CINRAVG_RX1	Read-only	0x00
0xFD7D		TXRXBUF_CINRAVG_RX2	Read-only	0x00
0xFD7E		TXRXBUF_CINRAVG_RX3	Read-only	0x00
0xFD7F	Maximum CINR Registers	TXRXBUF_CINRMAX_RX0	Read-only	0x00
0xFD80		TXRXBUF_CINRMAX_RX1	Read-only	0x00
0xFD81		TXRXBUF_CINRMAX_RX2	Read-only	0x00
0xFD82		TXRXBUF_CINRMAX_RX3	Read-only	0x00
0xFD83 - 0xFD86	RX Time Registers	TXRXBUF_RECTIME_RX0	Read-only	0x00..00
0xFD87 - 0xFD8A		TXRXBUF_RECTIME_RX1	Read-only	0x00..00
0xFD8B - 0xFD8E		TXRXBUF_RECTIME_RX2	Read-only	0x00..00
0xFD8F - 0xFD92		TXRXBUF_RECTIME_RX3	Read-only	0x00..00
0xFD93 - 0xFD96	Zero-Cross Time Registers	TXRXBUF_ZCT_RX0	Read-only	0x00..00
0xFD97 - 0xFD9A		TXRXBUF_ZCT_RX1	Read-only	0x00..00
0xFD9B - 0xFD9E		TXRXBUF_ZCT_RX2	Read-only	0x00..00
0xFD9F - 0xFDA2		TXRXBUF_ZCT_RX3	Read-only	0x00..00
0xFDA3 - 0xFDA4	Header EVM Registers	TXRXBUF_EVM_HD_RX0	Read-only	0x0000
0xFDA5 - 0xFDA6		TXRXBUF_EVM_HD_RX1	Read-only	0x0000
0xFDA7 - 0xFDA8		TXRXBUF_EVM_HD_RX2	Read-only	0x0000
0xFDA9 - 0xFDAA		TXRXBUF_EVM_HD_RX3	Read-only	0x0000
0xFDAB - 0xFDAC	Payload EVM Registers	TXRXBUF_EVM_PYLD_RX0	Read-only	0x0000
0xFDAD - 0xFDAE		TXRXBUF_EVM_PYLD_RX1	Read-only	0x0000
0xFDAF - 0xFDB0		TXRXBUF_EVM_PYLD_RX2	Read-only	0x0000
0xFDB1 - 0xFDB2		TXRXBUF_EVM_PYLD_RX3	Read-only	0x0000
0xFDB3 - 0xFDB6	Accumulated Header EVM Registers	TXRXBUF_EVM_HDACUM_RX0	Read-only	0x00..00
0xFDB7 - 0xFDBA		TXRXBUF_EVM_HDACUM_RX1	Read-only	0x00..00
0xFDBB - 0xFDBE		TXRXBUF_EVM_HDACUM_RX2	Read-only	0x00..00
0xFDBF - 0xFDC2		TXRXBUF_EVM_HDACUM_RX3	Read-only	0x00..00
0xFDC3 - 0xFDC6	Accumulated Payload EVM Registers	TXRXBUF_EVM_PYLACUM_RX0	Read-only	0x00..00
0xFDC7 - 0xFDCA		TXRXBUF_EVM_PYLACUM_RX1	Read-only	0x00..00
0xFDCB - 0xFDCE		TXRXBUF_EVM_PYLACUM_RX2	Read-only	0x00..00
0xFDCF - 0xFDD2		TXRXBUF_EVM_PYLACUM_RX3	Read-only	0x00..00
0xFDD3	Buffer Selection Register	TXRXBUF_SELECT_BUFF_RX	Read/Write	0x00
0xFDD4	RX Interrupts Register	TXRXBUF_RX_INT	Read/Write	0x00
0xFDD5	RX Configuration Register	TXRXBUF_RXCONF	Read/Write	0x02

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFDD6 - 0xFDD7	RX Initial Address Registers	TXRXBUF_INITAD_RX0	Read/Write	0x0000
0xFDD8 - 0xFDD9		TXRXBUF_INITAD_RX1	Read/Write	0x0000
0xFDDA - 0xFDDB		TXRXBUF_INITAD_RX2	Read/Write	0x0000
0xFDDC - 0xFDDD		TXRXBUF_INITAD_RX3	Read/Write	0x0000
0xFDDE - 0xFDE1	Reserved	-	-	0x00..00
0xFDE2 - 0xFDE5	Reserved	-	-	0x00..00
0xFDE6	Reserved	-	-	0x00
0xFDE7		-	-	0x00
0xFDE8		-	-	0x00
0xFDE9		-	-	0x00
0xFDEA - 0xFDEB	Reserved	-	-	0x0000
0xFDEC - 0xFDED		-	-	0x0000
0xFDEE - 0xFDEF		-	-	0x0000
0xFDF0 - 0xFDF1		-	-	0x0000
0xFDF2	Robust TX Control Register	TXRXBUF_TXCONF_ROBO_CTL	Read/Write	0x00
0xFDF3	Robust RX Mode Register	TXRXBUF_RXCONF_ROBO_MODE	Read-only	0x00
0xFDF4 - 0xFDF7	Reserved	-	-	0x00..00
0xFDF8 - 0xFDF9	Reserved	-	-	0x0000
0xFDFA	Reserved	-	-	0xE0
0xFDFB	Branch Selection Register	TXRXBUF_TXCONF_SELBRANCH	Read/Write	0x00
0xFDFC	Reserved	-	-	0x00
0xFDFD	Reserved	-	-	0x00
0xFDFE	Reserved	-	-	0x00
0xFDFF	Reserved	-	-	0x00
0xFE2A	PHY Layer Special Function Register	PHY_SFR	Read/Write	0x87
0xFE2C	System Configuration Register	SYS_CONFIG	Read/Write	0x04
0xFE30	Reserved	-	-	0x00
0xFE31		-	-	0x00
0xFE32		-	-	0x00
0xFE33		-	-	0x00
0xFE34		-	-	0x00
0xFE35		-	-	0x00
0xFE36		-	-	0x00
0xFE37		-	-	0x00
0xFE38	Reserved	-	-	0x40
0xFE39		-	-	0x40
0xFE3A		-	-	0x40
0xFE3B		-	-	0x40

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFE3C	Reserved	-	-	0x10
0xFE3D		-	-	0x10
0xFE3E		-	-	0x10
0xFE3F		-	-	0x10
0xFE47 - 0xFE4A	PHY Layer Timer Register	TIMER_BEACON_REF	Read-only	0x00..00
0xFE53 - 0xFE55	Reserved	-	-	0x000200
0xFE57	Reserved	-	-	0x1E
0xFE5C	Reserved	-	-	0x0C
0xFE5D		-	-	0x18
0xFE5F		-	-	0x26
0xFE60		-	-	0x2B
0xFE62 - 0xFE67	Sub Network Address Register	SNA	Read/Write	0x00..00
0xFE68	Reserved	-	-	0x5F
0xFE69 - 0xFE6A	Reserved	-	-	0xFFFF
0xFE6B - 0xFE6C		-	-	0xFFFF
0xFE6D - 0xFE6E		-	-	0xFFFF
0xFE6F - 0xFE70		-	-	0xFFFF
0xFE71 - 0xFE72		-	-	0xFFFF
0xFE73	Reserved	-	-	0x56
0xFE7D - 0xFE7E	Reserved	-	-	0x814C
0xFE7F	Reserved	-	-	0x00
0xFE80 - 0xFE81	Reserved	-	-	0x0000
0xFE8F	Reserved	-	-	0x03
0xFE90	TXRX Polarity Selector Register	AFE_CTL	Read/Write	0x00
0xFE91	Reserved	-	-	0x1E
0xFE92	Reserved	-	-	0x28
0xFE94	PHY Layer Error Counter Register	PHY_ERRORS	Read/Write	0x00
0xFE9D	Reserved	-	-	0x21
0xFE9E	Reserved	-	-	0x05
0xFE9F	Reserved	-	-	0x60
0xFEA0	Reserved	-	-	0x60
0xFEA1	Reserved	-	-	0x60
0xFEA2	Reserved	-	-	0x60
0xFEA3 - 0xFEA6	Reserved	-	-	0x77777777
0xFEAB - 0xFEAC	Reserved	-	-	0x5508
0xFEAD - 0xFEAE	Reserved	-	-	0x3C20
0xFEAF	Reserved	-	-	0x00
0xFEB0	Reserved	-	-	0x00
0xFEB4	Reserved	-	-	0x00
0xFEB5 - 0xFEB6	Reserved	-	-	0x0066

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFEB7	Reserved	-	-	0x00
0xFEBA - 0xFEBB	CRC32 Errors Counter Register	ERR_CRC32_MAC	Read-only	0x0000
0xFEBC - 0xFEBD	CRC8 Errors Counter Register	ERR_CRC8_MAC	Read-only	0x0000
0xFEC0 - 0xFEC1	CRC8 HD Errors Counter Register	ERR_CRC8_MAC_HD	Read-only	0x0000
0xFEC2 - 0xFEC3	CRC8 PHY Errors Counter Register	ERR_CRC8_PHY	Read-only	0x0000
0xFEC4	False Positive Configuration Register	FALSE_POSITIVE_CONFIG	Read/Write	0x10
0xFEC5 - 0xFEC6	False Positive Counter Register	FALSE_POSITIVE	Read-only	0x0000
0xFEC8	Reserved	-	-	0x3F
0xFEC9	Reserved	-	-	0x3F
0xFECA	Reserved	-	-	0x3F
0xFECB	Reserved	-	-	0x3F
0xFECC	Reserved	-	-	0x3F
0xFECD	Reserved	-	-	0x3F
0xFECE - 0xFECF	Reserved	-	-	0x0000
0xFED3	Reserved	-	-	0x40
0xFED5 - 0xFED6	Reserved	-	-	0x0000
0xFEDB	Reserved	-	-	0x00
0xFEDC - 0xFEDF	Reserved	-	-	0x00..00
0xFEE0	Reserved	-	-	0x02
0xFEE4 - 0xFEE5	Reserved	-	-	0x0000
0xFEE6 - 0xFEE7	Reserved	-	-	0x0000
0xFEE8	Reserved	-	-	0x00
0xFEE9	Reserved	-	-	0xFF
0xFEEA	Reserved	-	-	0x04
0xFEEB	Reserved	-	-	0x08
0xFEEC	Reserved	-	-	0x0C
0xFEED	Reserved	-	-	0x14
0xFEEE	Reserved	-	-	0x00
0xFEEF	Reserved	-	-	0x03
0xFEF0	Reserved	-	-	0x00
0xFEF1	Reserved	-	-	0x17
0xFEF2	Reserved	-	-	0x18
0xFEF3	Reserved	-	-	0x23
0xFEF4	CRC PRIMEPLUS Configuration Register	PRIMEPLUS_CRC_CONFIG	Read/Write	0x14
0xFEF5 - 0xFEF6	CRC PRIMEPLUS Polynomial Register	PRIMEPLUS_CRC_POLY	Read/Write	0x080F
0xFEF7 - 0xFEF8	CRC PRIMEPLUS Reset Value Register	PRIMEPLUS_CRC_RST	Read/Write	0x0000

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFEFA - 0xFEFD	Channel Selector Register	CTPS	Read/Write	0x000150C7
0xFEFE	Reserved	-	-	0x00
0xFF00 - 0xFF07	Reserved	-	-	0x411A1803 73D6893C
0xFF09 - 0xFF0A	Reserved	-	-	0x0EA5
0xFF0E - 0xFF11	Peripheral CRC Polynomial Register	VCRC_POLY	Read/Write	0x04C11DB7
0xFF12 - 0xFF15	Peripheral CRC Reset Value Register	VCRC_RST	Read/Write	0x00..00
0xFF16	Peripheral CRC Configuration Register	VCRC_CONF	Read/Write	0xC3
0xFF17	Peripheral CRC Input Register	VCRC_INPUT	Read/Write	0x00
0xFF18	Peripheral CRC Control Register	VCRC_CTL	Read/Write	0x00
0xFF19 - 0xFF1C	Peripheral CRC Value Register	VCRC_CRC	Read-only	0x00..00
0xFF1E	Zero Crossing Configuration Register	ZC_CONFIG	Read/Write	0x00
0xFF1F - 0xFF20	Reserved	-	-	0x051E
0xFF21 - 0xFF22	Reserved	-	-	0x8000
0xFF23	Zero Crossing Filter Register	ZC_FILTER	Read/Write	0xB2
0xFF24 - 0xFF27	Reserved	-	-	0x00030D40
0xFF28 - 0xFF2B	Reserved	-	-	0x00..00
0xFF2D	Reserved	-	-	0x01
0xFF33 - 0xFF36	Reserved	-	-	0x00..00
0xFF37 - 0xFF38	Reserved	-	-	0x0000
0xFF39	Reserved	-	-	0x14
0xFF3A	Reserved	-	-	0x80
0xFF3B	Reserved	-	-	0x70
0xFF3C	Reserved	-	-	0xC8
0xFF3D	Reserved	-	-	0x0A
0xFF3E	Reserved	-	-	0x02
0xFF3F	Reserved	-	-	0x04
0xFF40	Reserved	-	-	0x01
0xFF41	Reserved	-	-	0x01
0xFF42	Reserved	-	-	0x27
0xFF43	Reserved	-	-	0x0A
0xFF4C	Reserved	-	-	0xA8
0xFF51	Reserved	-	-	0x99
0xFF52	Reserved	-	-	0xC0
0xFF53	Reserved	-	-	0x00
0xFF54	Reserved	-	-	0x03
0xFF55	Reserved	-	-	0x99
0xFF56	Reserved	-	-	0x99

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFF57	Reserved	-	-	0xFF
0xFF58	Reserved	-	-	0x33
0xFF5E - 0xFF5F	Reserved	-	-	0x0000
0xFF61	Reserved	-	-	0x00
0xFF62	Reserved	-	-	0x10
0xFF63 - 0xFF64	Reserved	-	-	0x00BF
0xFF65 - 0xFF66	Reserved	-	-	0x03E8
0xFF67 - 0xFF68	Reserved	-	-	0x0400
0xFF69 - 0xFF6A	Reserved	-	-	0x0F20
0xFF6B - 0xFF6C	Reserved	-	-	0x01EE
0xFF6D - 0xFF6E	Reserved	-	-	0x00BF
0xFF6F - 0xFF70		-	-	0x0160
0xFF71 - 0xFF72		-	-	0x02F0
0xFF73 - 0xFF74		-	-	0x0450
0xFF75	Reserved	-	-	0x68
0xFF76	Reserved	-	-	0x80
0xFF77	Reserved	-	-	0x3B
0xFF78 - 0xFF79	Reserved	-	-	0x0000
0xFF7A - 0xFF7F	Reserved	-	-	0x00..00
0xFF80	Reserved	-	-	0x00
0xFF81	Reserved	-	-	0x30
0xFF82 - 0xFF83	Reserved	-	-	0x0600
0xFF84	Reserved	-	-	0x58
0xFF85	Reserved	-	-	0x99
0xFF86	Reserved	-	-	0x79
0xFF87 - 0xFF88	Reserved	-	-	0x0021
0xFF89	Reserved	-	-	0x03
0xFF8A	Reserved	-	-	0x01
0xFF8B	Reserved	-	-	0x02
0xFF8C	Reserved	-	-	0x04
0xFF8D	Reserved	-	-	0x7F
0xFF8E	Reserved	-	-	0x00
0xFF92	Reserved	-	-	0x14
0xFF93	Reserved	-	-	0x11
0xFF94	Reserved	-	-	0x80
0xFF95	Reserved	-	-	0x00
0xFF96	Reserved	-	-	0x00
0xFF97	Reserved	-	-	0x70
0xFF98	Reserved	-	-	0xC8
0xFF99	Reserved	-	-	0x0A
0xFF9A	Reserved	-	-	0x02
0xFF9B	Reserved	-	-	0x04

Table 7-1. Register Mapping

Address	Register	Name	Access	Reset
0xFF9C	Reserved	-	-	0x01
0xFF9D	Reserved	-	-	0x01
0xFF9E	Reserved	-	-	0x27
0xFF9F	Reserved	-	-	0x0A
0xFFA0 - 0xFFAF	Peripheral AES Key Register	AES_KEY	Read/Write	0x00..00
0xFFB0 - 0xFFBF	Peripheral AES Data Field Register	AES_DATA	Read/Write	0x00..00
0xFFC0	Peripheral AES Control Register	AES_CTL	Read/Write	0x04
0xFFE2 - 0xFFE3	Reserved	-	-	0x0424
0xFFE4 - 0xFFE5		-	-	0x0424
0xFFE6 - 0xFFE7		-	-	0x0424
0xFFE8 - 0xFFE9		-	-	0x0424
0xFFEA - 0xFFEB		-	-	0x0424
0xFFEC - 0xFFED		-	-	0x0424
0xFFEE - 0xFFEF		-	-	0x0424
0xFFF0 - 0xFFF1		-	-	0x0424
0xFFF2 - 0xFFF3		-	-	0x0424
0xFFF4 - 0xFFF5		-	-	0x0424
0xFFF6 - 0xFFF7		-	-	0x0424
0xFFF8 - 0xFFF9		-	-	0x0424

8. MAC Coprocessor

ATPL230A accelerators can be used to perform PRIME MAC-specific tasks by hardware, decreasing CPU load from the external MCU/MPU. For that purpose, Cyclic Redundance Check (CRC) and AES128 encryption blocks are available in ATPL230A.

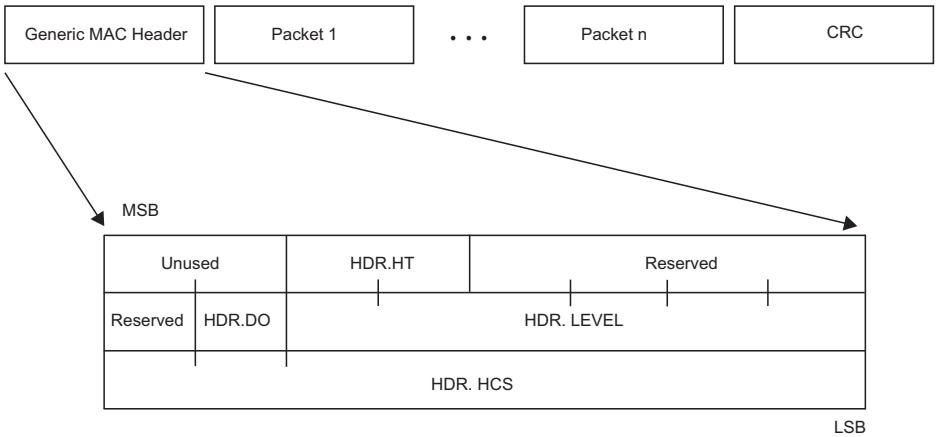
Please refer to Atmel doc43048 “Atmel PRIME Implementation” for Atmel software package detailed description and functionality.

8.1 Cyclic Redundancy Check (CRC)

8.1.1 PRIME v1.3 CRC

There are three types of MAC PDUs (generic, promotion and beacon) for different purposes, and each one has its own specific CRC. There is a hardware implementation of every CRC type calculated by the MAC layer. This CRC hardware-calculation is enabled by default. Note that the CRC included at the physical layer is also a hardware implementation available (enabled by default).

Figure 8-1. Example: Generic MAC PDU format and generic MAC header detail



In transmission all CRC bytes are real-time calculated and the last bytes of the MAC PDU are overwritten with these values, (provided that the field HT in the first byte of the MAC header in transmission data is equal to the corresponding MAC PDU type).

In reception the CRC bytes are also real-time calculated and these bytes are checked with the last bytes of the MAC PDU. If the CRC is not correct, then an error flag is activated, the complete frame is discarded, and the corresponding error counter is increased. These counters allow the MAC layer to take decisions according to error ratio.

For the Generic MAC PDU, there is an 8-bit CRC in the Generic MAC header, which corresponds to PRIME HDR.HCS. In reception, if this CRC doesn’t check successfully, the current frame is discarded and no interruption is generated.

This works in the same way as CRC for the PHY layer (CRC Ctrl, located in the PHY header, see PRIME specification for further information).

There is another CRC for the Generic MAC PDU which is the last field of the GPDU. It is 32 bits long and it is used to detect transmission errors. The CRC shall cover the concatenation of the SNA with the GPDU except for the CRC field itself. In reception, if the CRC is not successful then an internal flag is set and the error counter is increased.

For the Promotion Needed PDU there is an 8-bit CRC, calculated with the first 13 bytes of the header. In reception, if this CRC is not correct, then an internal flag is set and the corresponding error counter is increased.

For the Beacon PDU there is a 32-bit CRC calculated with the same algorithm as the one defined for the CRC of the Generic MAC PDU. This CRC shall be calculated over the complete BPDU except for the CRC field itself. In reception, if this CRC is not successful, then an internal flag is set and the same error counter used for GPDU is increased. The hardware used for this CRC is the same as the one used for GPDU.

8.1.2 Configurable CRC calculation

PRIME v1.3 version fixes the polynomial to calculate the CRCs. In case that these polynomials were modified, the CRC peripheral would be used. It is used as a peripheral unit, accessible using the system peripheral registers.

For example, to configure it for PRIME CRC8:

X^8 + X^2 + X + 1

VCRC_POLY = 0x00000007

VCRC_RST = 0x00000000

VCRC_CONF = 0xC0

And to configure it for PRIME CRC32:

x^32 + x^26 + x^23 + x^22 + x^16 + x^12 + x^11 + x^10 + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1

VCRC_POLY = 0x04C11DB7

VCRC_RST = 0x00000000

VCRC_CONF = 0xC3

A different set of registers can also be used to set CRC parameters:

X^12 + X^11 + X^3 + X^2 + X + 1

PRIMEPLUS_CRC_POLY = 0x080F

PRIMEPLUS_CRC_RST = 0x0000

PRIMEPLUS_CRC_CONFIG = 0x14

8.2 Advanced Encryption Standard (AES)

One of the additional security functionalities to PRIME v1.3 is the 128-bit AES encryption of data. ATPL230A includes a hardware implementation of this block, as a peripheral unit.

In transmission, data must be encrypted previously to the use of the PHY_DATA request primitive (see PRIME specification), in an independent way (note that Beacon PDU, Promotion PDU and Generic MAC header, as well as several control packets, are not encrypted).

In reception, data passed by the PHY layer is already encrypted and must be decrypted in a subsequent process.

To encrypt a data package with corresponding KEY, the process is as follows:

1. Write the KEY (128 bits long) in AES_KEY register. This step is only needed if a new key is going to be used (due to a key change or to a reset operation).
2. Write the 128 bits of data to be encrypted in AES_DATA register.
3. Set to ‘1’ the CIPHER control bit in AES_CTL register and then set to ‘1’ the START control bit to start the operation. This step could be realized as an atomic operation writing 0x03.
4. Wait until the READY bit in AES_CTL register becomes ‘1’ automatically. This bit indicates when the operation is completed.
5. After that, the encrypted (coded) data package is automatically stored in AES_DATA register.