

CAST

HDLC

HDLC Protocol Controller Core

Features

- LAPB/LAPD controlling machine providing
 - modulo 8 frame numbering
 - modulo 128 frame numbering
 - one- or two-byte addressing
 - automatically generated responses
- Serial Peripheral Interfaces
 - Bit stuffing
 - BOF and EOF flags generation
 - Support for RTS/CTS modem lines
 - Support for CD modem line
 - Collision detection in bus configuration
- Receive Length Check
- Three modes of receive operation
 - auto mode (with address recognition and control field insertion)
 - non-auto mode (with address recognition)
 - transparent mode (without address recognition)
- Receive and transmit blocks
 - Interrupt transfer mode
 - DMA transfer mode
- Separate FIFO's
 - 64-bytes long receive FIFO
 - 64-bytes long transmit FIFO
 - 16-bytes long address FIFO for storing up to 16 small frames in the receive FIFO
- Single or dual independent channel versions
 - Separate DMA lines for each channel
 - Common data bus and interrupt line
- Sophisticated self-checking Testbench (Verilog versions use Verilog 2001)

Configurability

- Two top-level architectures allowing implementing single or double channel cores available.

The HDLC core implements a single- or dual-channel controller for the High-Level Data Link Control (HDLC) protocol and its derivatives such as the Link Access Procedure, Balanced (LAPB) and Link Access Procedure, D channel (LAPD).

LAPB is used for public networks employing the X.25 communications protocol. LAPD is for ISDN applications.

The functional features of the core are based on the Siemens HSCX 82525 chip. Programs written for that chip can be used with the core with only minor changes.

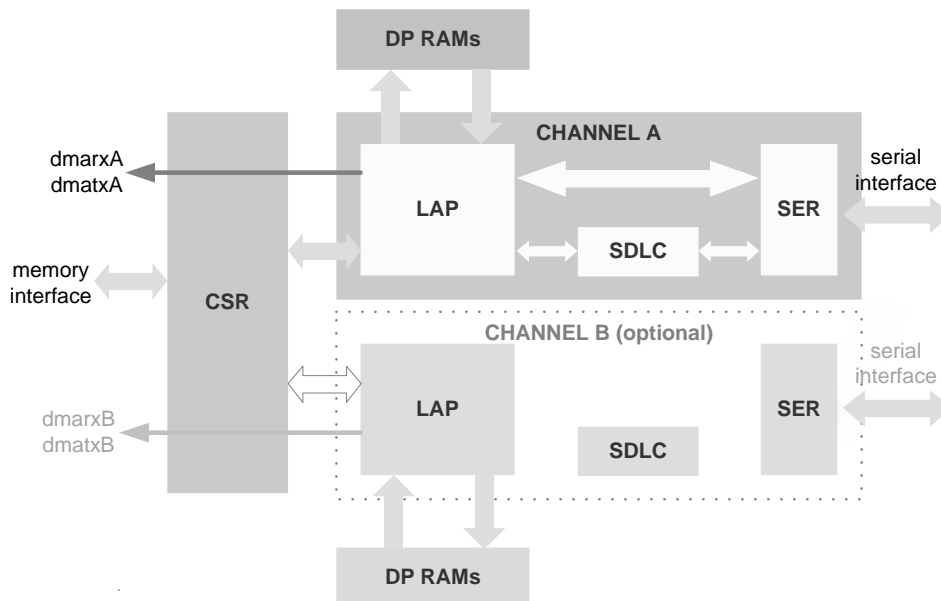
The design is strictly synchronous with positive-edge clocking, no internal tri-states and with a synchronous reset; therefore scan insertion is straightforward.

Applications

The core can be used for a variety of interface and communications applications, including:

- X.25 link control
- ISDN applications
- Physical link maintenance and quality monitoring of wide-area networks
- General purpose telecommunication applications

Block Diagram



Functional Description

The core is made up of several functional blocks, as shown in the block diagram and described below.

CSR Block

Converts host interface signals to the internal SFR interface, and switches some signals between channels. (The host interface is similar to the external memory interface used by CAST's R8051XC core.)

LAP Block

Controls transmission and provides LAPB/LAPD support. It has an internal timer, three FIFOs for data buffering, and a special function registers section for configuration controls. The LAP generates all interrupts and DMA requests, using an engine for automatic address and control field insertion and transfer engines for interrupt and DMA transfers. Eleven interrupt flag sources per channel determine the exact source of any interrupt.

SDLC Block

Provides serial data coding/decoding, bit stuffing, BOF and EOF generation, address recognition and CRC check/generation. It has small FIFO queues in both directions. The FIFOs are three-bytes deep and can be implemented as registers.

SER Block

Provides support for flow control and bus configuration. It also provides: RTS/CTS flow control, CD sense for enabling/disabling receive, collision detection in bus mod, bus IDLE state detection, and four different clock modes.

Serial Interface

Has three serial lines for input and output; one input can be used for feedback from the bus to detect collision. It also provides an enable line for the external tri-state buffer, which can be programmed to go active while a frame is being transmitted or only when "0" is transmitted. There are also clock inputs that are used in several clock modes. These are internally synchronized with the main host clock, meaning there is only single global clock in the design.

AFIFO, RFIFO, TFIFO interfaces

These are interfaces to DP RAM memories that should be implemented on the chip level. DP RAM memories are used by FIFOs in the HDLC core. RFIFO and TFIFO have a six-wire address bus and eight-bit data buses. AFIFO needs four-wire address buses and eight-bit data buses.

Implementation Results

HDLC reference designs have been evaluated in a variety of technologies. The following are sample ASIC results optimized for area with constraints set at clk=50 MHz.

| Channels | ASIC Technology | Approx. Area (gates) | Clock Constraint | Frequency |
|----------|-----------------|----------------------|------------------|-----------|
| Single | TSMC 0.09 um | 11,300 | 50 MHz | 216 MHz |
| Single | TSMC 0.13 um | 11,900 | 50 MHz | 160 MHz |
| Dual | TSMC 0.09 um | 26,917 | 100 MHz | 100 MHz |

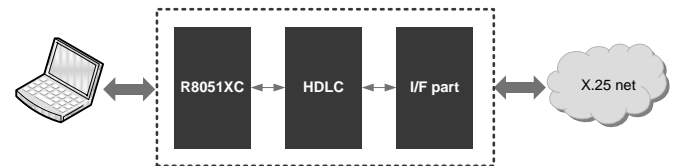
Support

The core as delivered is warranted against defects for ninety days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

Verification

The top level of the HDL testbench used for functional verification contains a MAC and additional environmental elements such as a stimulus vectors generator and output vectors comparator. Top-down simulation was performed with Aldec Active-HDL and MTI ModelSim based on the set of functional simulation tests.

Example Application – X.25 network



This example application uses the HDLC core to control an X.25 network for the LAPB protocol. The HDLC core is implemented together with internal dual port RAM, and an R8051XC microcontroller supports PLP.

Deliverables

The core includes everything required for successful implementation. The ASIC version includes:

- HDL RTL source code
- An example single-channel design
- Sophisticated self-checking Testbench (Verilog versions use Verilog 2001) that instantiates example design, test bench environmental design, external DP RAM, clock generator, and monitors that compare simulation results with expected results
- Simulation script, vectors, expected results, and comparison utility; Synthesis script
- Comprehensive user documentation, including detailed specifications and a system integration guide