The LIN core is a communication controller that transmits and receives complete LIN frames to perform serial communication according to the LIN Protocol Specification. The LIN controller can be implemented as a master or as a slave and operate on LIN 1.3, 2.0, 2.1 or 2.2 LIN network. It uses a single master/multiple slave concept for message transfer between nodes of the LIN network. The message transfer can be controlled via a micro controller interface and a LIN transceiver is needed for the connection to the LIN bus.

The LIN core is a microcode-free design developed for reuse in ASIC and FPGA implementations. The scan-ready design is strictly synchronous with positive-edge clocking and no internal tri-states. The robustly verified core has been production proven multiple times.

**Applications**

The LIN core can be utilized for a variety of applications including:
- low cost automotive networks
- interfaces for sensors and actuators

**FEATURES**

- Support of LIN specification 2.0, 2.1, and 2.2A
  - Backwards compatible with LIN specification 1.3
- Configurable for support of master or slave functionality
- Programmable data rate between 1 Kbit/s and 20 Kbit/s (for master)
- Automatic bit rate detection (for slave)
- 8-byte data buffer
- Generic 8-bit microcontroller interface
  - Wrappers converting the generic microcontroller interface to AMBA APB or AHB are offered with the core
- Slave can be implemented with or without clock synchronization
- Fully synchronous design, available in VHDL or Verilog, completely synthesizable
- The LIN Controller synthesizes to approximate 2500 to 3800 gates depending on the configuration
- Robustly verified and multiple times production proven IP core
Functional Description (cont.)

Control FSM
The finite control state machine is responsible for the behavior of the core depending on host controller commands and bus activity. It generates and processes the LIN frame fields according to the LIN protocol.

Bit Stream Processor
This module converts the data stream from parallel to serial (from transmit buffer to bus) and from serial to parallel (from bus to receive buffer).

Bit Timing Logic
The Bit Timing Logic is responsible for synchronizing the received data stream from the bus with the internal bit time clock.

Implementation Results
LIN reference designs have been evaluated in a variety of technologies. The following are sample Altera results.

<table>
<thead>
<tr>
<th>Family / Device</th>
<th>Mode</th>
<th>Logic Resources</th>
<th>Fmax (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arria 10</td>
<td>Master</td>
<td>228 ALMs</td>
<td>446</td>
</tr>
<tr>
<td>10AS016C3U192LG</td>
<td>Slave</td>
<td>361 ALMs</td>
<td>297</td>
</tr>
<tr>
<td>Cylcone 10 LP</td>
<td>Master</td>
<td>492 LEs</td>
<td>187</td>
</tr>
<tr>
<td>10CL010YU256C6G</td>
<td>Slave</td>
<td>833 LEs</td>
<td>161</td>
</tr>
<tr>
<td>Max 10</td>
<td>Master</td>
<td>497 LEs</td>
<td>148</td>
</tr>
<tr>
<td>10M08DAF484I7G</td>
<td>Slave</td>
<td>811 LEs</td>
<td>141</td>
</tr>
</tbody>
</table>

Notes:
1) Slave implemented with clock synchronization
2) Working frequency of LIN controller is 4 MHz

Core Modifications
The LIN core can be modified to include an acceptance filter. With that, a simple LIN slave that transmits response frames for only one identifier could be realized without the assistance of a host controller.

Please contact CAST, Inc. directly for any required modifications.

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 core has been verified through extensive synthesis, place and route, and simulation runs. It has been embedded in several shipping customer products, and is proven in both ASIC and FPGA technologies.

Deliverables
The core is available in FPGA netlist forms, and includes everything required for successful implementation, including a sophisticated self-checking testbench, simulation scripts, test vectors, expected results, synthesis scripts and comprehensive user documentation.