MEX-PROC
MODULAR EXPONENTIATION PROCESSOR

**Features**
- Performs large integer modular multiplication operations as used by many Public-Key cryptographic algorithms
- Uses multiple-precision arithmetic techniques yielding compact core size
- Parameterizable maximum modulus size (e.g. 1024 bits)
- Parameterizable arithmetic core size (e.g. 16 bits)
- Targeted at gate array, standard-cell and Synopsys CBA technologies

**Deliverables**
- Fully synthesizable VHDL- RTL source code.
- Synopsys Synthesis scripts
- VHDL test bench with functional test vectors.
- Synopsys CBA netlist.
- User documentation.

**Applications**
- Smartcard and embedded systems cryptographic coprocessor core
- Public-key algorithm coprocessor - suitable for RSA, DSS, DH, etc.
- Public-key parameter generation - primality testing, key generation, pseudo-random bit-sequence generation
- Privacy systems - secure distribution and management of encryption keys
- Non-Repudiation systems - digital signature generation and verification
- Authentication systems - secure identification and access control schemes
- Financial systems - digital currency and Secure Electronic Transaction (SET) implementations
**Basic Description**

The MEX-PROC is a fully synthesizable VHDL-RTL processing engine capable of performing large integer modular multiplication and exponentiation.

The device makes use of separate FISPbus and RAM interfaces; the FISPbus interface allows micro-controller access to the command and status registers of the device, while the RAM interface permits the connection of an external synchronous RAM used as a calculation workspace area. The micro-controller has read/write access to the RAM via the FISPbus interface.

![A simplified block diagram of the MEX-PROC is shown above.](image)

The heart of the device is the Arithmetic Unit which performs all modular arithmetic operations using multiple-precision arithmetic techniques. Note that the multiply-add circuitry of the Arithmetic Unit has been extracted into a separate component within the top-level of the MEX-PROC. This permits its easy replacement by a process-specific hard-macro should this be desired.

The natural word-size of the Arithmetic Unit can be specified independently of the maximum modulus size and may be set to 1, 2 or 4 times the size of the FISPbus data interface.

The Arithmetic Unit communicates with the external RAM via the RAM Interface. The RAM is assumed to be synchronous, offering single-cycle read/write access. The data-width of the RAM is the same as the natural word-size of the Arithmetic Unit, with the required number of words being equal to the number needed to store 4 maximum-sized large integers. For example, with a maximum modulus size of 1024 bits, a 16-bit Arithmetic Unit and an 8-bit FISPbus then the RAM configuration will be 256 words x 16 bits per word, but will appear to the FISPbus as a contiguous 512-byte address space.

The Control Unit shown in the diagram controls the operation of the Arithmetic Unit and the RAM Interface and is itself controlled by commands issued over the FISPbus interface. It can be instructed to perform a modular multiplication operation using the operands stored in the RAM. Since the RAM contains space for 4 large integers the multiply command specifies which of these fields are to be used for multiplier, multiplicand, modulus and result.

Modular exponentiation is performed under the control of the micro-controller by repeated multiplication and squaring. It is the responsibility of the micro-controller to scan the exponent and instruct the MEX-PROC as to the appropriate operations to perform. A typical smartcard implementation of the MEX-PROC would see a 1024-bit RSA digital signature performed in a fraction of a second.

Finally, all modular arithmetic within the device is performed using Montgomery's method. This permits an efficient and compact implementation of modular arithmetic functionality. Full details on how to use the MEX-PROC for both conventional and Montgomery multiplication as well as modular exponentiation can be found in the Product Specification document.

**Performance**

Gate count and speed figures are based on synthesis with Synopsys Design Compiler using a limited subset of the Synopsys CBA library. The gate counts exclude IO pads and RAM and ROM. Fmax is the maximum system clock speed that the design has been constrained for at synthesis, and Ffunc is the minimum system clock speed required for correct functionality of the chosen option. The FISP can run from a system clock anywhere within these two limits.

<table>
<thead>
<tr>
<th>OPTION</th>
<th>GATES</th>
<th>RAM</th>
<th>Ffunc (MHz)</th>
<th>Fmax (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8-bit FISPbus</td>
<td>7000</td>
<td>256 x 16-bit</td>
<td>N/A</td>
<td>40</td>
</tr>
<tr>
<td>16-bit AU</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1024-bit moduli</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Related Products**

- **DES-CORE** DES Encryption Core.
- **DES-PROC** DES Encryption Processor.