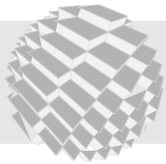


cranach SoC [FACT SHEET]

Network capable 'netpp on chip' processor solution for (small) FPGAs

The *cranach* System on Chip (SoC) is a compact network processor design targeted at small FPGAs.

It is part of the cCAP (configureable Custom Application Processor) family concept by section5.ch. The cCAP excel in high configureability and low resource usage on FPGA silicon. Independent of the FPGA technology, peripherals and IP cores can be tailored flexibly, starting with low-level register addresses up to number of peripheral instances (such as UART or SPI interfaces) or custom IP cores.

The cCAP family was particularely developed to meet specific requirements of robust and proveable functionality (like for safety relevant environments) while being easy to handle for application programmers at the same time, in order to facilitate handling of FPGA technology. The compact instruction set of the core architecture enables very compact program code as well as customer specific micro code extensions with DSP emulation functionality.

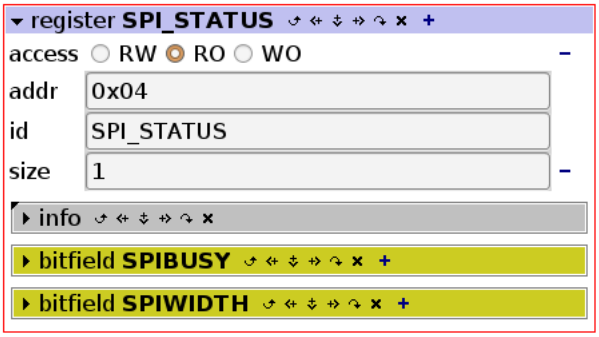


Fig. 1: SPI register description

1 Functional overview 'cranach'

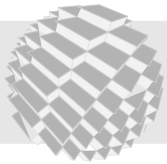
- 32 Bit processor 'ZPUng' v1.1, @50 Mhz
- Wishbone-Bus for address decoding and peripherals
- DMA for efficient data I/O without CPU intervention
- Programmable in C (GCC)
- Hardware Debugger (JTAG), GDB
- Ethernet, UDP/IP stack

For technical specifications of the standard configuration, see Table 1.

The default *netpp*-Operating System allows to define operation parameters ('Properties') by a XML device description language (see Fig. 1) and store them in a ROM area. The low level drivers for the corresponding interfaces are part of the firmware board supply package. External access to system properties can occur via python commands, a netpp client library or a webserver proxy. Attached sensor devices can this way be adressed at no programming overhead.

Technical specifications		
Memory configuration		
ROM	256kB	Overlay program memory (SPI) read only
L1RAM	16kB	Level one program and data segment
L1CACHE	16kB	Program memory or ROM cache (configureable)
SRAM	2kB	Dedicated Stack Memory
SPAD	8kB	Two ScratchPad RAM buffers for peripheral data I/O
Peripherals		
SCACHE	1	SPI flash cache
SIC	1	System Interrupt Controller with four peripheral channels and configureable priorities
DMA	2	Two DMA chanels for Ethernet MAC RX/TX packet queues
UART	1	UART console
MAC	1	Media Access Control Core (RGMII/MDIO), 100M/1G
SYSEXT	1	CRC16 accelerator

Tab. 1: cranach standard configuration



2 Simulation

The entire *cranach* SoC can be co-simulated in a virtual machine. Advantages:

- Cycle-accurate verification of correct functionality
- Efficient visual debugging in trace waveforms, optional on-chip trace debugging via Ethernet
- Co-Simulation with Python-Scripts for functional verification

3 Example applications

- Intelligent remote control of FPGA logic per Python scripting or LabView.
- Configuration processor for high speed Serdes interfaces
- Real time capable PWM controller with guaranteed network response latency
- DSP accelerators
- Redundant systems (Multi-Core)

Python example

```
import netpp

fpga = netpp.connect("UDP:192.168.0.5:2016")
r = fpga.sync()
if r.Status.get() == 1:
    r.LED.Green.set(1)
```

4 Reference platforms

The cranach-SoC is available as integrateable net list for the following platforms:

- Lattice ECP5 Versa Eval board (Diamond v3.9)
- Trenz TE-0600 Core-Modul (ISE v14.x)

Porting to other platforms requires adaptation of the MAC mdio driver.

5 Typical resource usage

Lattice ECP5	
SLICE	3370
Registers	3100
LUT4	4590
EBR	28

Tab. 2: Resource usage ECP5

Xilinx Spartan6	
Slice Registers	2395
Slice LUT	5780
LUT-FF	2364
BlockRAM	26

Tab. 3: Resource usage Spartan6

6 Further pointers

Overview cCAP-Family
http://section5.ch/index.php/dokumentation/masocist-soc/soc-overview/

cranach hardware reference `soc-cranach.pdf`
On request