Projects Overview 2014

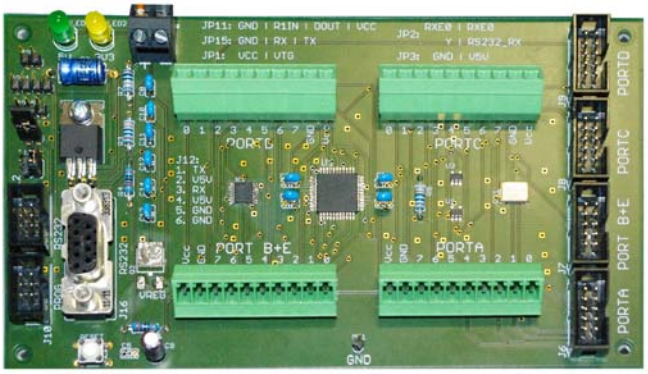
Release 1.1 (September 8, 2014)

1. Xmega Trainers

The Atmel Xmega microcontrollers are lavishly equipped with functional units. The processor core, however, is much more straightforward than, for example, an ARM or MIPS core. Hence those microcontrollers are well-suited for ambitious beginners and students who want to become familiar with more sophisticated controller ICs, where complicated I/O units are to be employed appropriately, problems of level translation are to be tackled and the like.

Type 1

The most straightforward module without level translation except for the serial interface. All I/O pins are easily accessible via pin headers or terminal blocks. One serial interface is provided. It can be configured for 5 V or RS-232 signal levels.

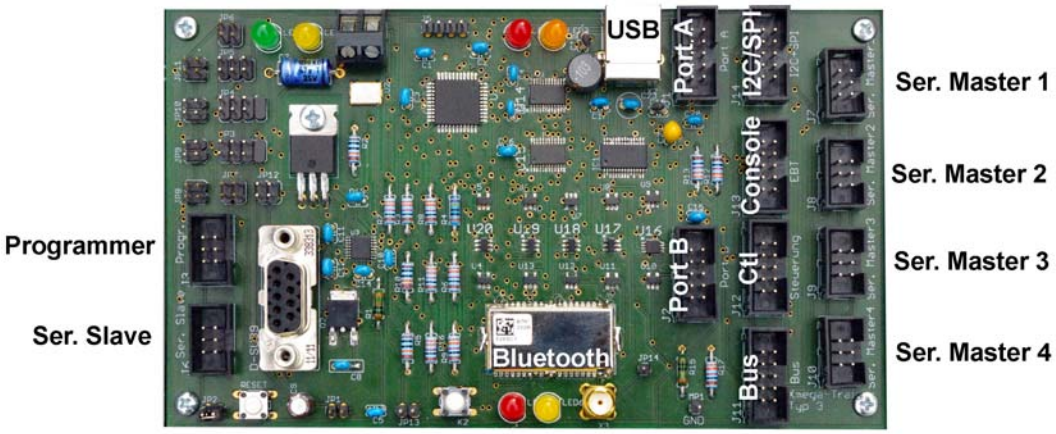


Type 2

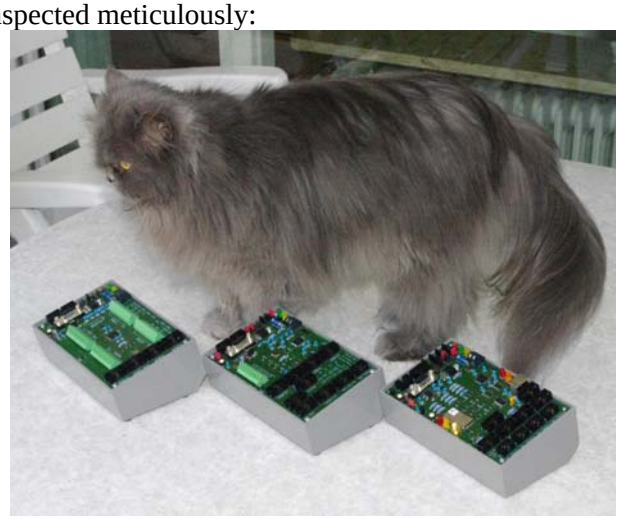
A module with level-translation provisions aimed at general-purpose applications. All I/O pins are easily accessible via pin headers. The 5-V-interfaces comprise one bidirectional 8-bit-bus, eight outputs, two serial interfaces and one SPI interface. An extra connector provides four bus and four control signals, so that a general-purpose human interface module can be attached. (The human interface module is thought as some kind of basic console.)

Type 3

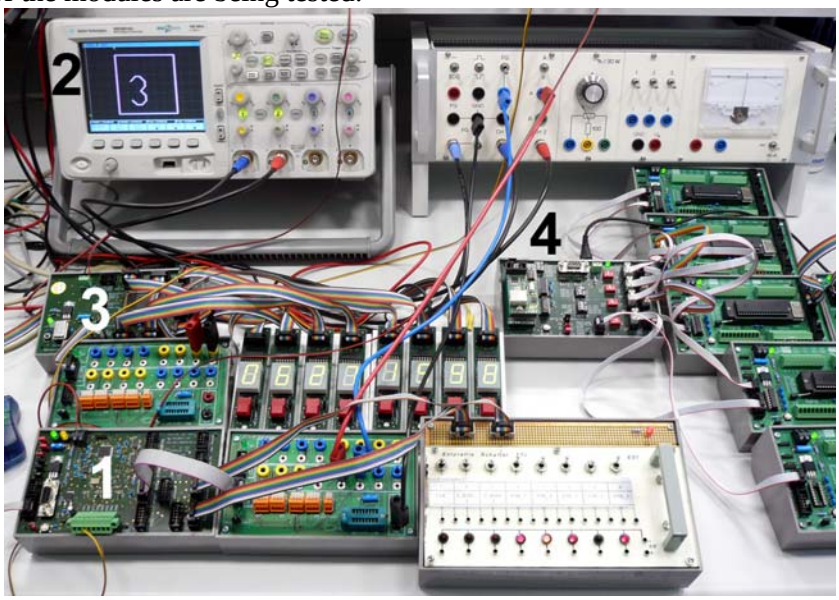
Above all, this module is an active hub to build multiprocessor systems with. It has one slave or upstream interface and four master or downstream interfaces (like USB, but much simpler). The slave interface can be configured as serial 5 V, RS-232, USB or Bluetooth. Additional interfaces are a bidirectional 8-bit-bus, eight outputs, SPI, and I2C. Similarly, to the type 2, a general-purpose human interface module can be attached, serving, for example, as console or troubleshooting aid.



Xmega trainers are inspected meticulously:

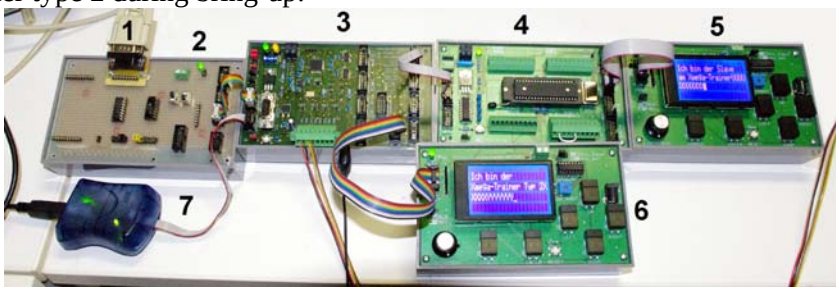


Here some of the modules are being tested:



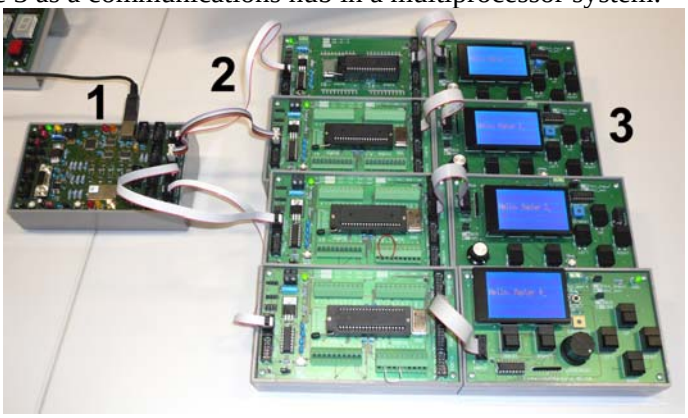
1 - Xmega trainer type 2; 2 - XY-presentation employing the Xmega's DACs; CPLD module (introduced below) serves as a time-of-day digital clock (24 hours, 0,01 s resolution); 4 - a multiprocessor configuration, consisting of four microcontroller modules attached to a passive hub.

Xmega trainer type 2 during bring-up:



1 - serial connection to a personal computer; 2 - passive hub (used for level translation); 3 - Xmega-Trainer; 4 - microcontroller module attached as a slave; 5, 6 - human interface modules, serving as debugging consoles.

Xmega trainer type 3 as a communications hub in a multiprocessor system:



1 - USB connection; 2 - slave modules. The cables provide for serial communication as well as for power supply. 3 - each of the slave modules has its own debugging console.

2. CPLD Modules

Complex FPGAs are terrible to design with. It is way too much for newcomers. Basic research is hampered as well. To investigate architectural principles, one needs a suitable environment to experiment. Computer simulation is too slow, FPGA implementations pose their own intricate problems like routing, clock delays and so on – especially when uncommon circuitry is to be implemented (the manufacturer's coding recommendations provide well for NIOS processors, ARM cores, FIFO buffers, digital filtering and the like, but obviously not for very special or even ReAl machines...).

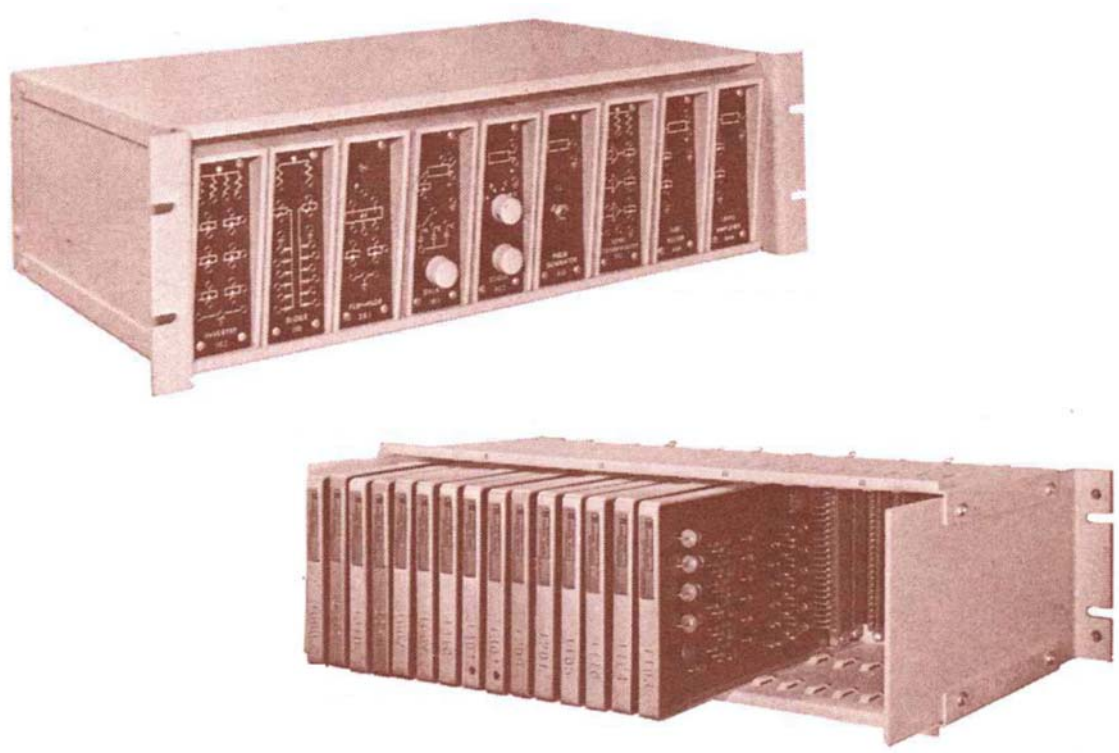
Hence we revive a venerable principle of hardware design. Instead of using a large FPGA, we implement the logic in comparatively small CPLDs (and some auxiliary modules) which are inserted in a backplane and connected by wire (wire-wrap technology).

It is basically a revival of the old modular plug-in principle. However, what was a card with many digital circuits in bygone times, is now a CPLD.

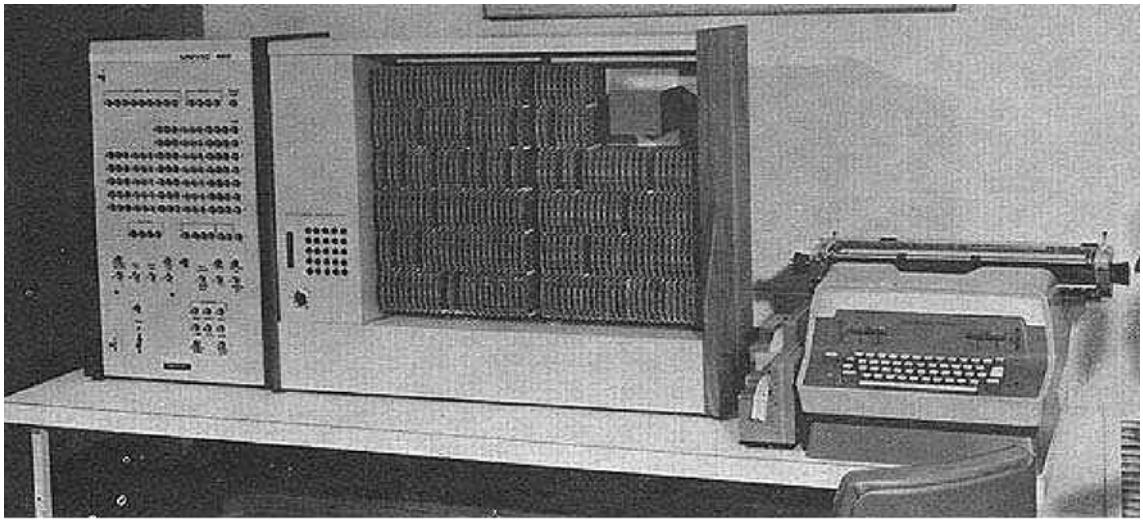
Thus, processors, programmable logic controllers and the like can be designed the traditional way. This also applies to bring-up and troubleshooting. It can be done with real scopes and logic analyzers instead of complex simulation software... Design bugs are corrected by reprogramming CPLDs and changing wire-wrapped connections.

From left to right: service processor (SVP) module (containing two Atmel Atmega microcontrollers), CPLD module (Xilinx 95108), SRAM module (128k • 24 or 512k • 24).

Legacy modules (DEC):

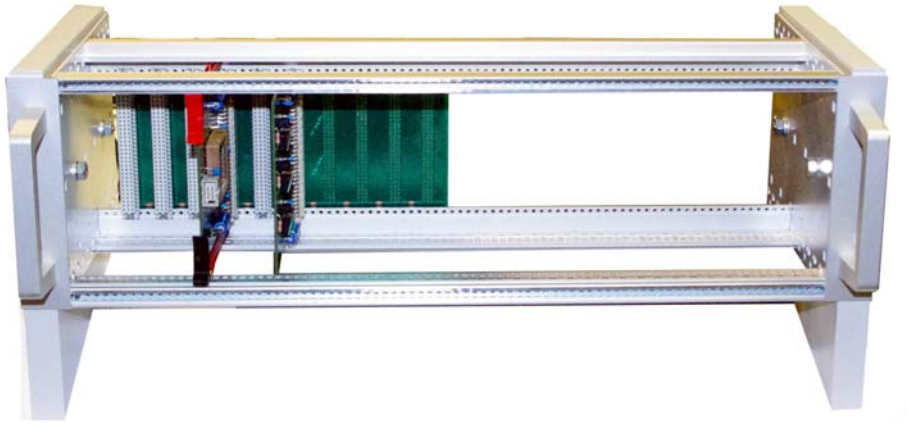


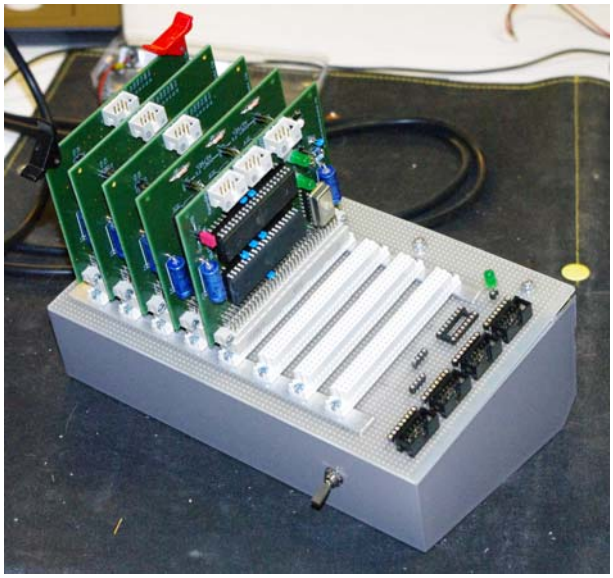
The Univac 422 Training Computer (UNIVAC Divison of Sperry Rand Corp.):

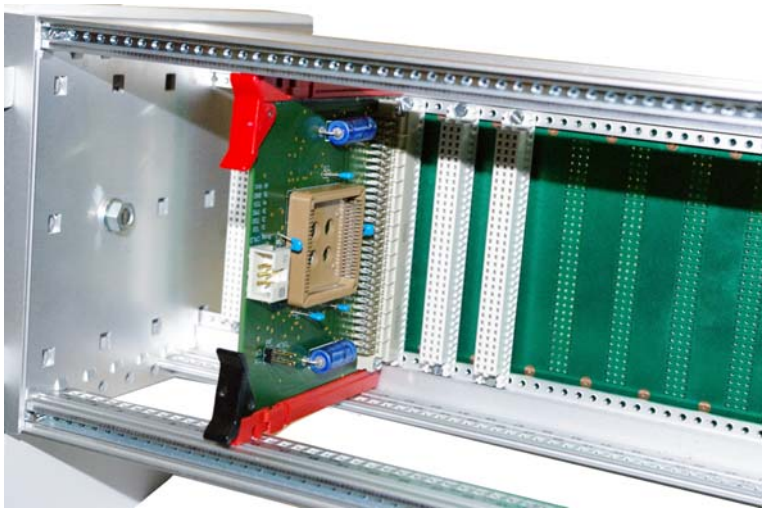


It consists of 1176 transistors and 4639 diodes. The machine word is 15 bits long. The memory capacity is 512 words. In a rough estimate, the logic circuitry corresponds to somewhat 3000 NAND gates or 800 7400 TTL circuits. Five Xilinx 95108 CPLDs should be sufficient...

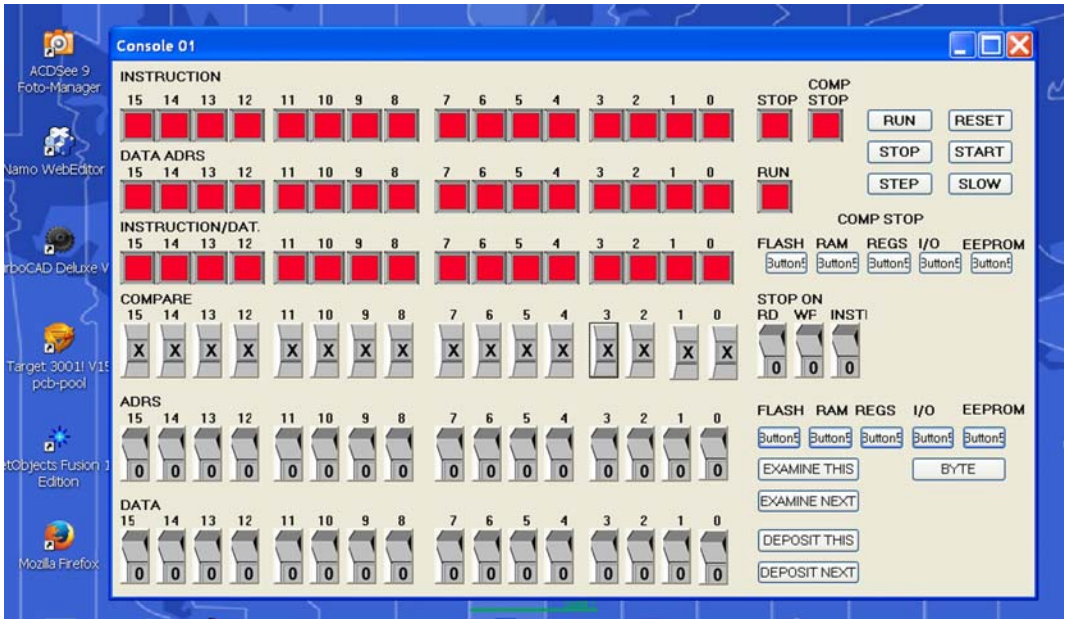
The form factor of our modules is 19", 3 U, half length of an eurocard. Thus we can rely on readily available 19" components.







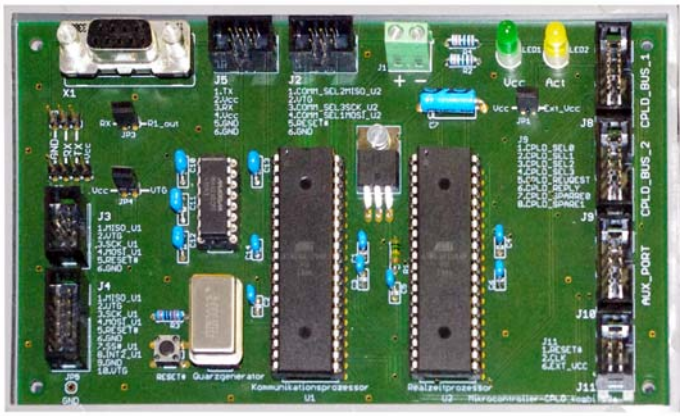
CPLD processors should be equipped with an old-fashioned console. However, the necessary switches, lamps and the like are more expensive than a tablet computer. Consequently, the console will not be built with sheet metal and wire, but programmed...



3. Platforms combining microcontrollers and CPLDs

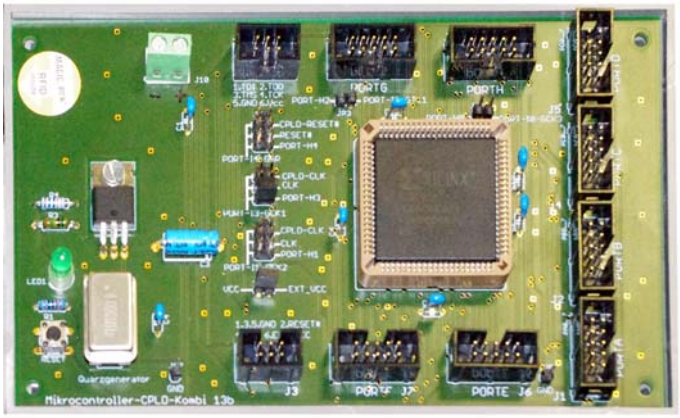
The first project is an educational compound platform, comprising two microcontrollers and a CPLD. In the beginning, we contemplated one PC board. However, two boards have the obvious advantage that they can be used independently. One of the microcontrollers is thought as the communications processor, the other as the applications or real-time processor. It can be connected to an CPLD. The CPLD serves as I/O extension or attached support processor.

The microcontroller module:



Both controllers are connected via an 8-bit bus interface and a serial interface (for initialization, debugging and the like).

The CPLD module:



It ist connected to the real-time processor via an 8-bit bus interface.