Practical programming of ARM microcontrollers using LabVIEW

Kenneth Frogner

Lund University

2009-03-10

Outlines

- Why to use ARM microcontrollers
- Applications for ARM microcontrollers
- LabVIEW for ARM programming
- Examples of simple implementations
- A practical demonstration
- Wishes for the future of the ARM module
- Conclusion

Microcontrollers

- Combines CPU, memory (RAM & ROM), I/O ports, counters, communication interfaces, etc. on one single chip

- Suited for embedded systems when space, power consumption and cost matters, rather than extreme computation speed

Customer Applications

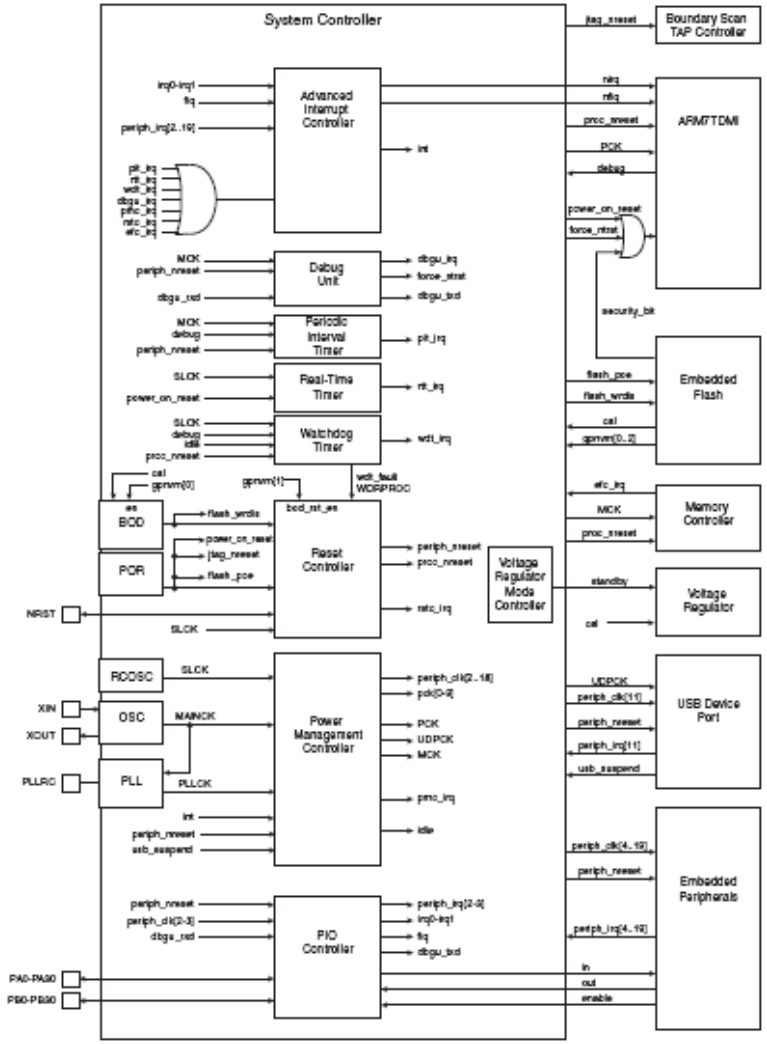
ARM

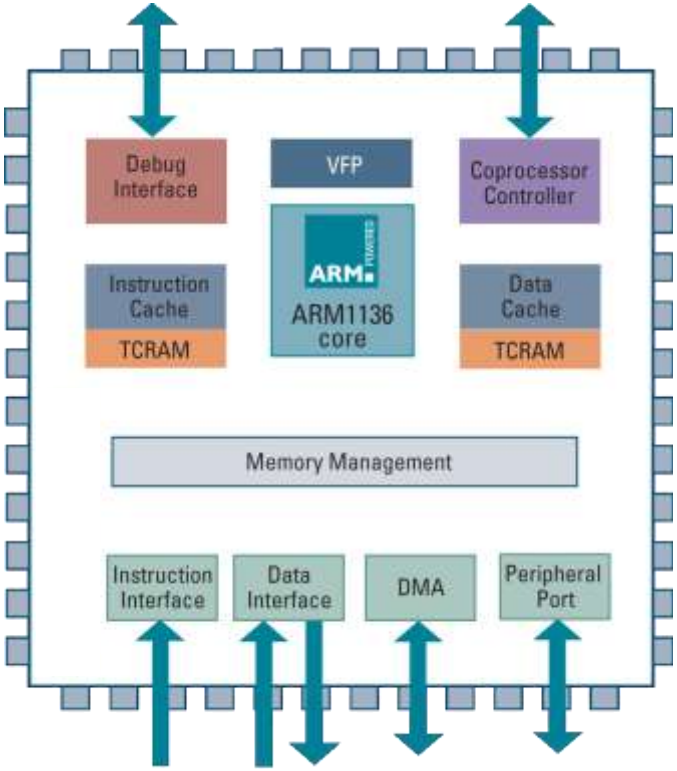
- Most common architecture in the industry
- More than 75 percent of the market share
- More than 10 000 000 000 processors shipped
- ARM licenses its technology to more than 200 world-leading semiconductor companies
- About 98 percent of all mobile phones use at least one ARM-designed core (2007)

ARM

- 32 bits architecture
- Ideal for control and interrupt-driven applications that require real-time response

LabVIEW ARM programming



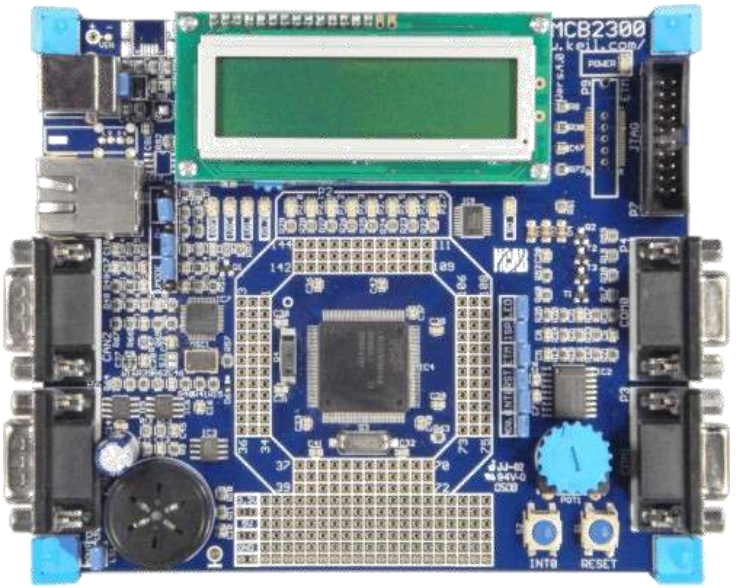


LabVIEW ARM module

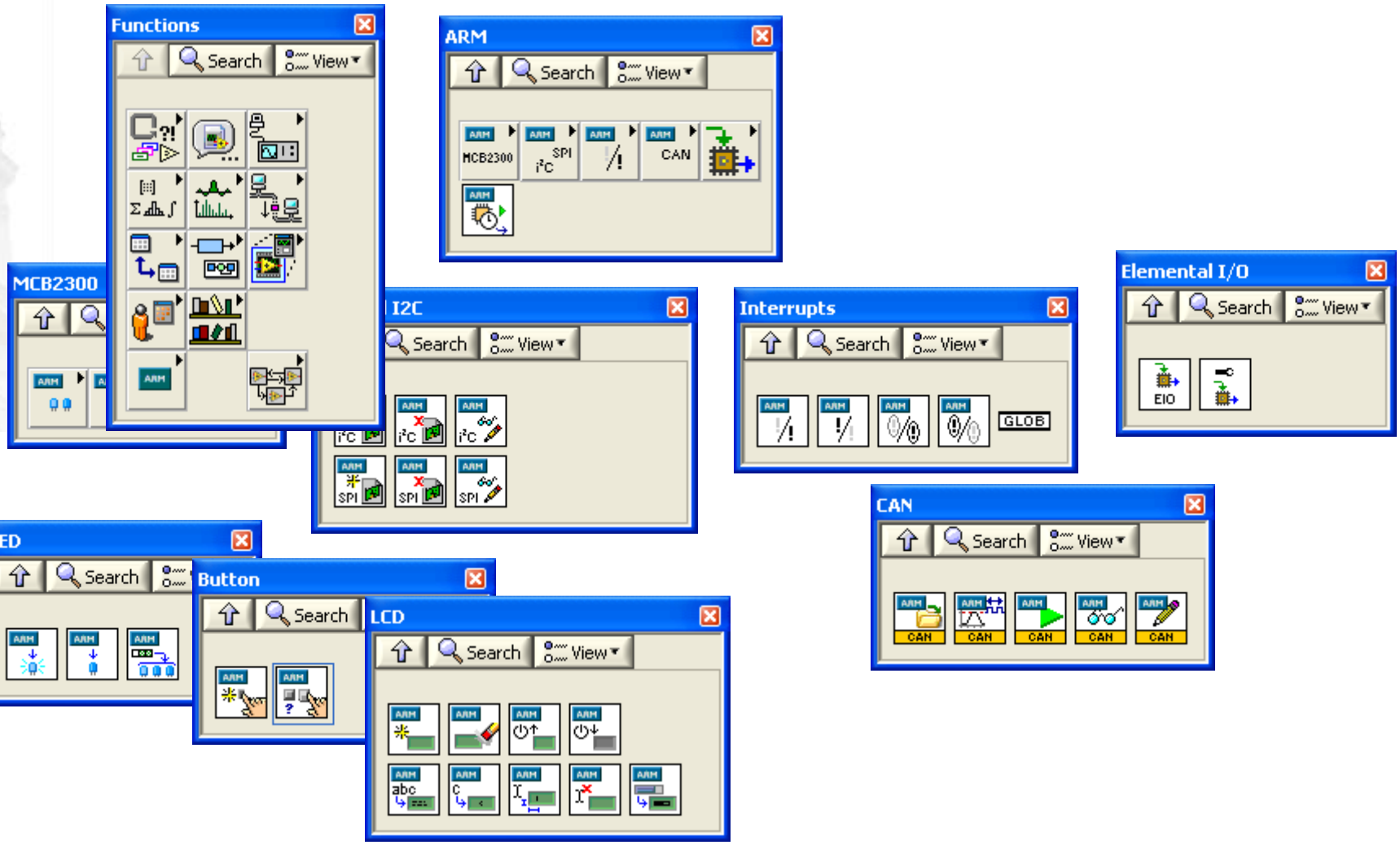
- ARM7, ARM9 and Cortex M3
- Programming, debugging and simulation
- Analog and digital I/Os, PWM, timing, counters
- Interrupts managing, communication interfaces

Demo Board

- NXP microcontroller with 512 kB flash memory, 72 MHz
- USB 2.0, serial ports, CAN ports and Ethernet
- LCD display, LEDs, Loudspeaker, buttons, potentiometer, etc.
- JTAG programming and debugging interface



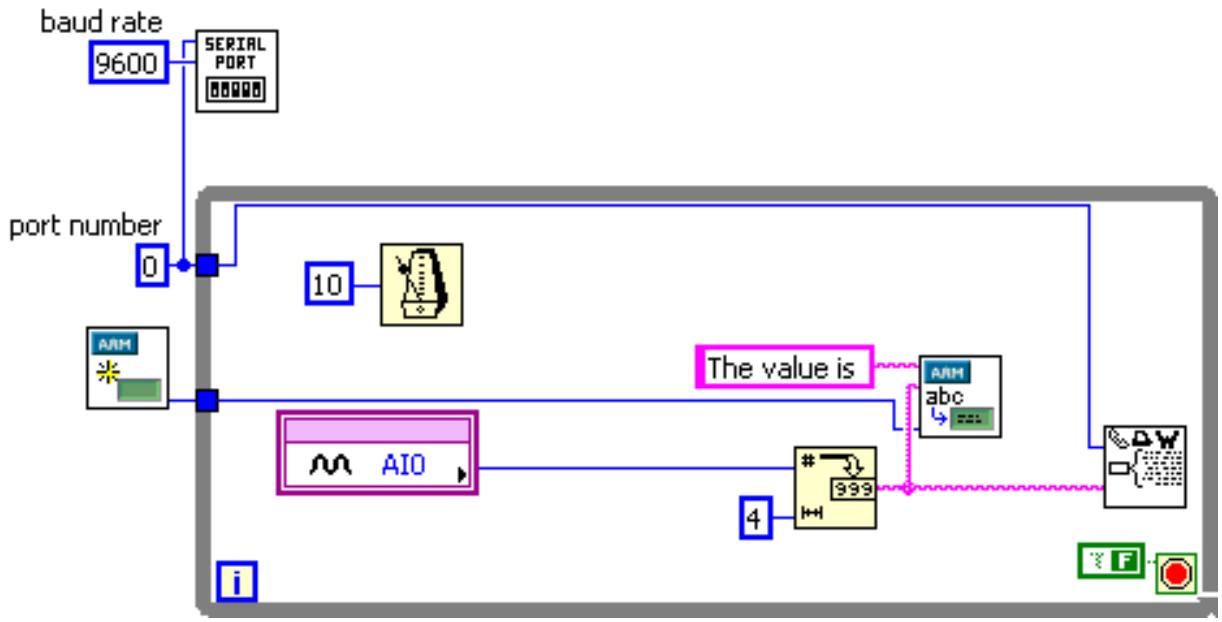
LabVIEW ARM Functions



LabVIEW ARM Example 1

Read in analog value every 100 ms

Write it to the LCD and send it out via rs232



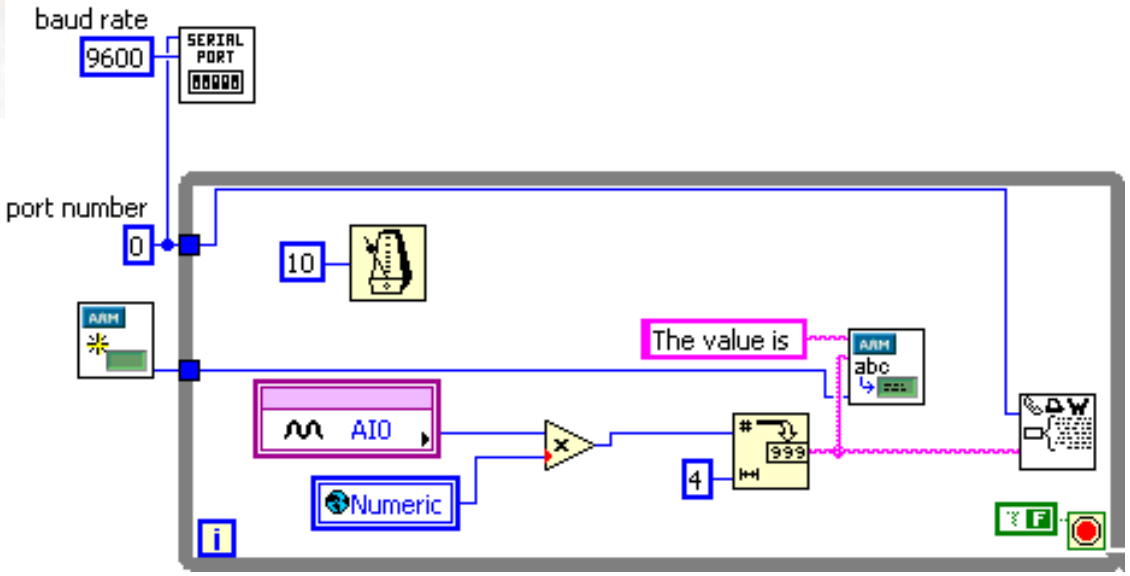
Interrupts

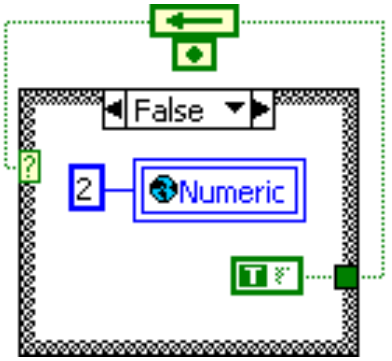
- Hardware generated signal
  - Timer interrupts
  - External events

- Interrupt handler
  - Context switch
  - Interrupt routine

Example continue, DEMO

Add interrupt





Compilation and optimization

- Cross compilation
- Optimization according to size and/or speed
- Debugging not available in optimized mode

Porting to other microcontrollers

- Support for more than 260 different microcontrollers
- Difficult, time consuming, low level work

Wishes for the future

- Simpler to port to other microcontrollers
- Support for USB interface
- Examples of using I2C and SPI
- Support for common peripherals like different displays etc.

Conclusion about ARM

- Common architecture, suitable for a lot of applications
  - Works with few external components
- Possible to program with LabVIEW
  - Fast and intuitive, but several things that can be improved

Thanks for Listening!