Technical details

CPU

ARM7 Atmel processor (AT91R4008)
Variable clock speed, 4 - 59 MHz
4 GB address space

Memory

8 MB Flash memory with Flash file system and direct access
4 MB DRAM
256 kB fast SRAM
Serial 512-byte EEPROM

Firmware

Java VM, Embedded Linux (optional)

Peripherals

2 serial interfaces, max. 921600 baud, TTL
PIF bus (5V-compatible), I²C bus
Maximum of 38 digital I/O ports, programmable as input or output
3 timers, externally accessible clock and I/O signals
Real-time clock, externally accessible alarm output
Programmable Watchdog timer
4 available IRQs, altogether

Power Management

3.3 V operating voltage
Energy consumption can be reduced by slowing or pausing the processor or various peripherals (such as USARTs, timers and PIO)
Active mode: 48 mA
Idle Mode (CPU clock off): 25 mA
Power-down mode: 7 mA
(all values measured at 59 MHz)

Housing

43 x 36,4 x 5 mm (CompactFlash Typ II)
50-pin connector

We're here for you!

Do you have your own ideas for an innovative use for the MicroARM?
Talk to the creative minds on our development team about the many possible applications for our embedded ARM!

taskit

taskit Rechnertechnik GmbH
www.taskit.de/en

embedded systems

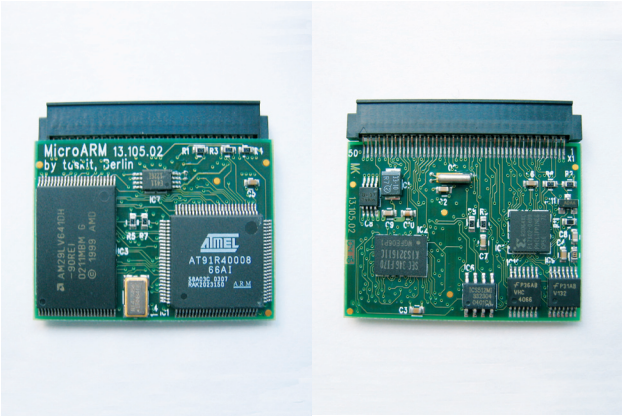
taskit

MicroARM

Embedded ARM and Java

The modern ARM architecture represents the most important manufacturer-independent processor architecture to date. The software support (from tools to operating systems) is correspondingly strong, including both Java and Embedded Linux.

The ARM implementation in MicroARM's AT91R40008 processor also offers the lowest energy consumption in this processor family – with full 32-bit operation.



MicroARM without housing

Simple connection to a PC

The MicroARM connects to a host PC by means of a terminal program. The Java shell enables you to start Java programs, and to copy data between the host PC and the MicroARM's flash file system.

Easy to program

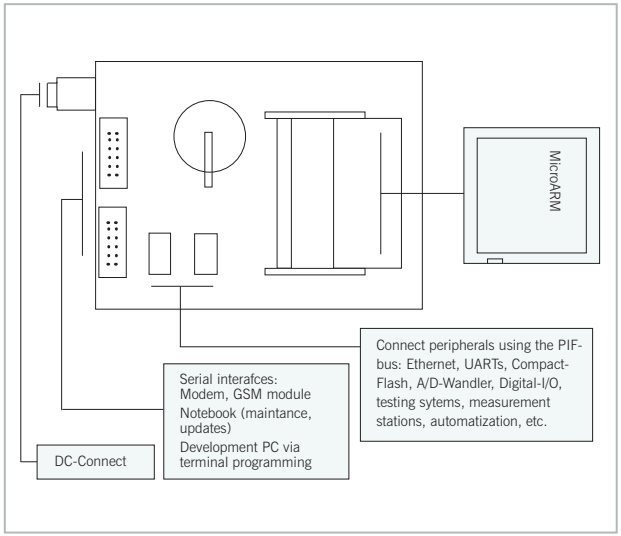
Java offers a free, hardware-independent programming environment. Access to the hardware interfaces is already implemented as part of the Java VM; the corresponding interface file just needs to be linked to the application program.

Easy connection of peripherals

The MicroARM makes it particularly straight-forward to design and connect peripherals without special, in-depth knowledge.
These easy-to-use interfaces include the PIF bus, I^2C bus, serial interfaces and digital ports. A wide variety of standard controllers for LCDs, networking or CompactFlash can be operated directly on the PIF bus. Implementation examples for digital I/O, AD/DA converters, LCDs, matrix keyboards and Ethernet are included in the starter kit.

Starter kit

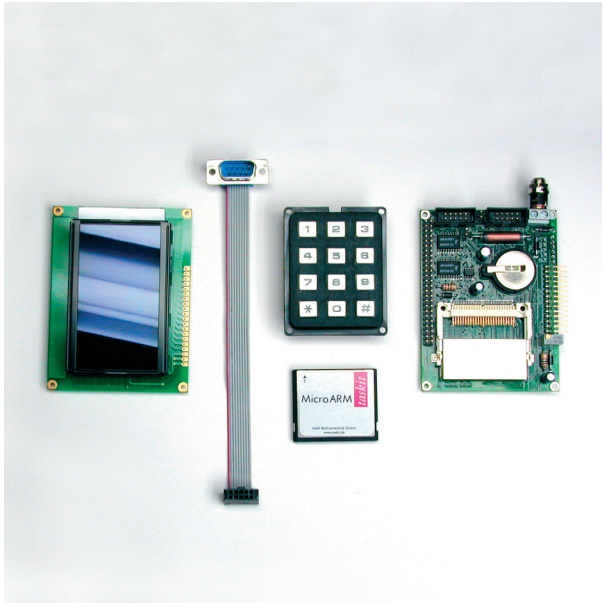
With the starter kit, you can put the MicroARM to use right away without additional hardware or software. The evaluation board just needs to be connected to a power supply and to the serial interface of a PC. After you start the terminal program, the MicroARM is ready to execute your commands and programs.



The Starter kit

Effective power management

Even at the maximum clock speed of 59 MHz, the MicroARM consumes less than 50 mA at 3.3 V. When using a single Mignon cell (e.g. NiMH battery) as a power source, this already constitutes a battery life of over 20 hours.



The starter kit

By slowing or pausing the processor core or peripherals, the energy consumption can be drastically reduced even further. As soon as an interrupt occurs, the MicroARM resumes its active state immediately.

Since, in practice, most applications spend the most time waiting for external results, this technique can minimize power consumption to a fraction of the nominal value without sacrificing computing power or significantly changing the response time.