Virtual eXecuter

Indhold

[Indledning 3](#_Toc198449658)

[Systembeskrivelse 4](#_Toc198449659)

[User interface 4](#_Toc198449660)

[Systemoversigt 4](#_Toc198449661)

[Technical considerations & choices made during the work 4](#_Toc198449662)

[Sources, references, links etc. 5](#_Toc198449663)

[Make CD with docs, sources, references, datasheets etc. 5](#_Toc198449664)

[Test results 5](#_Toc198449665)

[Conclusion 5](#_Toc198449666)

# Indledning

The VX virtual machine is designed for small embedded systems originally intended for 8 bit microcontrollers.  
When building the virtual machine (VM) core a user provided configuration file is used to configure the system. This configuration file specifies which type of program memory is available, how (if at all) errors must be presented and handled and all the specifics of the hardware.

The preliminary focus is to define and implement a working VM. Execution speed and core size is not the focus of the first version and so it’s not expected to outperform any existing controller.

The instruction set is designed to allow for a short leaning period and will therefore look fairly similar to that of a standard 8 bit microcontroller. Two exceptions from this are that the instructions are stack based and that threading is supported directly on the lowest level (threading is not supported yet!).

Inspiration for the VX has been taken from the AVR, the PIC, the 8051, the x86, and the ARM hardcore microcontrollers, and the MicroBlaze and PicoBlaze softcore microcontrollers. Java byte code has also contributed to the development.

Claus Andersen – 04296

# Systemoversigt

The main focus of the VX project is to make a virtual machine capable of running on the 8 bit AVR microcontroller. The chosen design enables programs of up to 16 MB in size to be loaded and executed.

The VX project consists not only of the VX itself but a complete package ready to run.

This package includes

|  |
| --- |
| **Brugerapplikationer** |
|  |
| **VX virtual machine** |
| SoftPeripherals |
|  |
| **MROS** |
| Terminal service  FileStore filsystem  DRAM controller |
|  |
| **Hardware platform** |
| ATmega64  USB port  Offline storage (EEPROM + DataFlash)  Arbejdshukommelse (1 MB 30 pin DRAM) |

# Systembeskrivelse

Block diagrams & figures

Define system architecture & user interface

# Systemkomponenter

## Operativsystem

### Trådhåndtering

### Uart

### Commander

### EEPROM

### Filsystem

### DRAM controller

## Brugergrænseflade

### Terminal

## Virtuel maskine

# Technical considerations & choices made during the work

(pros/cons of decisions)

# Sources, references, links etc.

# Make CD with docs, sources, references, datasheets etc.

# Test results

# Conclusion

(success or not?)