

μTasker project for the Freescale™ Kinetis Cortex™ M4



Embedding it better...

- **Small footprint** - The μTasker solution is especially suited to projects where an easy to use, slim-line operating system and TCP/IP stack are of importance. This makes it ideal for single chip applications where the control of resources is critical but a high level of developer/user comfort can't be compromised.
- **μTasker operating system** - efficient and easy to use operating system to enable very modular and manageable software. Inter-task queues with flexible software and HW timer support. Supports distributed processing features over an integrated fast network protocol.
- **μFileSystem and μParameterSystem** - Highly efficient FLASH, SPI FLASH or SPI EEPROM file system support, optimised for typical embedded applications.
- **μFAT** - FAT32 compatible file system with SD card interface support, optionally integrated into FTP and HTTP servers supporting reading LFN (long file names).
- **User Files support** - Allows embedded files in code to be used or mixed with μFileSystem files, including utility for creating an embedded set of user files for simple upload to targets.
- **Device drivers** - UARTs, PC, CAN, LCD, Matrix keyboard, ADC.
- **USB support** - user friendly configuration and interfacing, with buffered streaming mode as well as firmware upload capability.
- **Special support** - DMA library routines (uMemcpy, uMemset), Low power task.
- **TCP/IP stack** - ARP, RARP, ICMP, IP, UDP, DHCP, DNS, TFTP, TCP, dynamic HTTP, HTTP post, FTP, SMTP, POP3, TELNET, NetBIOS and VLAN plus various useful utilities for real-world Internet work. Integrated with the μTasker operating system for optimal performance.
- **Graphical LCD support** - μGLCDLib and graphical LCD simulator.
- **Boot Loader** - The μTasker project has integrated support for the "Bare-Minimum" boot loader which enables complete and secure software uploads via Ethernet/Internet (FTP and HTTP POST) as well as USB, and occupies less than 2k of FLASH space! Optionally it allows uploads to external SPI FLASH as well as encryption.
- **Serial Loader** - The μTasker project supports an SREC based serial loader for deleting and loading new application over a UART, optionally encrypted.
- **Unique μTasker simulator** - allows the supported devices and peripherals to be simulated in "real-time" on the PC and can dramatically reduce development times compared to classical debug methods on the target.
- **Ready-to-run with pre-configured projects** - The μTasker project is delivered with setups for GCC, IAR, Keil uVision, Rowley CrossWorks and CodeWarrior. The μTasker project is a high quality demonstration project which can be used as base for professional work.

The μTasker project targets single-chip embedded control and communication projects where reliability and fast development are of importance.

M.J.Butcher Consulting is Freescale™ Design Alliance Partner and offers training and coaching for the μTasker project and its supported processors as well as design and development services, with the emphasis on optimising processor resources and reducing software development times.

Modbus extension package:

- **MODBUS Serial RTU/ASCII**
 - Multiple serial master and slave interfaces
 - Multiple slaves at each UART interface (each with its own slave address, resources and application interface)
 - Routing of received messages to other MODBUS master interfaces based on function codes, access ranges or other decisions
 - RS485 RTS control
 - Serial/serial bridge function
 - ASCII slave support via USB
- **MODBUS/TCP**
 - Multiple master and slave TCP ports
 - Each slave TCP port supports multiple sessions (multiple master connections)
 - Configurable idle connection timeout on each TCP port
 - Multiple slaves on each TCP port (each with its own slave address, resources and application interface)
 - Slave gateway operation to TCP or serial MODBUS masters
 - Slave routing operation to multiple TCP or serial masters via a single TCP port
- **General**
 - Autonomous handling of commands and requests or with application intervention
 - Coils and discrete elements can be optimally overlaid with registers (architecturally independent)
 - Simple generation of public MODBUS function data content, simplifying application interface
 - Supports delayed slave responses while external data is accessed
 - Complete MODBUS projects can be simulated in the μTasker simulator

Processor support:

- ATMEL™ SAM7 – ARM7
- ATMEL™ AVR32 – UC3A/UC3B
- Luminary Micro™/Texas Instruments™ Stellaris – LM3Sxxx Cortex™ M3
- Freescale™ – M522xx Coldfire V2 (incl. M520x, M521x, M528x)
- NXP™ – LPC200X ARM® 7
- NXP™ – LPC17XX Cortex™ M3
- ST Micro™ STM32 Cortex™ M3
- ST Micro™ STR9XX ARM®9

Visit the μTasker web site for more information, documentation and user's forum:

www.uTasker.com / www.uTasker.com/forum

Contact details in Switzerland at www.mjbc.ch