

μEZ[®] Rapid Development Platform

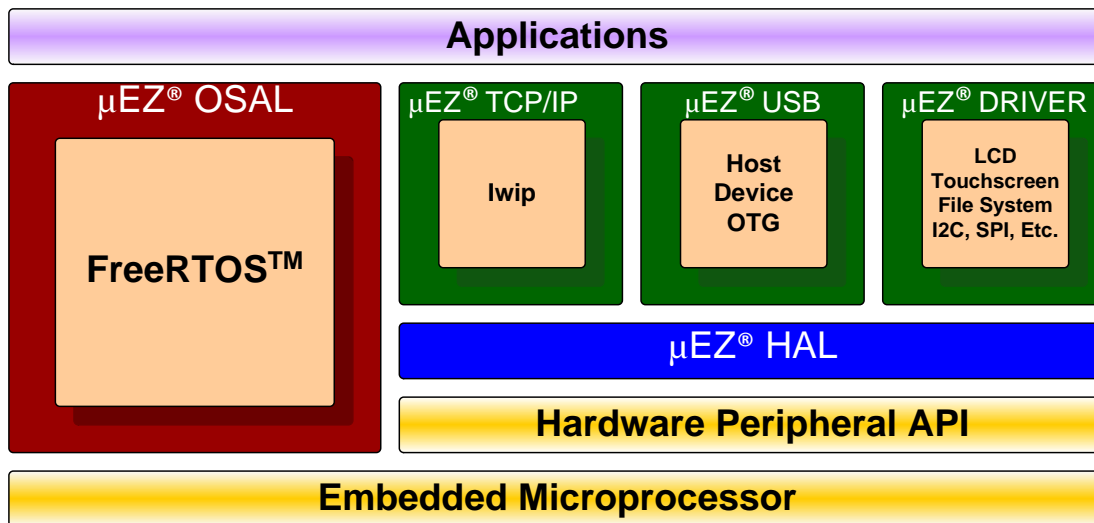
SMALL * FAST * PORTABLE * AFFORDABLE



μEZ takes its name from the Muses of Greek mythology. A Muse was a goddess who inspired the creation process for the arts and sciences. Like its ancient Greek namesake, the μEZ platform inspires rapid development by supplying customers with an extensive library of open source software, drivers, and processor support - all under a common framework. μEZ development works on the premise of "design once, reuse many times". This provides an open source standard for embedded developers to build upon and support. μEZ allows companies to focus on innovation and on their own value-added applications while minimizing development time and maximizing software reuse.

The diagram below shows a typical embedded application stack. μEZ has three primary categories of components that help simplify embedded application development:

- 1. Operating System Abstraction Layer (μEZ OSAL)**
- 2. Sub-system drivers (μEZ TCP/IP, μEZ USB, μEZ Driver)**
- 3. Hardware Abstraction Layer (μEZ HAL)**



The selection of an RTOS can be one of the most daunting aspects of an embedded system development. With μEZ the primary features of common multi-tasking operating systems are abstracted, thus easing the transition to an open source or low-cost RTOS. The μEZ OSAL provides applications access to the following features in an OS-independent fashion:

- Pre-emptive multitasking
- Stack overflow detection
- Unlimited number of tasks
- Queues
- Semaphores (binary, counting, mutex)

The μEZ sub-system drivers utilize the OSAL functions to provide protected access to the processor peripherals. The sub-system driver API functions are typically protocol layer interfaces (TCP/IP, USB, etc) designed as high-level access routines such as open, close, read, write, etc. where possible.

The HAL functions provide single-threaded unprotected access to the processor peripherals. Customers can use the μEZ HAL routines provided by FDI or they can write their own. The HAL routines provide for RTOS/μEZ independence and allow portability within a family of processors.

μEZ is ideally suited for Embedded Systems with standard features such as:

- Processor and Platform BSPs (Board Support Packages)
- Real Time Operating System (RTOS)
- Memory Management
- NAND/NOR Flash
- SDRAM and DDR Memory
- TCP/IP stack
- USB Device/Host Libraries
- Mass Storage Devices
- LCD Displays with Touch Screen
- Input / Output Devices

μEZ® Frequently Asked Questions

DESIGN ONCE REUSE MANY TIMES
DESIGN ONCE REUSE MANY TIMES

What is μEZ and how does it help me?

μEZ is a Rapid Development Platform designed to simplify embedded software development. It should help you get your product to market faster and with a lower development cost than other options. When you get right down to it, what's more important than saving time and \$\$\$!

The goal of the μEZ platform is to provide underlying RTOS and processor abstraction, enabling the programmer to focus on the value-added features of their product. μEZ enhances portability of application code to multiple ARM® platforms with high reusability.

OPEN SOURCE = FREE

Is μEZ expensive - what does it cost?

μEZ is an open source, middleware platform so there is *no cost* to the user. Customers with the necessary engineering resources, a compatible compiler suite, and a little free time can directly integrate μEZ into their embedded application. For customers with time or resource limitations, FDI offers affordable integration services that are customized to your hardware and software requirements. These FDI integration services are full turnkey, can cost as little as \$10K, and may require only 4 weeks of schedule time to complete and test on your hardware.

FDI

Where do I get μEZ and who provides support for the platform?

The μEZ source code and documentation can be downloaded from www.sourceforge.net/projects/uez. Marketing updates and details on technical support are available at www.teamfdi.com/uez.



Why not use Linux?

Linux is a powerful OS with a much larger code and memory footprint and requiring a processor with a dedicated MMU (Memory Management Unit). μEZ is designed to work with lower cost processors with smaller memories and, whenever possible, to use only the processor's internal ROM and RAM. A recent study undertaken by FDI reviewed the cost increases likely for a Linux based system over a μEZ based system. The study found a potential hardware cost savings of \$25 - \$40 per unit for the μEZ system. This savings is significant since it might represent 25 - 50% of the total embedded hardware cost.

In addition, setting up a full Linux development system can be a source of frustration for many developers. μEZ uses a single source tree for all its applications and drivers, thus providing a build system compatible with the IDE of choice. Compiling μEZ is as easy as compiling a single application.

I don't need all of the features you listed, can μEZ be trimmed down?

μEZ was specifically designed and structured to allow developers to pick and choose only the features that are needed for their particular application. Typical code and data memory footprints for μEZ are 300KB code and 50KB data memory (excluding data memory for LCD or display Frame Buffers).

Can we use another RTOS?

All μEZ components are made to connect through the μEZ OSAL (Operating System Abstraction Layer) to the RTOS ensuring compatibility with many different RTOS's. Currently all μEZ commercial development is focused on FreeRTOS™ since it satisfies the low cost tool requirement because it is "free". A μEZ SafeRTOS version is also available for mission critical applications. Other RTOS products can also be used with μEZ.



Which compiler suites do you support?

Currently, most μEZ development by FDI has been focused on the low cost Rowley CrossWorks compiler, the IAR EWARM tool suite and the Keil uVision but GNU and other compilers can be used with μEZ.

What debug tools are available?

Since μEZ uses the debug tools that are provided in the customer's compiler suite, it can be used with any of the tools listed above.

Which processors are supported?

Even though μEZ is processor independent, all of our initial development has been focused on various members of the ARM and Renesas RX families. We currently support the NXP LPC24xx family, the NXP LPC17xx, and the Renesas RX62N family.



μEZ® is a registered trademark of Future Designs, Inc.
FreeRTOS™ is a registered trademark of Richard Barry

www.teamfdi.com