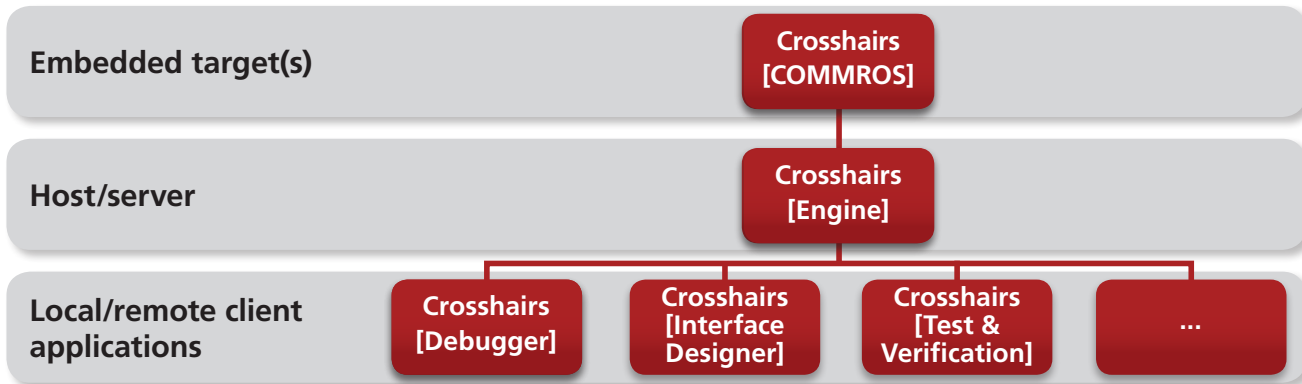


Crosshairs Embedded [Engine]



Engine

- The Crosshairs Embedded [Engine] provides unsurpassed flexibility and scalability for low-level embedded systems interaction.
- All communication from engine to clients is done with TCP/IP making remote operation seamless.
- Uses data-driven architecture which makes supporting custom hardware and new architectures very easy.
- Supports multiple concurrent clients as well as multiple concurrent embedded target connections.

Technical

- Written in portable C++.
- Efficient, multi-threaded architecture to maximize data throughput.
- Communication medium agnostic – serial and Ethernet supported out-of-the-box.
- Uses authentication for client applications.
- Minimal hardware requirements (low CPU usage, less than 32 Mb memory consumption under load).

```
2010-Feb-23 18:33:44 - Default BinPath:C:/Program Files/Crosshairs Embedded/Crosshairs [debugger] 1.
2010-Feb-23 18:33:44 - Default dataPath: C:/Users/Espen/Documents/Crosshairs Embedded/Debugger/data,
2010-Feb-23 18:33:44 - Trying to locate old log-file: C:/Users/Espen/Documents/Crosshairs Embedded/I
2010-Feb-23 18:33:44 - Notice: We are trying to delete old Engine logFile-file
2010-Feb-23 18:33:44 - Deleted old Engine log-file
2010-Feb-23 18:33:44 - Crosshairs Embedded [engine] - Version 1.2 (build 3777) (c) Crosshairs Embedde
2010-Feb-23 18:33:44 - Crosshairs [engine] has 2 target(s)
```

Crosshairs Embedded [COMMROS]

COMMROS

- Small, configurable and portable communications kernel that provides non-intrusive hardware access to all Crosshairs Embedded products.
- Sizes range from 800 bytes to about 8Kb depending on configuration.
- Simple and flexible integration, either in standard mainloop-style applications or with an RTOS.
- Requires very minimal code changes (three lines of code added to end-user application).
- Available as C or C++ libraries with full integration examples.
- Currently available for TI C2000, MSP430 and Cortex M3.

Well suited for hard real time systems

Crosshairs Embedded is non intrusive. It does not require a lot of adaptation of target code. Variables are accessible, referred to by the name used in the source code.

Kjell Ljokelsoy

Kjell Ljokelsoy
Research Scientist
Sintef Energy Research
www.sintef.no

Table 1: Commros versions and features

	Tiny*	Small	Regular	Full
Ping	X	X	X	X
Reset	X	X	X	X
Read/Write memory (not atomic)	X	X	X	X
Download Code	X	X	X	X
Run functions	X	X	X	X
Packet interface	X	X	X	X
Serial Comm. Interface		X	X	X
Atomic read/write of variables			X	X
Start/Stop			X	X
Crc 32			X	X
Data logger			X	X
Read multiple variables at the same time (increased update speed)				X
Operators				X
Bit fields support				X
External Flash Programming				
Program Words (flash) **	0x1C5 (C) 0x1C5 (C++)	0x357 (C) 0x357 (C++)	0x87D (C) 0x883 (C++)	0x9EA (C) 0x9F0 (C++)
Data words (RAM)	0x90 (C) 0x90 (C++)	0x13E (C) 0x13E (C++)	0x194 (C) 0x194(C++)	0x194 (C) 0x194 (C++)

* This version does not support serial communication

** Texas Instruments C2000 Code Generation Tools 5.2.1