EEMBC's Automotive/Industrial Microprocessor Benchmarks

June 4, 2004

EEMBC's Automotive/Industrial Benchmark Suite

16 different algorithms used in automotive applications

- Angle-to-time conversion
- Basic floating point
- Bit manipulation
- Cache buster
- CAN remote data request
- FFT
- FIR
- IIR
- IDCT
- Matrix arithmetic
- Pointer chasing
- Pulse width modulation
- Road speed calculation
- Table lookup and interpolation
- Tooth-to-spark

Auto$_{mark}$ score aggregates the individual performance measures

EEMBC Adds Suite of Benchmarks for 8- and 16-bit Microcontrollers

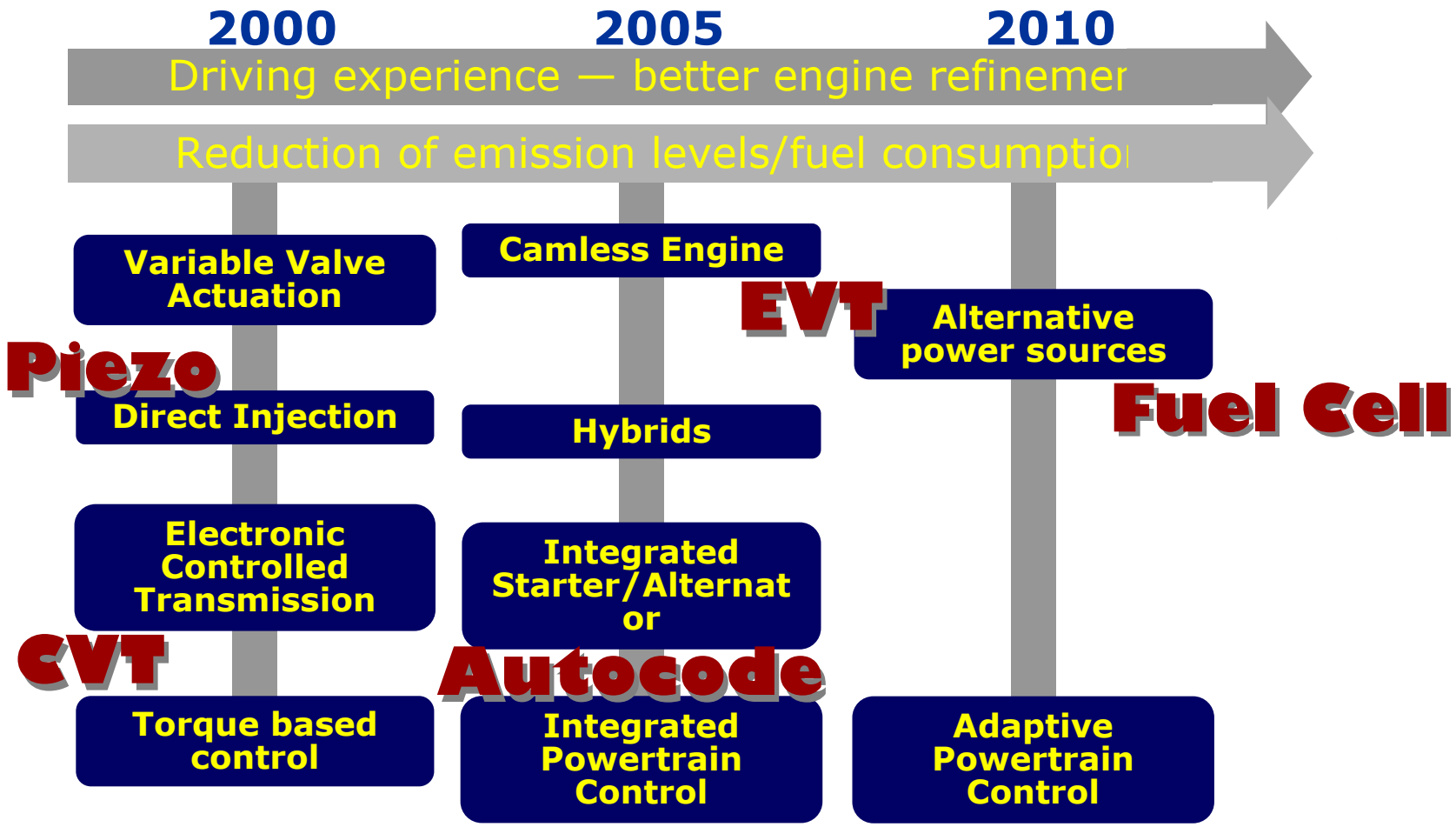
In the automotive world, 8- and 16-bit microcontrollers remain an important category of processing devices

- Industry's first certifiable benchmarks for 8- and 16-bit microcontrollers
- Converted from automotive/industrial benchmarks
- Shrunk for smaller memory footprints
- No floating-point
- New task-based benchmark

Why high performance microcontrollers are crucial for automotive.

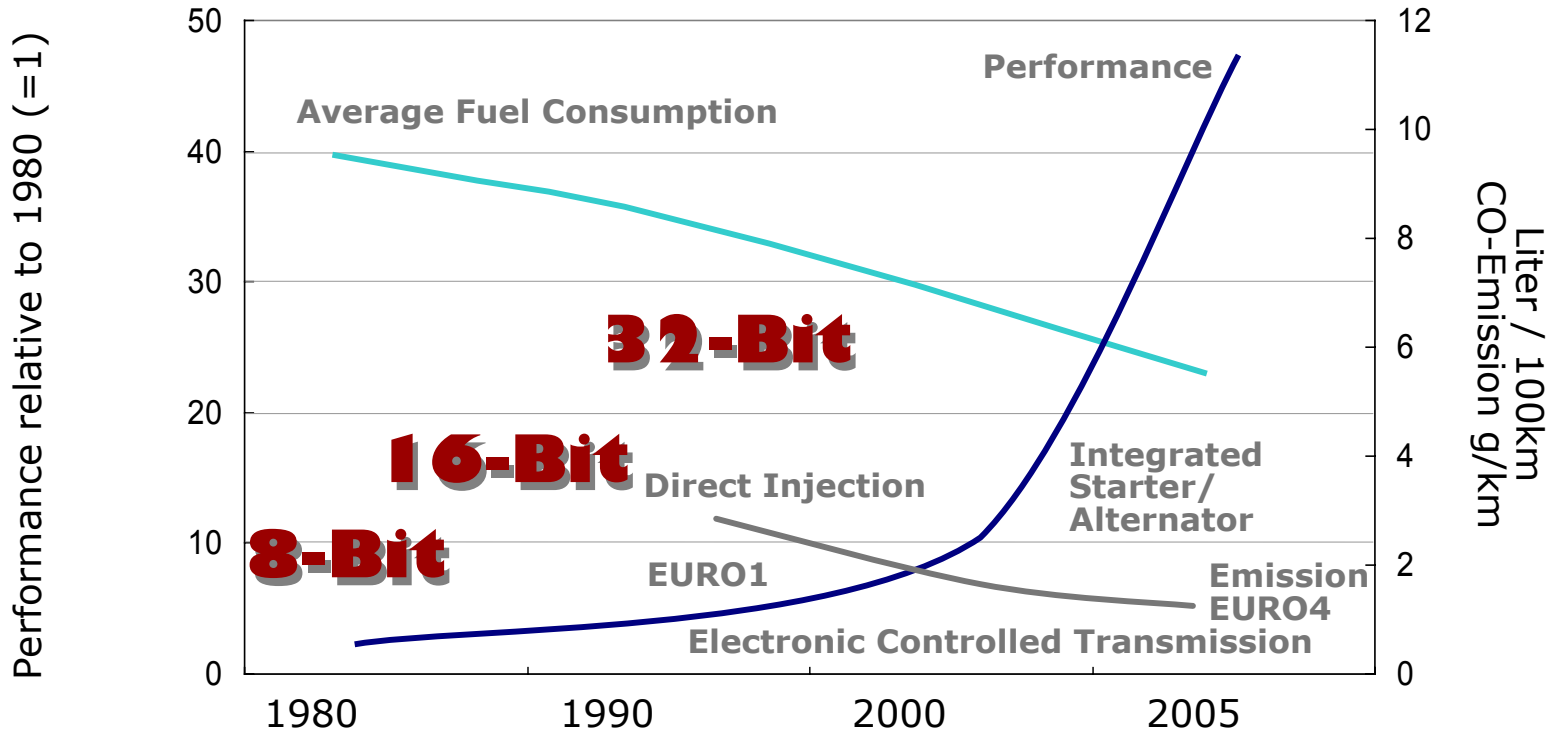
- Microcontrollers help to reduce emissions and fuel consumption because they provide tighter feedback and allow the system to more rapidly and accurately track the operating environment and conditions
- Though processors are already used for widespread applications, there are still a number of areas where processors can bring new capabilities and lower prices
- The burgeoning performance of 16- and 32-bit processors permits using chips for new control functions, particularly for safety and entertainment applications inside the car
- Additionally, for the telematics industry to grow as predicted, inexpensive yet efficient processors must perform many tasks, such as voice recognition

Powertrain – Driving Forces and System Trends



Source: Infineon Technologies

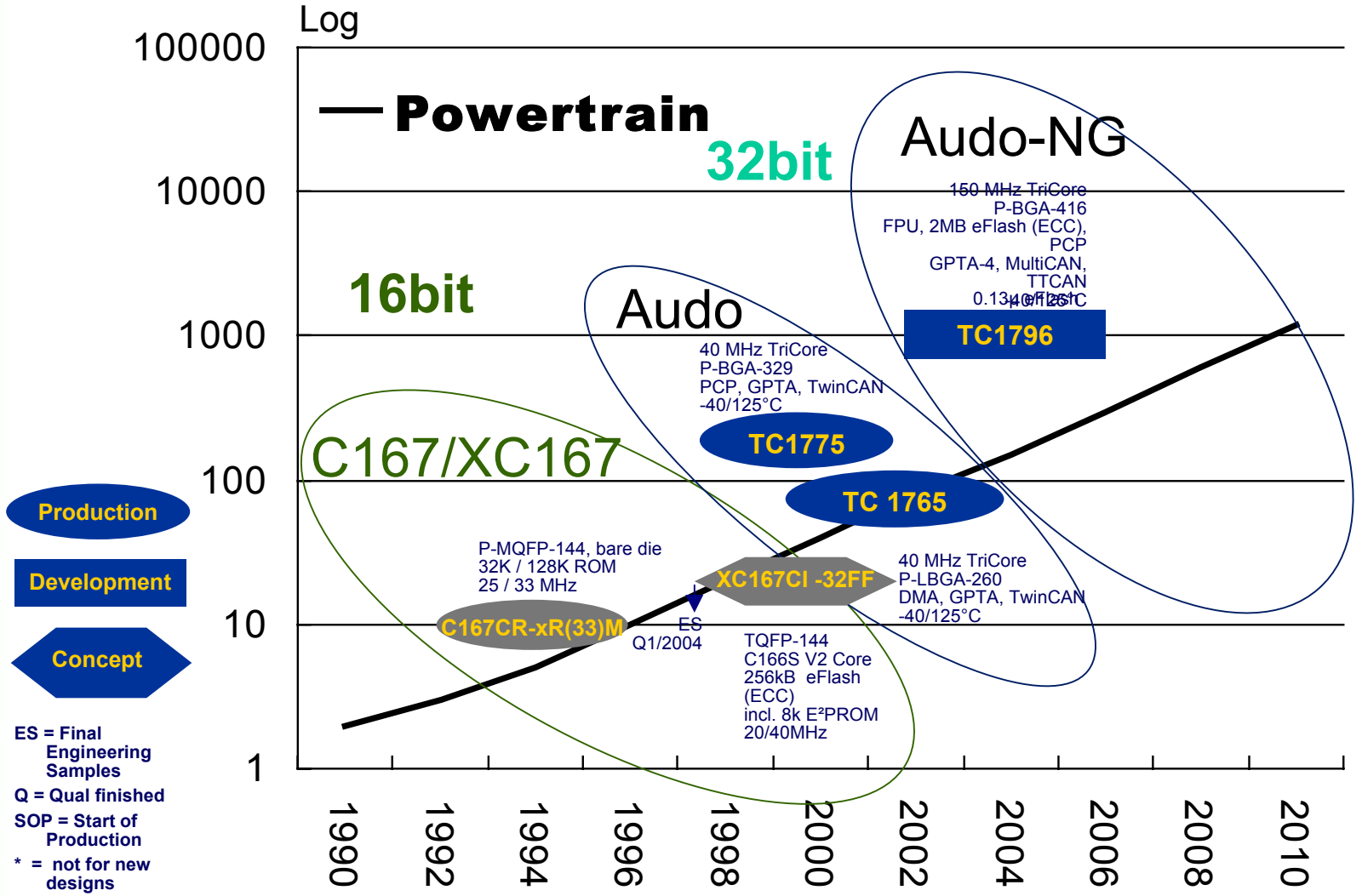
Powertrain Driving Forces for Higher Performance



Source: VDA, Infineon Technologies

EUROx = European Emission Standards

Powertrain Driving Forces for Higher Performance



Source: Infineon Technologies

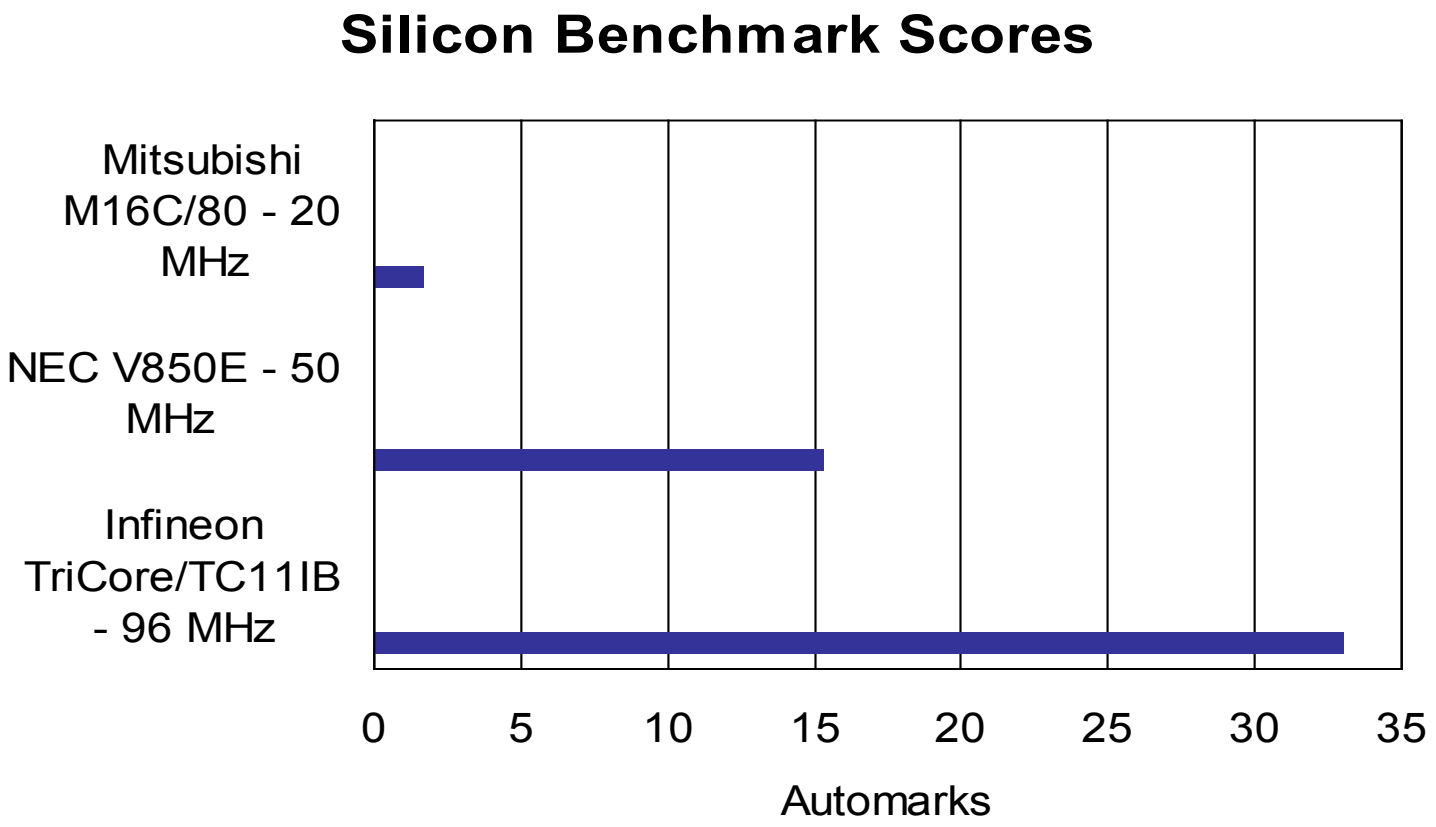
Changing Requirements for Microcontrollers

Feature	1994	2004
Architecture	16-bit	32-bit
Frequency	33 MHz	150 MHz
Instruction cycle time	60 ns	4.44 ns (typical)
Flash Memory	0	2 MB
RAM Memory	4 kB	192 kB
Input/ Output Handling	Peripheral Event Controller	32-bit I/O Processor
Timer for Motor Control	Capture/Compare (9 Timers)	GPTA (96 Timers)

This table summarizes the information provided in the previous graph, showing the dramatic change in microcontroller requirements over the past 10 years.

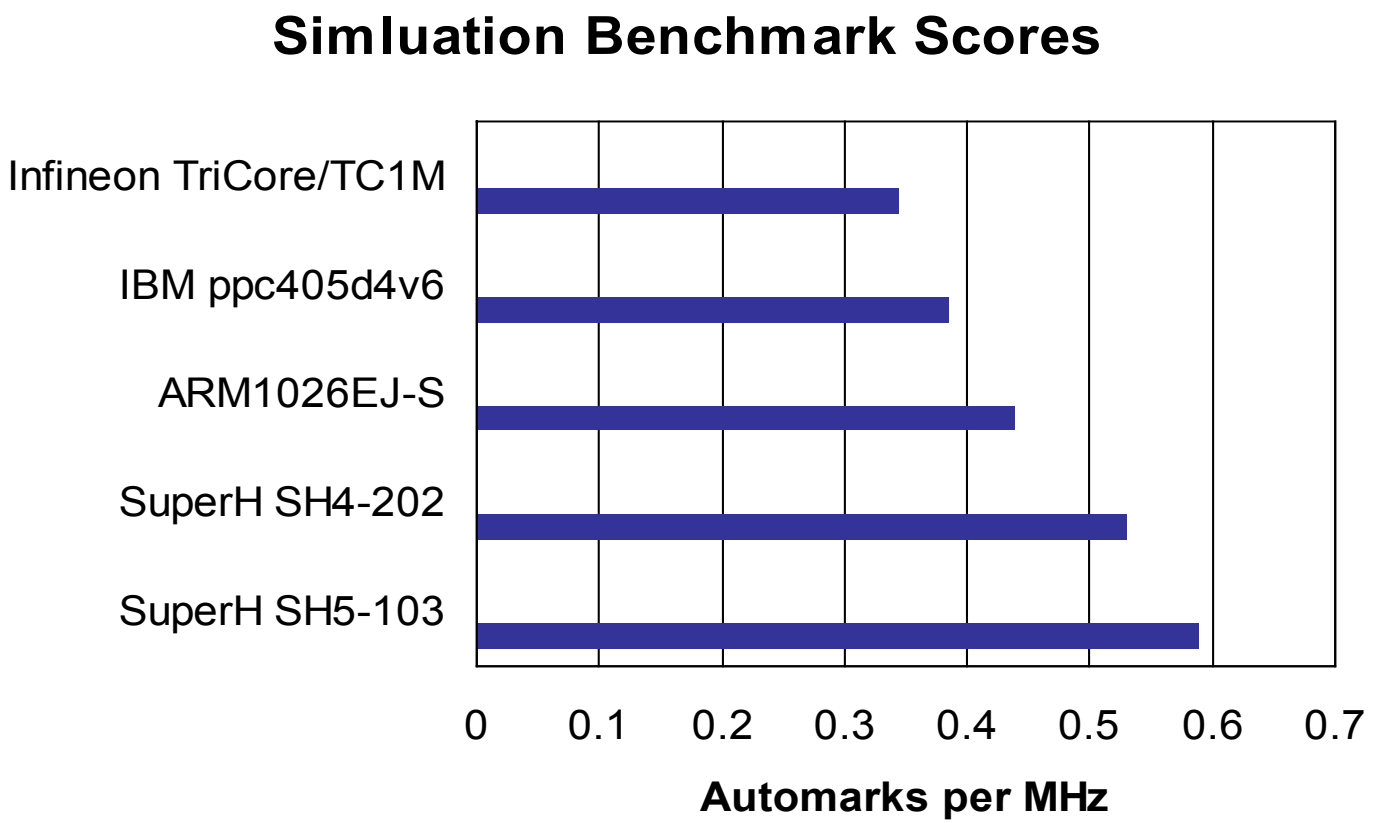
Source: Infineon Technologies

EEMBC Benchmark Results



This chart compares three processors using EEMBC's Automark (The Automark is calculated by taking the geometric mean of each of the individual scores within the Automotive/Industrial benchmark suite). These results are more or less 'expected' because a) the Mitsubishi device is a 16-bit microcontroller compared to the other two devices which are 32-bit microcontrollers; b) operating frequency is a big factor in performance measurements; c) after normalizing for operating frequency, the Infineon device is still approximately 10% faster than the NEC device, this is most likely related to the former's enhanced ability to perform signal processing.

EEMBC Benchmark Results



Similar to the explanation on the previous slide, this chart compares 5 processor architectures using EEMBC's Automarks. One of the significant differences on this chart is that these scores are all based on running the benchmarks using simulators for these processors and all scores are normalized to 1MHz. Without going into the specific details for each processor, it's important to note that the ARM and SuperH devices support hardware floating-point units that will help boost their performance on several of the Automotive/Industrial benchmarks.