CE Requirements from a Supplier Perspective

The Open Group
Real-time & Embedded Systems Forum
2005/JAN/23
Shozo TAKEOKA
AXE,Inc.

Who am I

(short self introduction)

AXE provides embedded OS

- Sharp Zaurus (PDA)
 XTAL:
 AXE's embedded OS

- Sharp (DoCoMo)
 Mobile Phone;
 DSPBridge: communication middleware ARM-DSP

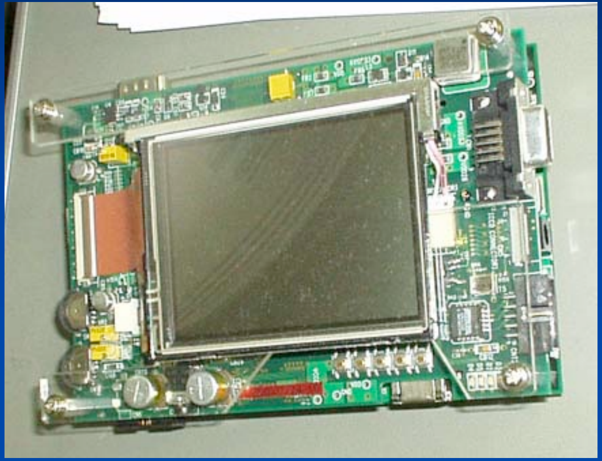
- Olympus
 Digital Camera;
 XTAL

- KonicaMinolta
 WebCamera; axLinux

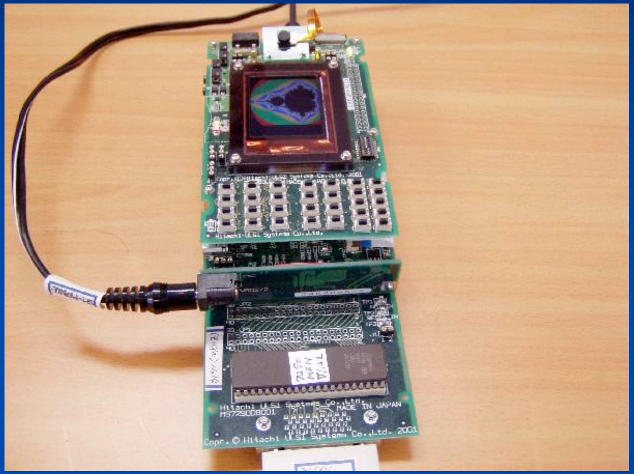
- Panasonic Projector
 axLinux

- Toshiba Projector
 axLinux

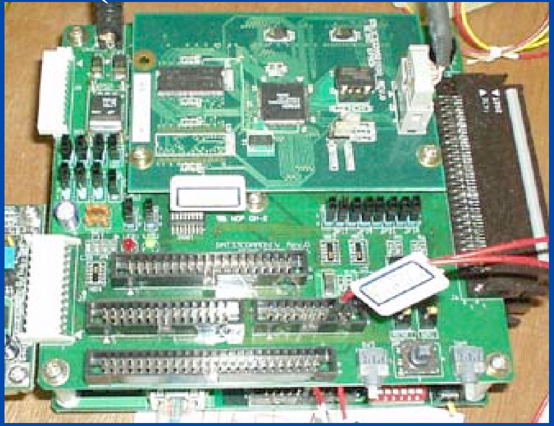
Work with Japanese SemiconCorp. (authorized axLinux partner)



Fujitsu FR/V
(32bit VLIW for embedded)

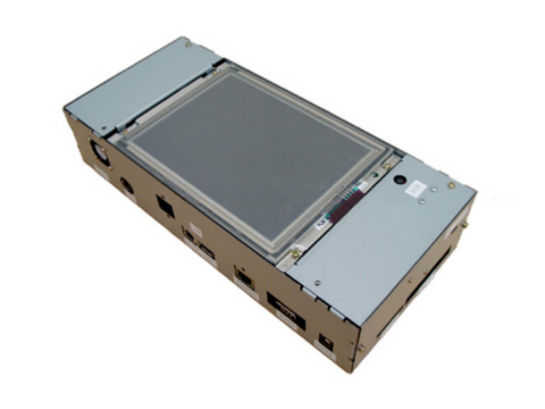


Hitachi (Renesus) SH-Mobile



Epson C33

Sharp LH795xx(ARM)



Sanyo LC690132(ARM)

Cowork with Printer makers

- Canon, Epson, FujiXeroxPrinting etc…

  AXE is working with Canon, Epson in UNIX/Linux printing system(for desktop & server)

- Only AXE providing InkJet-printer-driver for non-x86 Linux (Canon , Epson)

Requirement from Japanese CE Area

My Focusing points

- Consumer Electronics
- Advanced automobile devices, car navigation, car audio

- Embedded single-chip multi-processor
  Multicore
  SMP on single chip

- AXE provides embedded OS to
  Panasonic, Sharp, Sony, Olympus, Konica-Minolta,
  VodafoneK.K., KDDI …
  for advanced CE devices, Digital Camera, Mobile Phone

in Consumer Electronics (and advanced automobile devices)

- No hard realtime required.

  OS for advanced devices:

  Linux, NetBSD

  WinCE

  T-Engine (future), (iTRON has less middleware for new age)

- CPU power resource management is required.

  This is not solved by realtime.(such as priority)

- response time not recommended hardly in ordinary application.

  Of course, some application requires realtime.

Highend systems and Communication Devices

- hard realtime required, of course

- Mobile Phone needs hard realtime in some part.
  hard realtime things are processed in special CPU
  for communication, its OS is iTron, VxWorks

Mobile Phones use Application processor(s) for
WebBrowser, E-mail, Movie(MPEG4).
Those OS is required
NO realtime
well user interaction (very soft realtime? No deadline)
fast startup

Mutlicore CPU (SMP)

- Japanese silicon vendor‘s embedded processors are becoming multi-processor.
- Multicore
- SMP

Japanese Embedded users wait new APIs

- QoS API
- API or some framework for Multicore and SMP.

required QoS API standard for multimedia of CE

- HDD or DVD Video Recorder/Player

  HDD Video Recorder/Player with network.

  two or more processes runs at same time.

  Some HDD Recorder troubled without QoS.

  Network Protocol Stack consumes large CPU power on receiving many packet,and MPEG-player works less frame rate.

- Application combination requires QoS; for example

  MPEG and Network,

  Video and Audio.

  CPU-power-ratio must be changed dynamically (by combination of applications)

- AXE is already providing QoS under axLinux

  specify PID and percentage of CPU-power via /proc/qos special device (specify to running process.)

Priority vs QoS in CE

- Priority is used specifying really High priority process

- QoS is resource control

  resource:

    CPU power

    Network bandwidth

required API for Multicore, SMP.

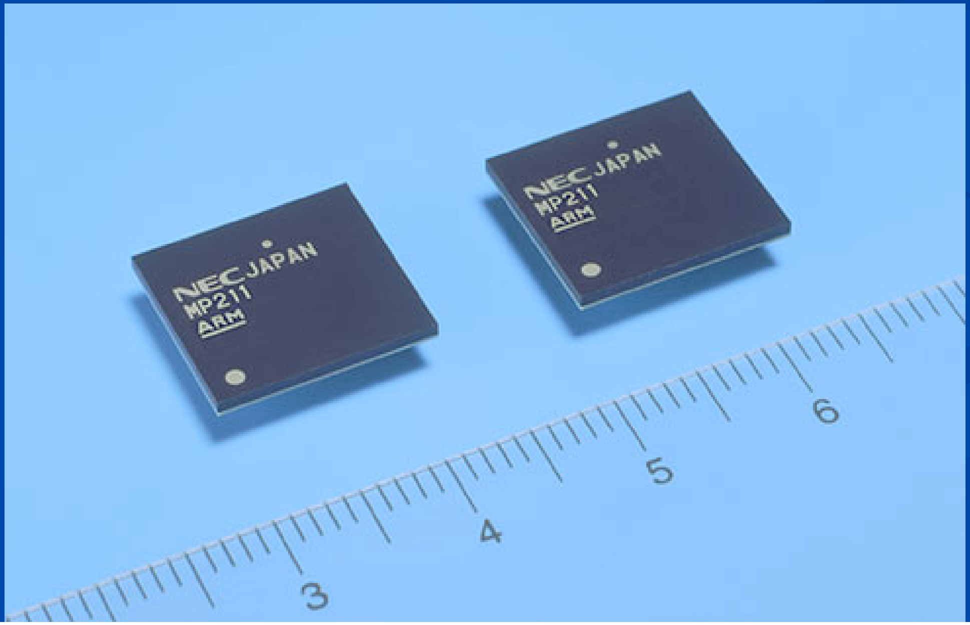
- Japanese silicon vendor's embedded processors are becoming multi-processor. (NEC, Fujitsu will ship Multicore CPU)
- INRIA (French advanced computer project) reports
  importance of software platform for Multicore CPU
- Traditional RT or embedded OS's API is just for single CPU.
  User studied established technique for SMP in 20 century.
- Muticore is not SMP.
  SMP has cache coherency.
  Some Multicore dose not have it always.
- processing an I/O device by Multicore has some problem.
  Synchronization:
    not useful the technique for SMP, cache is incohearent.
  holding the processing data on cache

Multicore

Multicore CPU (real chip)

- NEC　MP211
- 3 ARM cores on single chip
- No coherent cache
- Targeting MultiMedia with low power:

  Digital TV, Mobile Video Phone(already serviced in Japan), MP3 audio in MobilePhone
- http://www.necel.com/ja/news/archive/0409/2701.html



Multicore report in France

- French INRIA Report

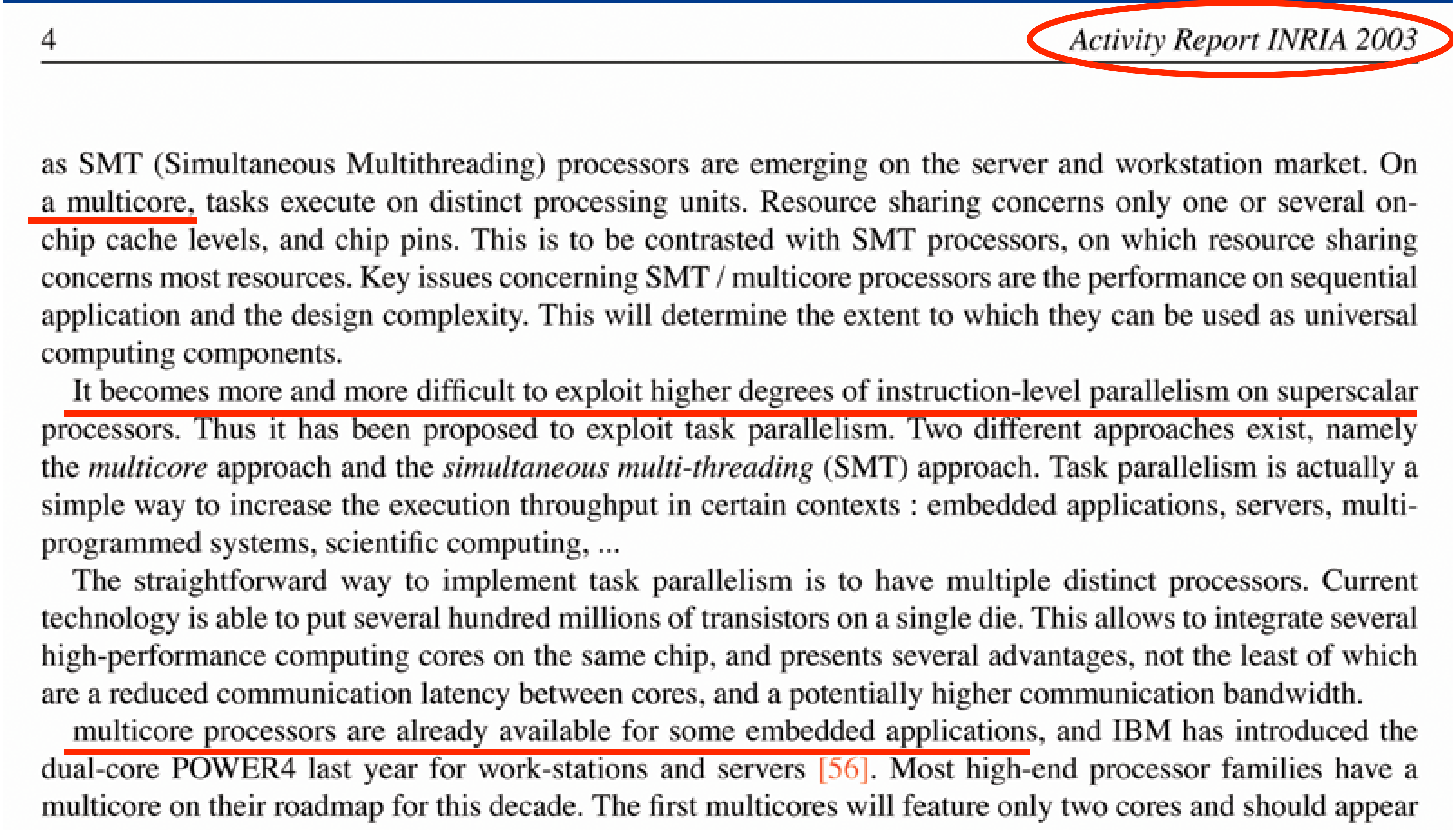
http://www.inria.fr/rapportsactivite/RA2003/caps2003/module5.html
http://www.inria.fr/recherche/equipes/caps.en.html
http://www.irisa.fr/caps/people/michaud/hpca2004.pdf

- ST micro is in France.

Multicore report in France

INRIA Project-Team caps Activity Report 2003

http://www.inria.fr/rapportsactivite/RA2003/caps/caps.pdf



4 Activity Report INRIA 2003

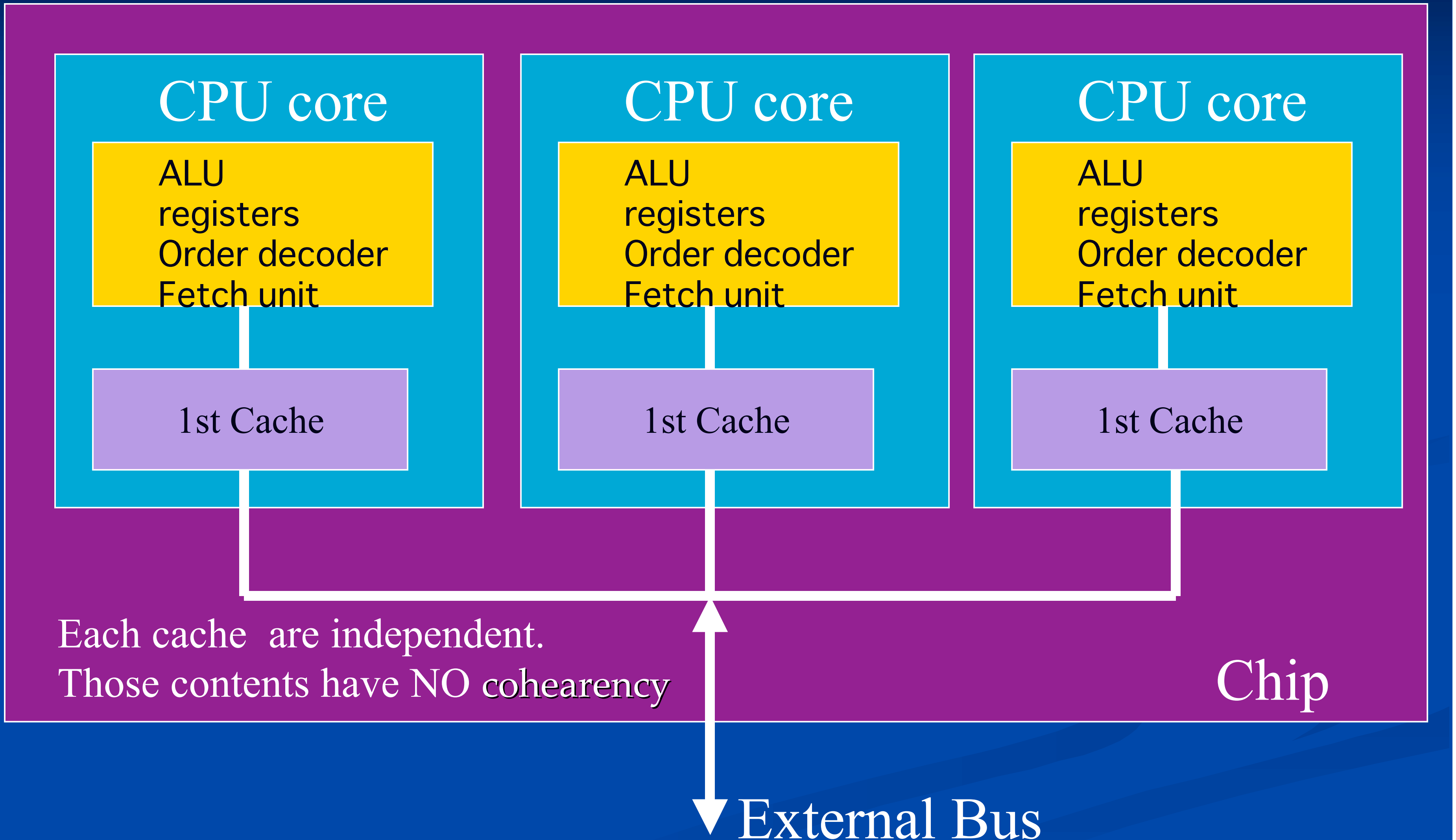
as SMT (Simultaneous Multithreading) processors are emerging on the server and workstation market. On a multicore, tasks execute on distinct processing units. Resource sharing concerns only one or several on-chip cache levels, and chip pins. This is to be contrasted with SMT processors, on which resource sharing concerns most resources. Key issues concerning SMT / multicore processors are the performance on sequential application and the design complexity. This will determine the extent to which they can be used as universal computing components.

It becomes more and more difficult to exploit higher degrees of instruction-level parallelism on superscalar processors. Thus it has been proposed to exploit task parallelism. Two different approaches exist, namely the *multicore* approach and the *simultaneous multi-threading* (SMT) approach. Task parallelism is actually a simple way to increase the execution throughput in certain contexts : embedded applications, servers, multi-programmed systems, scientific computing, ...

The straightforward way to implement task parallelism is to have multiple distinct processors. Current technology is able to put several hundred millions of transistors on a single die. This allows to integrate several high-performance computing cores on the same chip, and presents several advantages, not the least of which are a reduced communication latency between cores, and a potentially higher communication bandwidth.

multicore processors are already available for some embedded applications, and IBM has introduced the dual-core POWER4 last year for work-stations and servers [56]. Most high-end processor families have a multicore on their roadmap for this decade. The first multicores will feature only two cores and should appear

Multicore CPU



CPU core

ALU
registers
Order decoder
Fetch unit

1st Cache

CPU core

ALU
registers
Order decoder
Fetch unit

1st Cache

CPU core

ALU
registers
Order decoder
Fetch unit

1st Cache

Each cache are independent.
Those contents have NO cohearency

Chip

External Bus

New API for Multicore, SMP.
My Idea

- Specify Stick a process on 1 CPU, or Grabbing CPU
  process migration is high cost on Multicore (than SMP)

- New dynamic memory allocation primitive with specifying storage class.

```
void *malloc_with_mclass(int bytes, int STORAGE_CLASS)
```

Storage class : for example;
High speed on chip SRAM,
on board (off chip) SRAM,
DRAM (ordinary main memory),
unchachable DRAM

Unchachable area used for holding Mutexes on Multicore CPU.
Unchachable area is good for non-chache- coherency-system.
Unchachable area can be used for DMA.

pthread library under Linux(UNIX) on Multicore meets same problem.

End

www.axe-inc.co.jp