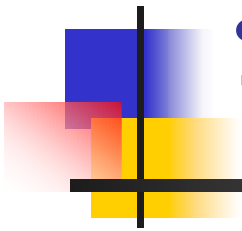
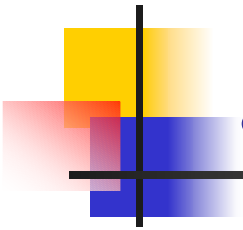


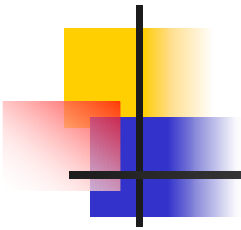
กรณีศึกษาของ T-Kernel RTOS บนแพลทฟอร์ม T-Engine

ผศ.ดร.ศุภชัย วรพจน์พิศุทธิ์
คณะวิศวกรรมศาสตร์
มหาวิทยาลัยธรรมศาสตร์



ภาพรวมในการนำเสนอ

- แนะนำแพลทฟอร์ม T-Engine
- การพัฒนาซอฟต์แวร์บน T-Engine
- ระบบปฏิบัติการ T-Kernel
- การศึกษาประสิทธิภาพของ T-Kernel
- สรุป



แพลทฟอร์ม T-Engine

- เทคโนโลยีของระบบสมองกลฝังตัว ซึ่ง Prof. Ken Sakamura คิดค้นต่อเนื่องจากระบบปฏิบัติการ μITRON
- มีข้อกำหนดชัดเจนทั้งในระดับของฮาร์ดแวร์และในระดับของซอฟต์แวร์
- แบ่งออกเป็น 4 ระดับ ได้แก่ standard, micro, nano, pico

แพลทฟอร์ม T-Engine

Standard T-Engine
75x120 ม.ม

μT-Engine
60x85 ม.ม.

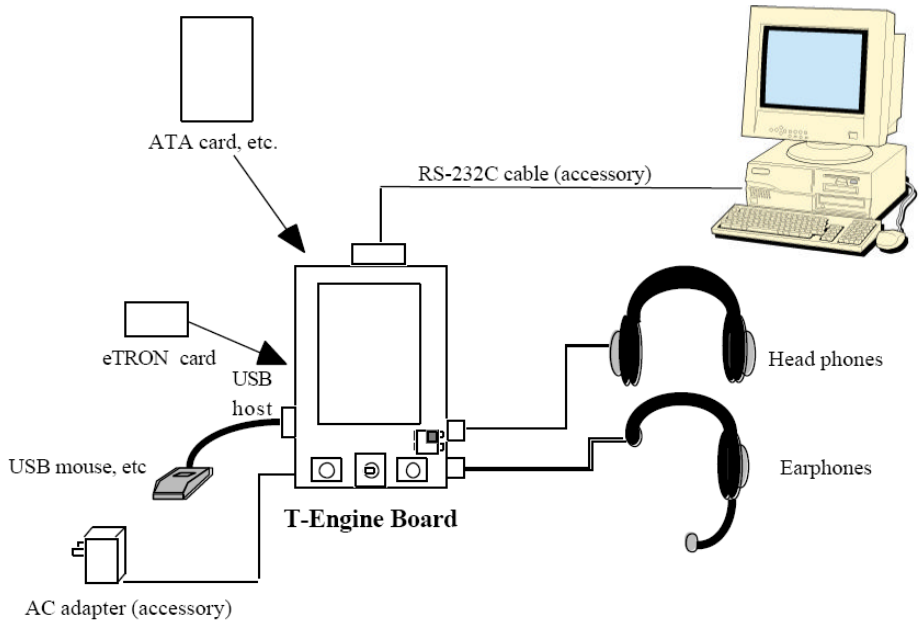
nT-Engine
28x33 ม.ม.

pT-Engine
20x20 ม.ม.

ชุดพัฒนา SH7727

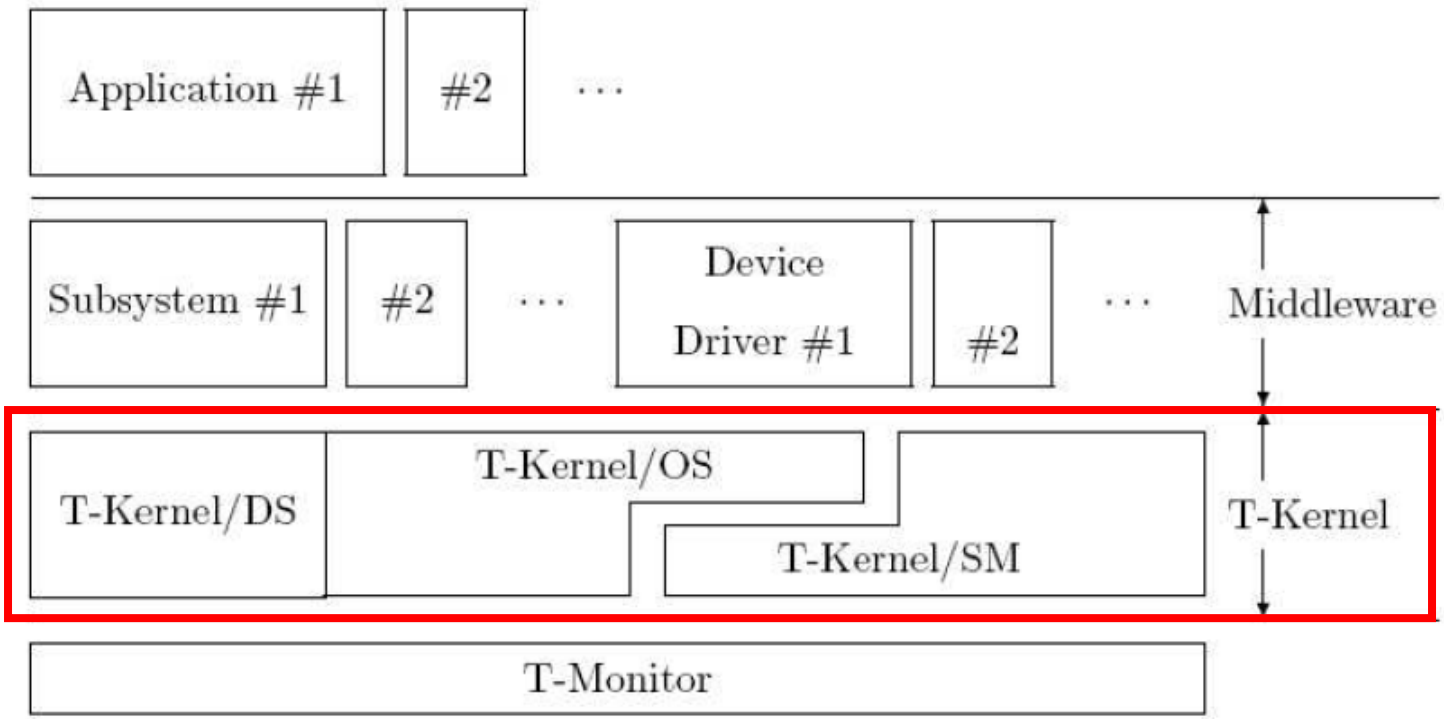
- หน่วยประมวลผลแบบ DSP รุ่น SH7727 ของ Renesas
- หน่วยความจำ
  - Flash 8MB
  - RAM 32 MB
- จอภาพ LCD 240x320 แบบสัมผัส
- ช่องเสียบอุปกรณ์แบบ USB และ PCMCIA

การพัฒนาซอฟต์แวร์บน T-Engine



- ติดตั้งซอฟต์แวร์ gcc, terminal software
- เชื่อมต่อคอมพิวเตอร์กับ T-Engine ผ่าน RS232C
- สร้างดิสค์ (USB,PCMCIA) สำหรับใช้ระหว่างพัฒนา
- พัฒนาซอฟต์แวร์ / ถ่ายข้อมูลผ่าน RS232C และสั่งทำงานผ่าน CLI

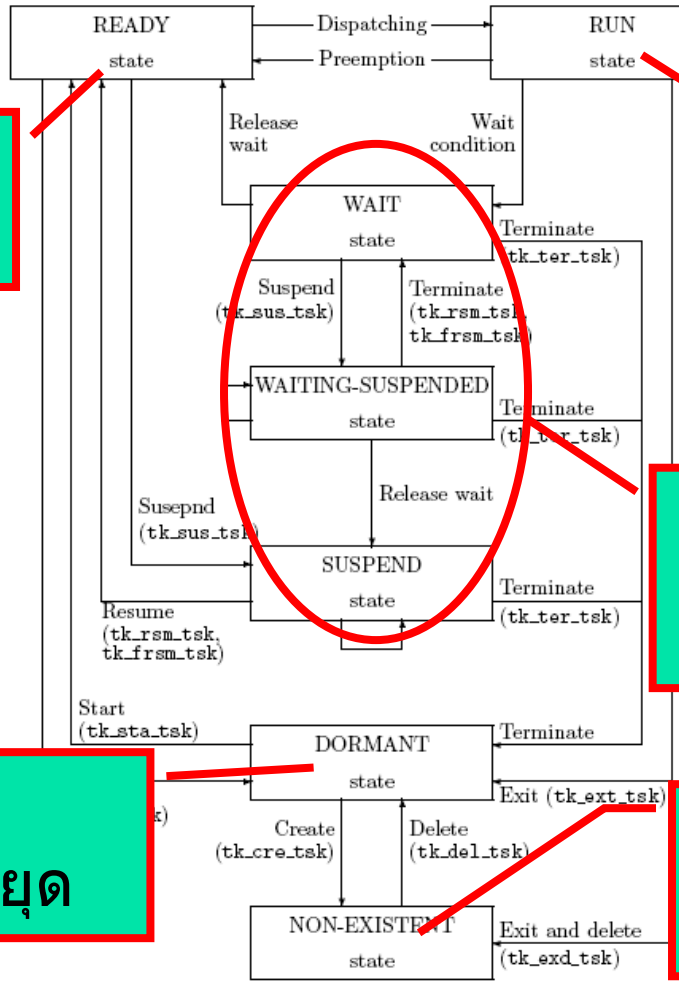
โครงสร้างซอฟต์แวร์บน T-Engine



ระบบปฏิบัติการ T-Kernel

- Pre-emptive priority-based scheduling RTOS
- ใช้หลักการของ FCFS (first-come first-serve) ในกรณีความสำคัญเท่ากัน
- ประกอบด้วย 3 ส่วนหลัก คือ T-Kernel/OS, T-Kernel/SM และ T-Kernel/DS
- มีส่วนขยายคือ T-Kernel Extension

สถานะของ task ใน T-Kernel



READY
พร้อมทำงาน

RUN
กำลังประมวลผล

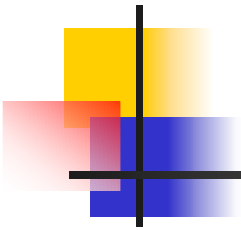
WAIT-SUSPEND
รอเงื่อนไข

DORMANT
เพิ่งถูกสร้าง/หยุด

NON-EXISTENT
ไม่อยู่ในระบบ

การทำงานแบบเวลาจริง

- การทำงานแบบขึ้นกับการตั้งเวลา
  - Delay
    tk_dly_tsk( )
  - Cyclic handler
    tk_cre_cyc( )
  - Alarm handler
    tk_cre_alm( )
- การทำงานแบบขึ้นกับ task อื่น
  - Sleep-Wakeup
    tk_wup_tsk( )
  - Semaphore
    tk_wai_sem( )
  - Mailbox / Message
    tk_snd_mbx( )
    tk_snd_mbf( )
  - Etc.



การทดสอบการทำงาน

- แบ่งเป็น 2 สถานการณ์
  - Idle ไม่มี task อื่น ทำงาน ณ ขณะนั้น
  - Load มี task ที่ทำการคำนวณอีก 100 tasks
- ใช้ 32-bit counter 2 ของ TMU ใน SH7727
  - 0.1334 μsec (ทดสอบด้วยฟังก์ชัน WaitUsec)
- ทดสอบในแง่ของ
  - ความถูกต้องของช่วงเวลา
  - ความเร็วในการตอบสนอง

ผลการทดสอบ #0

- การสอบเทียบ counter ด้วย WaitUsec (กรณี 100,000 μsec)
  - ค่าเฉลี่ยของ counter 749,515.8
  - ค่าเบี่ยงเบนมาตรฐาน 0.6
  - ค่า max ของ counter 749,520
  - ค่า min ของ counter 749,515
  - คาบเวลา 0.133419 μsec

ผลการทดสอบ #1

- การใช้ tk_dly_tsk( ) หน่วงเวลา 1 หน่วย
- กรณี idle
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    16006.61 μsec / 1.18 μsec
  - ค่า max / ค่า min
    16008.53 μsec / 15997.19 μsec
- กรณี load
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    16006.52 μsec / 3.97 μsec
  - ค่า max / ค่า min
    16017.47 μsec / 15995.19 μsec

ผลการทดสอบ #2

- การใช้ tk_cre_alm( ) เพื่อตั้งเวลา 1 หน่วย
- กรณี idle
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    15996.68 μsec / 1.39 μsec
  - ค่า max / ค่า min
    15998.79 μsec / 15987.85 μsec
- กรณี load
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    15991.97 μsec / 3.49 μsec
  - ค่า max / ค่า min
    16001.19 μsec / 15985.59 μsec

ผลการทดสอบ #3

- การใช้ tk_cre_cyc( ) เพื่อสร้างฐานเวลา 1 หน่วย
- กรณี idle
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    7992.78 μsec / 1.67 μsec
  - ค่า max / ค่า min
    7999.33 μsec / 7984.39 μsec
- กรณี load
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    7973.67 μsec / 4.14 μsec
  - ค่า max / ค่า min
    7989.59 μsec / 7958.51 μsec

ผลการทดสอบ #4

- การกลับเข้าสู่สภาวะ RUN จาก tk_wup_tsk( )
- กรณี idle
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    9.39 μsec / 0.80 μsec
  - ค่า max / ค่า min
    14.00 μsec / 9.07 μsec
- กรณี load
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    12.44 μsec / 1.13 μsec
  - ค่า max / ค่า min
    14.94 μsec / 10.67 μsec

ผลการทดสอบ #5

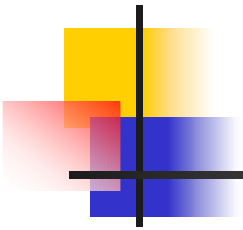
- การตอบสนองต่อการได้รับข้อมูล tk_rcv_mbf( )
- กรณี idle
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    11.09 μsec / 0.89 μsec
  - ค่า max / ค่า min
    14.40 μsec / 10.53 μsec
- กรณี load
  - ค่าเฉลี่ย/ค่าเบี่ยงเบน
    14.57 μsec / 1.28 μsec
  - ค่า max / ค่า min
    18.01 μsec / 12.67 μsec

ประสิทธิภาพแบบเวลาจริง

	เวลาเฉลี่ย	ค่าเบี่ยงเบน
tk_dly_tsk( )	16006.61 / 16006.52	1.18 / 3.97
tk_cre_alm( )	15996.68 / 15991.97	1.39 / 3.49
tk_cre_cyc( )	7992.78 / 7973.67	1.67 / 4.14
tk_wup_tsk( )	9.39 / 12.44	0.80 / 1.13
tk_rcv_mbf( )	11.09 / 14.57	0.89 / 1.28

Context Switching ของ RTOS อื่น

	เวลาเฉลี่ย	ค่าเบี่ยงเบน	ค่า max
RT Linux	8.7 / 11.2	0.5 / 4.5	33.1 / 193.9
RTEMS	2.3 / 10.4	3.7 / 2.0	16.4 / 51.3
VxWorks	3.1 / 9.5	0.3 / 3.2	19.0 / 38.8



สรุป

- ประสิทธิภาพแบบเวลาจริงของ T-Kernel เป็นที่น่าพอใจ โดยอยู่ในระดับของ μsec
- ความน่าเชื่อถือในแง่ของช่วงเวลา มีค่าน้อยกว่า 10 μsec
- ความเร็วในการตอบสนองในกรณีช้าสุด มีค่าน้อยกว่า 20 μsec
- การอยู่ภายใต้ภาระโหลดไม่ทำให้ค่าประสิทธิภาพแบบเวลาจริงแย่ลงมากนัก