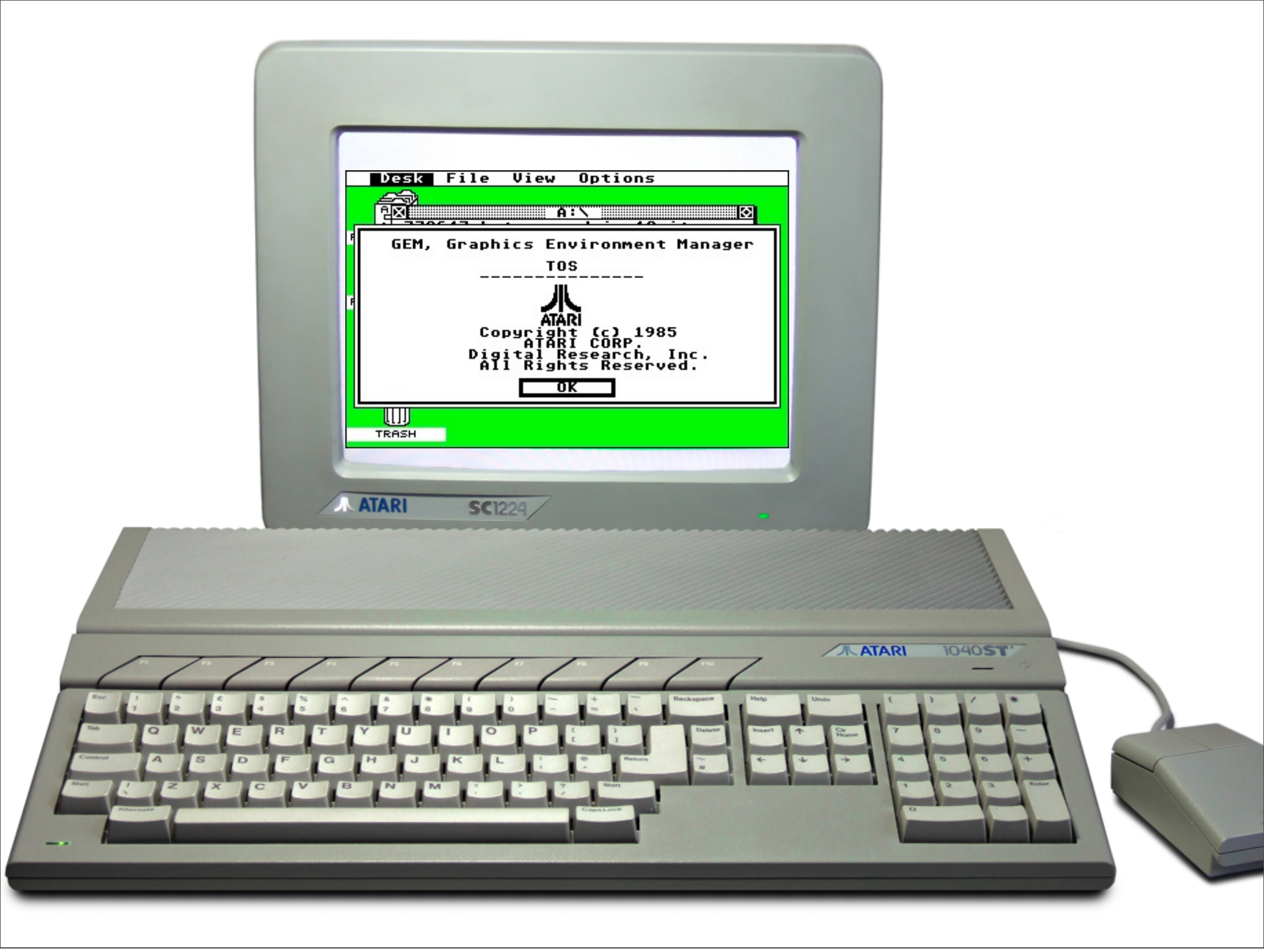
QEMU's recompilation engine

About Me

- Alexander Graf
- Freelance developer for SUSE and Freescale
- KVM and Qemu developer
- Server class PowerPC KVM port
- S390x Qemu guest support
- x86 Mac OS X in KVM
- Nested SVM
- ...

What is Emulation


Desk File View Options
A:\
GEM, Graphics Environment Manager
TOS
ATARI
Copyright (c) 1985
ATARI CORP.
Digital Research, Inc.
All Rights Reserved.
OK
TRASH
ATARI SC1224
ATARI 1040ST

Disks
Documentation
Home
Preferences
Trash
Terminal
Welcome-DE.html
Welcome-ES.html
Welcome-NL.html
Welcome.html

Hatari v1.4.0

Desk File View Options

A:\

GEM, Graphics Environment Manager

TOS

ATARI

Copyright (c) 1985
ATARI CORP.
Digital Research, Inc.
All Rights Reserved.

OK

TRASH

A: B: HD: FS: 0 8MHz/1MB ST, EmuTOS REC:

```
We alrea
We alrea
We alrea
Support
Building
Trying t
Connecte
Start pl
AAC init
CD Digit
CD Audio
FFMpeg i
OSS devi
OSS: Una
OSS Audi
RAW init
Screen V
Media Se
Ogg Vorb
HDA: nVIDIA HD-Audio (MCP51) pcm
nVIDIA HD-Audio (MCP51) pcm initialized
Plugin scan finished
Device opened
NVRAM not found at '/home/root/.hatari/hatari.nvram'
```

13:44:22

Donnerstag, 15. August 13


```
File   Options
LOGON LINUX026
NIC 0700 is created; devices 0700-0702 defined
NIC 0800 is created; devices 0800-0802 defined
z/VM Version 5 Release 4.0, Service Level 0801 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 04:29:18 EST FRIDAY 06/04/10
z/VM V5.4.0    2009-03-04 14:19
i 150
DMSACP723I A (191) R/O
DMSWSP108W Shared S-STAT not available
PROFILE EXEC: CALLING LINUX026 EXEC TO INIT THE MACHINE...
CTCA 0600 DEFINED
CTCA 0601 DEFINED
NIC 0500 is created; devices 0500-0502 defined
CTCA 0600 COUPLED TO ROUTER01 0635
CTCA 0601 COUPLED TO ROUTER01 0634
HCPNDF2787E NIC 0500 not connected; LAN SYSTEM Z44LAN01 does not exist
HCPNIC2781E NIC 0700 not created; network devices 0700-0702 could not be defined
HCPNDF2787E NIC 0700 not connected; LAN SYSTEM Z44LAN02 does not exist
5000-5004 ATTACHED TO LINUX026
Ready; T=0.01/0.01 04:30:05
zIPL v1.8.0 interactive boot menu

 0. default (LinuxV1)

 1. LinuxV1
 2. Linux
 3. ipl
 4. Failsafe

Note: VM users please use '#cp vi vmsg <number> <kernel-parameters>'

Please choose (default will boot in 10 seconds):
Booting default (LinuxV1)...
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 2.6.27.19-5-default (geeko@buildhost) (gcc version 4.3.2 [gcc-4_3-
branch revision 141291] (SUSE Linux) ) #1 SMP 2009-02-28 04:40:21 +0100
setup.1a06a7: Linux is running as a z/VM guest operating system in 64-bit mode
                                                          MORE...   ZVMV5R40
                                                                    042/001
```


File
Options
LOGON LINUX02b
NIC 0700 is created; devices 0700-0702 defined
NIC 0800 is created; devices 0800-0802 defined
z/VM Version 5 Release 4.0, Service Level 0801 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES: NO RDR, NO PRT, NO PUN
LOGON AT 04:24:18 EST FRIDAY 06/04/10
z/VM V5.4.0 2009-03-04 14:19
IPL 150
DMSWGP723I A (191) R/O
DMSQSP100W Shared S-STAT not available
PROFILE EXEC: CALLING LINUX02b EXEC TO INIT THE MACHINE...
CTCA 0600 DEFINED
CTCA 0601 DEFINED
NIC 0500 is created; devices 0500-0502 defined
CTCA 0600 COUPLED TO ROUTER01 0634
CTCA 0601 COUPLED TO ROUTER01 0634
HCPNDF2787E NIC 0500 not connected; LAN SYSTEM 244LAN01 does not exist
HCPNIC2781E NIC 0700 not created; network devices 0700-0702 could not be defined
HCPNDF2787E NIC 0700 not connected; LAN SYSTEM 244LAN02 does not exist
5600-5604 ATTACHED TO LINUX02b
Ready; T=0.01/0.01 04:30:05
zIPL v1.8.0 interactive boot menu
0. default (LinuxV1)
1. LinuxV1
2. Linux
3. ipl
4. Failsafe
Note: VM users please use '#cp vi vmsg <number> <kernel-parameters>'
Please choose (default will boot in 10 seconds):
Booting default (LinuxV1)...
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 2.6.27.19-5-default (geeko@buildhost) (gcc version 4.3.2 [gcc-4_3-branch revision 141291] (SUSE Linux)) #1 SMP 2009-02-28 04:40:21 +0100
setup.1a06a7: Linux is running as a z/VM guest operating system in 64-bit mode
MORE... ZVMV5R40
042/001
iMac


x05
000000000
493
Wii


x05
000000000
493
Wii


Dolphin
File Emulation Options Tools View Help
Browse
Pause
Stop
FullScr
ScrShot
Config
Graphics
DSP
GCPad
Wiimote
3x05
00000000
493
Wii

CPU Instructions

CPU Instructions

Register	Value
r0	0
r1	0
...	0

CPU Instructions

Register	Value
r0	0
r1	0
...	...

```
code:
38 20 00 01  li       r1,1
7c 20 0b 78  mr       r0,r1
38 21 00 01  addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	0
...	...

```
code:
38 20 00 01  li       r1,1
7c 20 0b 78  mr       r0,r1
38 21 00 01  addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	0
...	...

```
code:
38 20 00 01  li       r1,1
7c 20 0b 78  mr       r0,r1
38 21 00 01  addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	0
...	...

```
code:
38 20 00 01   li        r1,1
7c 20 0b 78   mr        r0,r1
38 21 00 01   addi      r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	1
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	1
...	...

```
code:
38 20 00 01  li       r1,1
7c 20 0b 78  mr       r0,r1
38 21 00 01  addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	1
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	1
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	1
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	0
r1	1
...	...

```
code:
38 20 00 01   li       r1,1
7c 20 0b 78   mr       r0,r1
38 21 00 01   addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	1
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	1
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	1
...	...

```
code:
38 20 00 01   li       r1,1
7c 20 0b 78   mr       r0,r1
38 21 00 01   addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	1
...	...

1

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	1
...	...

1 + 1 =

```
code:
38 20 00 01   li       r1,1
7c 20 0b 78   mr       r0,r1
38 21 00 01   addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	1
...	...

1 + 1 = 2

```
code:
38 20 00 01   li       r1,1
7c 20 0b 78   mr       r0,r1
38 21 00 01   addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li       r1,1
7c 20 0b 78   mr       r0,r1
38 21 00 01   addi     r1,r1,1
```

CPU Instructions

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01  li       r1,1
7c 20 0b 78  mr       r0,r1
38 21 00 01  addi     r1,r1,1
```

Different ISAs

Register	Value
r0	1
rl	2
...	...

```
code:
38 20 00 01  li      r1,1
7c 20 0b 78  mr      r0,r1
38 21 00 01  addi    r1,r1,1
```

Different ISAs

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li      r1,1
7c 20 0b 78   mr      r0,r1
38 21 00 01   addi    r1,r1,1
```

```
code:
bb 01 00 00 00  mov $0x1,%ebx
89 d8           mov %ebx,%eax
83 c3 01        add $0x1,%ebx
```

Register	Value
eax	0
ebx	0
...	...

Different ISAs

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li      r1,1
7c 20 0b 78   mr      r0,r1
38 21 00 01   addi    r1,r1,1
```

```
code:
bb 01 00 00 00  mov $0x1,%ebx
89 d8           mov %ebx,%eax
83 c3 01        add $0x1,%ebx
```

Register	Value
eax	0
ebx	0
...	...

Different ISAs

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li      r1,1
7c 20 0b 78   mr      r0,r1
38 21 00 01   addi    r1,r1,1
```

```
code:
bb 01 00 00 00  mov $0x1,%ebx
89 d8           mov %ebx,%eax
83 c3 01        add $0x1,%ebx
```

Register	Value
eax	0
ebx	1
...	...

Different ISAs

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li      r1,1
7c 20 0b 78   mr      r0,r1
38 21 00 01   addi    r1,r1,1
```

```
code:
bb 01 00 00 00   mov $0x1,%ebx
89 d8            mov %ebx,%eax
83 c3 01         add $0x1,%ebx
```

Register	Value
eax	1
ebx	1
...	...

Different ISAs

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li      r1,1
7c 20 0b 78   mr      r0,r1
38 21 00 01   addi    r1,r1,1
```

```
code:
bb 01 00 00 00  mov $0x1,%ebx
89 d8           mov %ebx,%eax
83 c3 01        add $0x1,%ebx
```

Register	Value
eax	1
ebx	2
...	...

Different ISAs

Register	Value
r0	1
r1	2
...	...

```
code:
38 20 00 01   li      r1,1
7c 20 0b 78   mr      r0,r1
38 21 00 01   addi    r1,r1,1
```

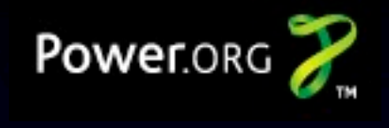
```
code:
bb 01 00 00 00  mov $0x1,%ebx
89 d8           mov %ebx,%eax
83 c3 01        add $0x1,%ebx
```

Register	Value
eax	1
ebx	2
...	...

Converting code

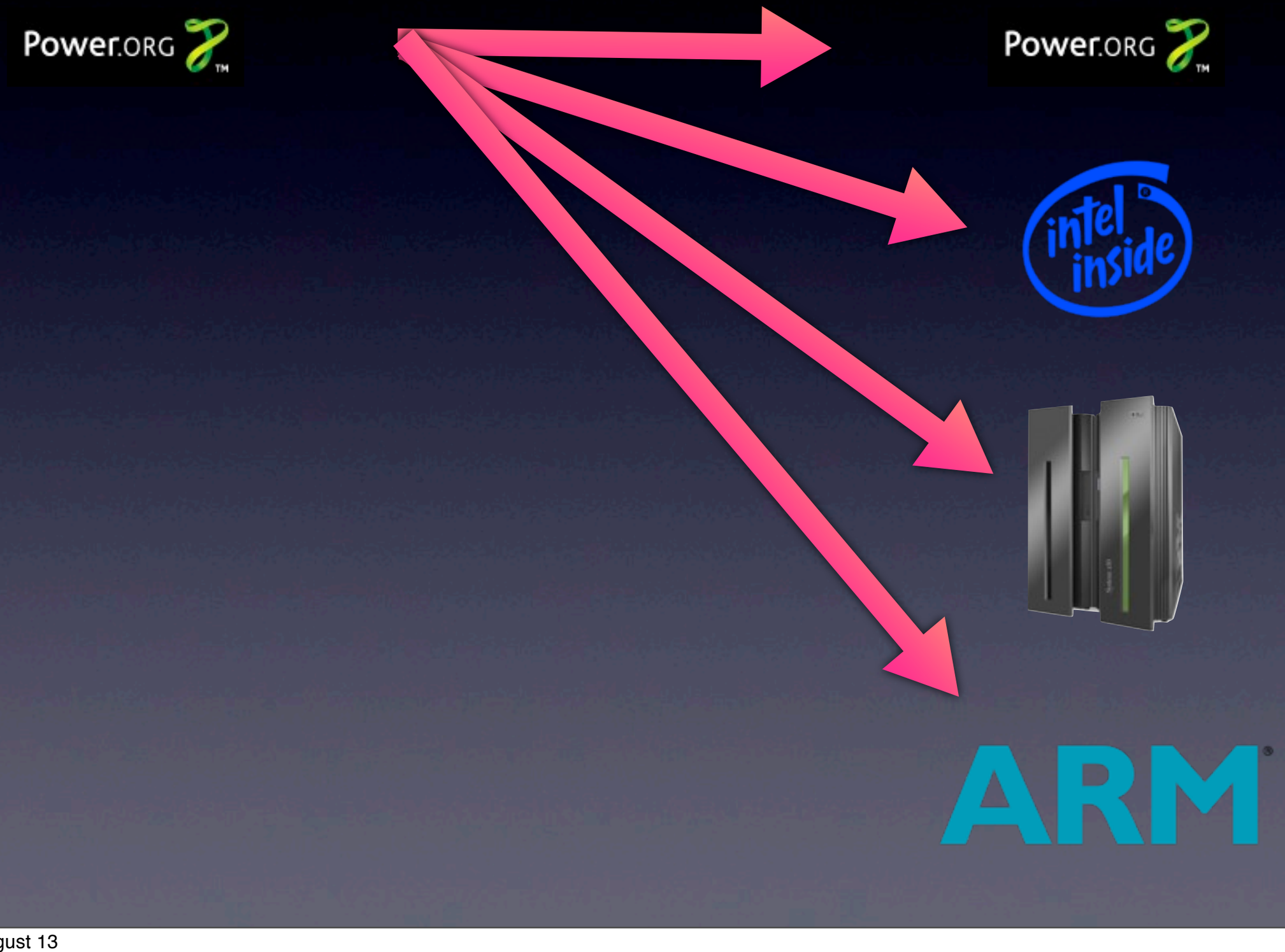
Converting code

Converting code

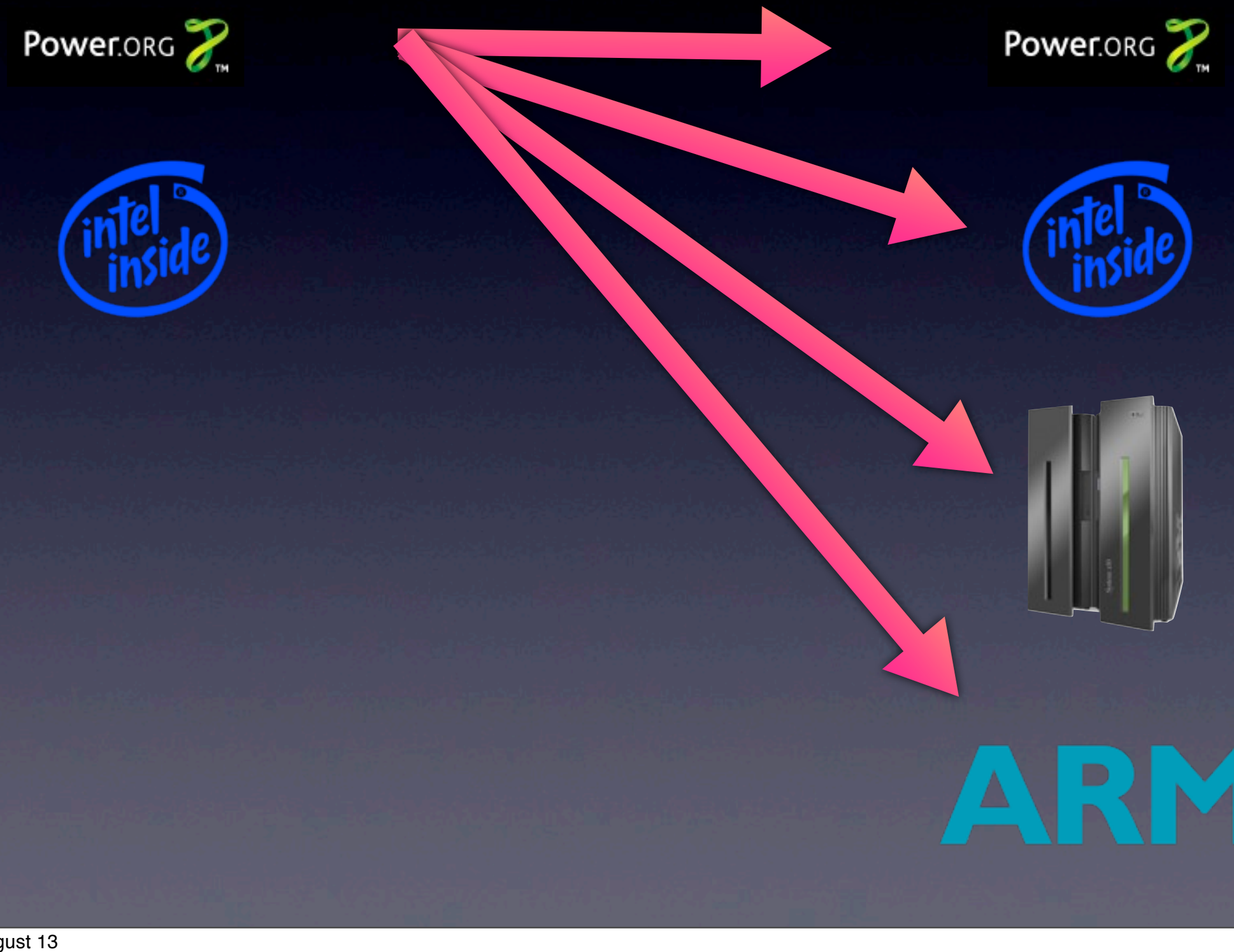


ARM

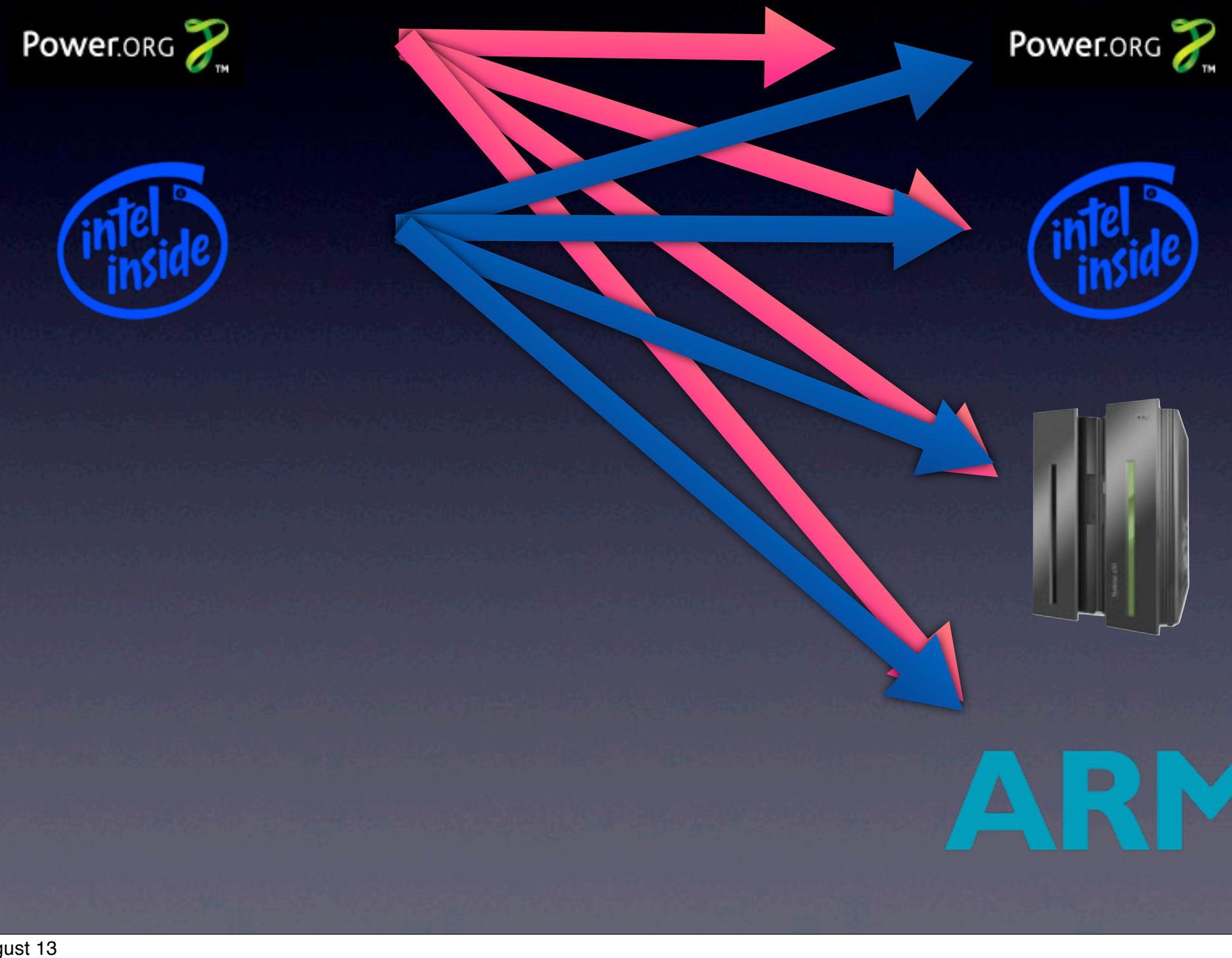
Converting code



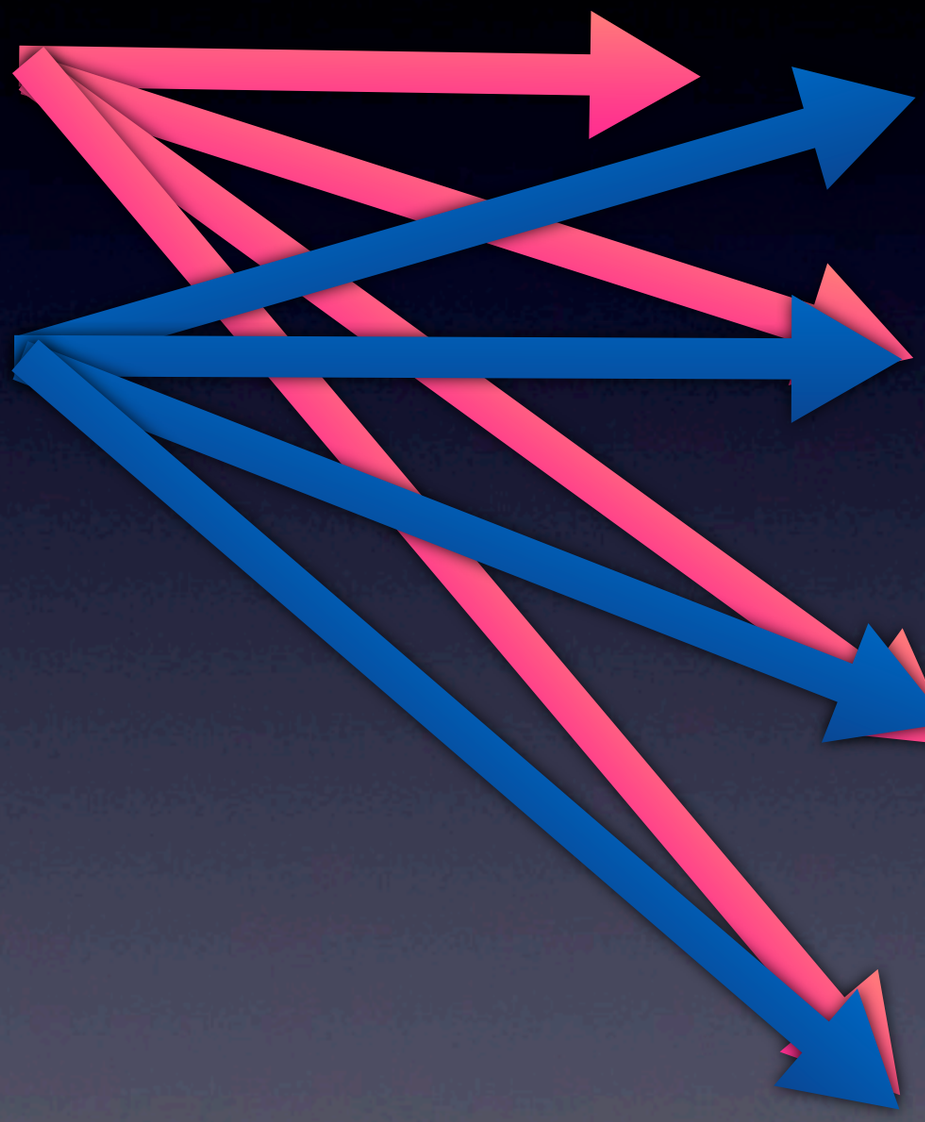
Converting code



Converting code

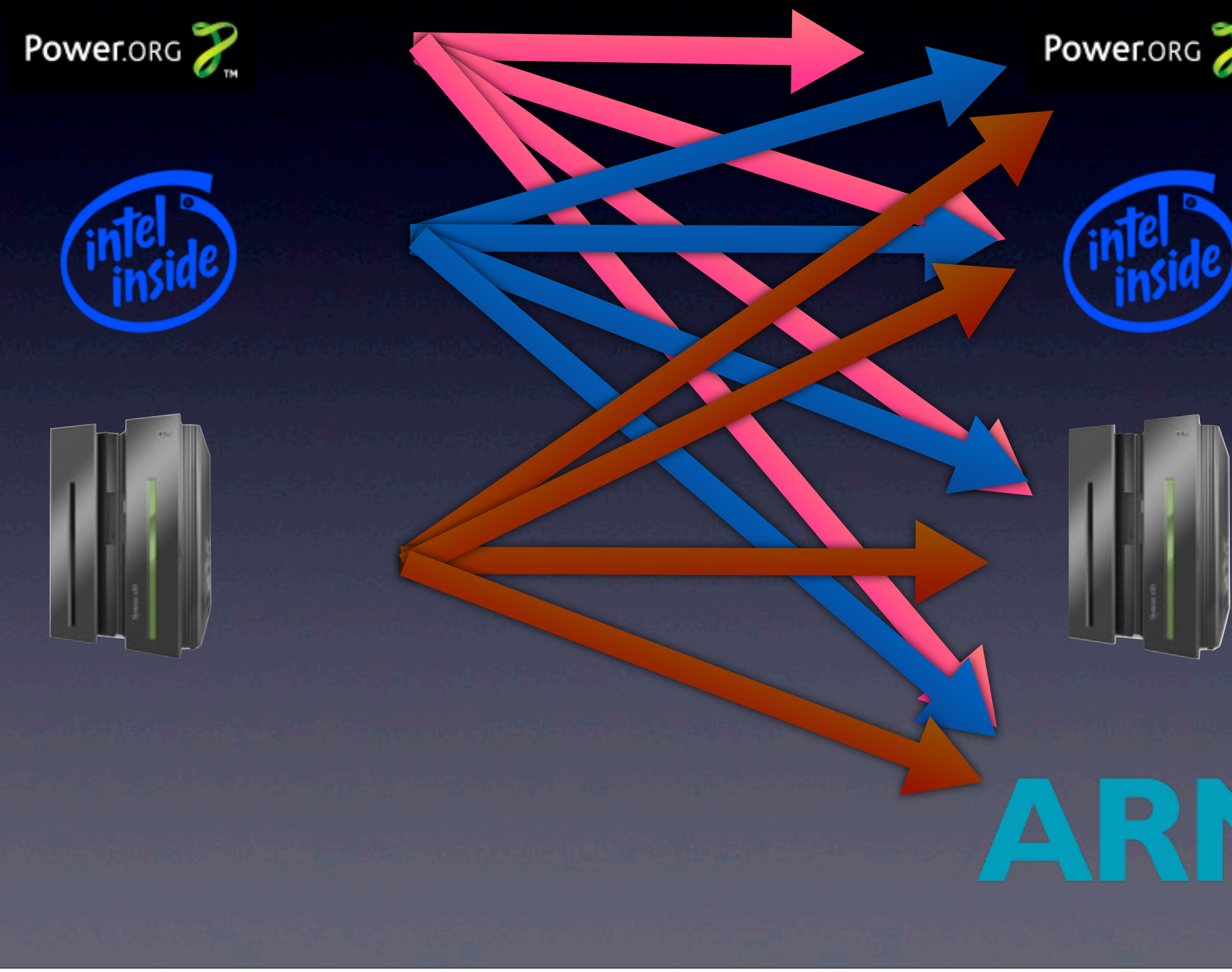


Converting code

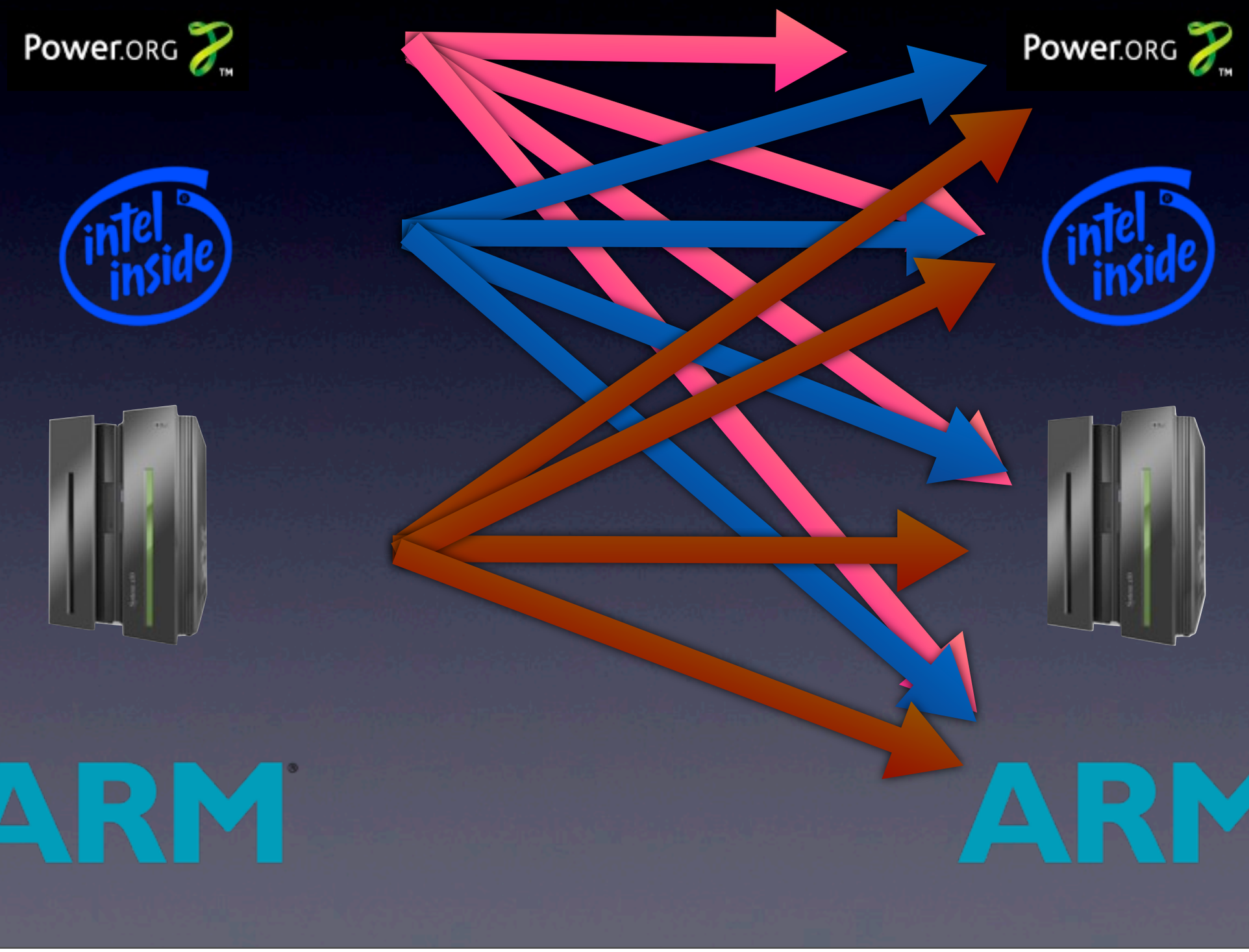


ARM

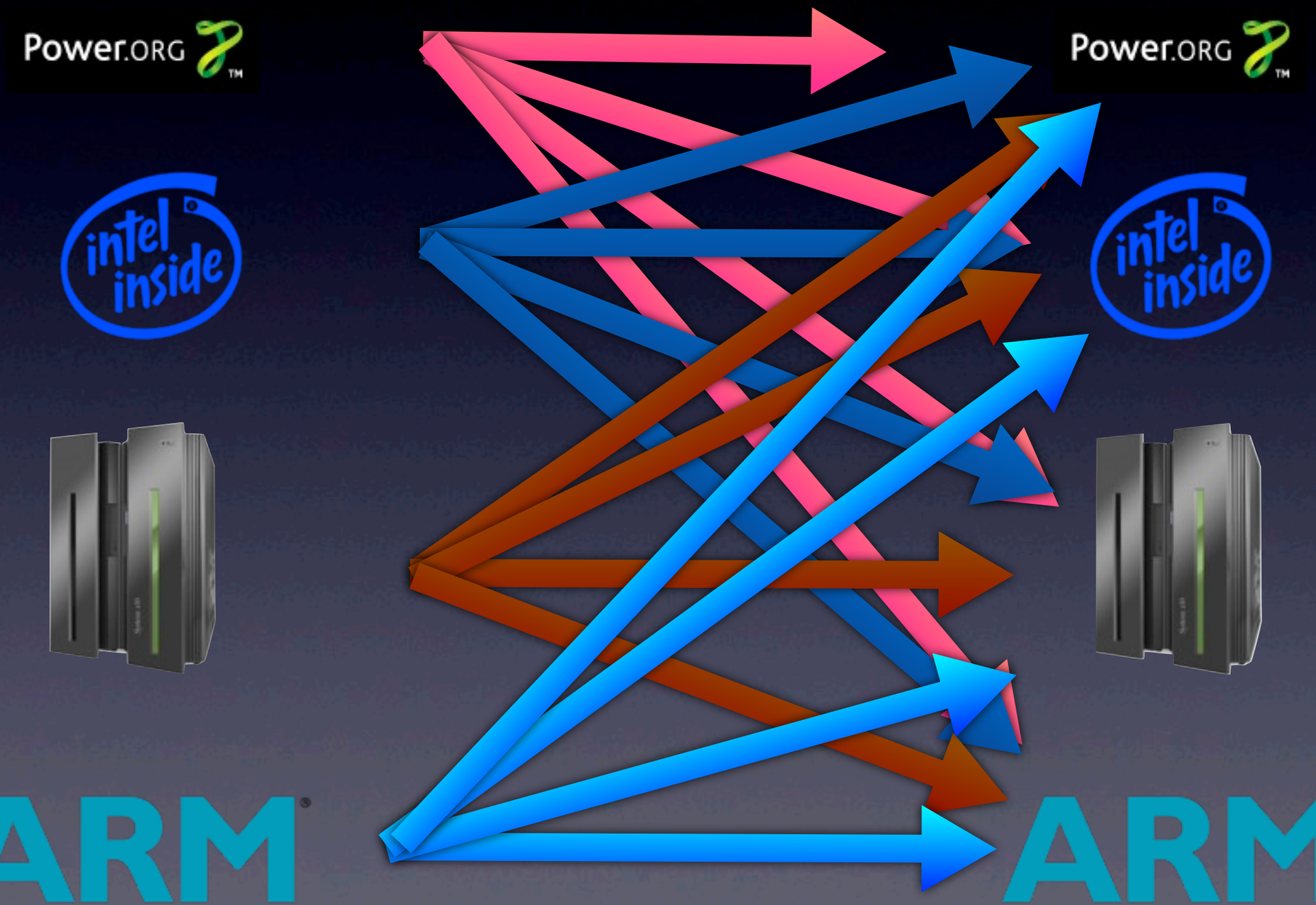
Converting code



Converting code



Converting code



TCG

Power.ORG

Power.ORG

intel inside

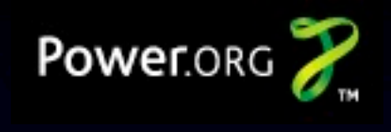
intel inside

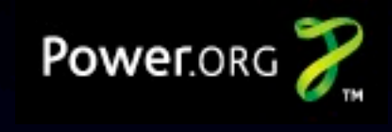
TCG

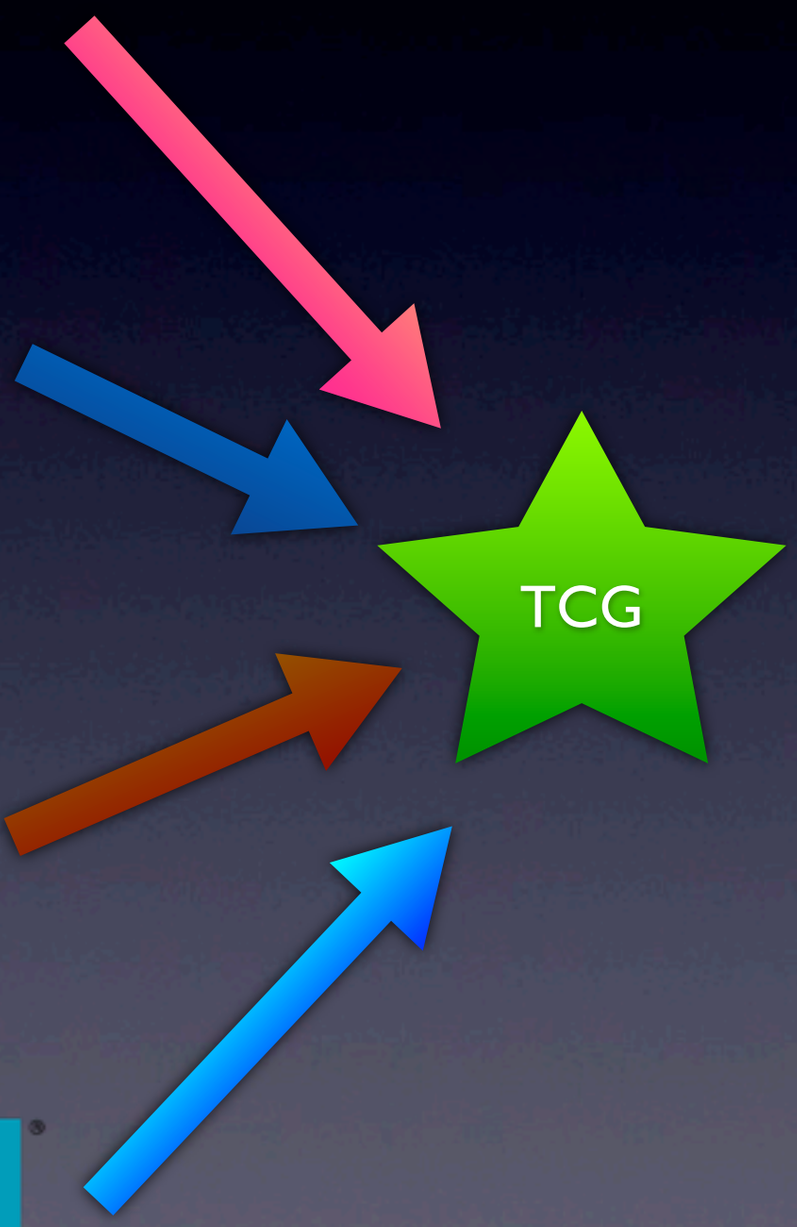
ARM

ARM

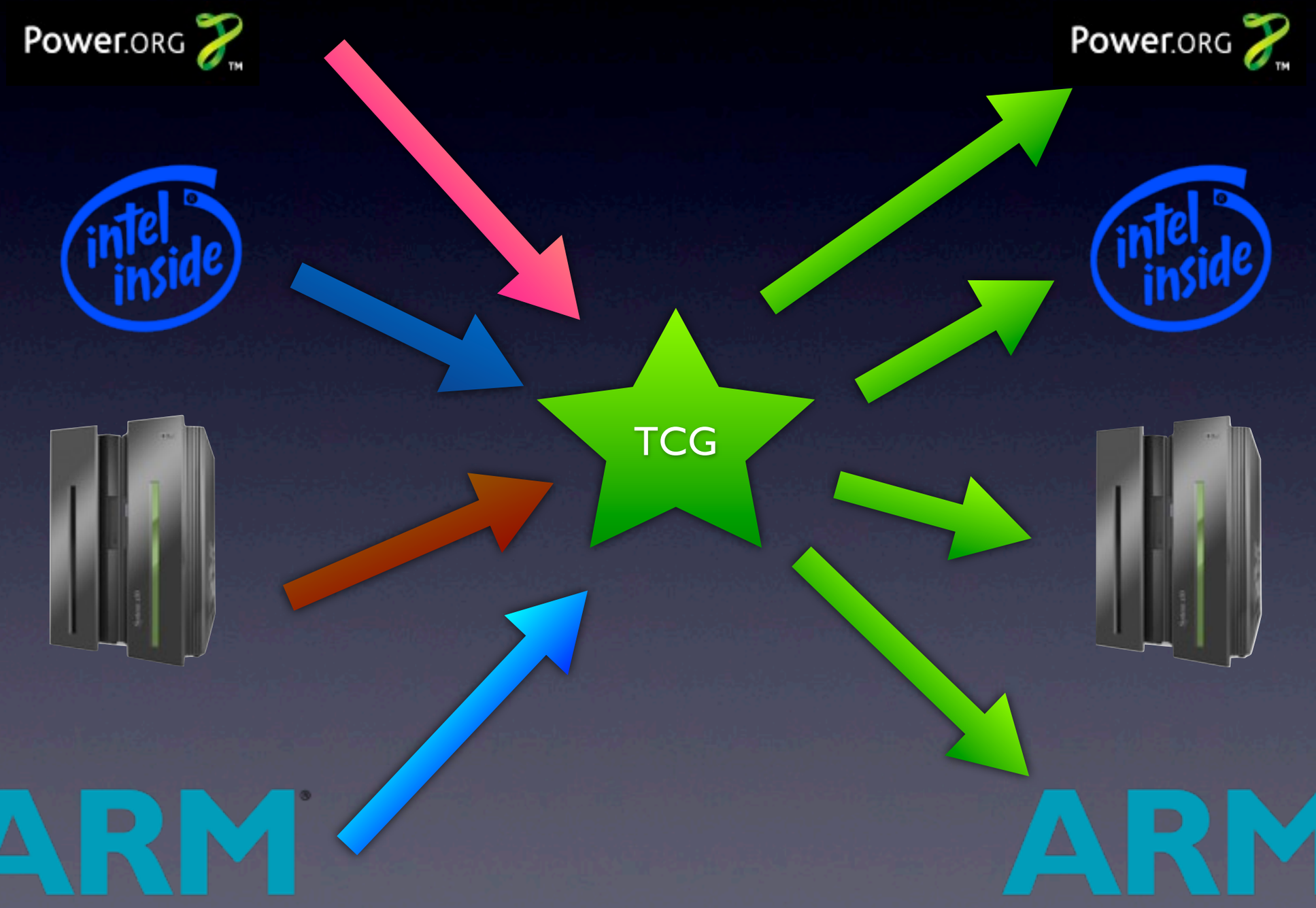
TCG







TCG



TCG micro-ops

```
38 21 00 01  addi    r1,r1,1
```

TCG micro-ops

```
38 21 00 01  addi    r1,r1,1
```

TCG micro-ops

```
38 21 00 01  addi    r1,r1,1
```

```
tcg_gen_addi_tl(cpu_gpr[rD(ctx->opcode)], cpu_gpr[rA(ctx->opcode)], simm);
```

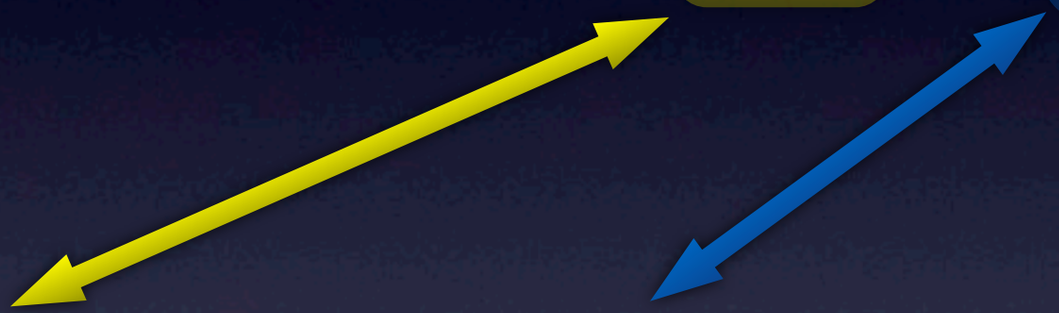
TCG micro-ops

```
38 21 00 01  addi    r1,r1,1
```

```
tcg_gen_addi_tl(cpu_gpr[rD(ctx->opcode)], cpu_gpr[rA(ctx->opcode)], simm);
```

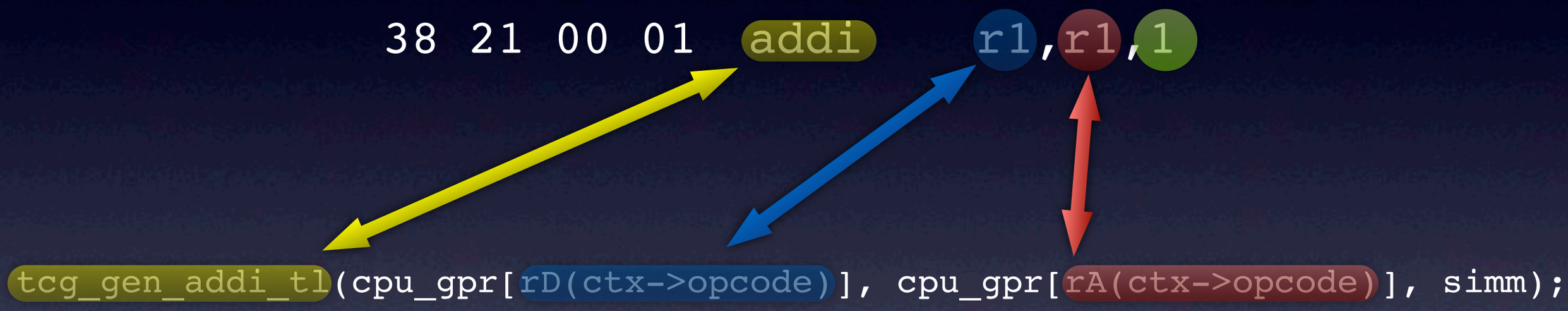
TCG micro-ops

```
38 21 00 01   addi     r1,r1,1
```

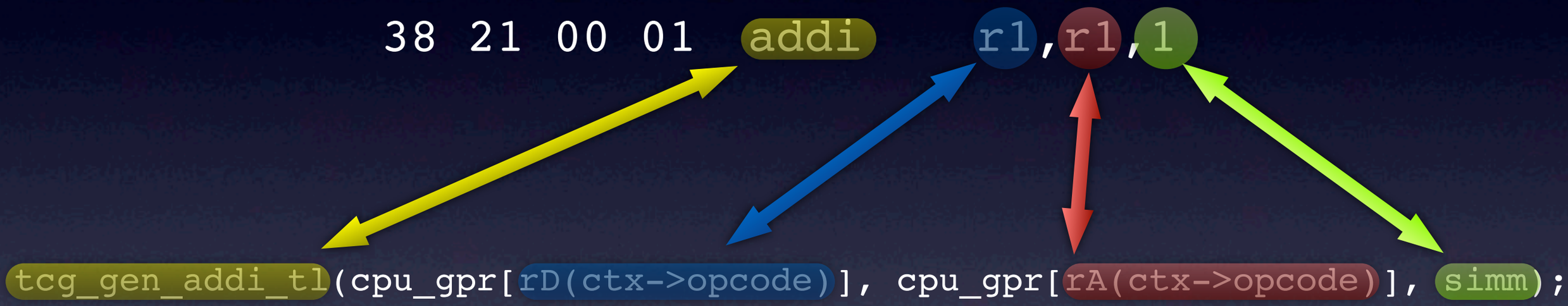


```
tcg_gen_addi_tl(cpu_gpr[rD(ctx->opcode)], cpu_gpr[rA(ctx->opcode)], simm);
```

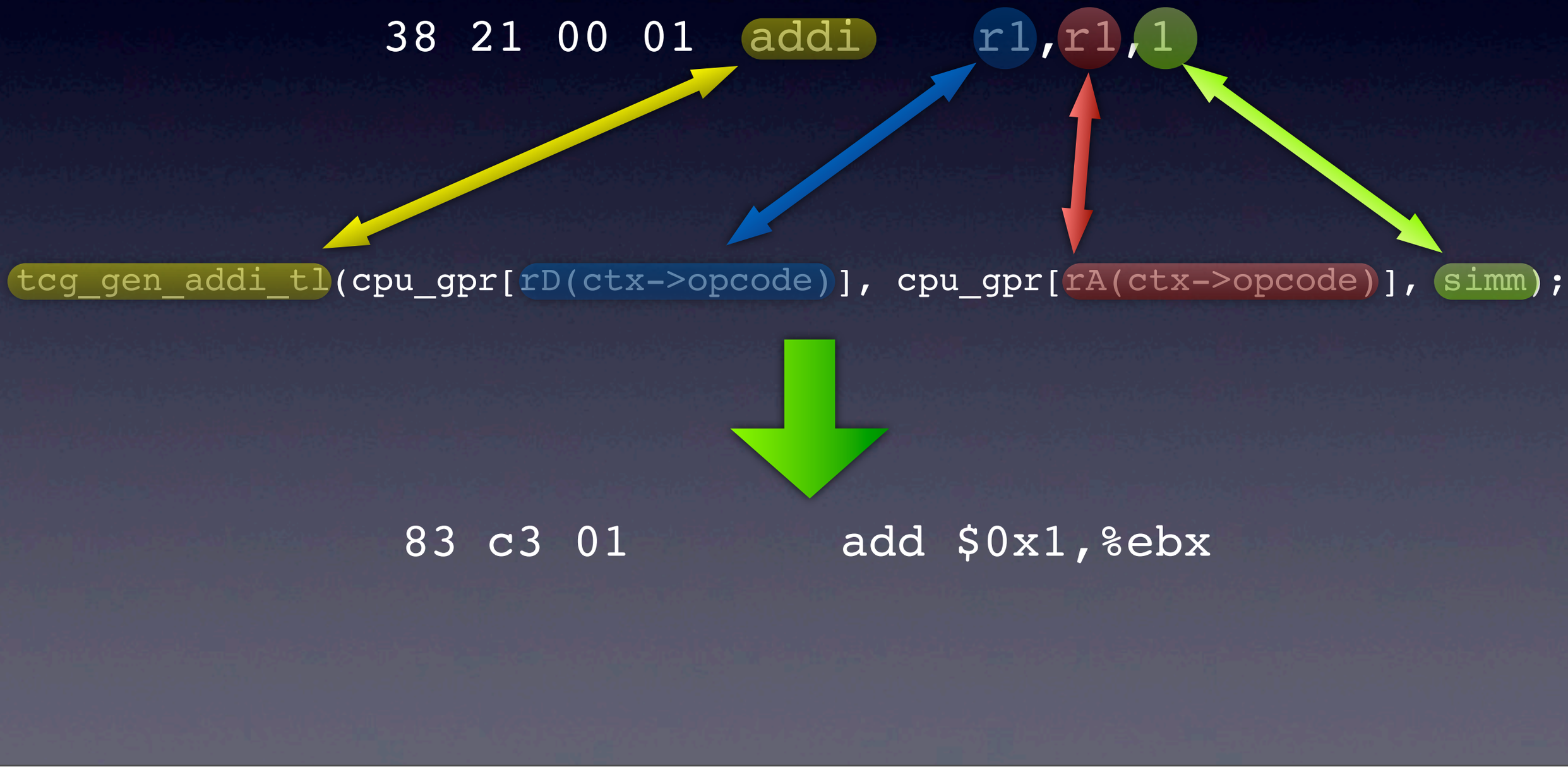
TCG micro-ops



TCG micro-ops



TCG micro-ops



TCG micro-ops

```
7c 22 18 96  mulhw  r1,r2,r3
```

TCG micro-ops

```
7c 22 18 96  mulhw  r1,r2,r3
```

TCG micro-ops

```
7c 22 18 96  mulhw  r1,r2,r3
```

```
t0 = tcg_temp_new_i64();
t1 = tcg_temp_new_i64();
tcg_gen_ext32s_tl(t0, cpu_gpr[rA(ctx->opcode)]);
tcg_gen_ext32s_tl(t1, cpu_gpr[rB(ctx->opcode)]);
tcg_gen_mul_i64(t0, t0, t1);
tcg_gen_shri_i64(cpu_gpr[rD(ctx->opcode)], t0, 32);
tcg_temp_free_i64(t0);
tcg_temp_free_i64(t1);
```

TCG micro-ops

```
7c 22 18 96  mulhw  r1,r2,r3
```

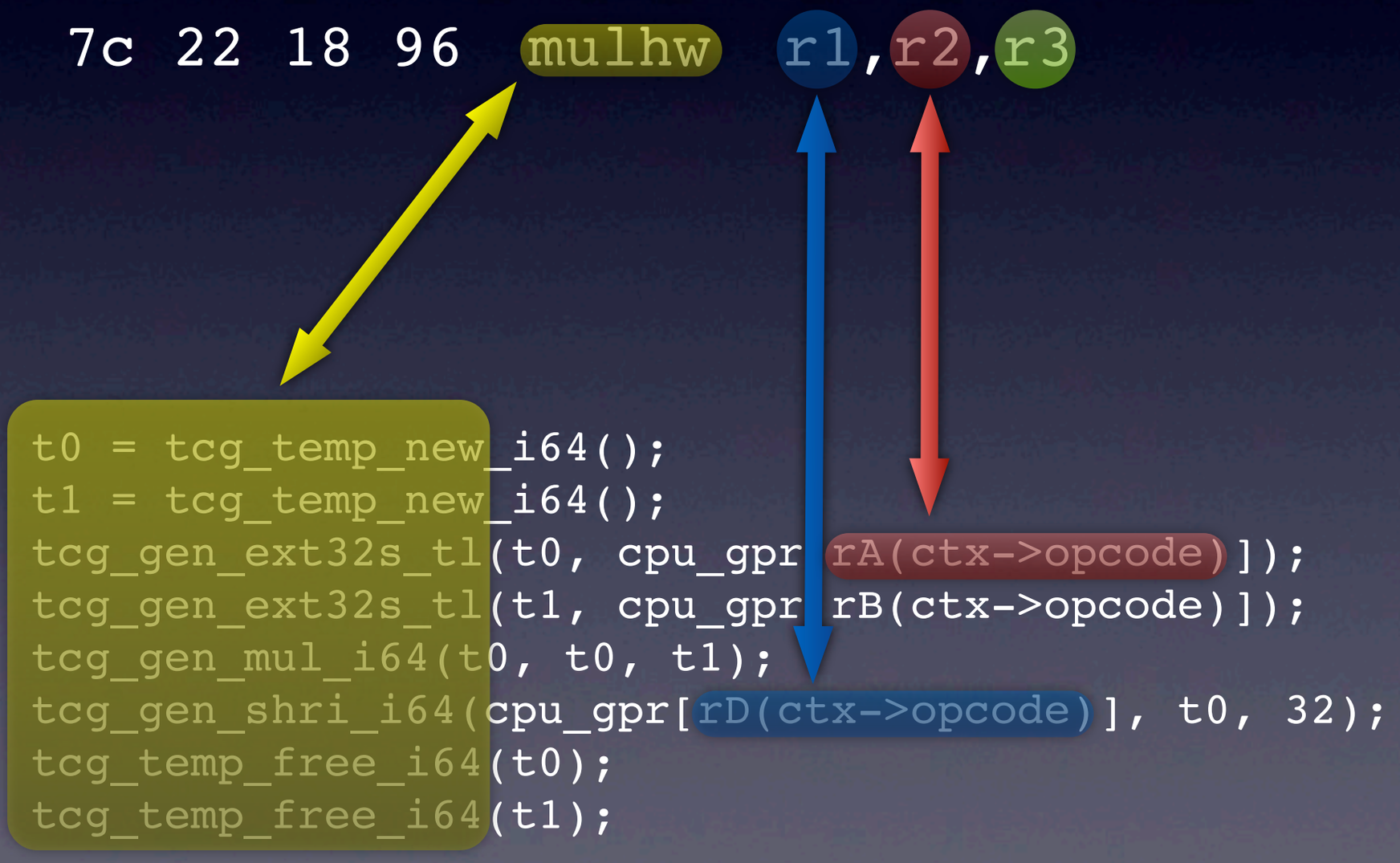
```
t0 = tcg_temp_new_i64();
t1 = tcg_temp_new_i64();
tcg_gen_ext32s_tl(t0, cpu_gpr[rA(ctx->opcode)]);
tcg_gen_ext32s_tl(t1, cpu_gpr[rB(ctx->opcode)]);
tcg_gen_mul_i64(t0, t0, t1);
tcg_gen_shri_i64(cpu_gpr[rD(ctx->opcode)], t0, 32);
tcg_temp_free_i64(t0);
tcg_temp_free_i64(t1);
```

TCG micro-ops

```
7c 22 18 96  mulhw  r1,r2,r3
```

```
t0 = tcg_temp_new_i64();
t1 = tcg_temp_new_i64();
tcg_gen_ext32s_tl(t0, cpu_gpr[rA(ctx->opcode)]);
tcg_gen_ext32s_tl(t1, cpu_gpr[rB(ctx->opcode)]);
tcg_gen_mul_i64(t0, t0, t1);
tcg_gen_shri_i64(cpu_gpr[rD(ctx->opcode)], t0, 32);
tcg_temp_free_i64(t0);
tcg_temp_free_i64(t1);
```

TCG micro-ops



TCG micro-ops

```
7c 22 18 96  mulhw  r1,r2,r3
```

```
t0 = tcg_temp_new_i64();
t1 = tcg_temp_new_i64();
tcg_gen_ext32s_tl(t0, cpu_gpr[rA(ctx->opcode)]);
tcg_gen_ext32s_tl(t1, cpu_gpr[rB(ctx->opcode)]);
tcg_gen_mul_i64(t0, t0, t1);
tcg_gen_shri_i64(cpu_gpr[rD(ctx->opcode)], t0, 32);
tcg_temp_free_i64(t0);
tcg_temp_free_i64(t1);
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

TCG helper

```
7c 41 00 34   cntlzw   r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

```
e8 e3 df 00 00  callq  helper_cntlzw
```

TCG helper

```
7c 41 00 34  cntlzw  r1,r2
```

```
gen_helper_cntlzw(cpu_gpr[rA(ctx->opcode)], cpu_gpr[rS(ctx->opcode)]);
```

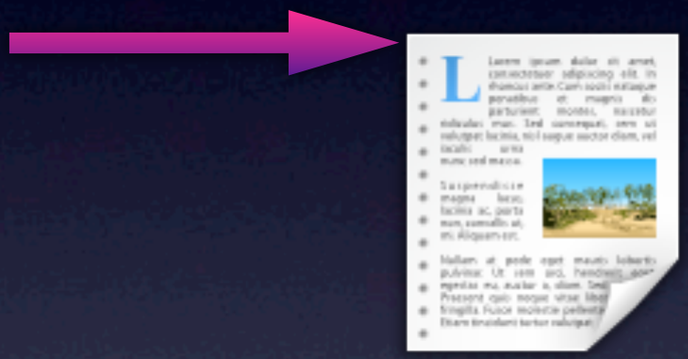
```
e8 e3 df 00 00  callq  helper_cntlzw
```

```
target_ulong helper_cntlzw (target_ulong t)
{
    return __builtin_clz(t);
}
```

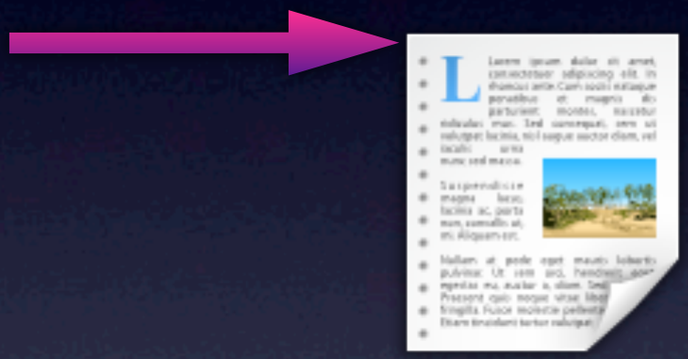
Executing Code

Translation Blocks

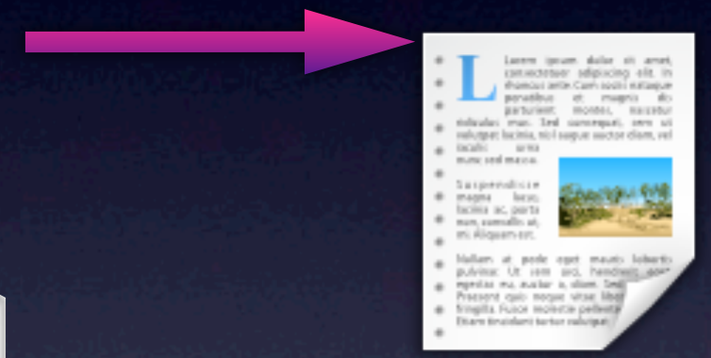
Translation Blocks



Translation Blocks



Translation Blocks



Translation Blocks

```
0:38 20 00 01  li      r1,1
4:7c 20 0b 78  mr      r0,r1
8:38 21 00 01  addi    r1,r1,1
```

Translation Blocks

```
 0:38 20 00 01  li      r1,1
 4:7c 20 0b 78  mr      r0,r1
 8:38 21 00 01  addi    r1,r1,1
 c:2c 01 00 02  cmpwi   r1,2
10:41 82 ff f0  beq+    0x0
```

Translation Blocks

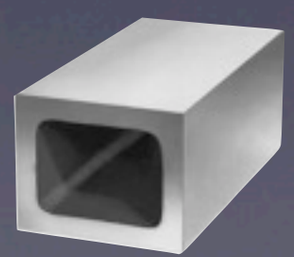
```
 0:38 20 00 01   li      r1,1
 4:7c 20 0b 78   mr      r0,r1
 8:38 21 00 01   addi    r1,r1,1
 c:2c 01 00 02   cmpwi   r1,2
10:41 82 ff f0   beq+    0x0
```

Translation Blocks

```
 0:38 20 00 01   li      r1,1
 4:7c 20 0b 78   mr      r0,r1
 8:38 21 00 01   addi    r1,r1,1
 c:2c 01 00 02   cmpwi   r1,2
10:41 82 ff f0   beq+    0x0
```

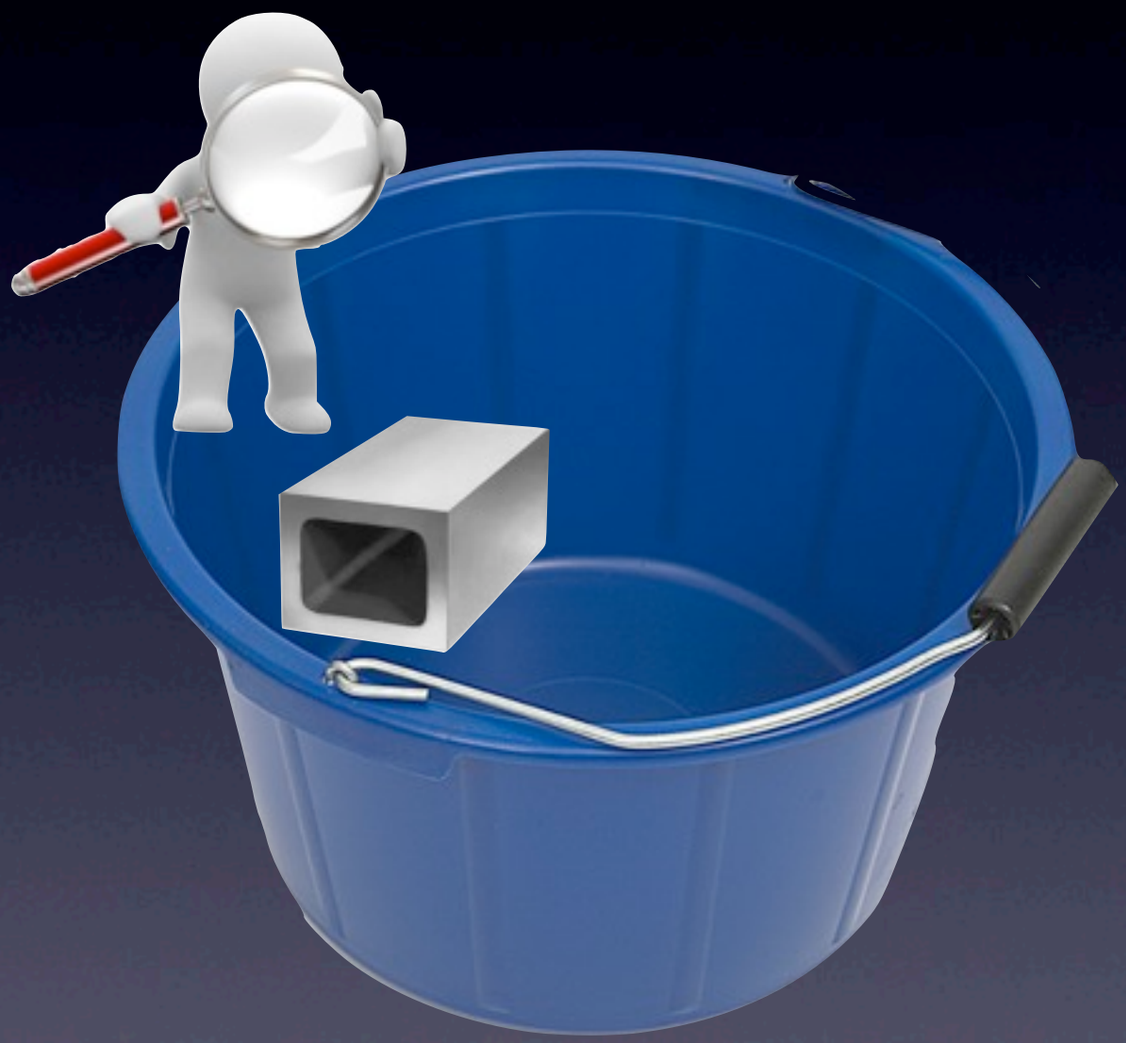
Translation Blocks

```
 0:38 20 00 01   li      r1,1
 4:7c 20 0b 78   mr      r0,r1
 8:38 21 00 01   addi    r1,r1,1
 c:2c 01 00 02   cmpwi   r1,2
10:41 82 ff f0   beq+    0x0
```



Translation Blocks

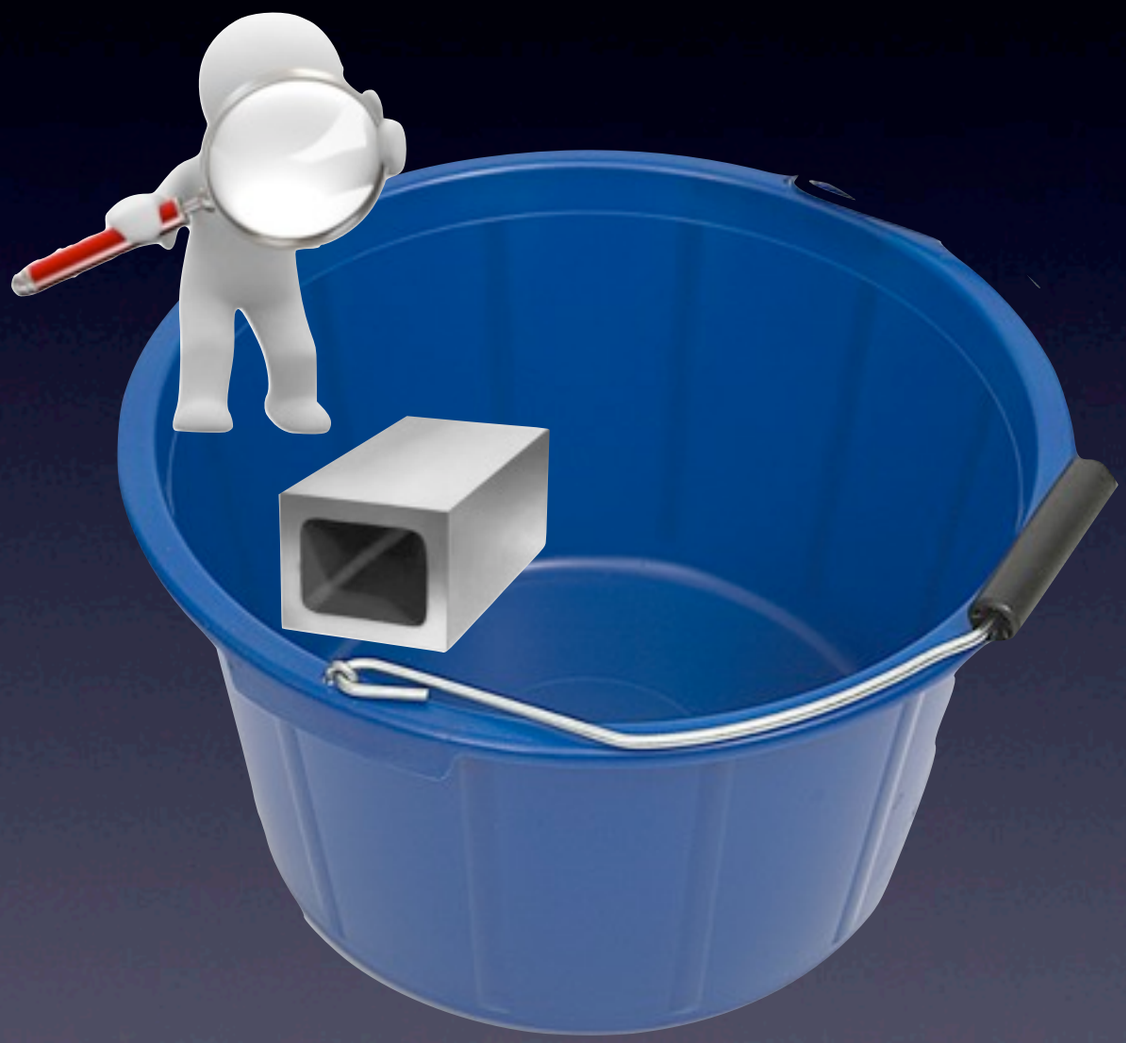
Translation Blocks



Translation Blocks

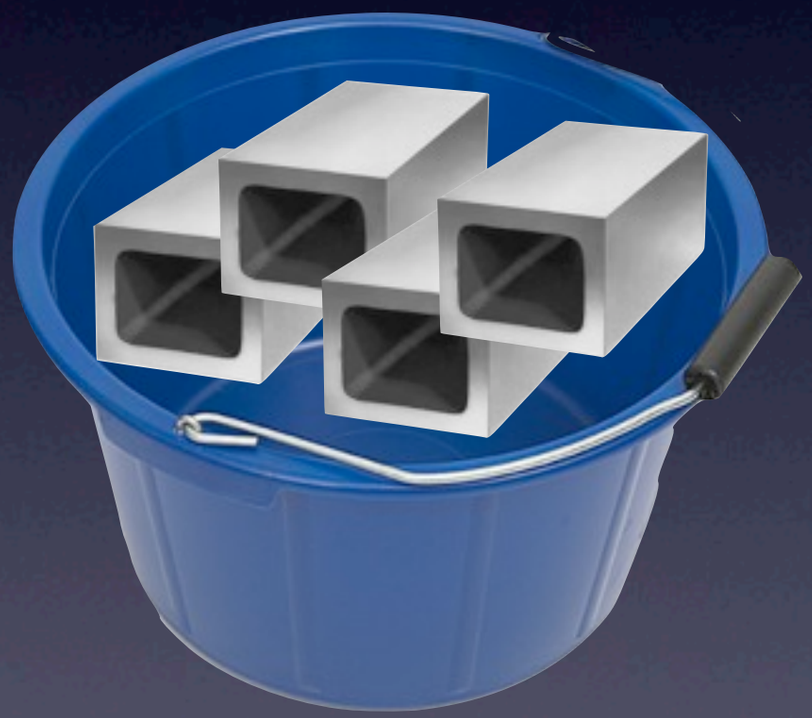
Translation Blocks

Translation Blocks

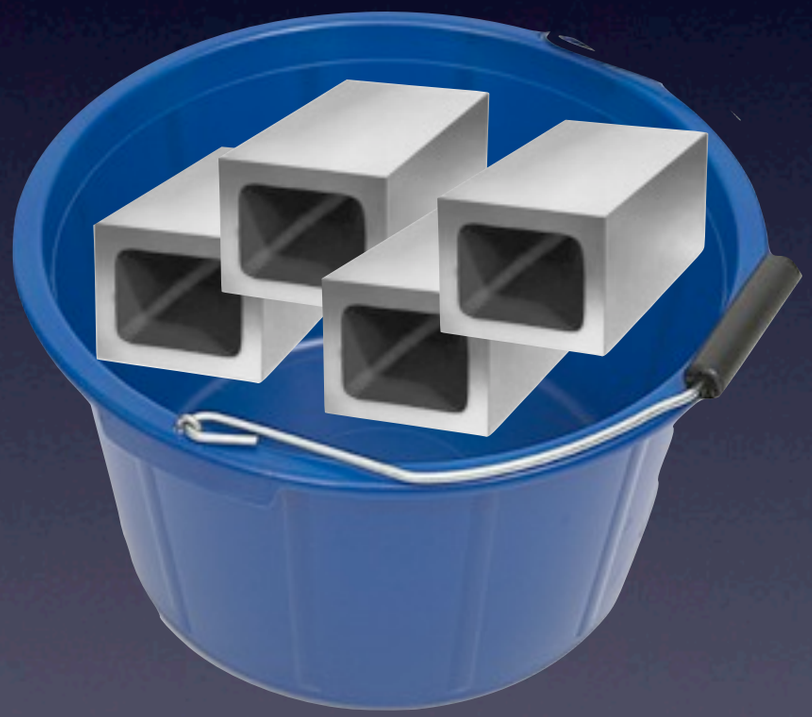


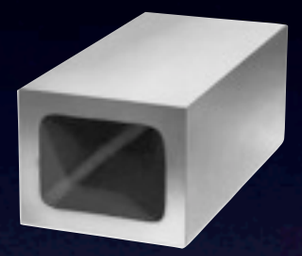
Translation Blocks

TB Chaining

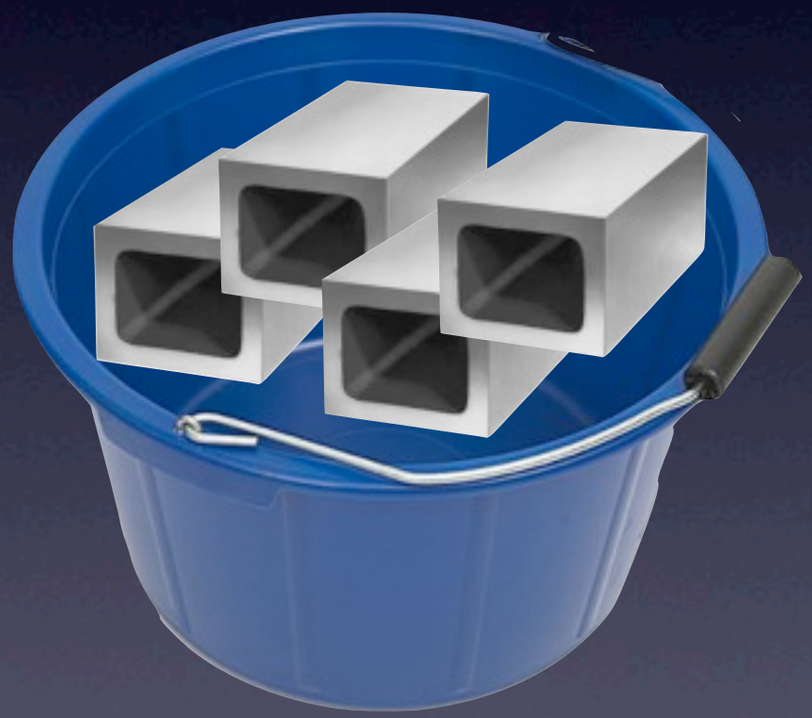


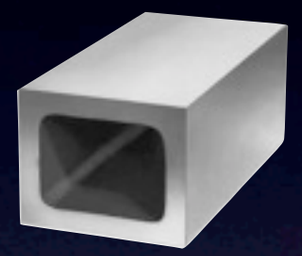
TB Chaining

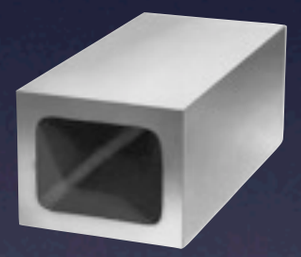




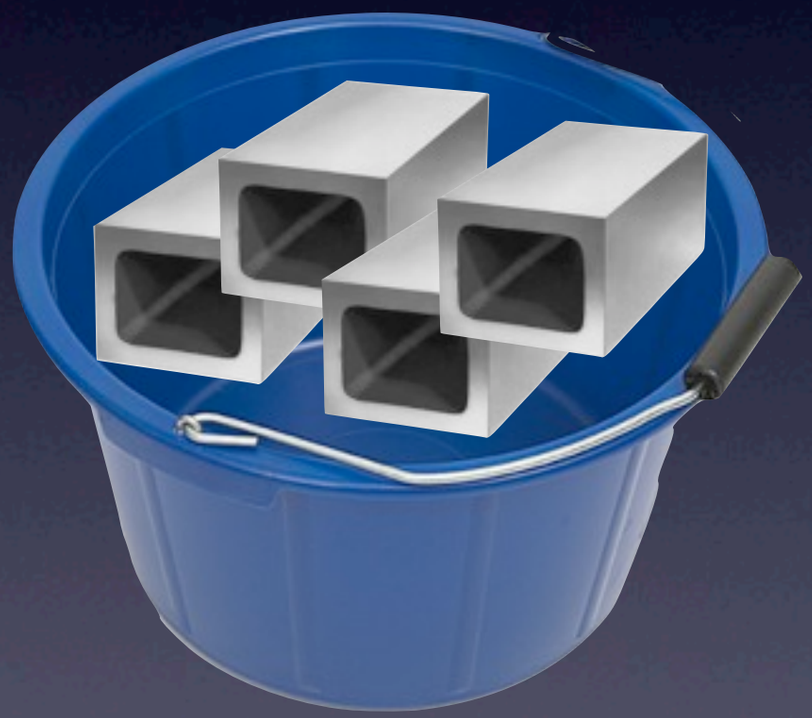
TB Chaining

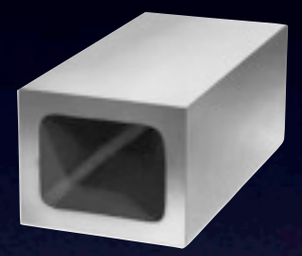


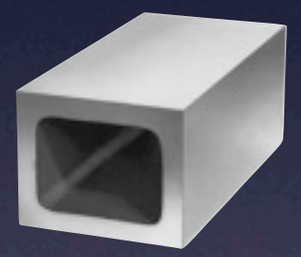


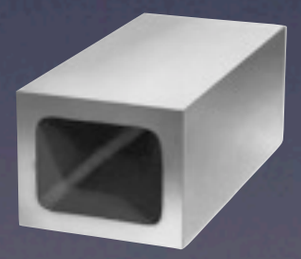


TB Chaining

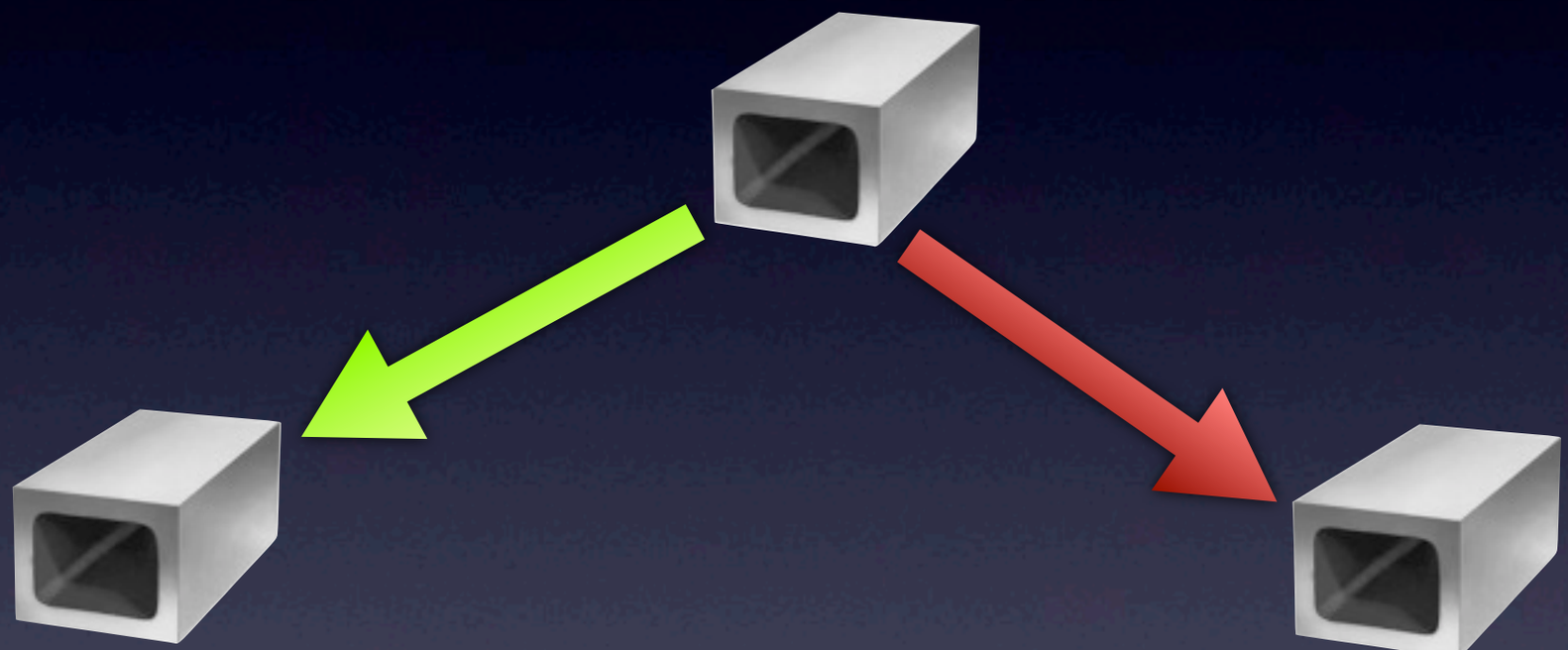




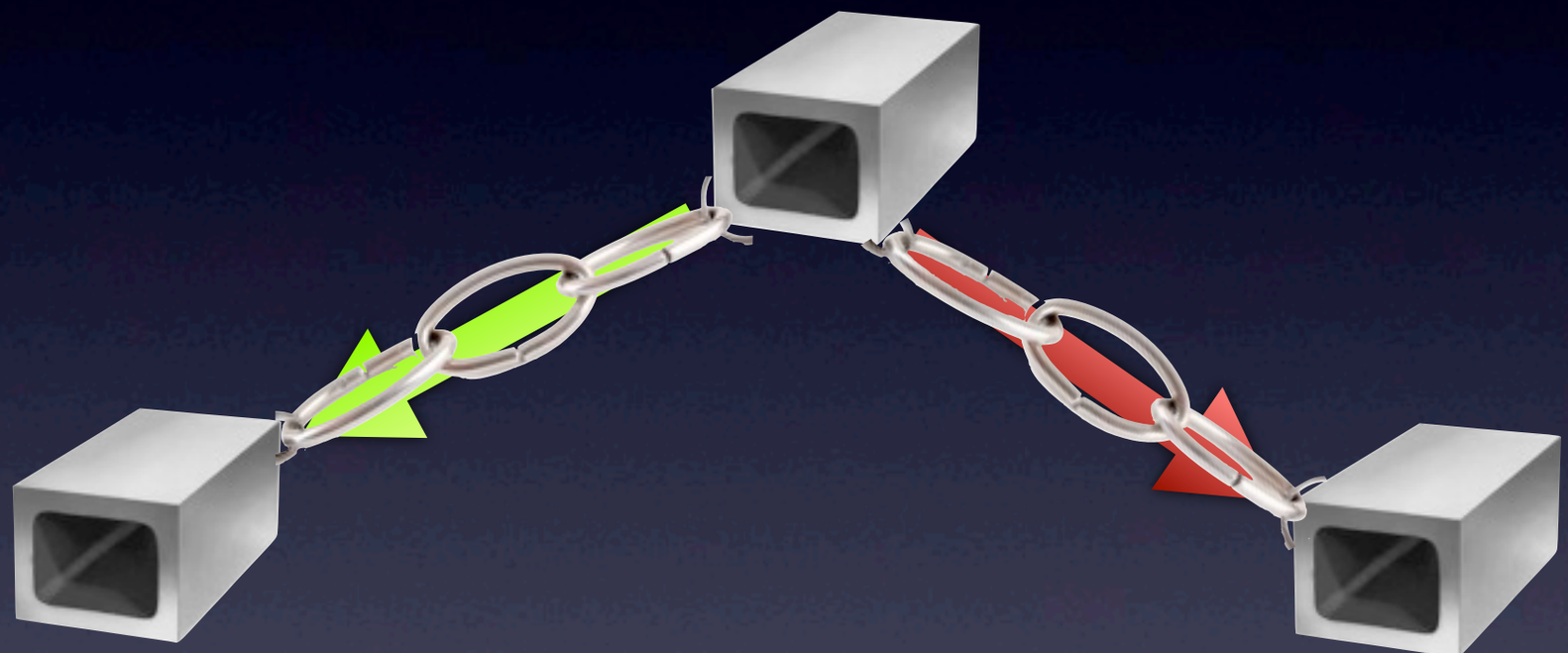




TB Chaining

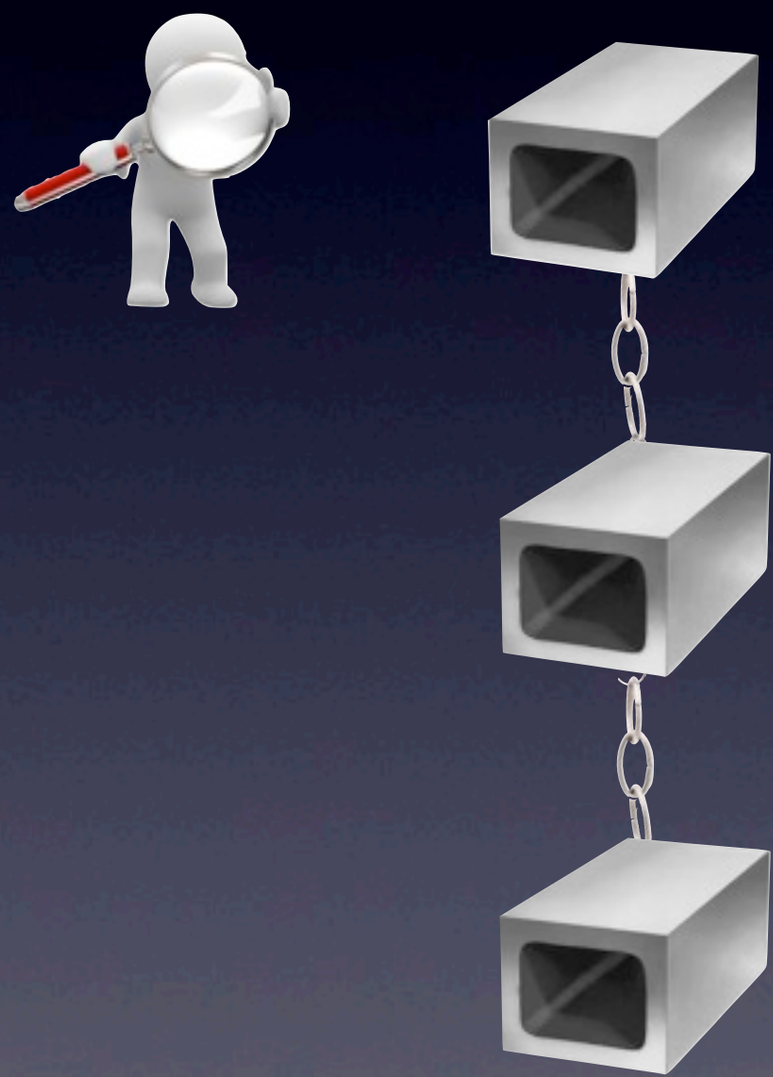


TB Chaining



TB Chaining

TB Chaining



TB optimization

```
38 20 00 01   li        r1,1
38 21 00 02   addi      r1,r1,2
38 20 00 09   li        r1,9
```

TB optimization

```
38 20 00 01   li      r1,1
38 21 00 02   addi    r1,r1,2
38 20 00 09   li      r1,9
```

TB optimization

```
38 20 00 01   li      r1,1
38 21 00 02   addi    r1,r1,2
38 20 00 09   li      r1,9
```

TB optimization

```
38 20 00 01   li      r1,1
38 21 00 02   addi    r1,r1,2
38 20 00 09   li      r1,9
```

TB optimization

```
38 20 00 01  li      r1,1
38 21 00 02  addi    r1,r1,2
38 20 00 09  li      r1,9
```

TB optimization

```
38 20 00 01  li      r1,1
```

```
C HELPER: r1=output, no input
```

```
38 20 00 09  li      r1,9
```

TB optimization

```
38 20 00 01  li       r1,1
```

```
C HELPER: r1=output, no input

38 20 00 09  li       r1,9
```

TB optimization

```
38 20 00 01  li        r1,1
```

PURE

```
C HELPER: r1=output, no input

38 20 00 09  li        r1,9
```

TB optimization

```
38 20 00 01  li       r1,1
```

PURE ~~C HELPER. r1-output, no input~~

```
38 20 00 09  li       r1,9
```

TB optimization

```
38 20 00 01  li       r1,1
```

```
C HELPER: r1=output, no input

38 20 00 09  li       r1,9
```

TB optimization

```
38 20 00 01  li        r1,1
```

CONST C HELPER: r1=output, no input

```
38 20 00 09  li        r1,9
```

TB optimization

```
38 20 00 01  li        r1,1
```

CONST

```
C HELPER: r1=output, no input

38 20 00 09  li        r1,9
```

TB optimization

```
38 20 00 01  li       r1,1
```

CONST

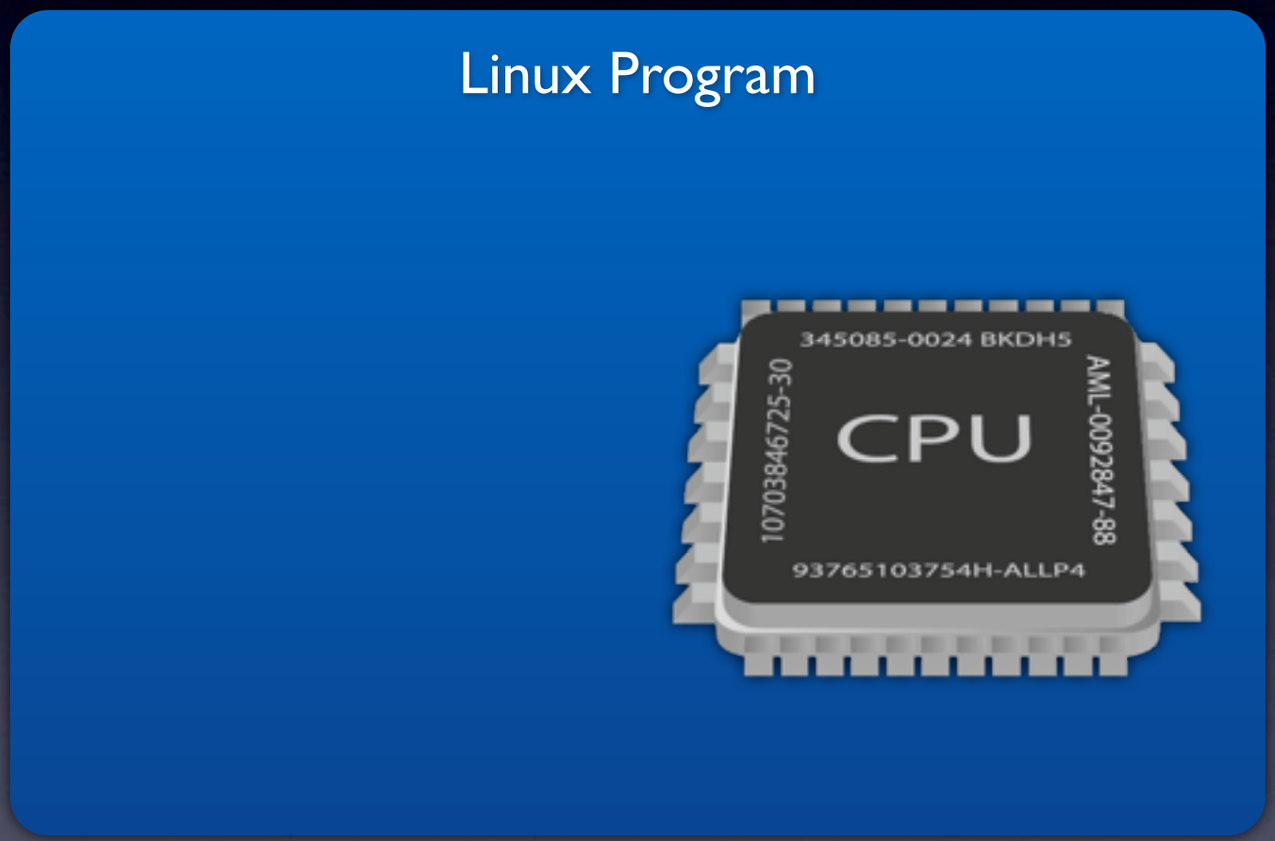
```
C HELPER: r1=output, no input

38 20 00 09  li       r1,9
```

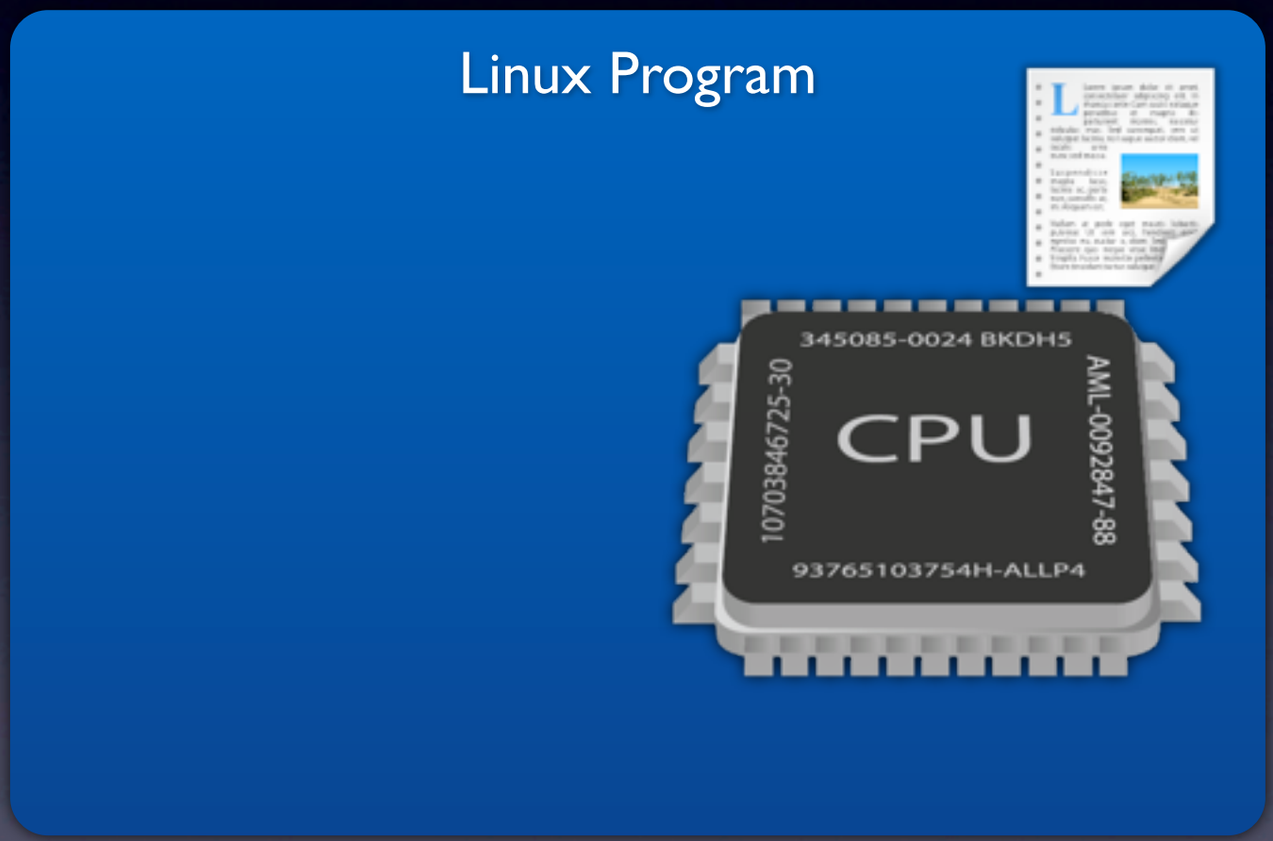
Donnerstag, 15. August 13

Donnerstag, 15. August 13

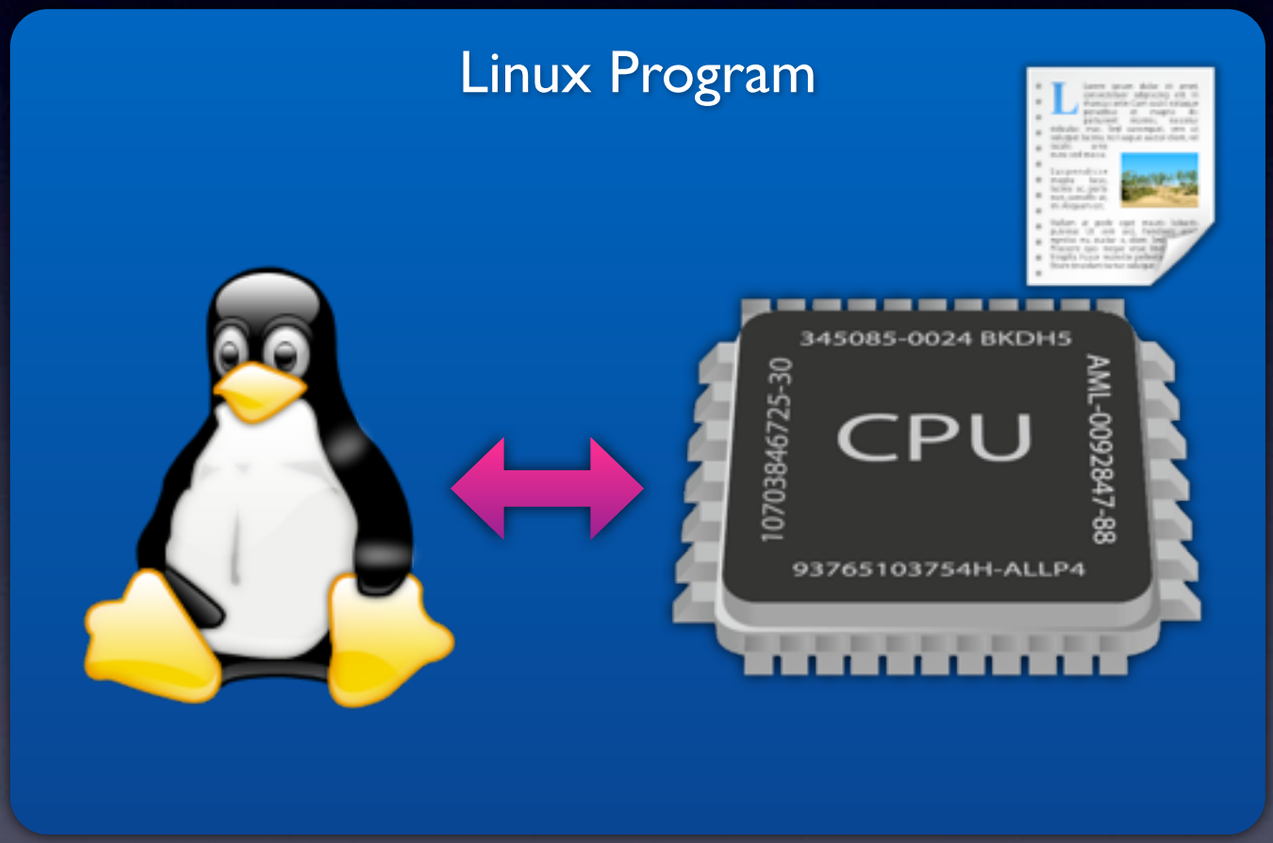
Linux-user emulation



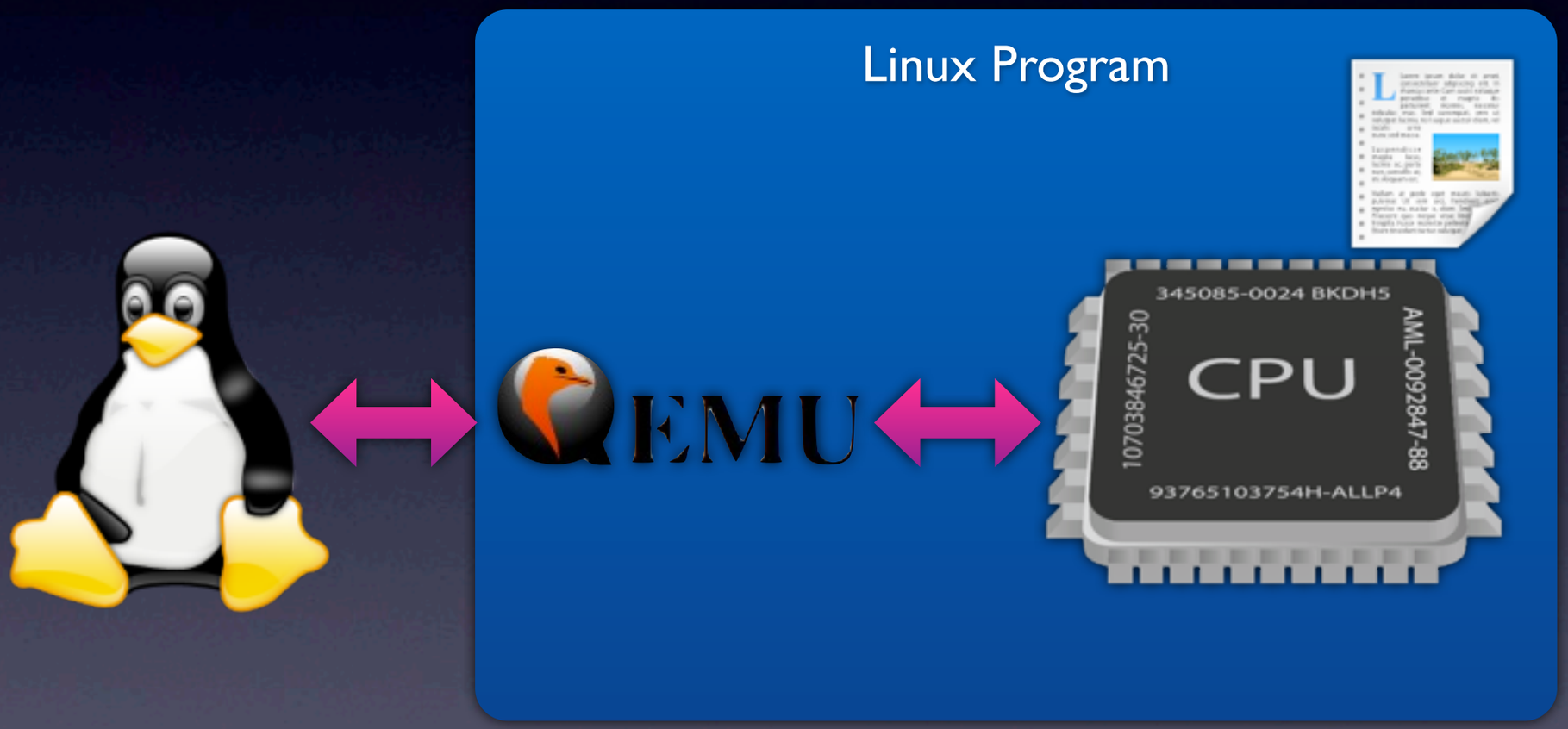
Linux-user emulation



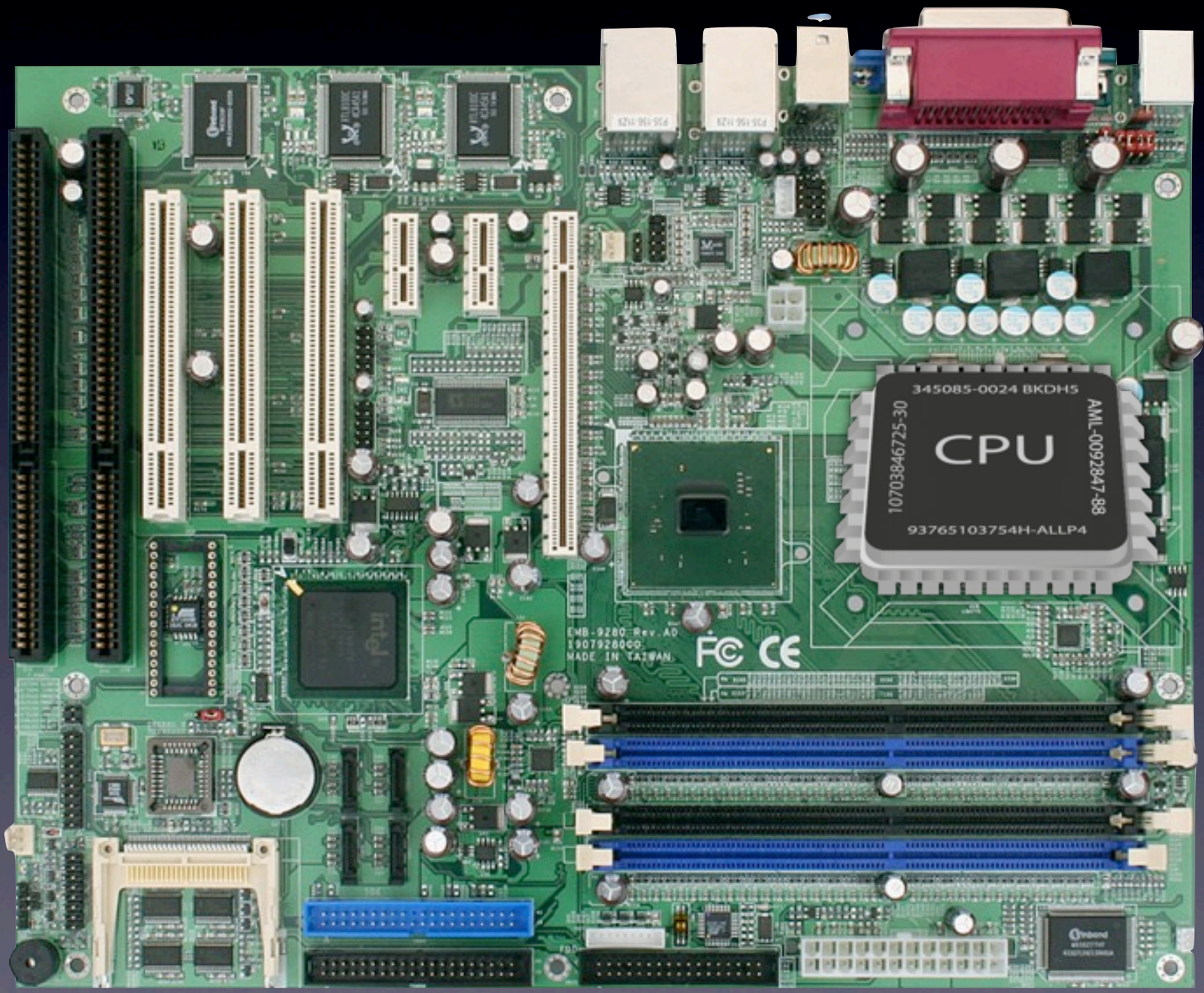
Linux-user emulation



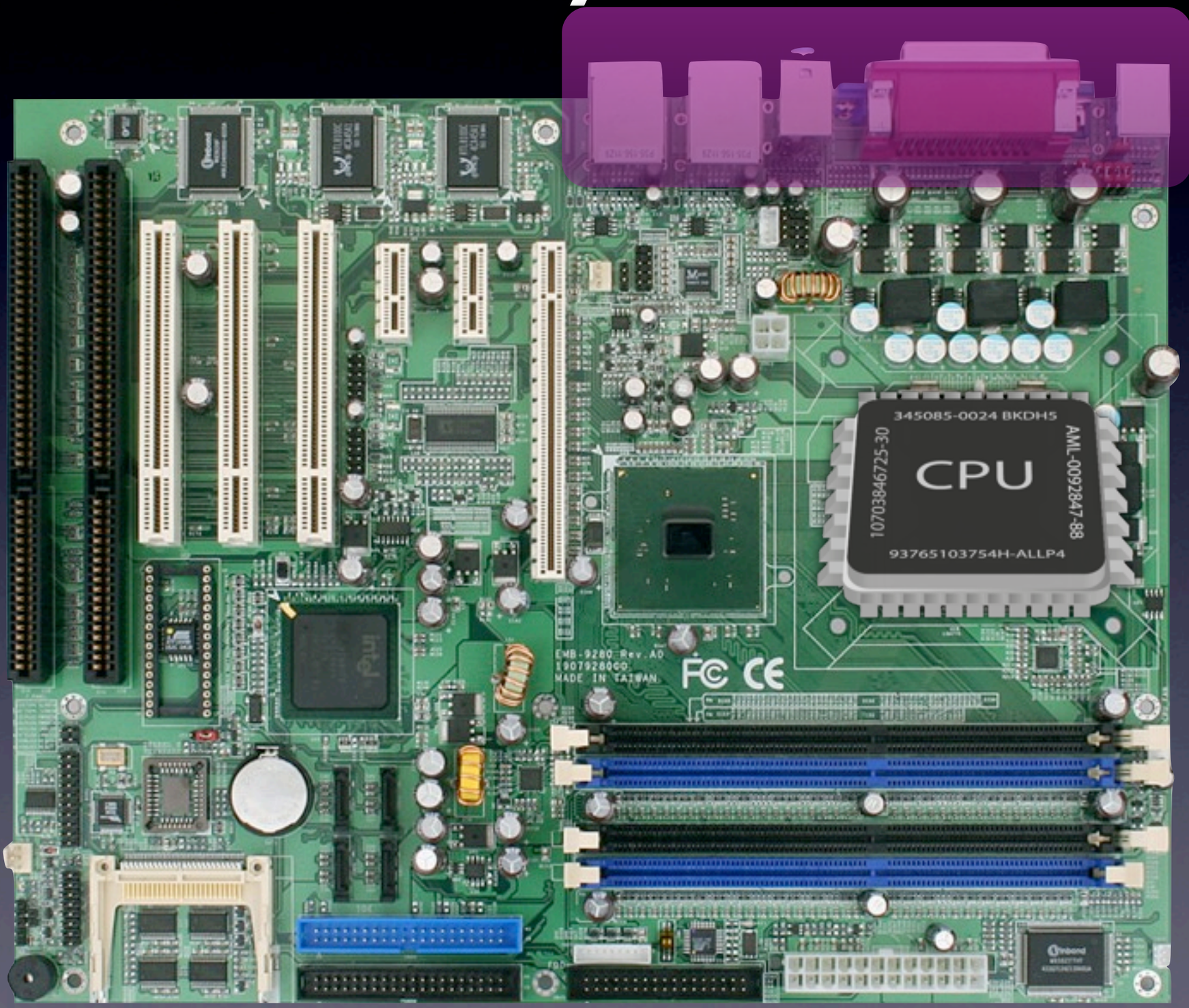
Linux-user emulation



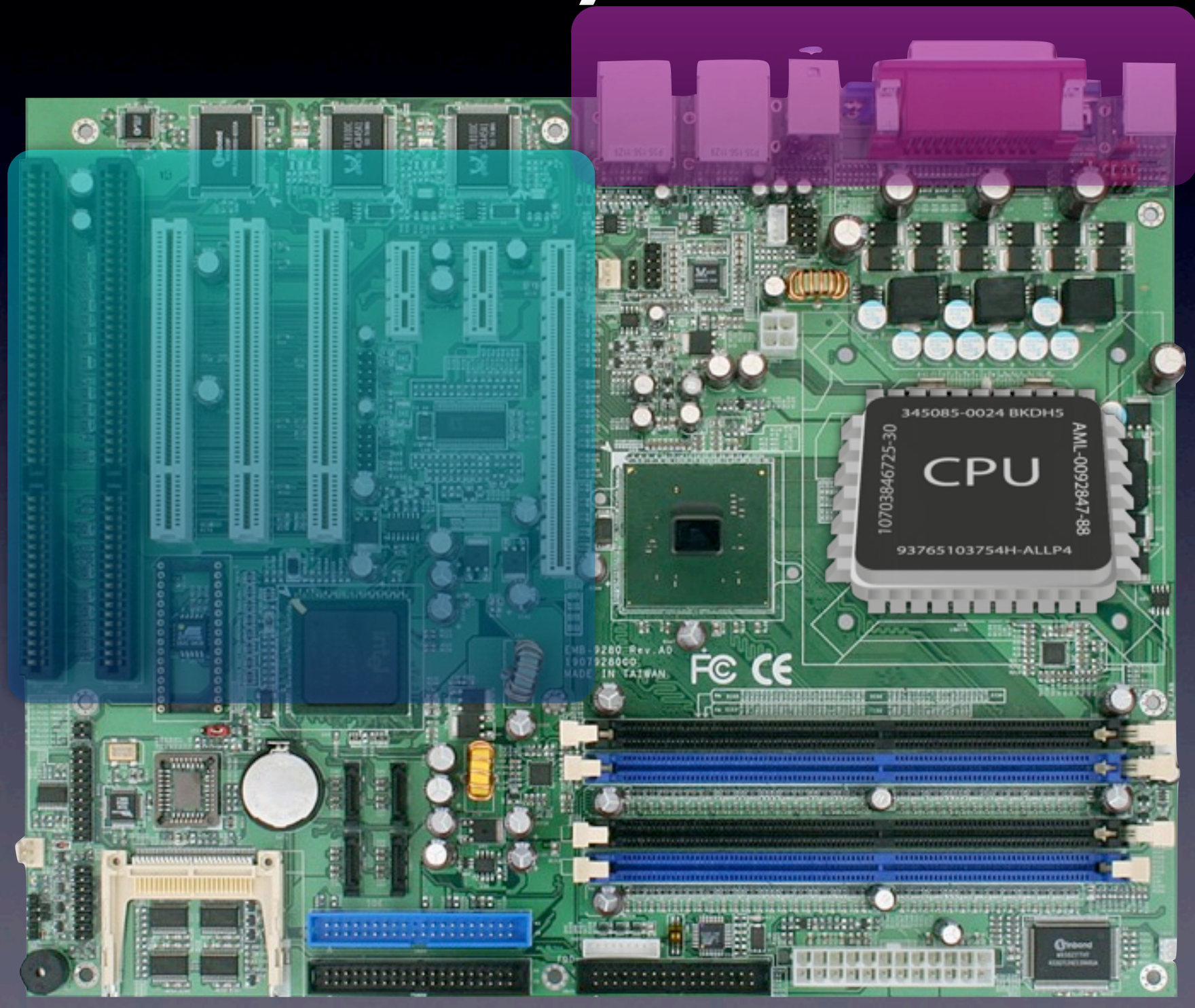
Full system


345085-0024 BKDH5
CPU
AML-0092847-88
93765103754H-ALLP4
EMB-9280 Rev.A0
MADE IN TAIWAN

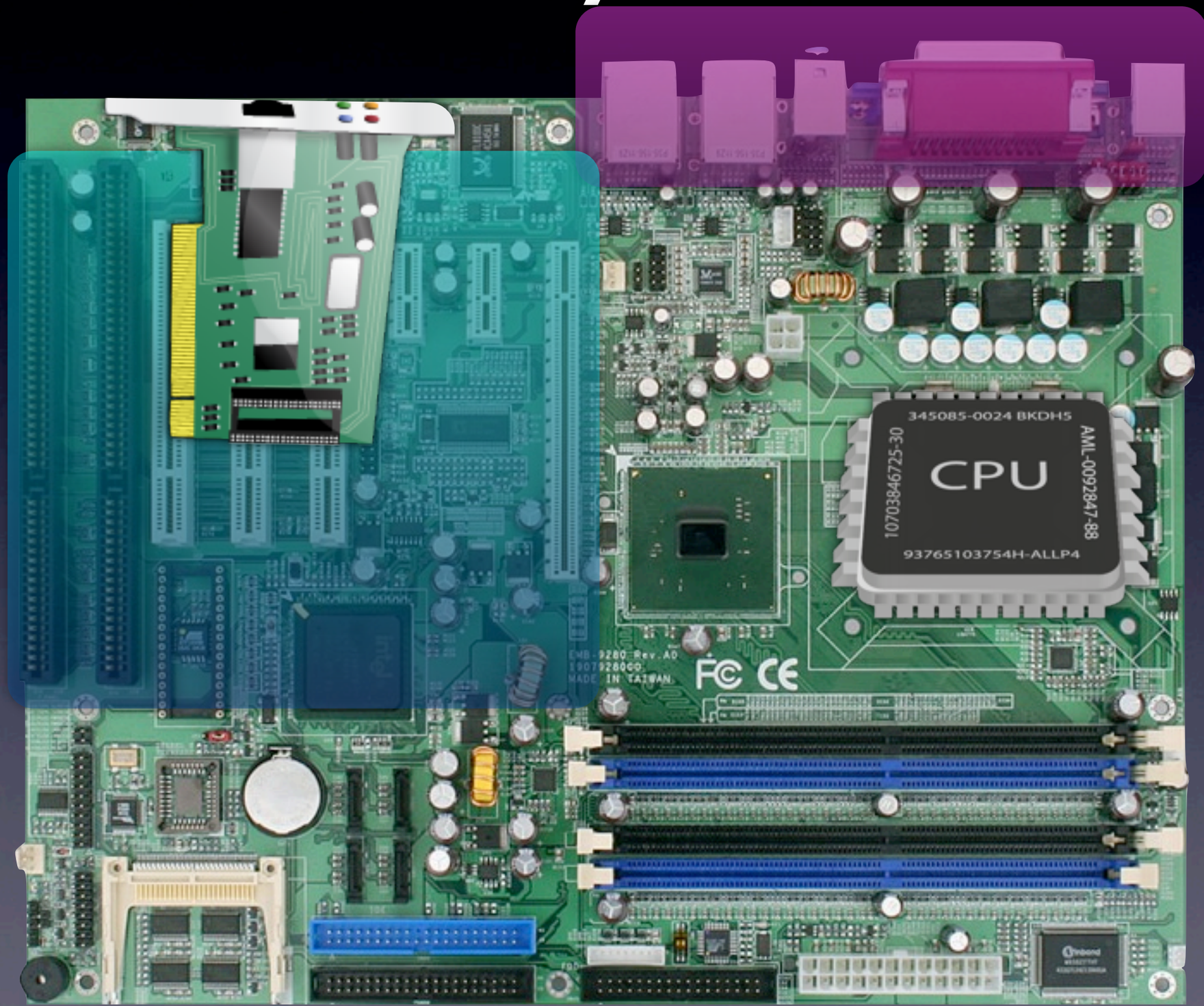
Full system


345085-0024 BKDH5
CPU
AML-0092847-88
93765103754H-ALLP4
EMB-9280 Rev.A0
MADE IN TAIWAN

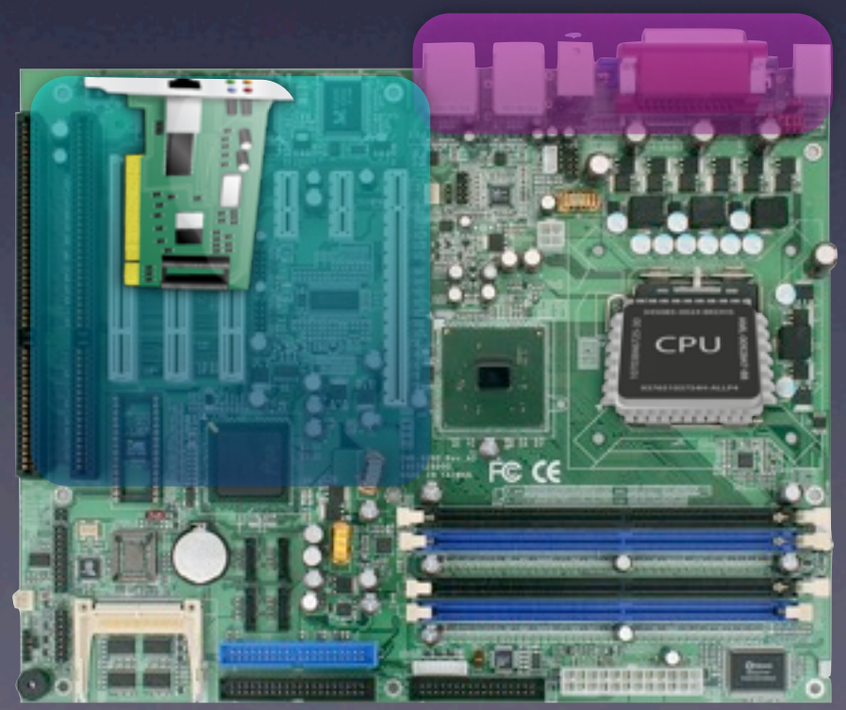
Full system


345085-0024 BKDH5
CPU
10703846725-30
AML-0092847-88
93765103754H-ALLP4
EMB-9280 Rev.A0
MADE IN TAIWAN

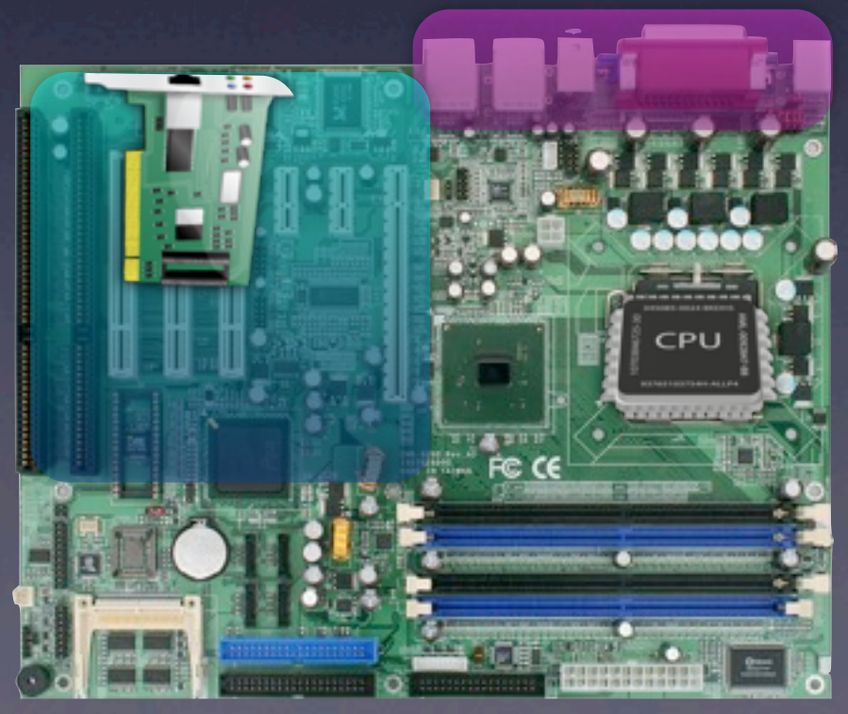
Full system


345085-0024 BKDH5
CPU
10703846725-30
AML-0092847-88
93765103754H-ALLP4
EMB-9280 Rev.A0
MADE IN TAIWAN

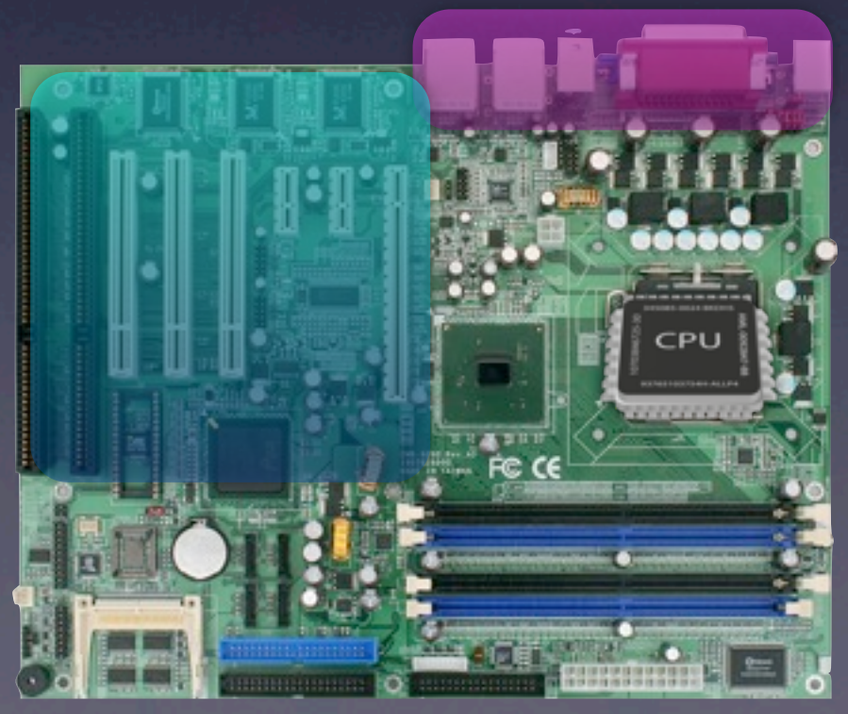
Full system



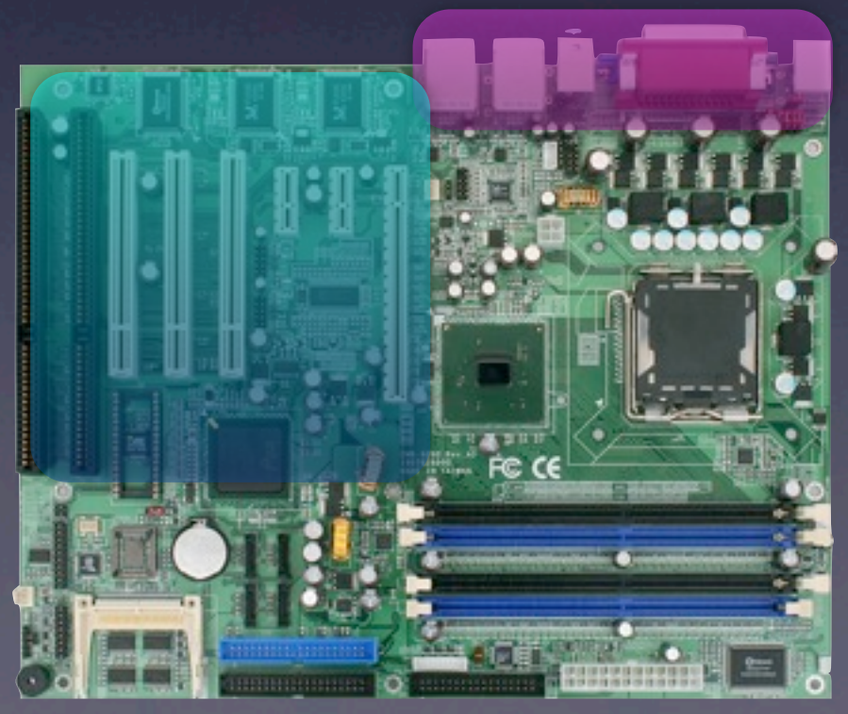
Full system



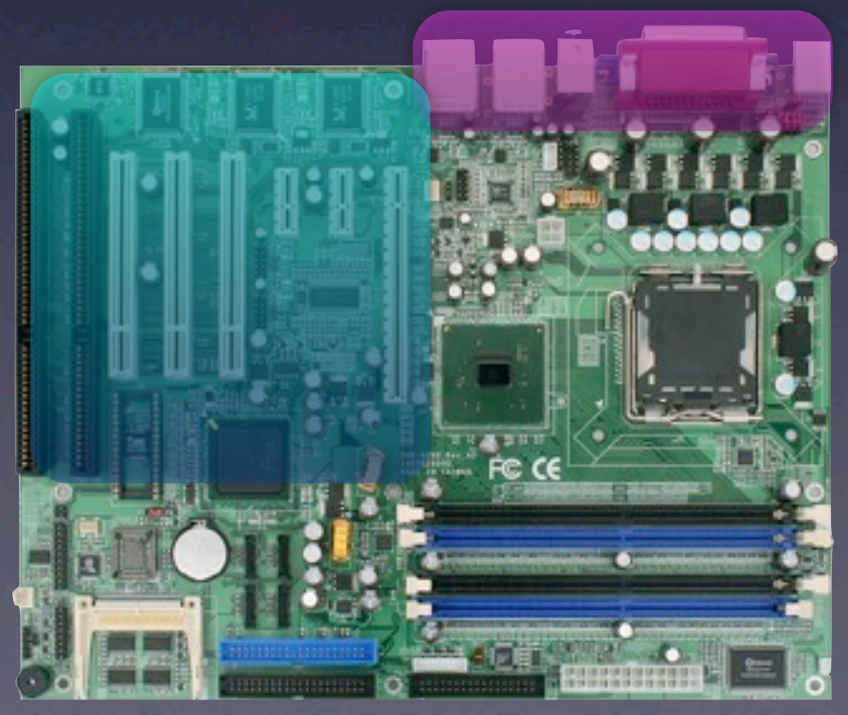
Full system



Full system



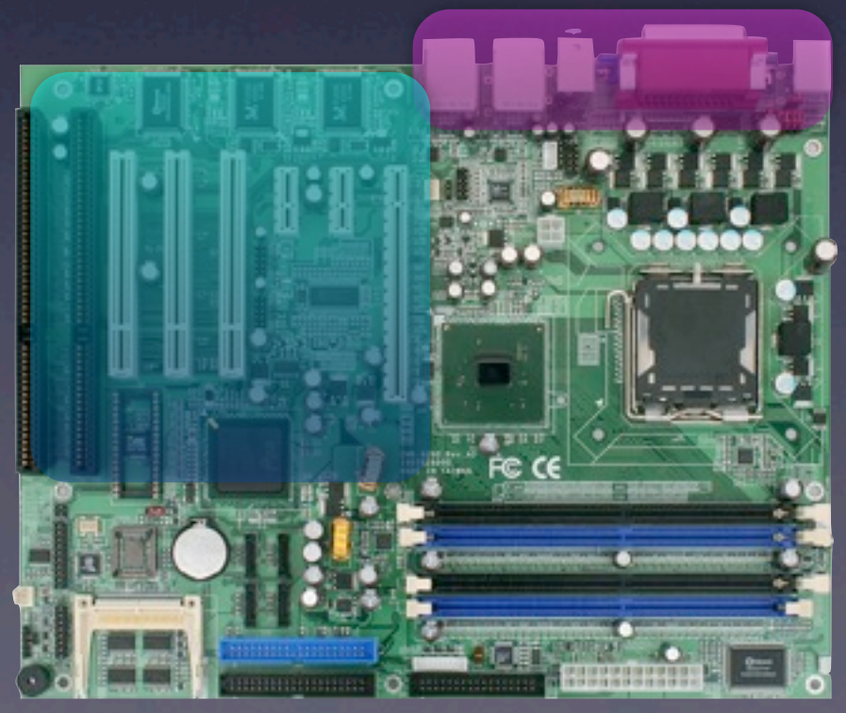
Full system



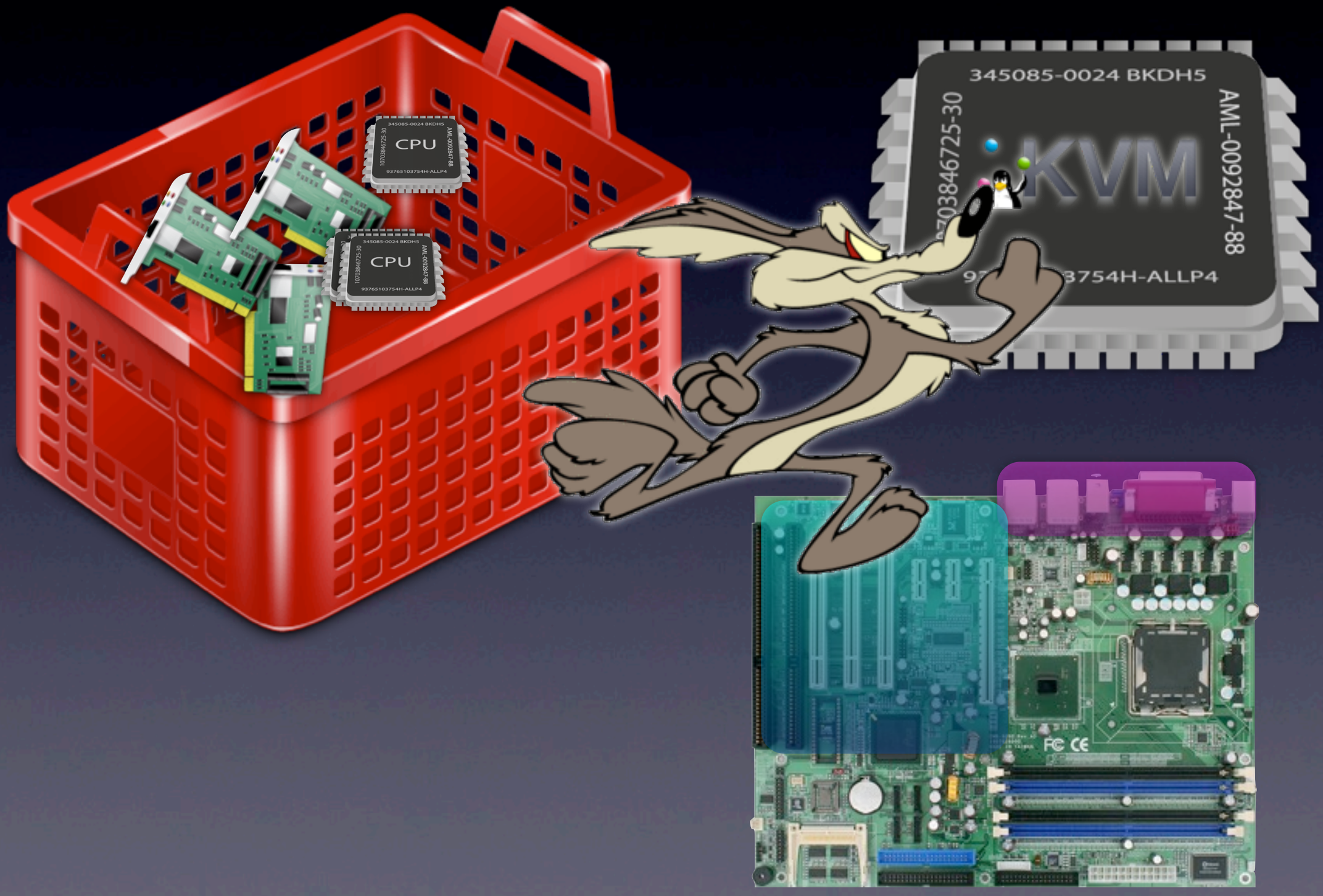
Full system


345085-0024 BKDH5
CPU
10703846725-30
AML-0092847-88
93765103754H-ALLP4


345085-0024 BKDH5
10703846725-30
AML-0092847-88
KVM
93765103754H-ALLP4



Full system


345085-0024 BKDH5
CPU
10703846725-30
AML-0092847-88
93765103754H-ALLP4
345085-0024 BKDH5
CPU
10703846725-30
AML-0092847-88
93765103754H-ALLP4
345085-0024 BKDH5
10703846725-30
KVM
AML-0092847-88
3754H-ALLP4
FC CE

Future Development

- Computer Systems Lab, Institute of Information Science @ Taiwan
- Uli's optimizations

Conclusion

- QEMU is very versatile
- A lot of areas for improvement
- Join in and hack away!