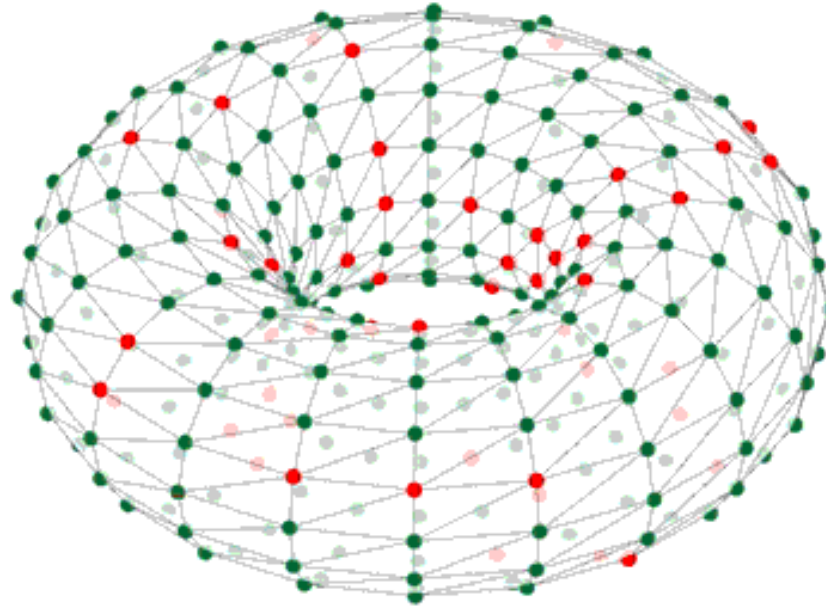


SpiNNaker Chip Resources



Steve Temple

SpiNNaker Workshop – Manchester – Sep 2015



European Research Council
Established by the European Commission



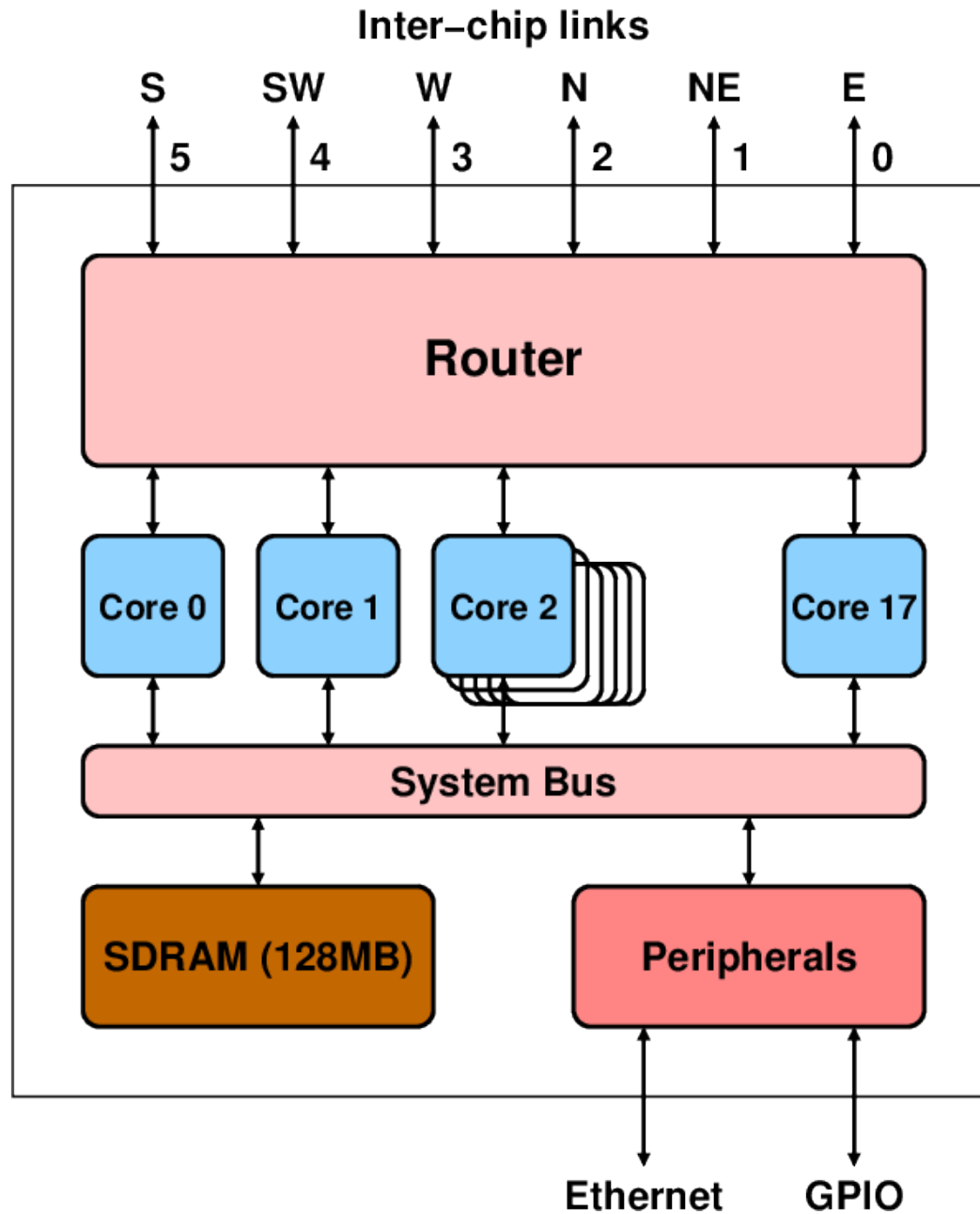
Human Brain Project



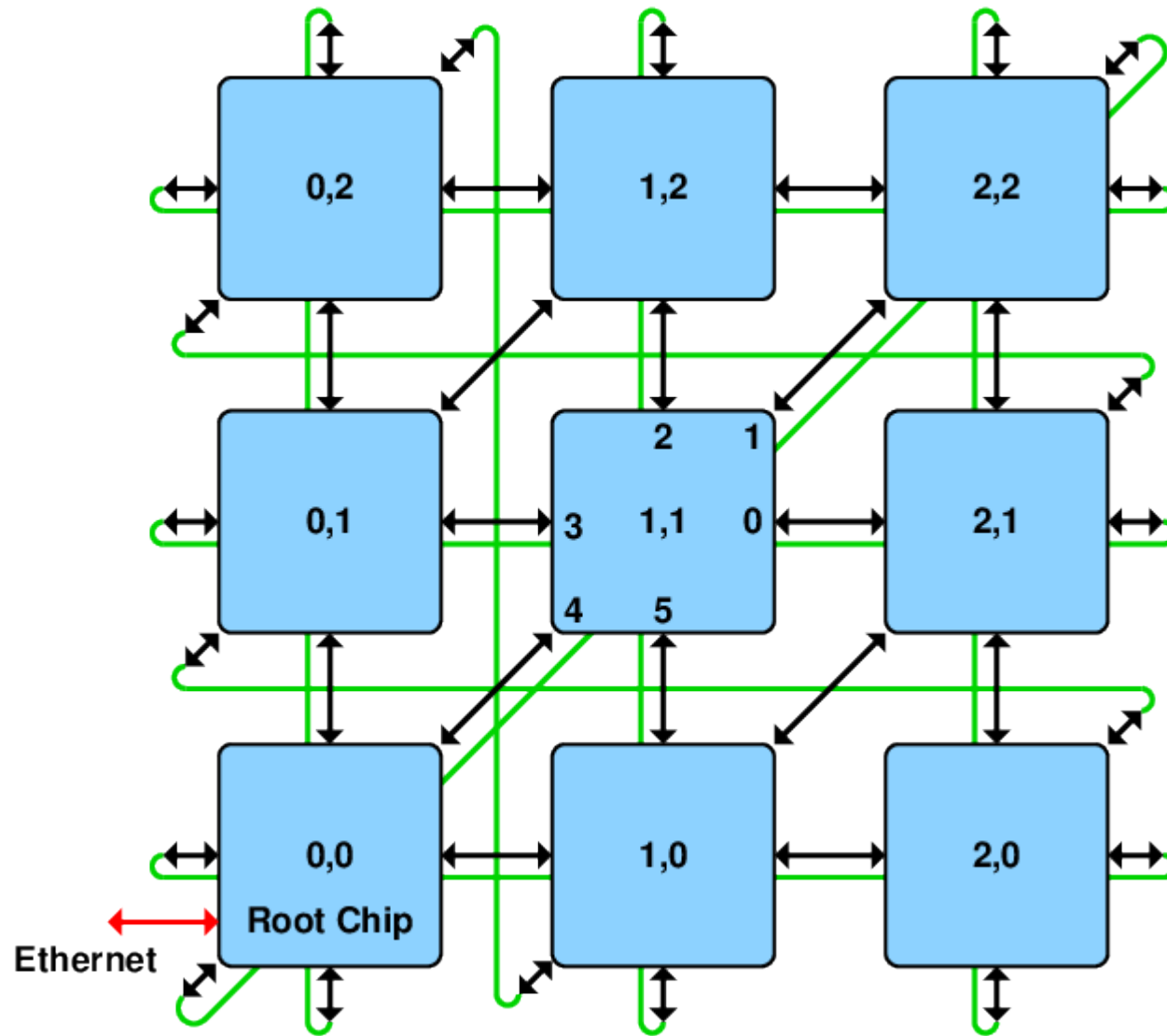
- Chip Architecture
- Core Architecture
- Low-level Communication
 - Packet formats
 - Multicast routing
- High-level Communication – SDP
- Hardware Limitations

Please interrupt if you have a question!

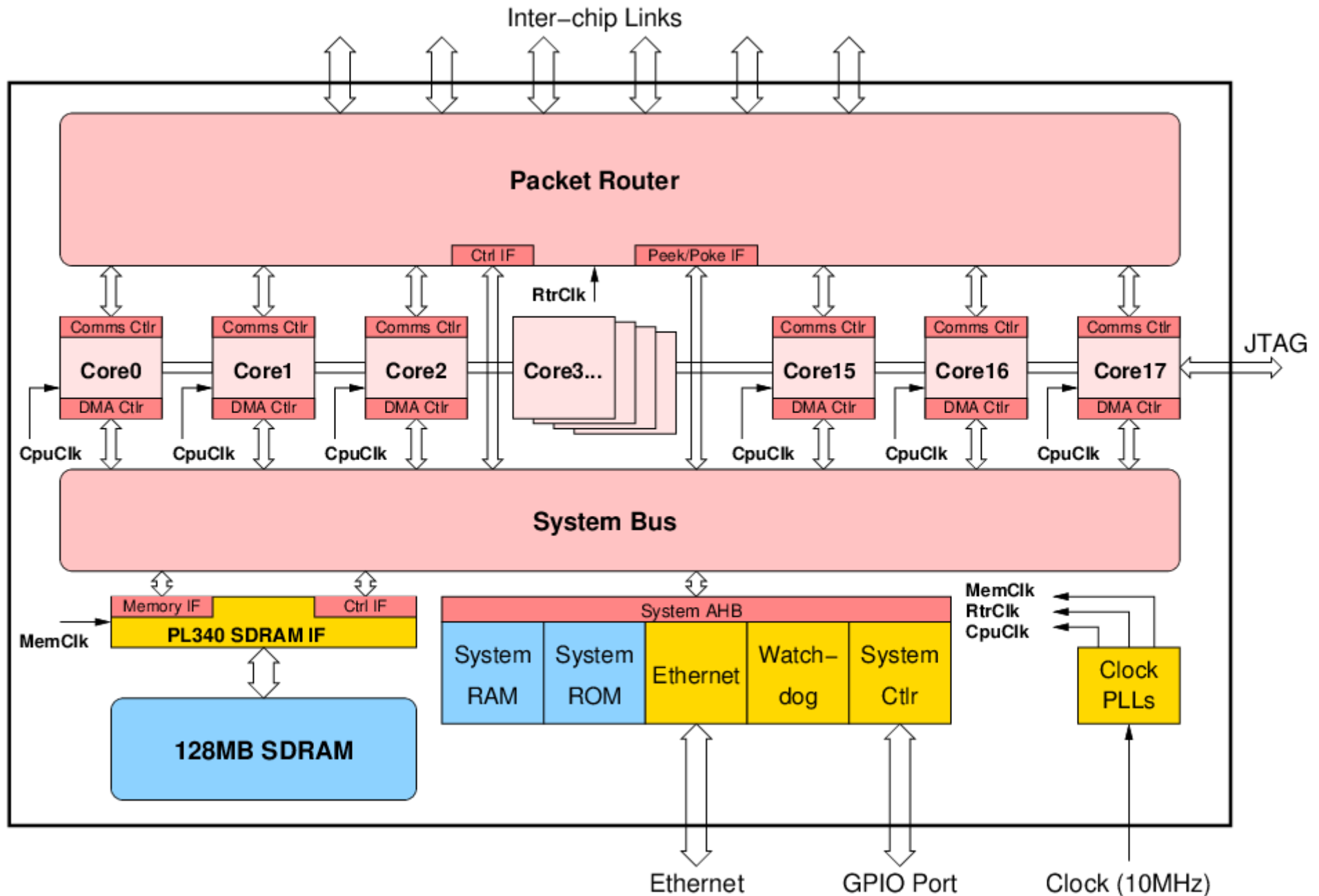
SpiNNaker Chip Outline



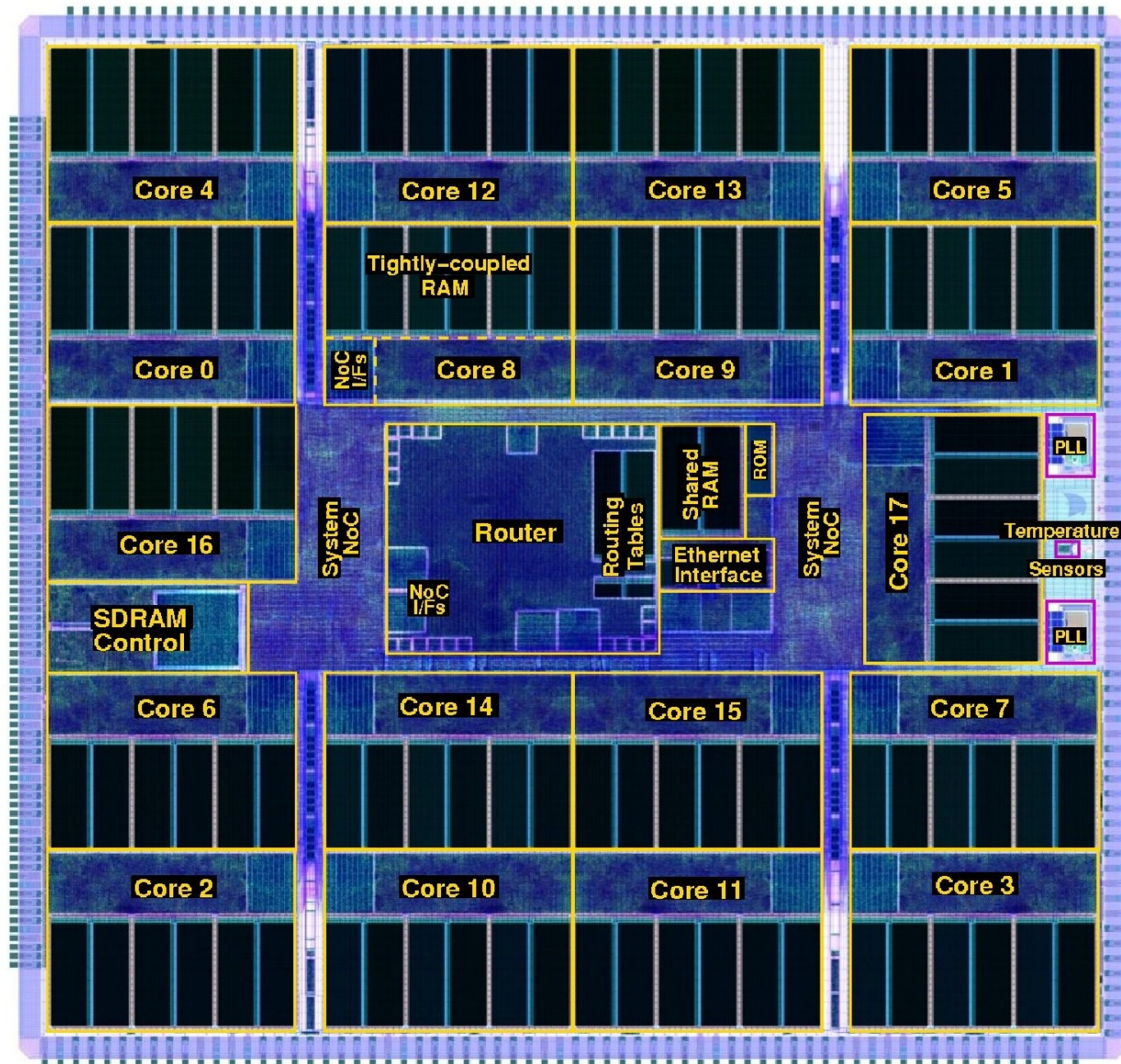
Chip Interconnect



SpiNNaker Chip Details



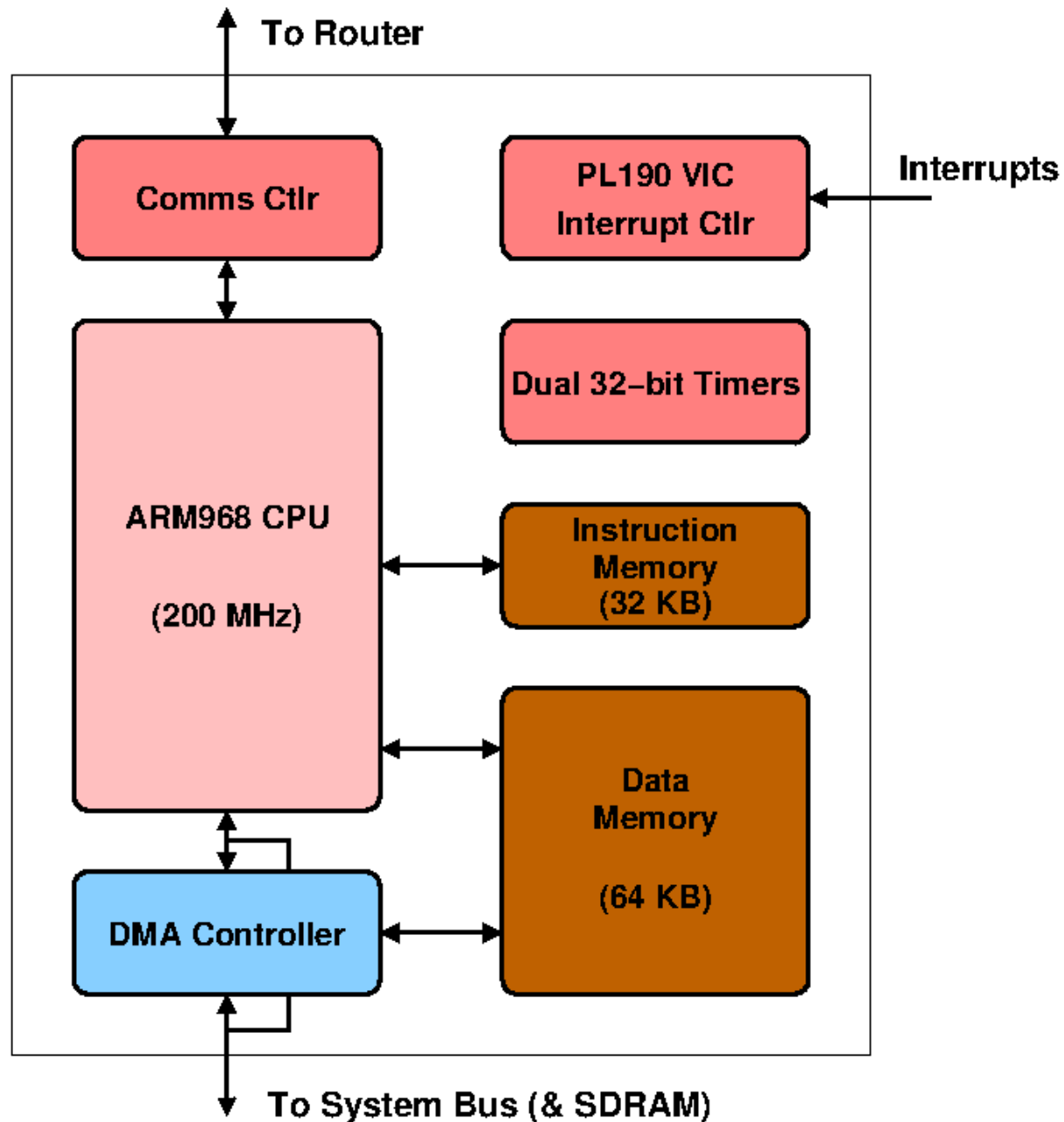
SpiNNaker Chip Layout



- 130nm process
- 10 x 10 mm
- 18 ARM cores with 96K SRAM
- Router
- SDRAM controller
- Asynchronous NoC



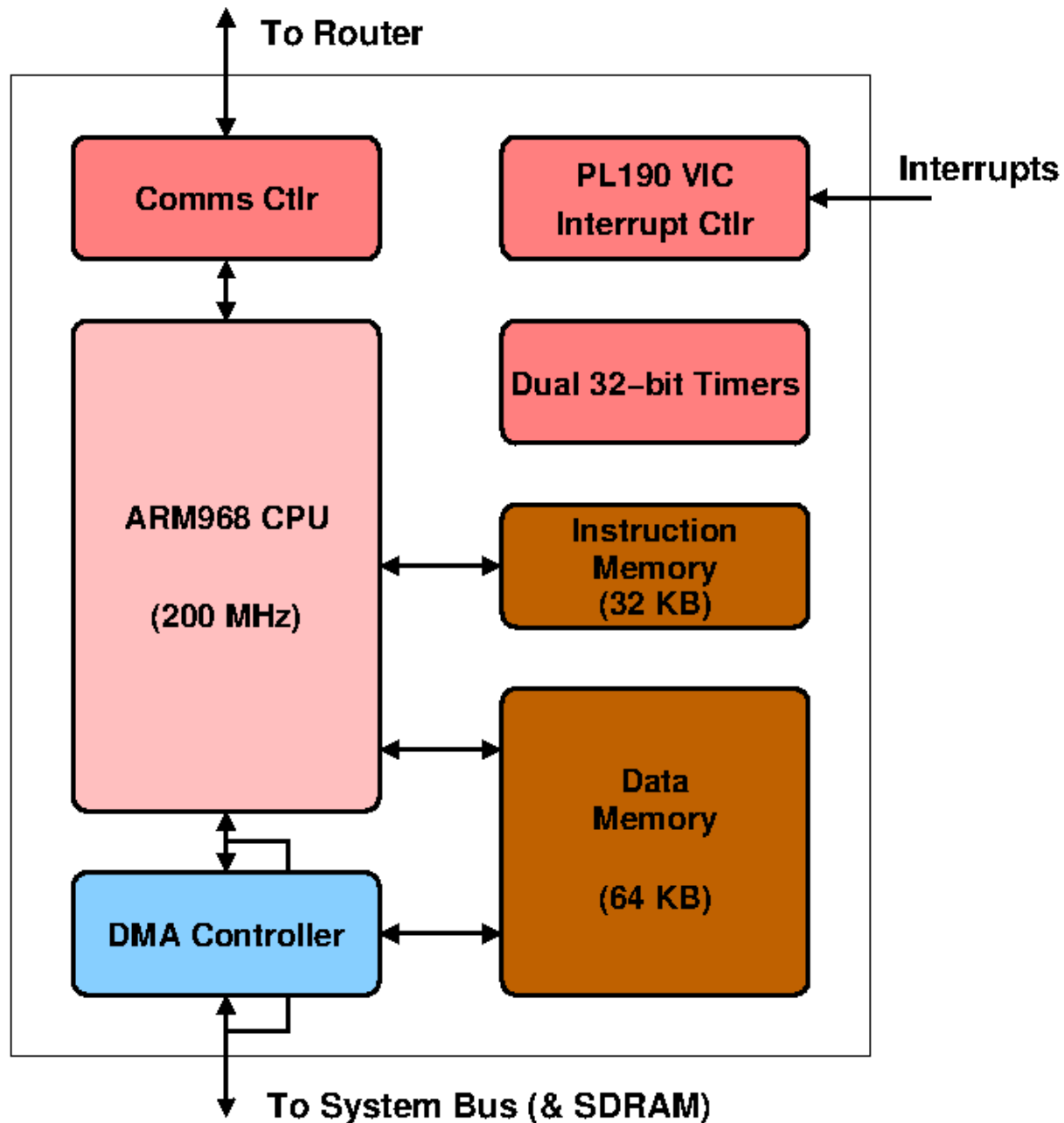
SpiNNaker Core



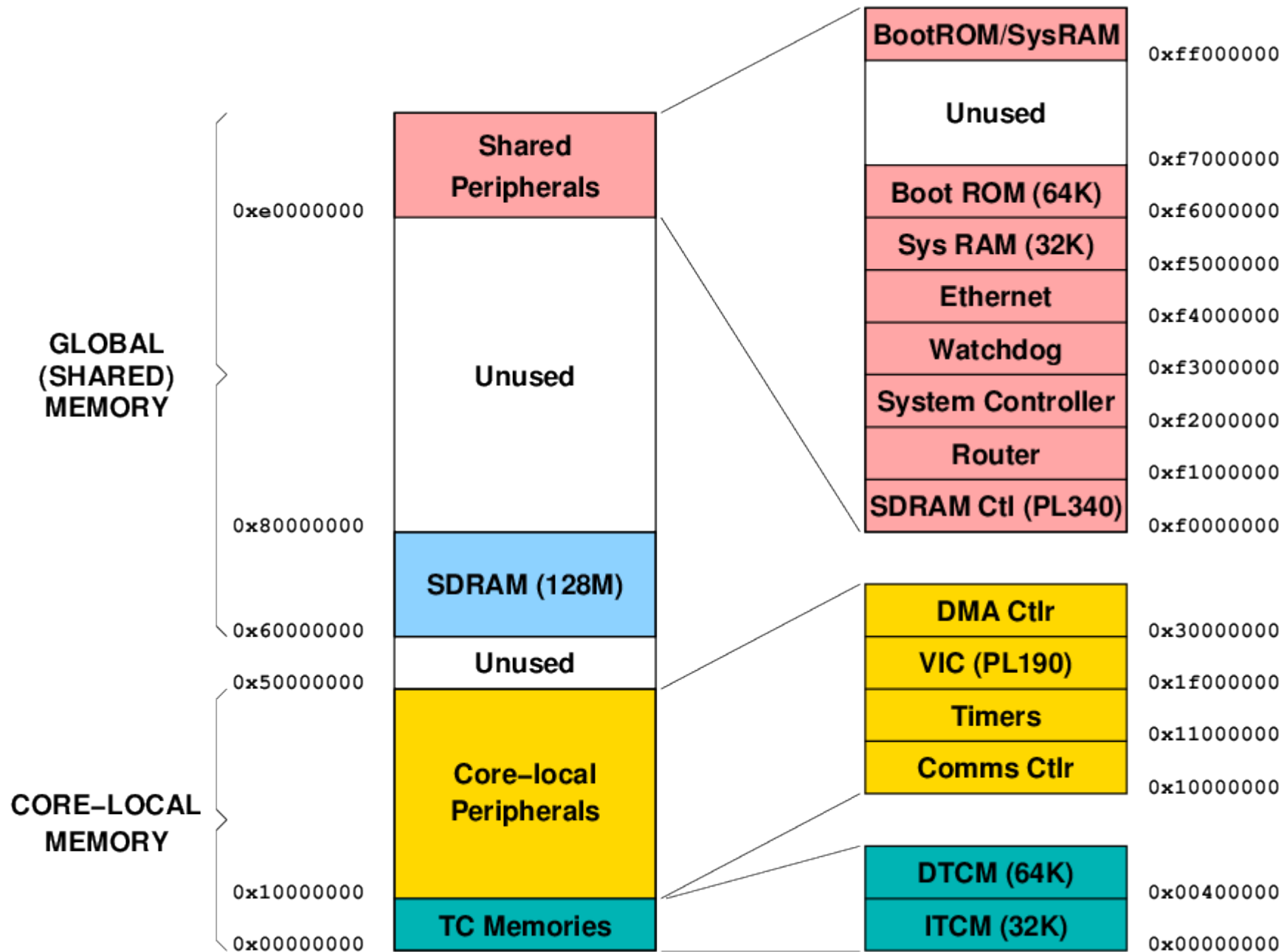
ARM968 CPU

- ARM9 CPU clocked at 200 MHz
- ARM v5TE architecture
 - Supports 32-bit ARM and 16-bit Thumb code
 - Some DSP instruction support - saturated arithmetic, extended multiplies
 - **No floating point hardware!**
- Two Tightly Coupled Memory (TCM) blocks
 - Single cycle (5 ns) access time
 - 32 KB Instruction TCM (ITCM)
 - 64 KB Data TCM (DTCM)
- DMA interface into both TCMs

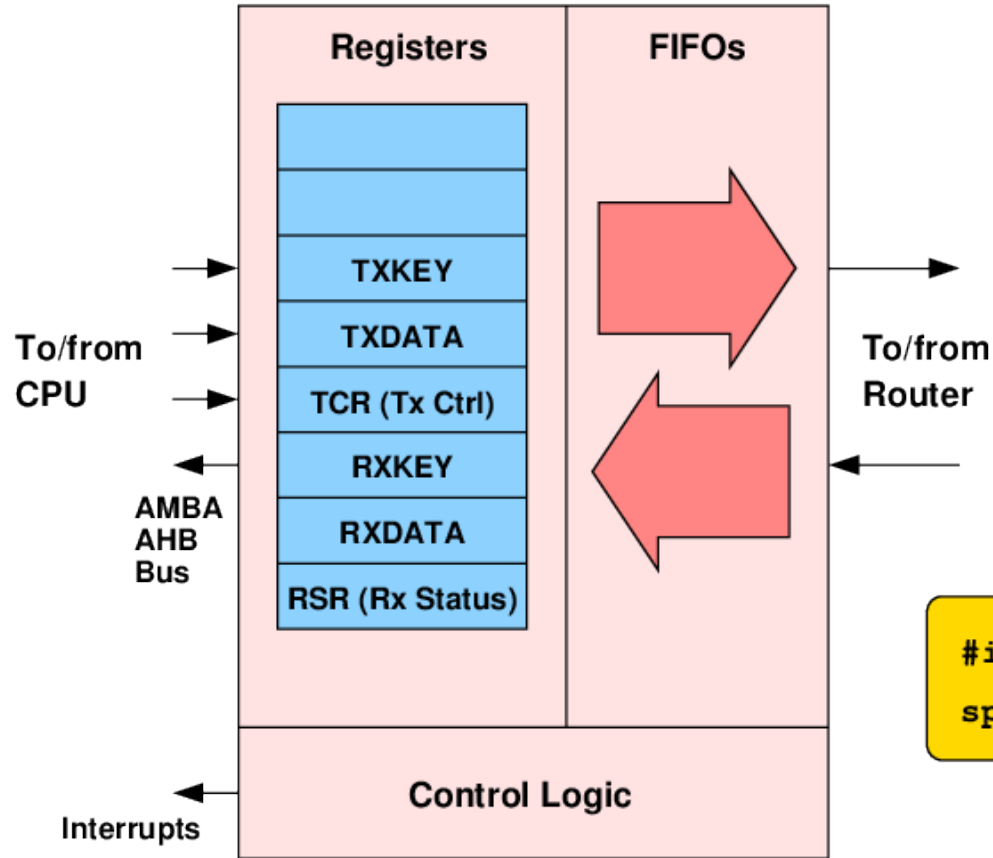
SpiNNaker Core



SpiNNaker Memory Map



Communications Controller



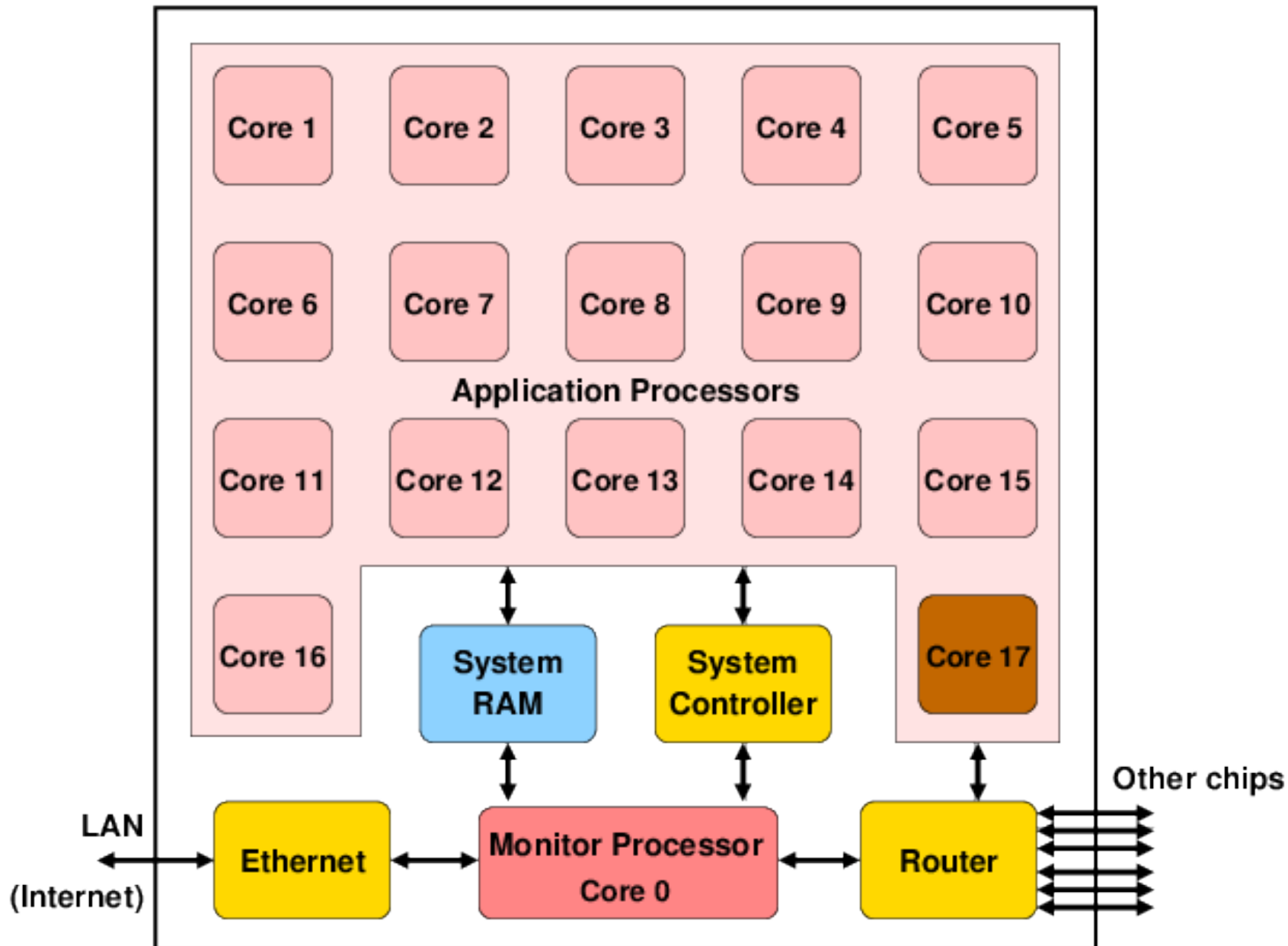
```
#include "spinnaker.h"

void send_mc_pkt (uint key, uint data)
{
    while (cc[TCR] & TX_FULL)
        continue;
    cc[TCR] = PKT_MC_PL;
    cc[TXDATA] = data;
    cc[TXKEY] = key;
}
```

```
#include "spin1_api.h"

spin1_send_mc_pkt (key, data, WITH_PAYLOAD);
```

Monitor Processor & Virtual Cores



SpiNNaker Packet Types

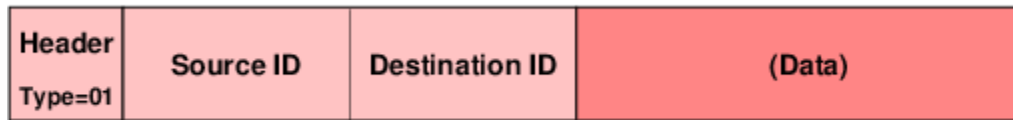
Multicast (MC)

Any core



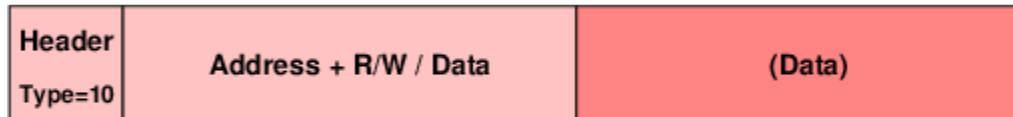
Point-to-point (P2P)

Monitor Processor



Nearest-neighbour (NN)

Monitor Processor



Fixed Route (FR)

Any core

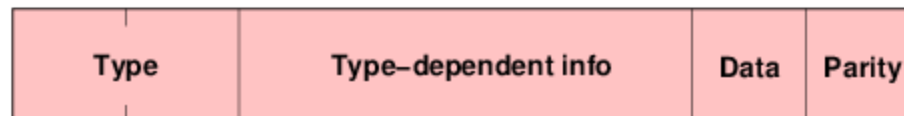


8 bits

32 bits

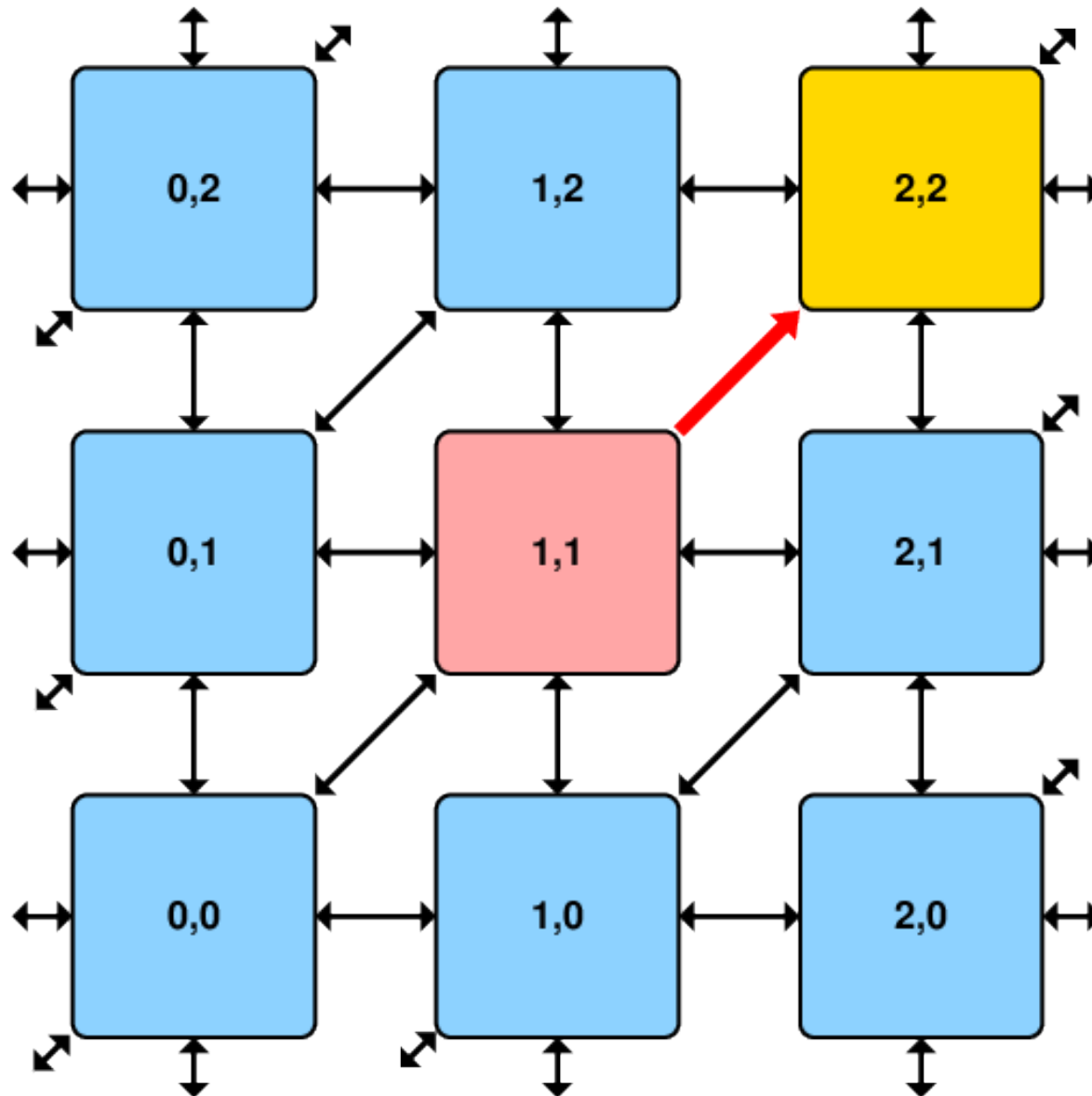
32 bits

Header

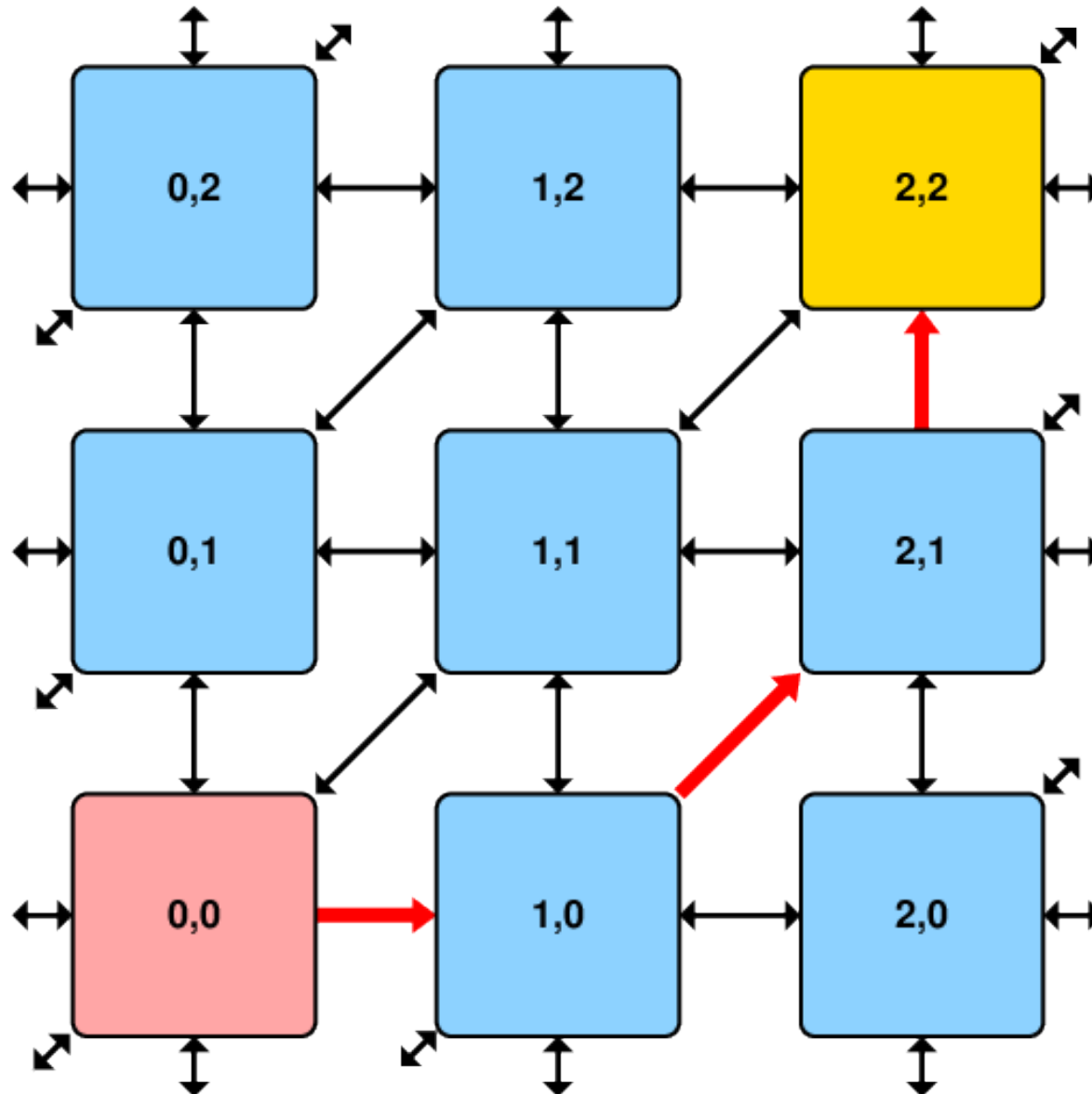


```
uint spin1_send_mc_pkt (uint key, uint data, uint payload);
```

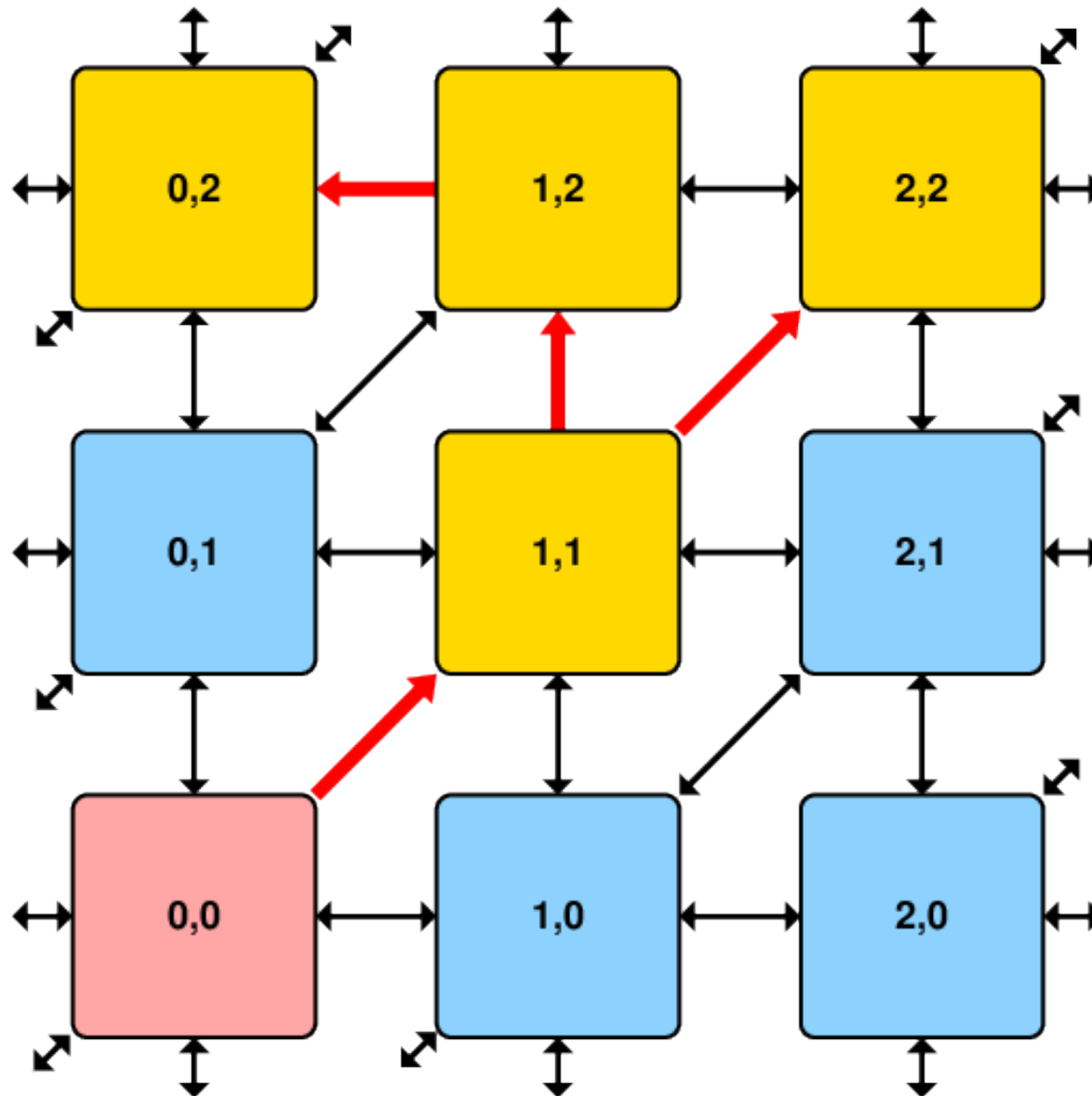
Nearest-neighbour packets



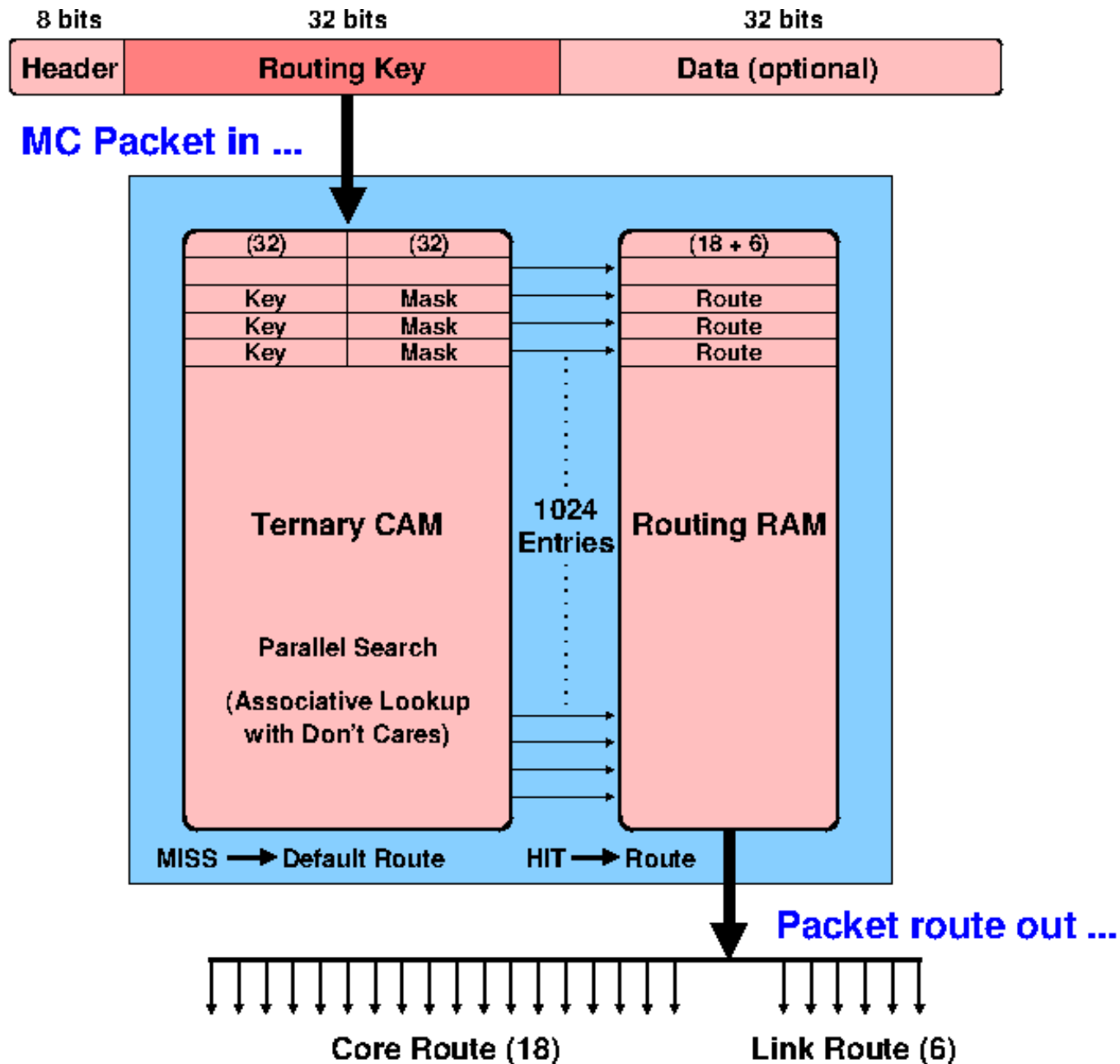
Point-to-point packets



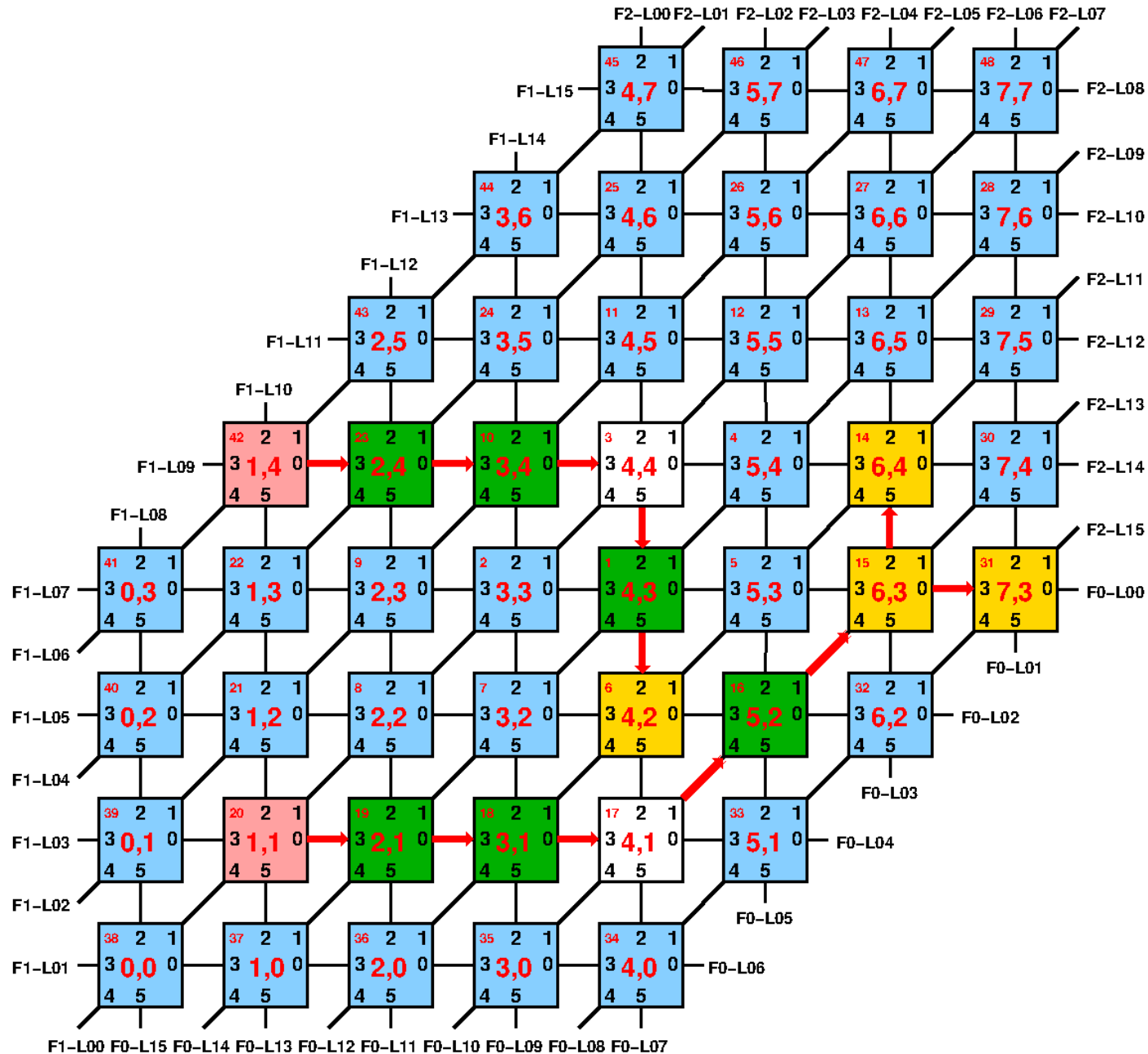
Multicast packets



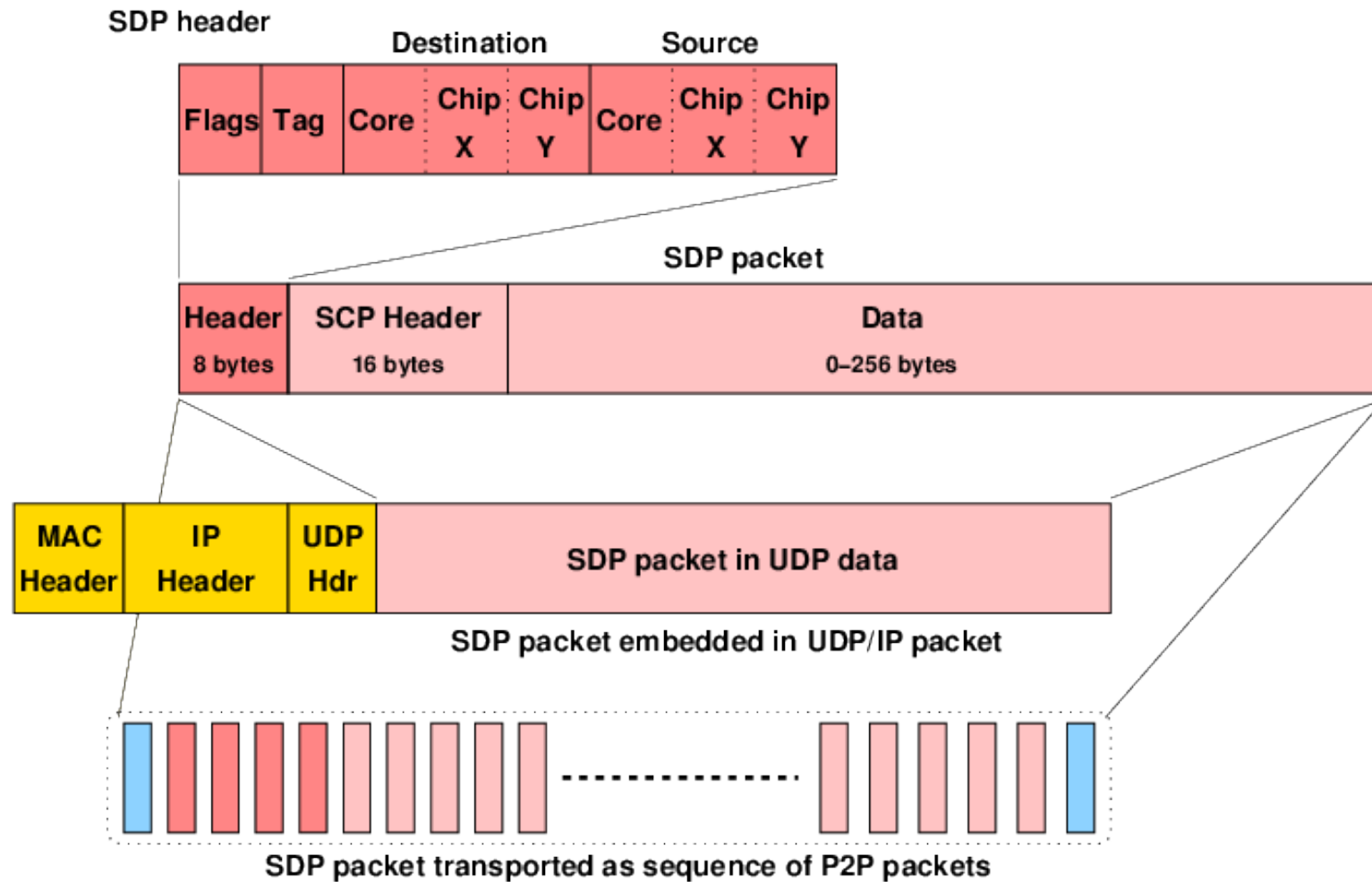
Multicast Packet Router



Multicast Packet Routing

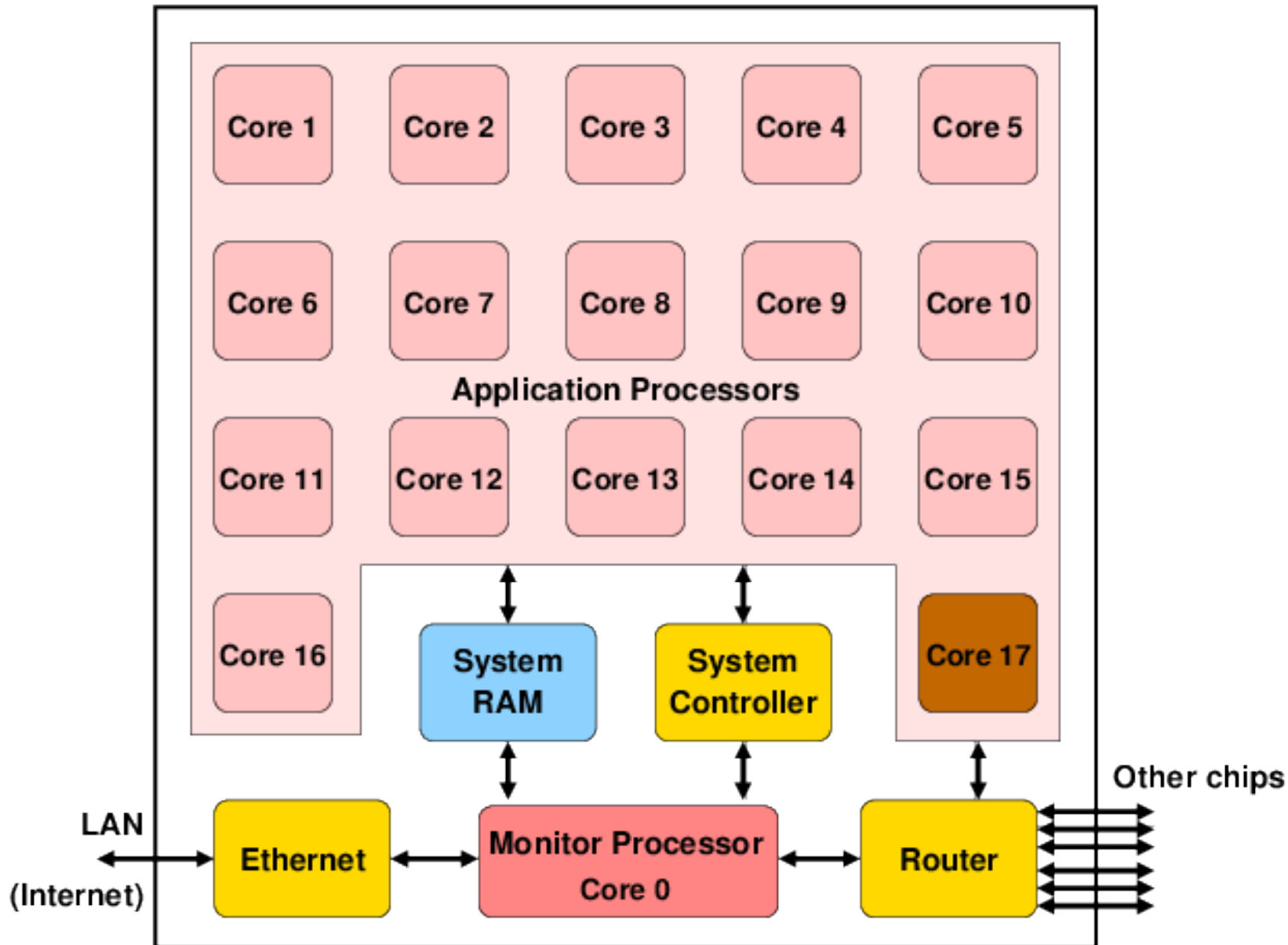


SpiNNaker Datagram Protocol



```
uint spin1_send_sdp_msg (sdp_msg_t *msg, uint timeout);
```

SDP Routing



SpiNNaker Hardware Limits

- Processors – 16/17 per chip (but scalable to thousands of chips)
- ARM968 – ARM9 at 200MHz – 220 DMIPS
- Local memory – very limited
 - Instruction memory – 32K bytes
 - Data memory – 64K bytes
- Local Memory access time - 5 ns
- Per chip memory – 128M bytes (shared)
- Shared memory access time
 - Individual accesses - > 100 ns (NB write buffer)
 - DMA accesses ~ 15ns per word

SpiNNaker Arithmetic Limits

- ARM968 has no floating point hardware
- Options
 - Soft Floating Point – slow and memory hungry
 - Fixed point – uses integer ops
 - Limited range before precision lost
 - Some GCC compiler support (but slowish)
 - Or hand code (C or assembly) for best performance (some libraries available)
- ARM968 has some DSP extensions
 - Saturation, MAC, double operations, CLZ
 - Accessible via compiler intrinsics

SpiNNaker Packet Limits

- Packet payload is small – typically 32 bits
- Packet bandwidth is limited
- Chip-to-chip links ~ 250M bit/s (5 or 3 M pkt/s)
 - Currently 50% slower via board-to-board links
- CPU packet processing overhead typically 200-1000ns
- Packets can get lost (dropped) in case of congestion – can be “re-injected” in some cases
- Multicast router table is not infinite!

SpiNNaker Bandwidth Limits

- Overall I/O bandwidth into the machine is limited
- Currently most external I/O is by 100 Mbit/s Ethernet (and only one interface per board)
- High level I/O via SDP is limited by software overheads
 - Around 10 Mbyte/s to Ethernet-attached chip
 - Around 2 Mbyte/s to 'unattached' chips (via P2P packets)
- Potential for higher I/O bandwidth via SATA links on FPGAs but currently unexploited

That's all for now – any questions?