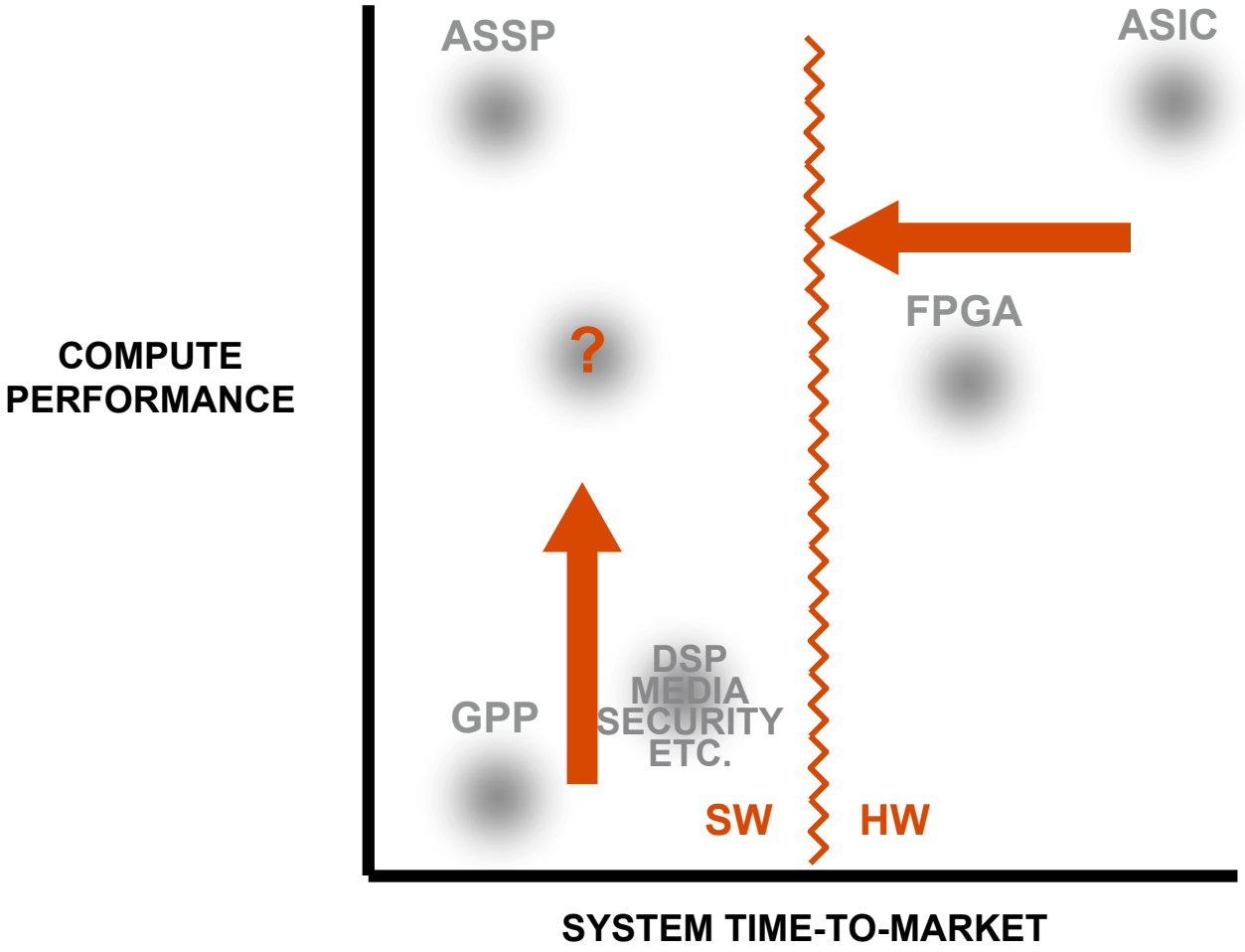
Software Configurable Processors Change System Design

Ricardo E. Gonzalez

© 2005 Stretch Inc. All rights reserved.

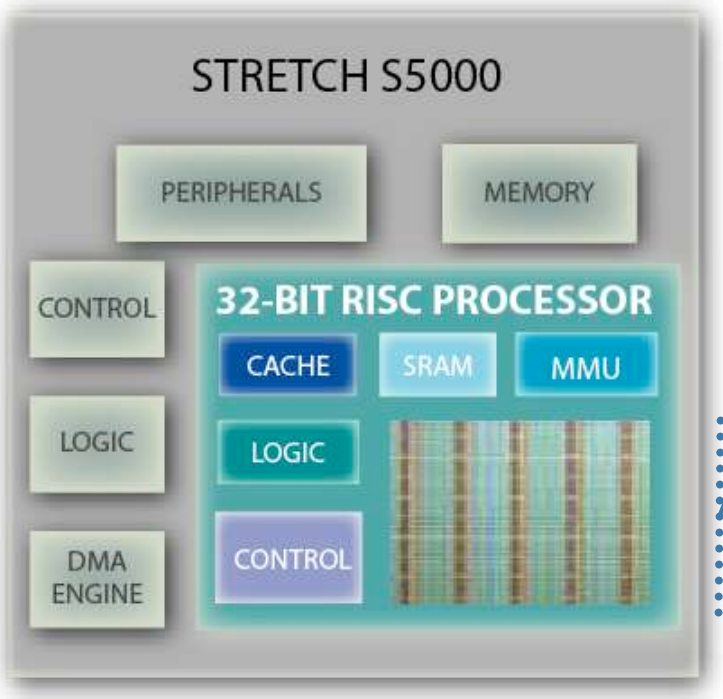
Embedded System Design Dilemma



© 2005 Stretch Inc. All rights reserved.

Stretch

Solving Compute Intensive Problems

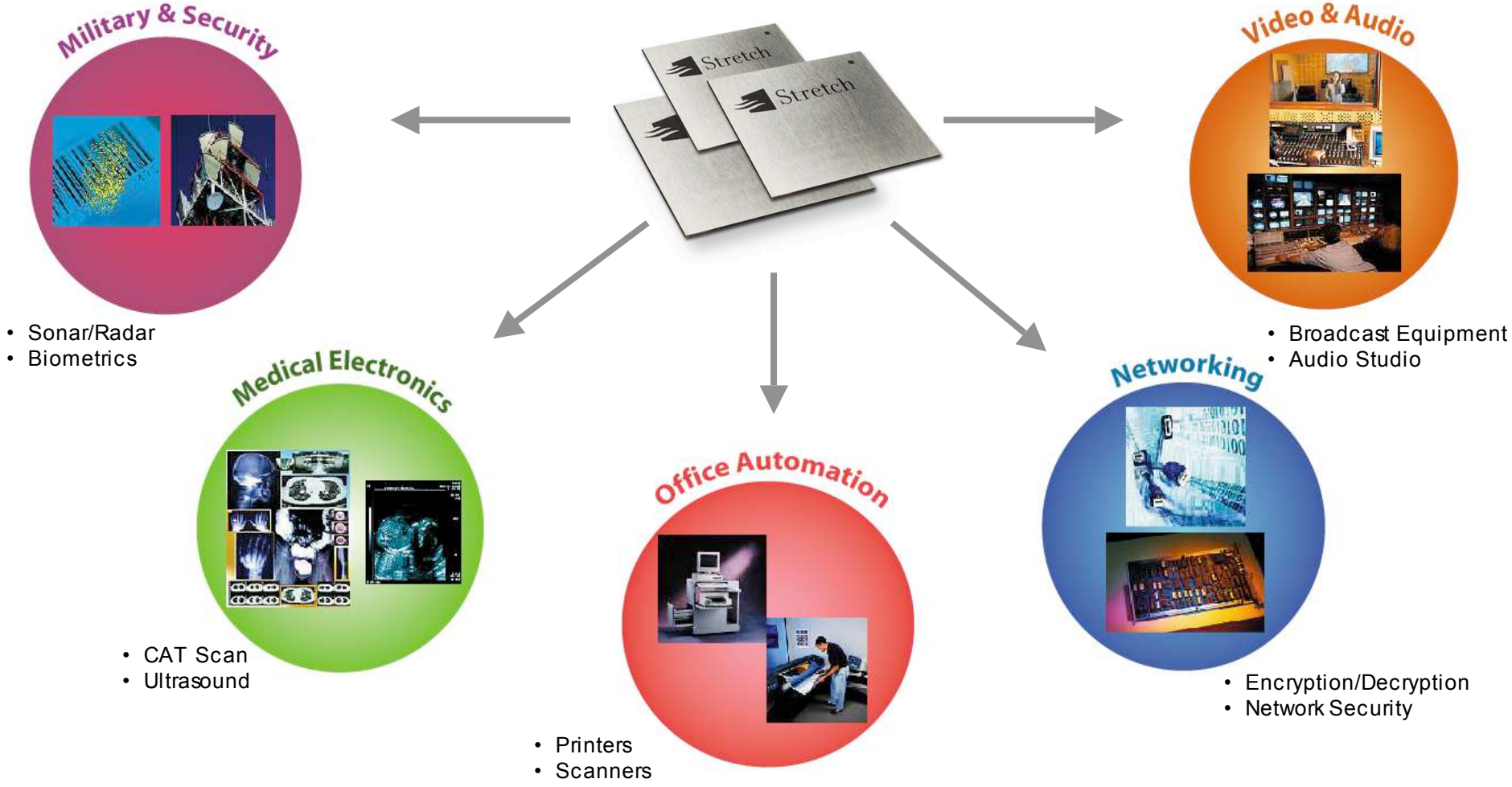


- Processor with embedded programmable logic
  - Lots of compute resources
- Utilize fabric by extending the ISA
  - Design your own FU
  - Add processor state
- Simple programming model
  - Issue instructions to perform computation
- Tailored for high-throughput applications
  - Compute intensive

© 2005 Stretch Inc. All rights reserved.

Stretch

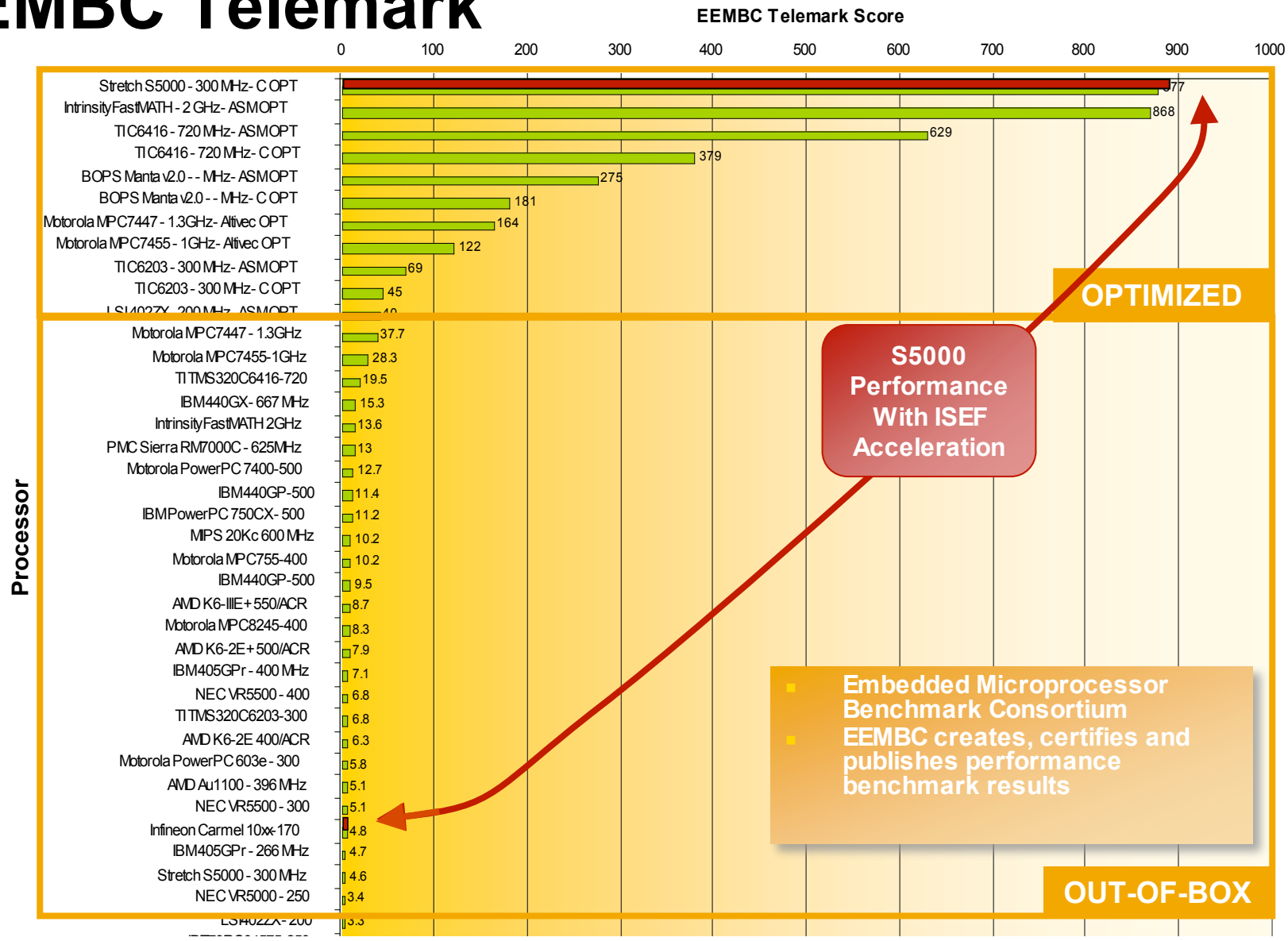
Compute Intensive Markets & Applications



© 2005 Stretch Inc. All rights reserved.

Stretch

EEMBC Telemark



© 2005 Stretch Inc. All rights reserved.

Stretch

Extending the Possibilities > www.stretchinc.com

Application Acceleration Process

© 2005 Stretch Inc. All rights reserved.

RGB → YC_bC_r

Color Conversion Function:

```
SE_FUNC              /* Tells Stretch C-Compiler to reduce this function to an instruction */
Void RGB2YCBCR (WR A, WR *B) {
  char Y[4], Cb[4], Cr[4];
  char R[4], G[4], B[4];

  R[0] = A(23,16); G[0] = A(15,8); B[0] = A(7,0);
        /* and so on. . . */

  for (i = 0; i < 4; i++) {
    Y[i]  = (77*R[i] + 150*G[i] + 29*B[i]) >> 8;
    Cb[i] = (32768 - 43*R[i] - 85*G[i] + (B[i] << 7)) >> 9;
    Cr[i] = (32768 + (R[i] << 7) - 107*G[i] - 21*B[i]) >> 9;
  }
  *B = (Y[3],Cr[3]+Cr[2],Y[2],Cb[3]+Cb[2],Y[1],Cr[1]+Cr[0],Y[0],Cb[1]+Cb[0]);
}
```

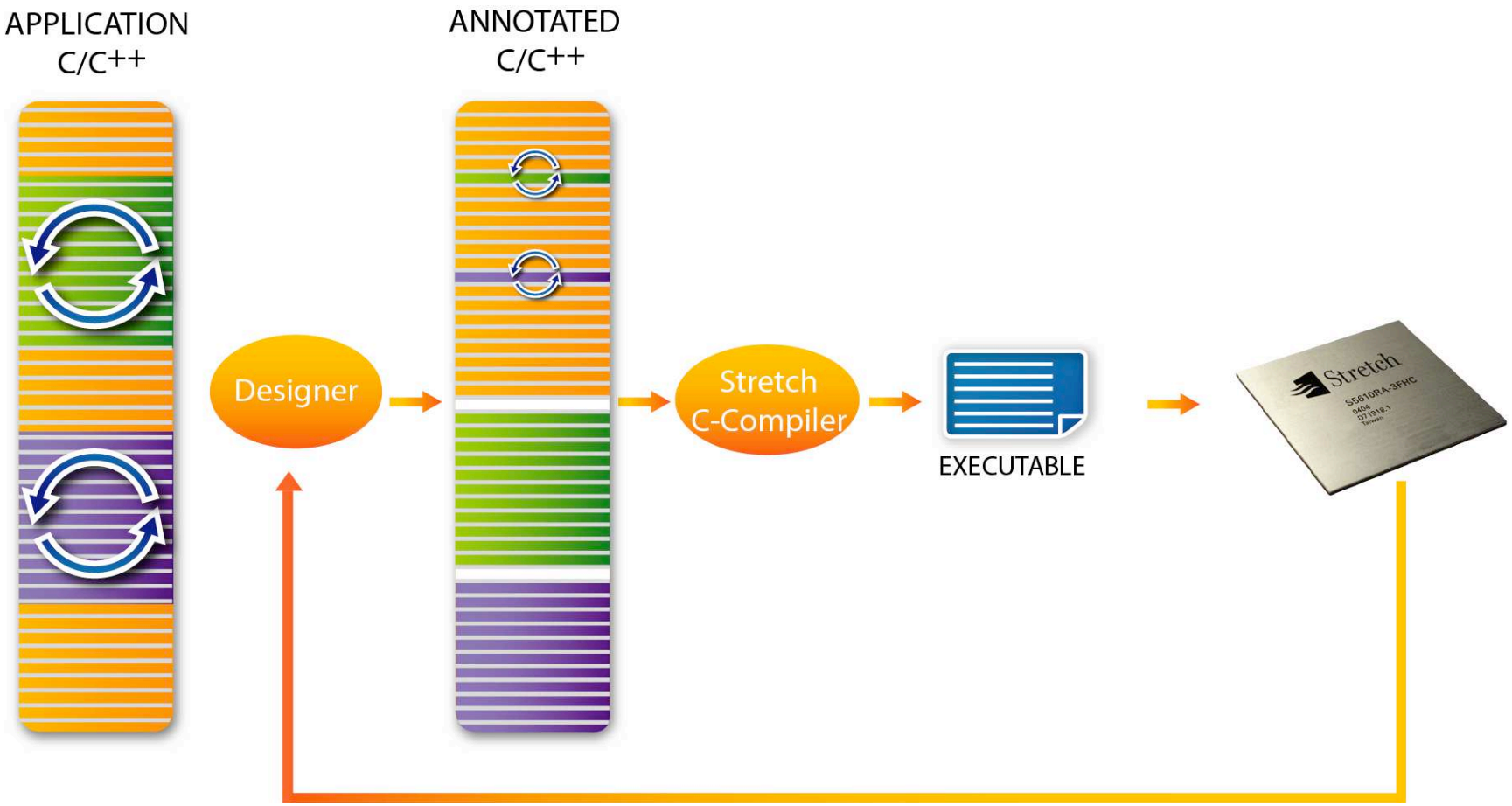
Program Loop:

```
for (…) {
    WRGET0(&A, 12);       /* Load 12 bytes (4 RGB pixels)  */
    RGB2YCBCR(A, &B);     /* Convert 4 pixels              */
    WRPUT0(B, 8);         /* Store 8 bytes (4 YCbCr pixels */
}
```

© 2005 Stretch Inc. All rights reserved.

Stretch

Programming Flow

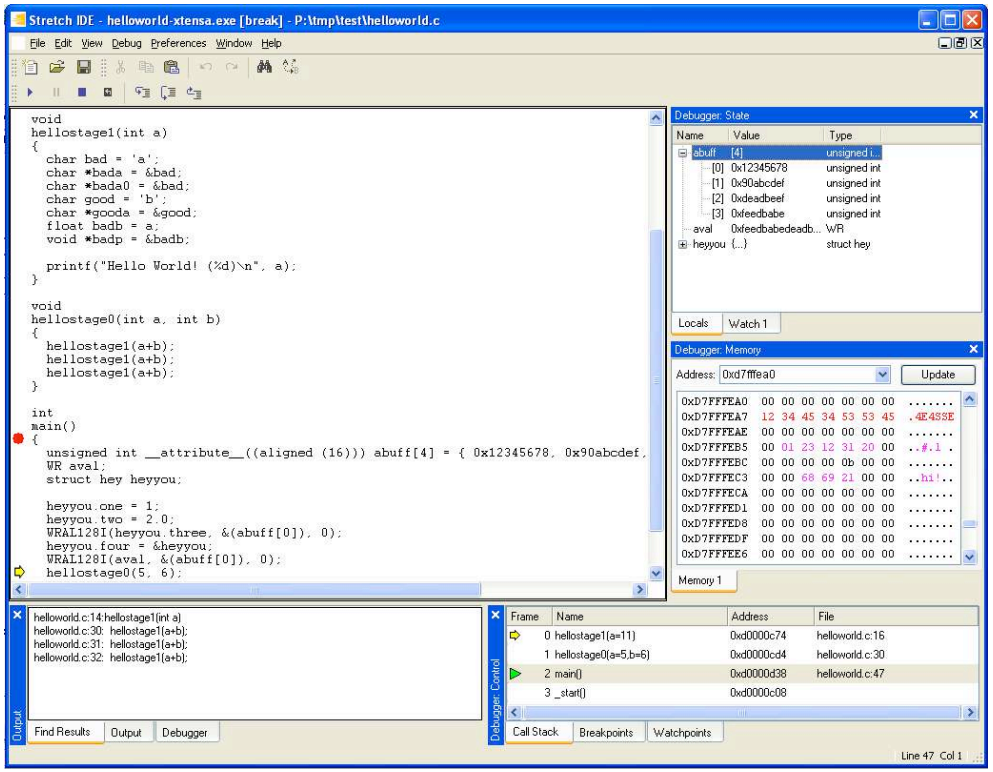


© 2005 Stretch Inc. All rights reserved.

Stretch

Development & Debug

- Programmers develop & debug using familiar tools
- Faster debug cycle
- Port & accelerate apps within Stretch IDE
- No hardware design experience required!



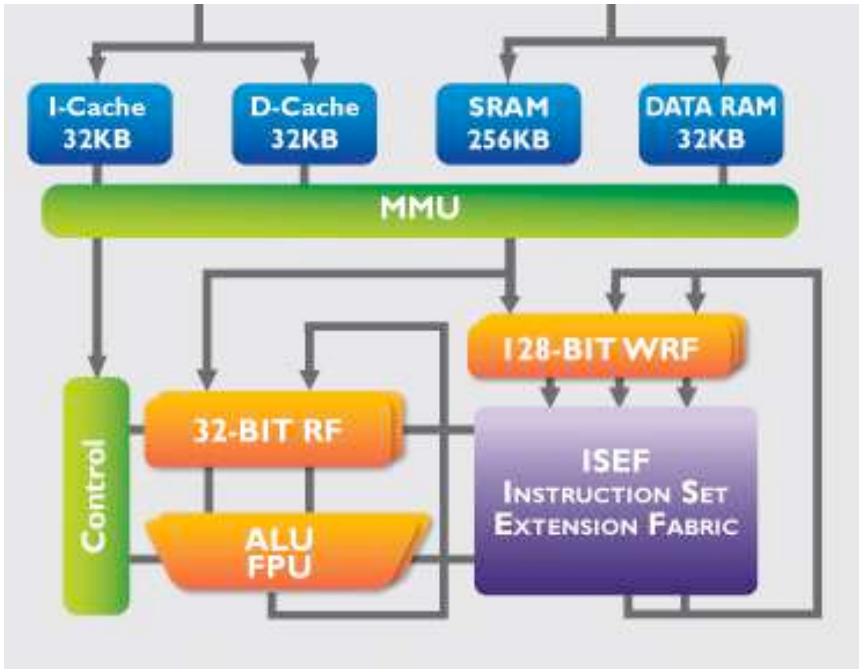
© 2005 Stretch Inc. All rights reserved.

Extending the Possibilities > www.stretchinc.com

Hardware Overview

© 2005 Stretch Inc. All rights reserved.

S5 Datapath and Key Parameters



Wide Register File
- 32 128-bit entries

Load/store unit
- 128-bit load/store
- Auto increment/decrement
- Immediate, indirect, circular
- Variable-byte load/store
- Variable-bit load/store

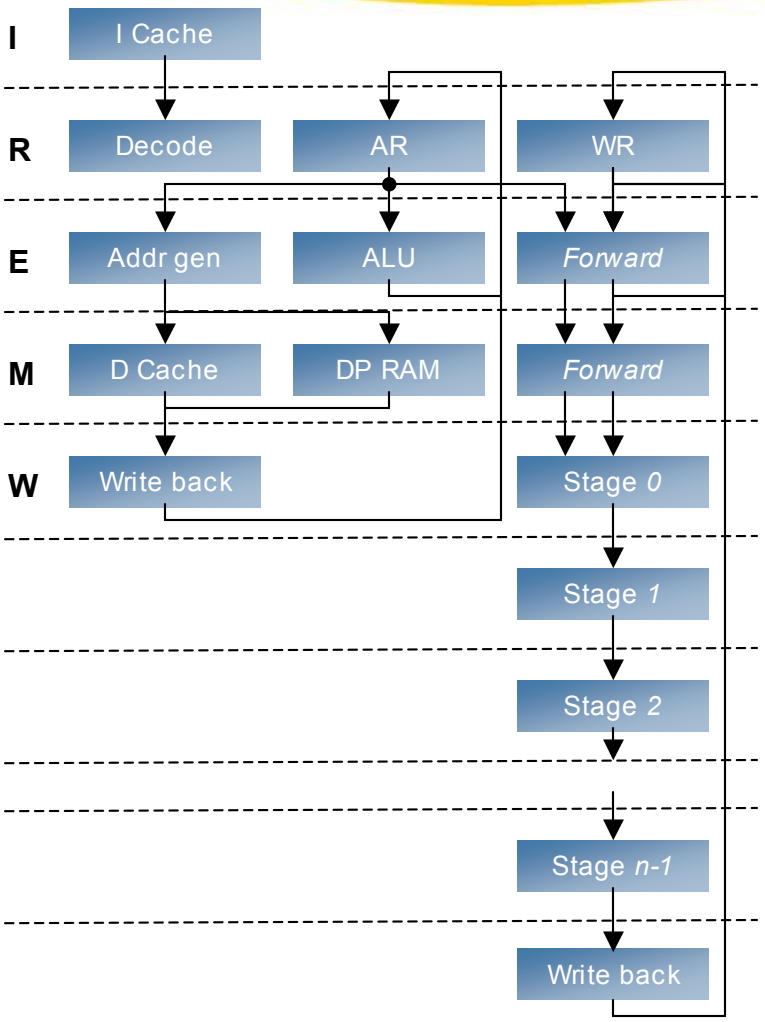
ISEF
- 3 inputs and 2 outputs
- Pipelined, interlocked
- 32 16-bit MACs and 256 ALUs
- Bit-sliced for arbitrary bit-width
- Allow control logic
- Allow internal processor state

RISC Processor
- Tensilica – Xtensa V
- 32 KB I & D Cache
- On-Chip Memory, MMU
- 24 Channels of DMA, FPU

© 2005 Stretch Inc. All rights reserved.

Stretch

S5 Pipeline

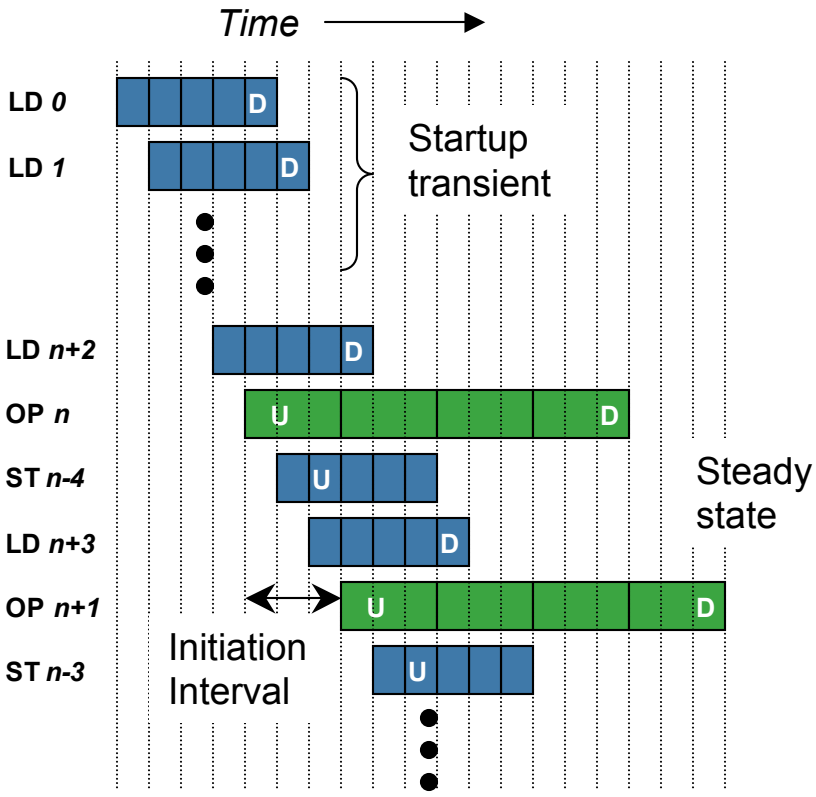


- Standard 5-stage RISC pipeline
- Extension instruction may be *much* longer
  - Fully pipelined
- Compiler schedules *all* instructions
- Hardware interlocks ensure correctness
  - Instruction groups subdivided by use/def requirements
  - Interlocks driven by config table in Instruction Unit

© 2005 Stretch Inc. All rights reserved.

Stretch

Pipeline Schedule



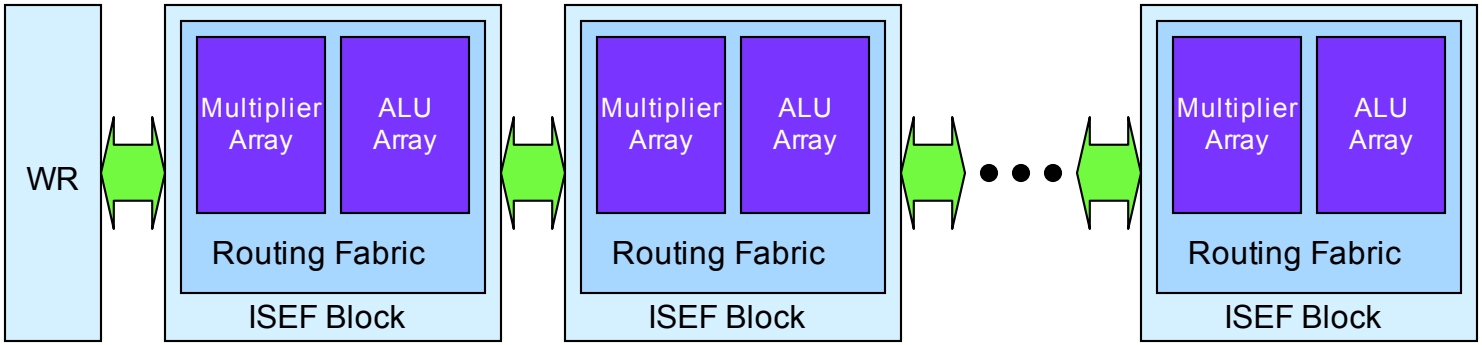
- Extension Instructions may have different latencies
- Initiation interval of EI a function of:
  - ISEF clock ratio
  - WR and state use/def
- Statically scheduled
  - Pipeline behavior is completely known at compile time
- Optimal schedule typically has initiation interval > 1
  - No performance penalty for lower clock rate!

© 2005 Stretch Inc. All rights reserved.

Stretch

ISEF Architecture

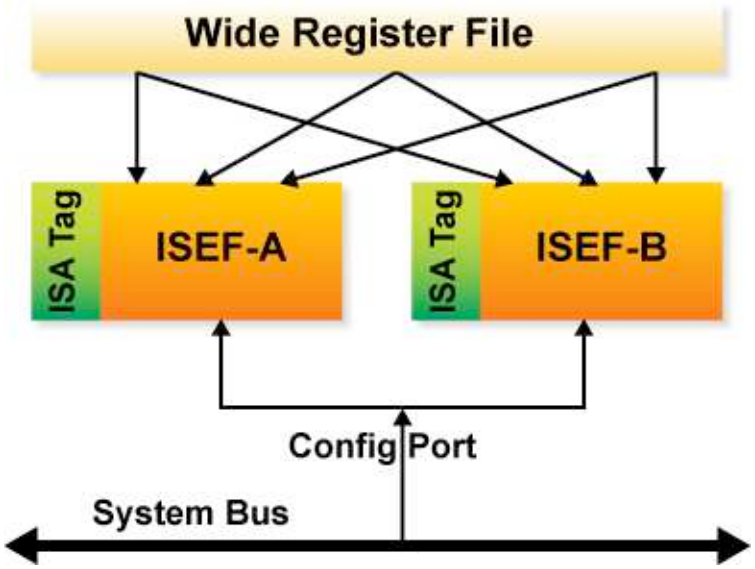
- Array of 64-bit ALUs
  - May be broken at 4-bit or 1-bit boundaries
  - Conditional ALU operation: `Y = C ? (A op1 B) : (A op2 B)`
  - Registers to implement state, pipelining
- Array of 4x8 multipliers, cascadable to 32x32
  - Programmable pipelining
- Programmable routing fabric
- Execution clock divided from processor clock: 1:1, 1:2, 1:3, 1:4
- Up to 16 instructions per ISEF configuration
- Many configurations per executable
- Reconfiguration
  - User directed or on demand
  - <100 µsec for complete reconfiguration



© 2005 Stretch Inc. All rights reserved.

Stretch

Dynamic ISEF Configuration



Two ISEFs
- Each holds up to 16 instructions
- Configured via system bus
- Managed by DMA
- Independently configurable

On-Demand Configuration
- Auto managed by HW and OS
- Handled like cache miss

Configuration Preloading
- Managed by application
- Handled like instruction pre-fetch

On-Demand Configuration

Configuration Preloading

© 2005 Stretch Inc. All rights reserved.

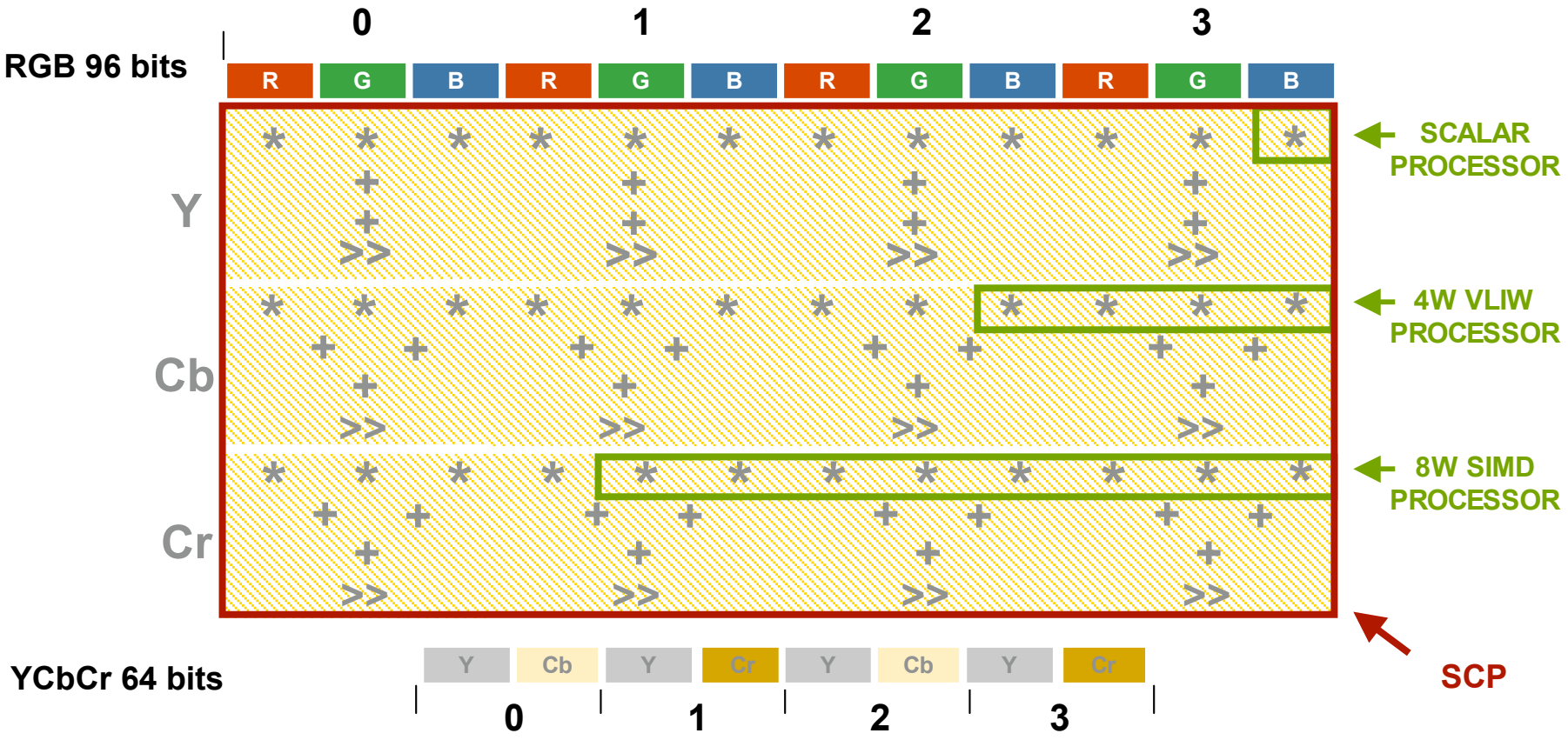
Stretch

Stretch

Examples of Extension Instructions

© 2005 Stretch Inc. All rights reserved.

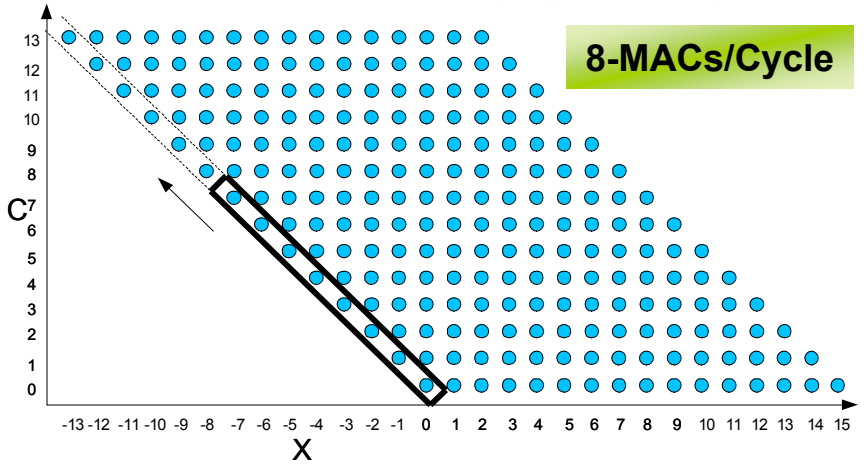
RGB2YCbCr: SCP Instruction Efficiency

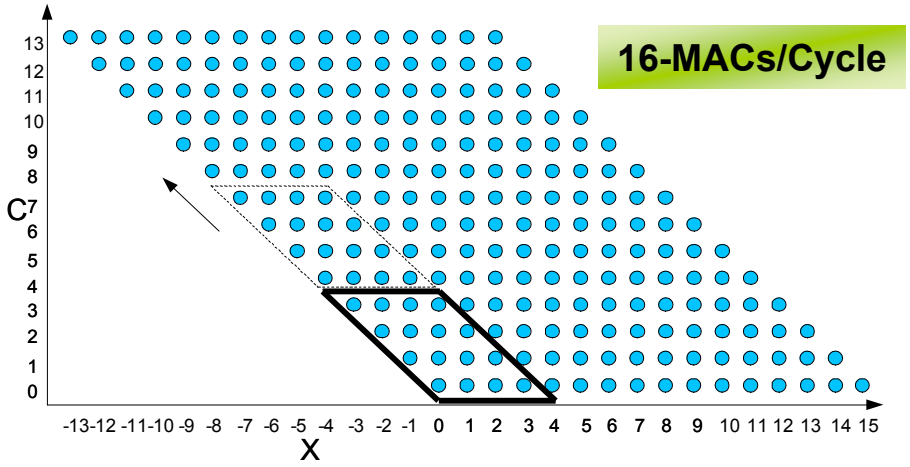


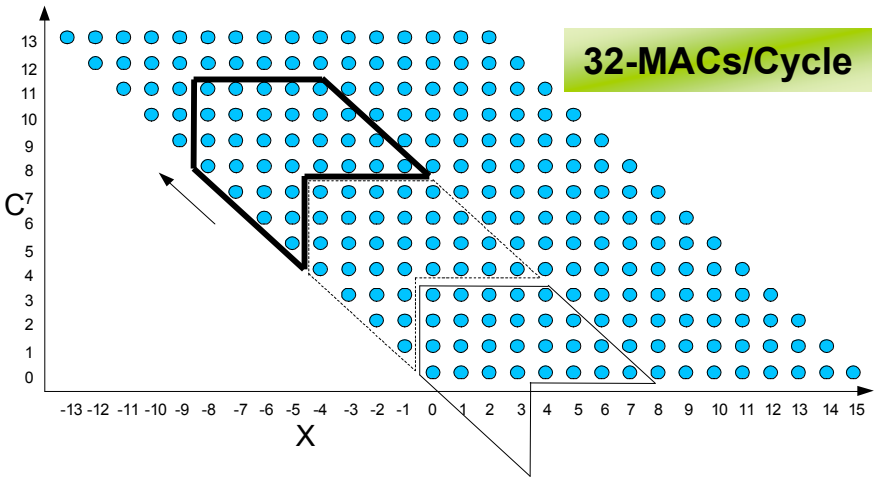
© 2005 Stretch Inc. All rights reserved.

Stretch

Flexibility in FIR Implementations







- S5000 Strengths
  - Up to 32-MAC/Cycle
  - Flexibility to trade-off ISEF usage vs. compute speed
  - Wide Load & Stores feed the compute array

© 2005 Stretch Inc. All rights reserved.

Stretch

Summary

- C/C++ Programmers can tailor the processor for their application
  - Continue to develop & debug in a familiar environment (IDE/gdb)
- Application-specific instruction provide 10-100X speedup
  - Modest porting effort
- C/C++ programmers can easily design high-performance systems
  - Enabling a new system design methodology

© 2005 Stretch Inc. All rights reserved.