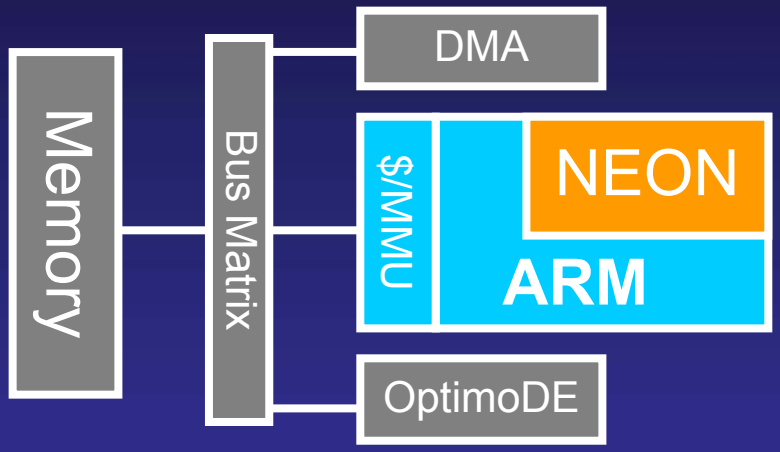
ARM NEON™ Technology

Simon Ford
Technical Lead

ARM NEON™ Technology

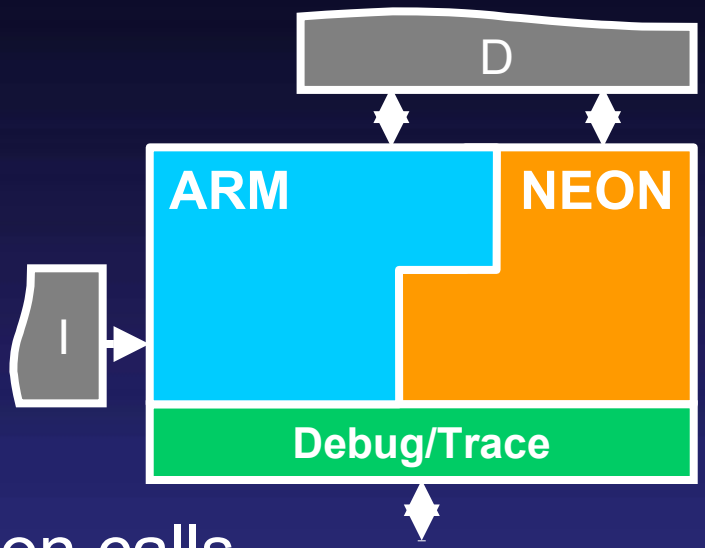
- **NEON** is a 64/128-bit hybrid SIMD architecture
  - Available in ARM and Thumb-2 Instruction Sets
  - Implemented as a tightly coupled DSP engine



- **NEON** can be built in to next generation cores
  - Applications Processors : 1GHz, Superscalar
  - Embedded Processors : 300MHz, Single Issue

Programmers Model

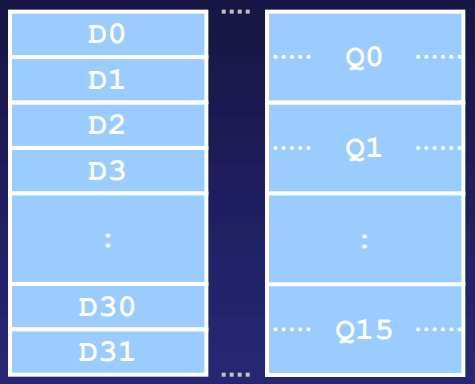
- Single instruction stream
- Single view of memory
- Single debug and trace



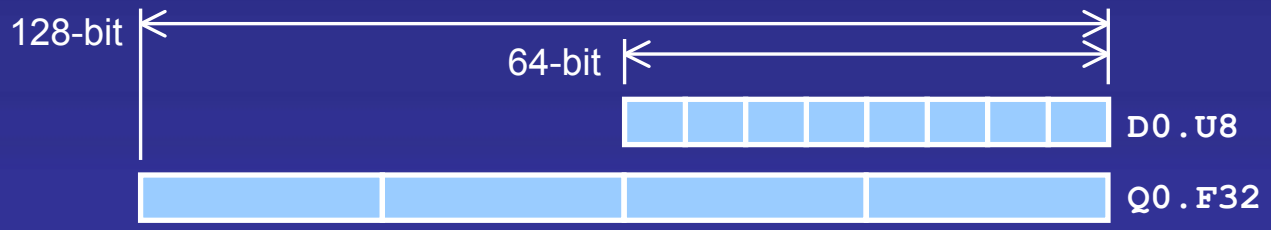
- **ARM** handles control plane
  - Loops, branches and function calls
  - Address calculation
  - Hardware optimised for tight control
- **NEON** handles data plane
  - Integer, Fixed-point and Floating-point DSP
  - Hardware optimised for high throughput

Register File

- A 256-byte register file distinct from the core
- Two explicitly aliased views
  - 32 x 64-bit registers (D0-D31)
  - 16 x 128-bit registers (Q0-Q15)
  - Vector length vs. entry trade-off

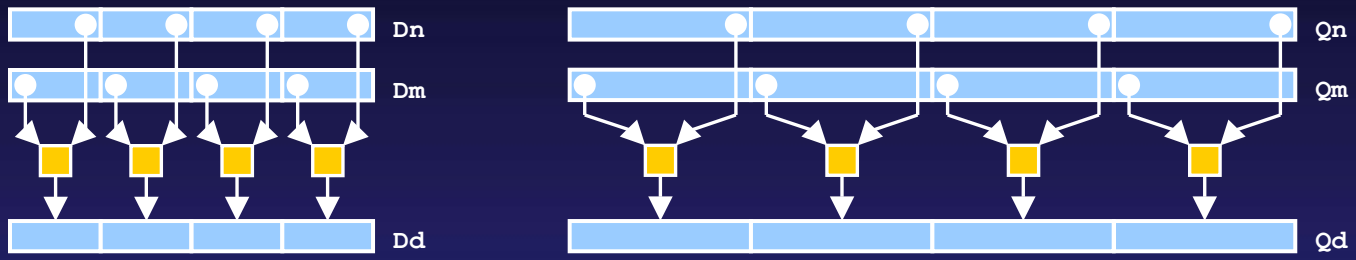


- Vector described by a `<register>.<type>` pair



Instruction Format

- Instructions perform same operation in parallel



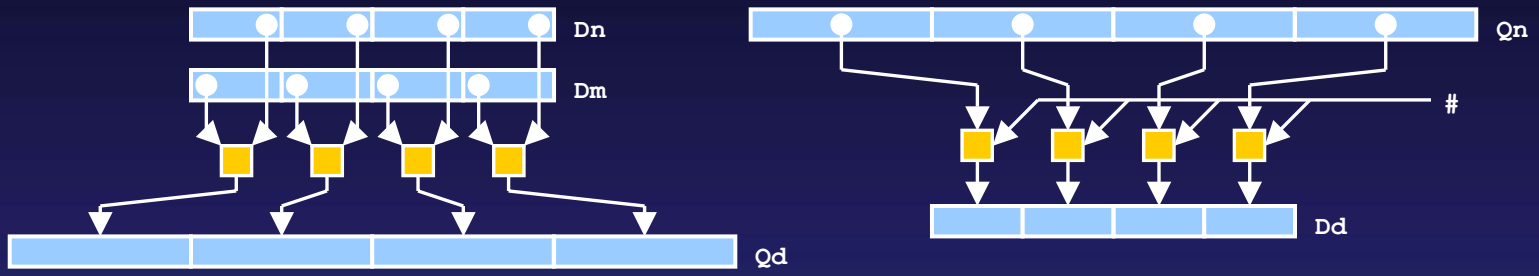
- Orthogonal instruction format and options
  - Consistent framework for compilers

```
V{<mod>}<op>.<type>   <Vd>, <Vn>, <Vm>
```

Add		Shift Right	
VADD.I32	D0, D1, D2	VSHR.U16	D0, D1, #5
VADD.F32	Q0, Q1, Q2	VSHR.S16	D0, D1, #1
VQADD.S16	Q0, Q1, Q2	VRSHR.S32	Q0, Q1, #7

Instruction Format (2)

- Promotion and demotion included in operations



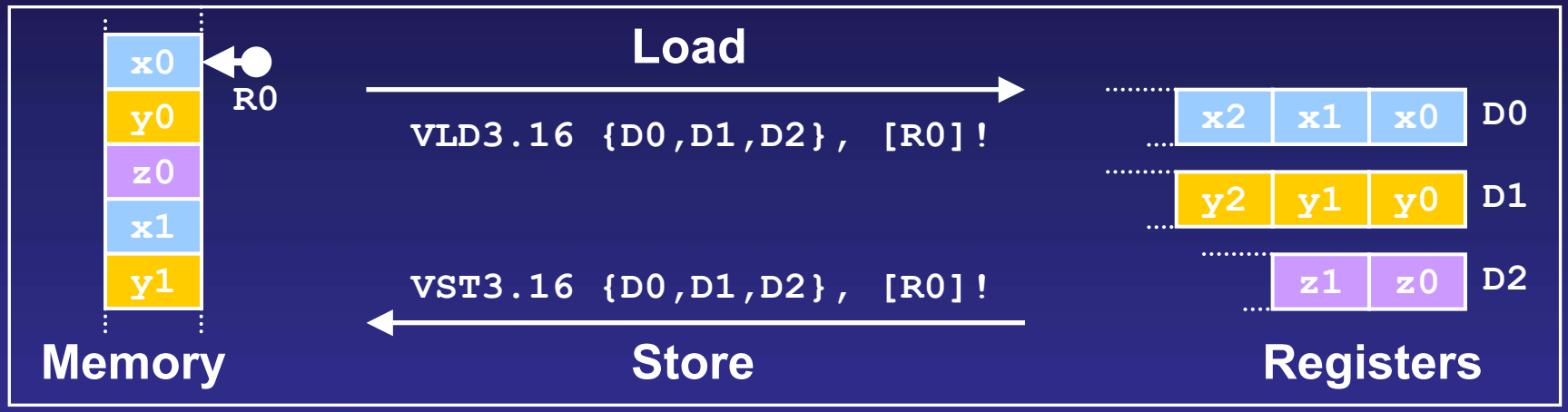
- Enables efficient use of register file
- Eliminates packing and unpacking overhead

```
V{<mod>}<op>.<type>   <Vd>, <Vn>, <Vm>
```

Long Multiply		Narrowing Shift Right	
VMUL.I32.S16	Q0, D2, D3	VSHR.I8.I16	D0, Q1, #5
VMUL.I16.U8	Q0, D2, D3	VQSHR.S16.S32	D0, Q1, #1
VQDMUL.S32.S16	Q0, D2, D3	VRSHR.I16.I32	D0, Q1, #7

Memory Access

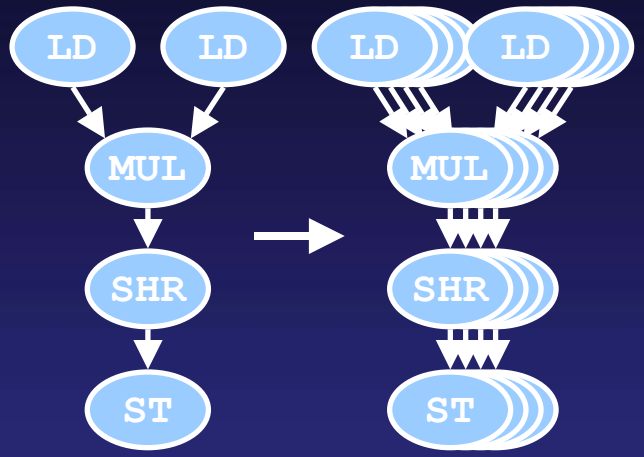
- Native support for structures
  - e.g. Complex Numbers, Pixels, Co-ordinates
  - Memory treated as an Array of Structures



- Eliminates overhead
  - Memory accessed as single block transfer
  - Data arranged for efficient SIMD processing

Compiler

- **NEON** provides a consistent algorithm mapping
  - Apply narrowing analysis
  - Vectorise over loop iterations



- Enabled by architectural model
  - Orthogonal instruction framework
  - Few inter-lane operations
  - Support for type promotion and demotion in operations

- Designed in conjunction with compiler technology
  - Ensure architecture optimised for this compiled mode
  - Benefits of CSE, unrolling, scheduling, register allocation
  - Portable solutions by avoiding hand coding or intrinsics

Code Example

- Colour space conversion
  - 3 and 4 element structures

```
struct rgb {
 u8 r, g, b;
};
struct cmyk {
 u8 c, m, y, k;
};
```

```
void rgb2cmyk(cmyk* out, rgb* in, int n) {
 for(int i=0 ; i<n ; i++ ) {
  u8 r = in[i].r;     /* red   */
  u8 g = in[i].g;     /* green */
  u8 b = in[i].b;     /* blue  */

  u8 k = r;
  if (k<g) k = g;
  if (k<b) k = b;

  out[i].c = k - r;   /* cyan    */
  out[i].m = k - g;   /* magenta */
  out[i].y = k - b;   /* yellow  */
  out[i].k = ~k;      /* black   */
 }
}
```

```
#define r D0.U8
#define g D1.U8
#define b D2.U8
#define c D3.U8
#define m D4.U8
#define y D5.U8
#define k D6.U8

#define out R0
#define in  R1
#de
```

```
loop:
  VLD3 {r, g, b}, [in]!
  VMAX k, r, g
  VMAX k, k, b
  VSUB c, k, r
  VSUB m, k, g
  VSUB y, k, b
  VNOT k, k
  VST4 {c, m, y, k}, [out]!
  SUBS n, n, #8
  BGT  loop
```

Code Example (2)

- Complex fractional multiply
  - Adds promotion/demotion

```
struct c16 {
 s16 re, im;
};
#define SCALE 12
```

```
void mul(c16* z, c16* a, c16* b, int n) {
 for(int i=0 ; i<n ; i++ ) {
  s16 a_re = a[i].re;
  s16 a_im = a[i].im;
  s16 b_re = b[i].re;
  s16 b_im = b[i].im;

  s32 t_re = a_re * b_re - a_im * b_im;
  s32 t_im = a_re * b_im + a_im * b_re;

  s16 z_re = t_re >> SCALE;
  s16 z_im = t_im >> SCALE;

  z[i].re = z_re;
  z[i].im = z_im;
 }
}
```

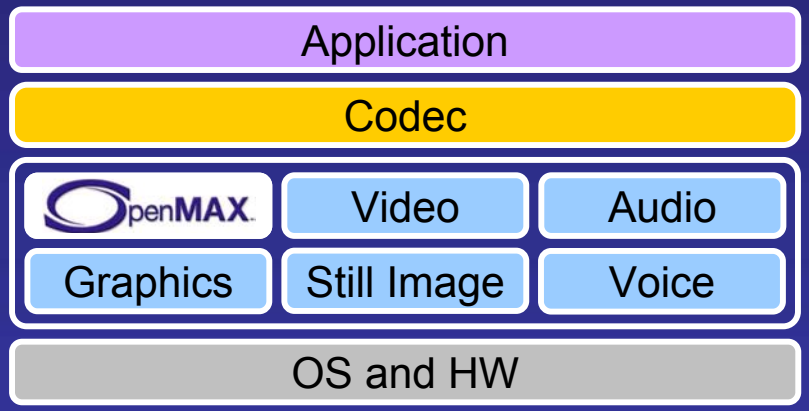
```
#define a_re D0.S16
#define a_im D1.S16
#define b_re D2.S16
#define b_im D3.S16
#define z_re D4.S16
#define z_im D5.S16
#define t_re Q10.S32
#define t_im Q11.S32

#define z R0
#define a R1
#de
#de
```

```
loop:
  VLD2 {a_re, a_im}, [a]!
  VLD2 {b_re, b_im}, [b]!
  VMUL t_re, a_re, b_re
  VMLS t_re, a_im, b_im
  VMUL t_im, a_re, b_im
  VMLA t_im, a_im, b_re
  VSHR z_re, t_re, #SCALE
  VSHR z_im, t_im, #SCALE
  VST2 {z_re, z_im}, [z]!
  SUBS n, n, #4
  BGT  loop
```

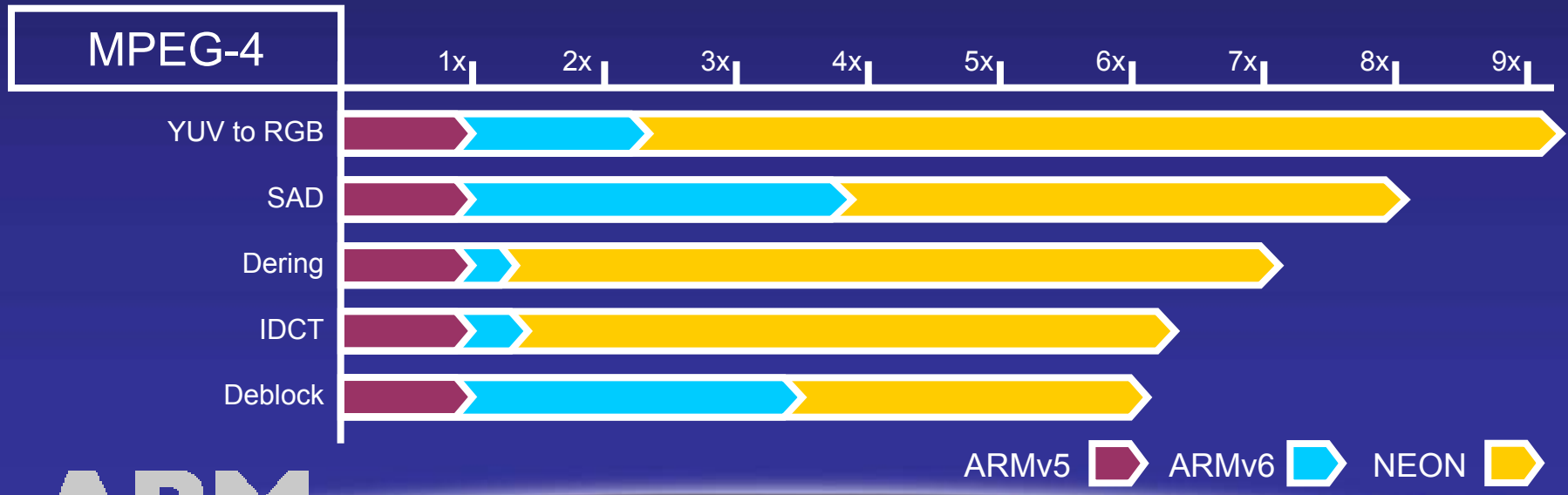
OpenMAX™ Libraries

- Target **NEON** using media kernel libraries
- **OpenMAX** is an open, non-proprietary API
  - Defined within Khronos group (like OpenGL-ES)
  - Focused on key 'hotspot' functions and kernels
  - Off-the-shelf, portable and verified solutions
- Retarget using library
  - **NEON**
  - ARMv6
  - OptimoDE
  - Other HW/DSP

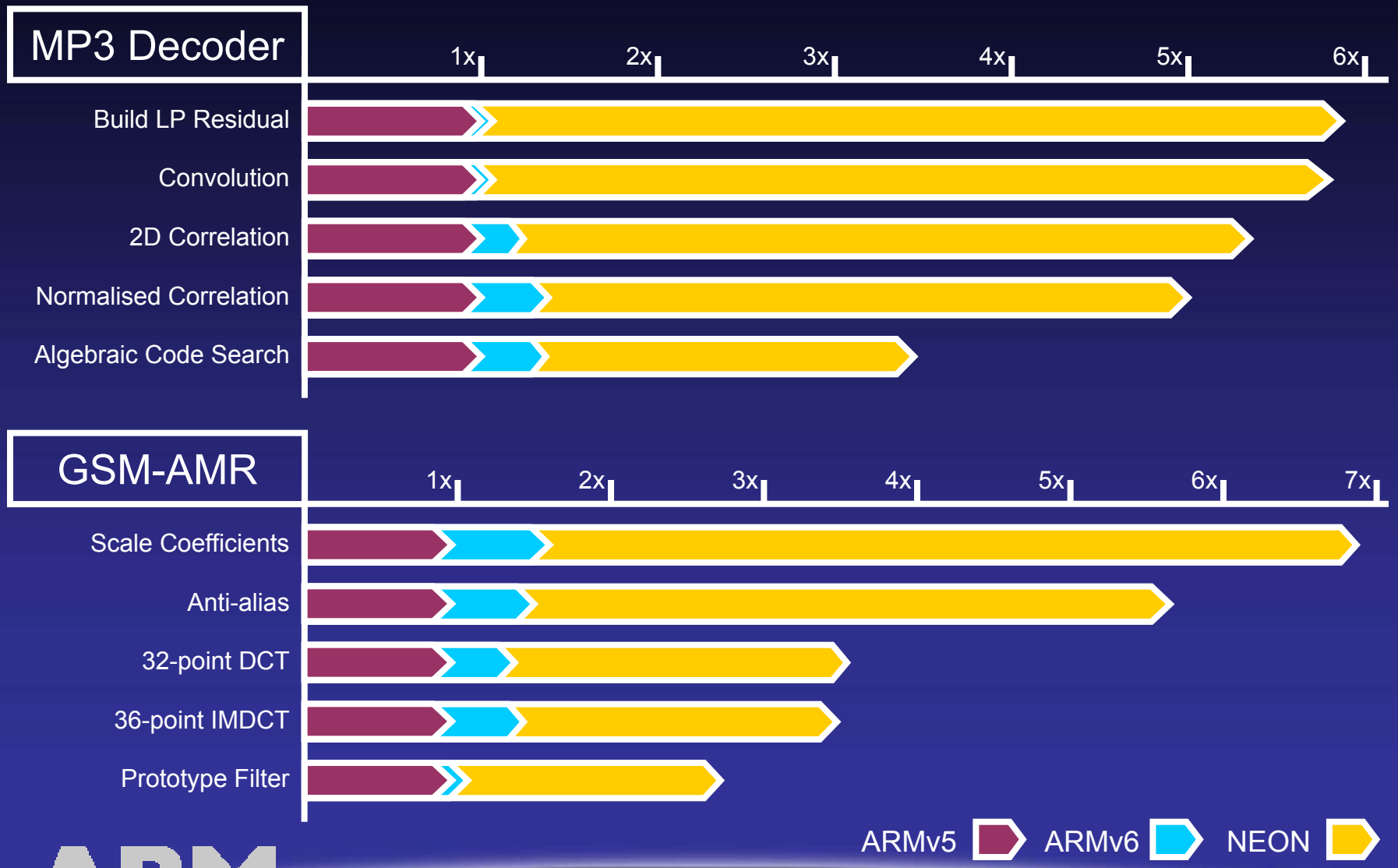


Kernel Performance

- **NEON** accelerates a wide range of kernels
  - Audio, Video, 3D Graphics, Speech, Still Image
- Figures based on future applications processor
  - Hand coded kernels, instructions/data in cache



Kernel Performance (2)



Real System Performance

- Translates to real application performance

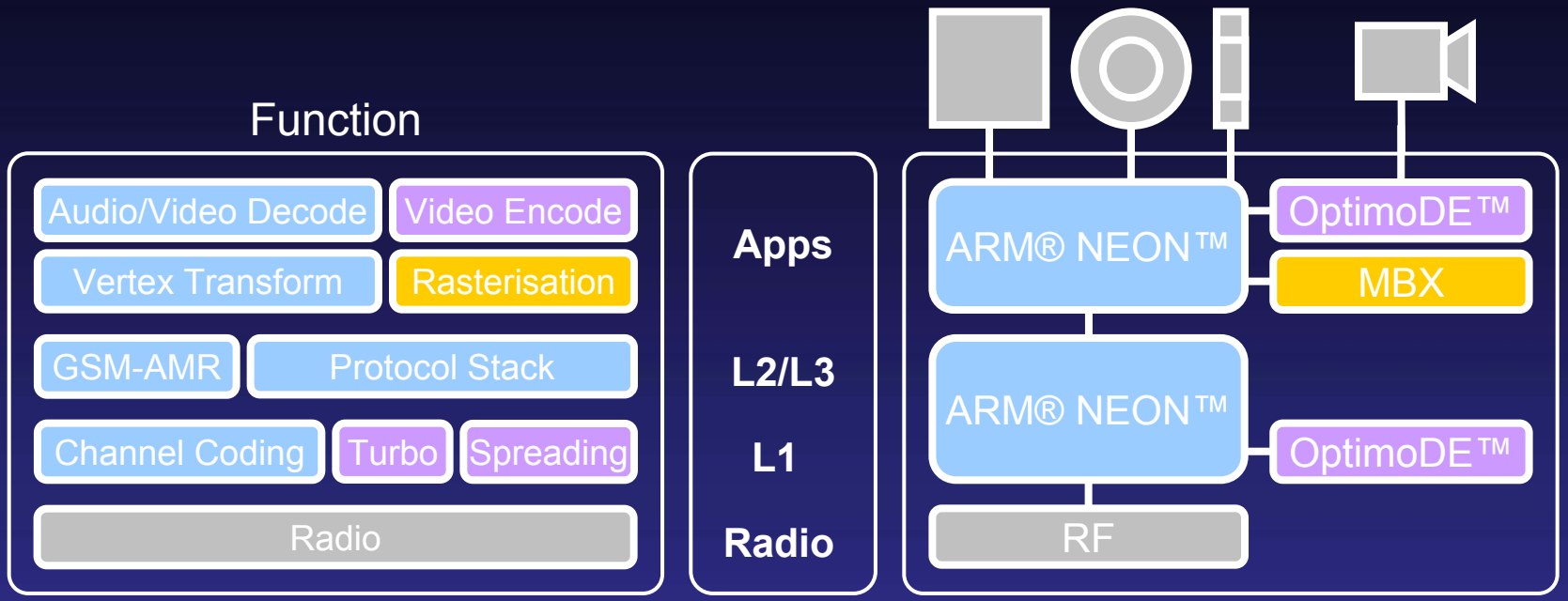
MPEG-4 Full Duplex CIF @ 30fps[1]	225 MHz
GSM-AMR (worst case)[2]	13 MHz
MP3 Decoder 320kbps 48kHz[3]	9.4 MHz

1) MPEG-4 Simple Profile @ 30fps 515kbps CIF, 133MHz SDRAM 10-1-1-1-1-1-1-1 memory

2) GSM-AMR (worst case), 3 cycle per word memory

3) MP3 Decoder @ 320kbps 48kHz (worst case), 133MHz SDRAM 10-1-1-1-1-1-1-1 memory

NEON Enabled Platform



- Example: Flexible Smartphone Platform
  - **NEON** enabled applications processor
  - **NEON** enabled baseband processor
  - OptimoDE accelerators

ARM NEON™ Technology

- A general purpose programmable DSP solution
  - Supports efficient data access and manipulation
  - Leverages the ARM core infrastructure
- Simple to target
  - Single instruction stream and view of memory
  - Single core debug and trace
- Simple to reuse
  - Enables fast reactions to changes in application
  - Compiler avoids intrinsics or hand coding

ARM NEON™ Technology

provides the basis for a high-performance, standardized and flexible media and DSP platform