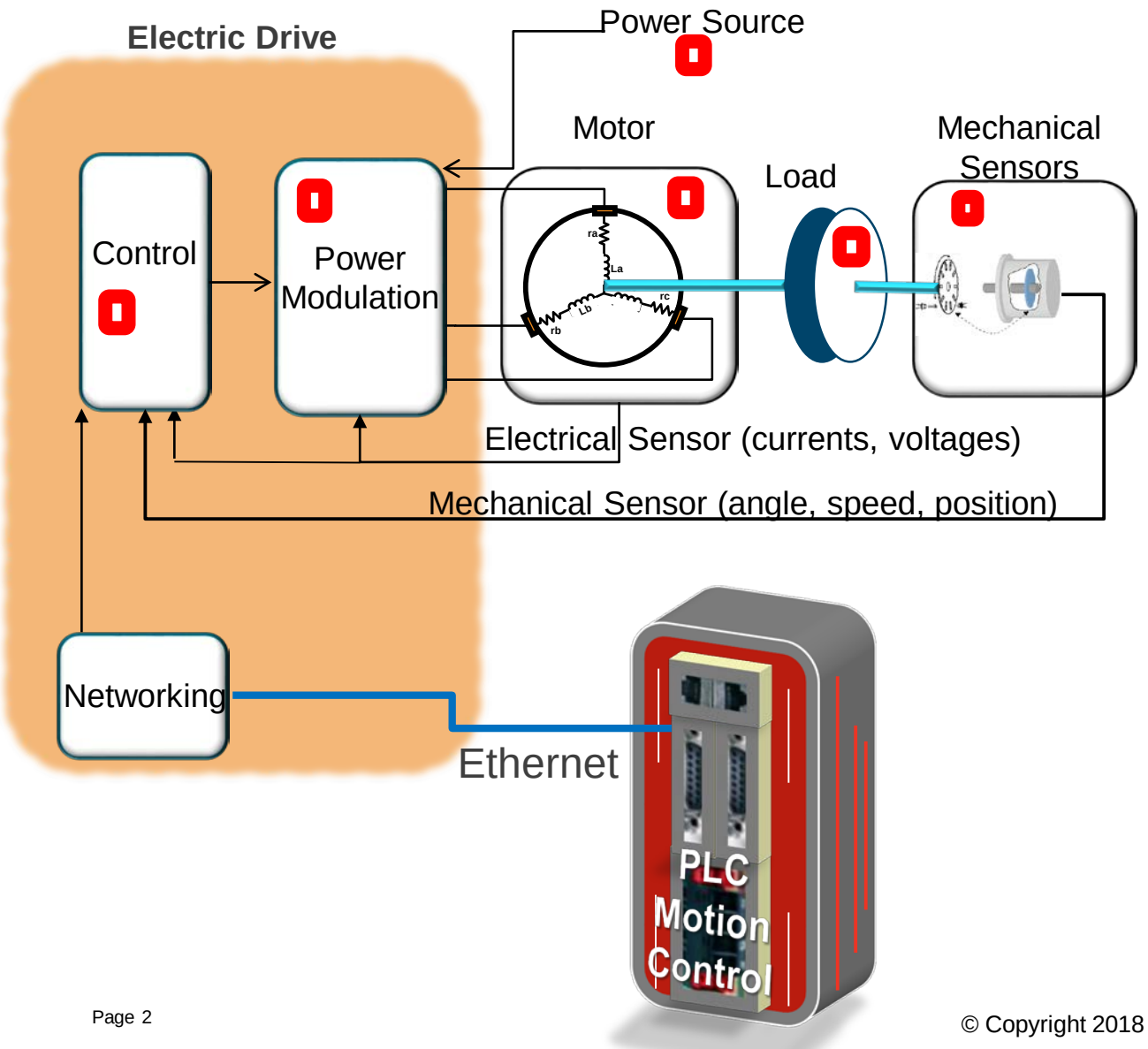
XILINX

Scalable IIoT Platforms
from Edge to Cloud

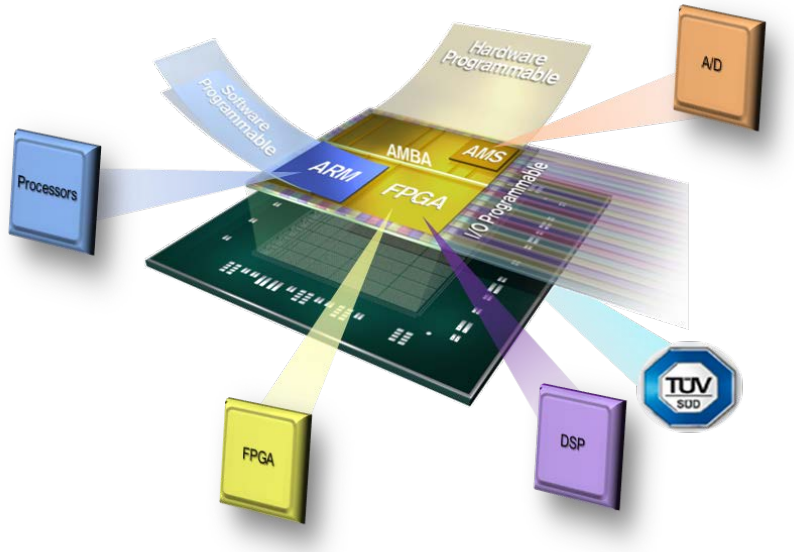
Getting Started with the Electric Drives Demo Platform

Module 1: Introduction to Modern Electric Drives

The Electric Drive



Zynq-7000 All Programmable SoC



© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

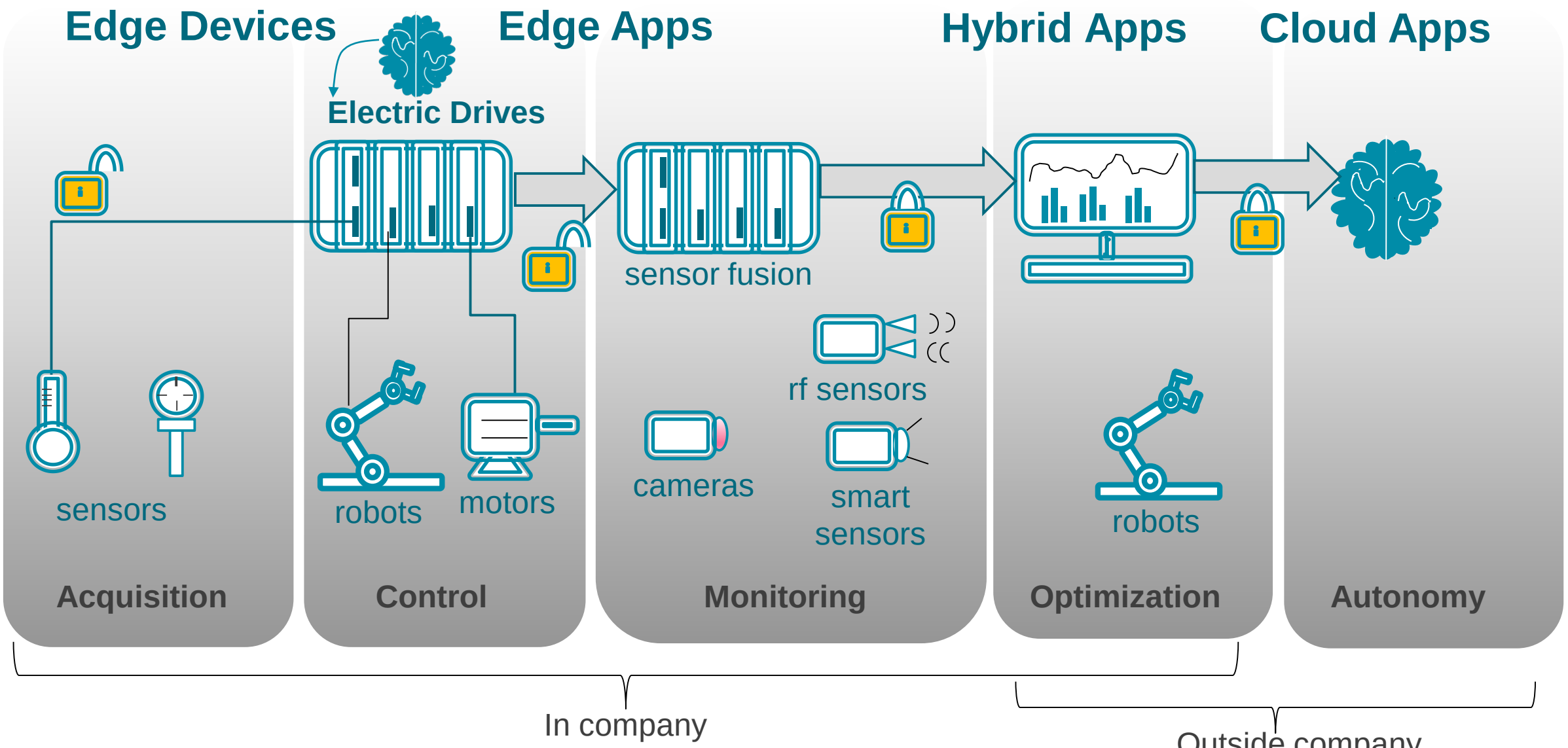
Industrial IoT / Industrie 4.0 Electric Drive

- Need better diagnostic
- Improved fault resilience
- Real-time optimization
- IoT enabled
- Real Time data gathering
- Safety
- Security
- Fast development time

Major requirements for Industrie 4.0 Electric Drives

© Copyright 2018 Xilinx

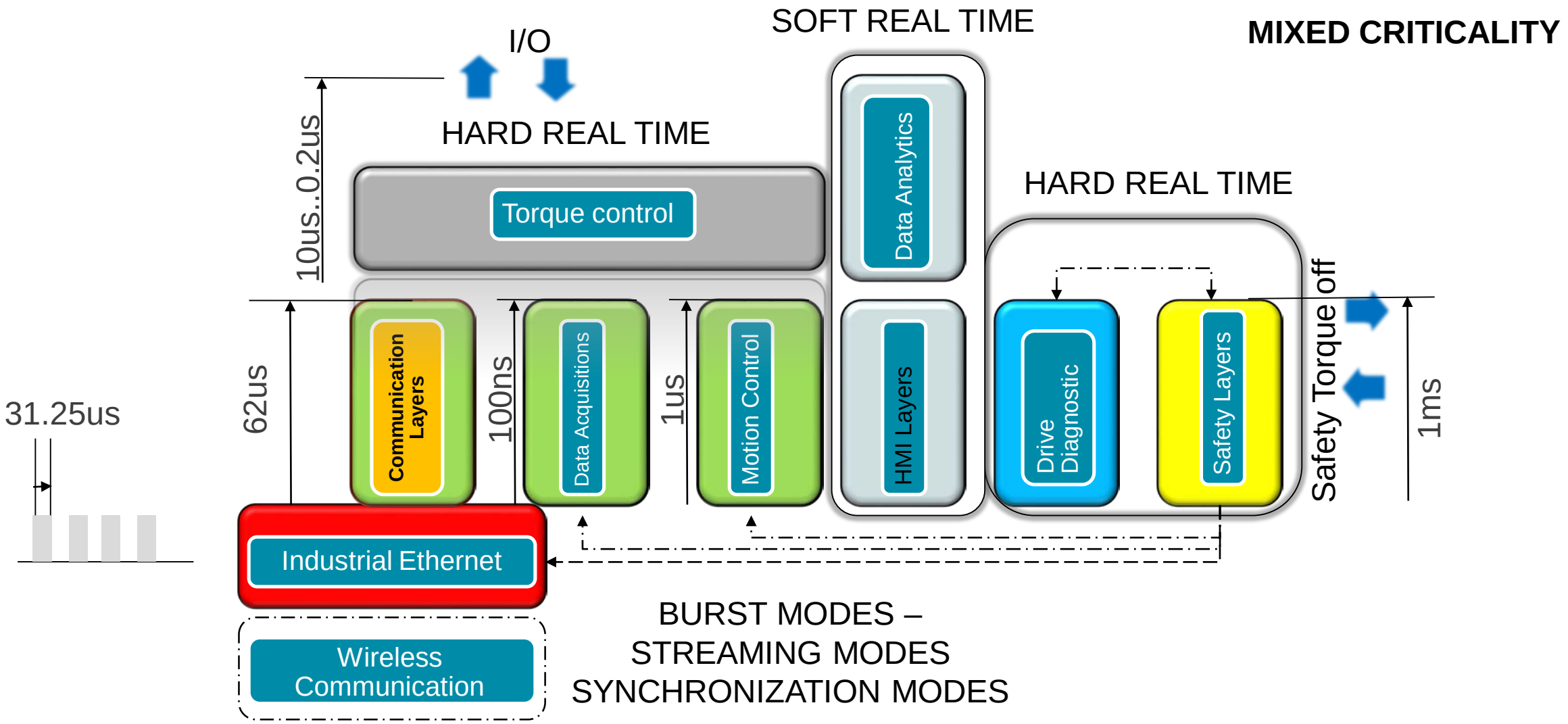
Where Electric Drives sit in Industrial Internet of Things



© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

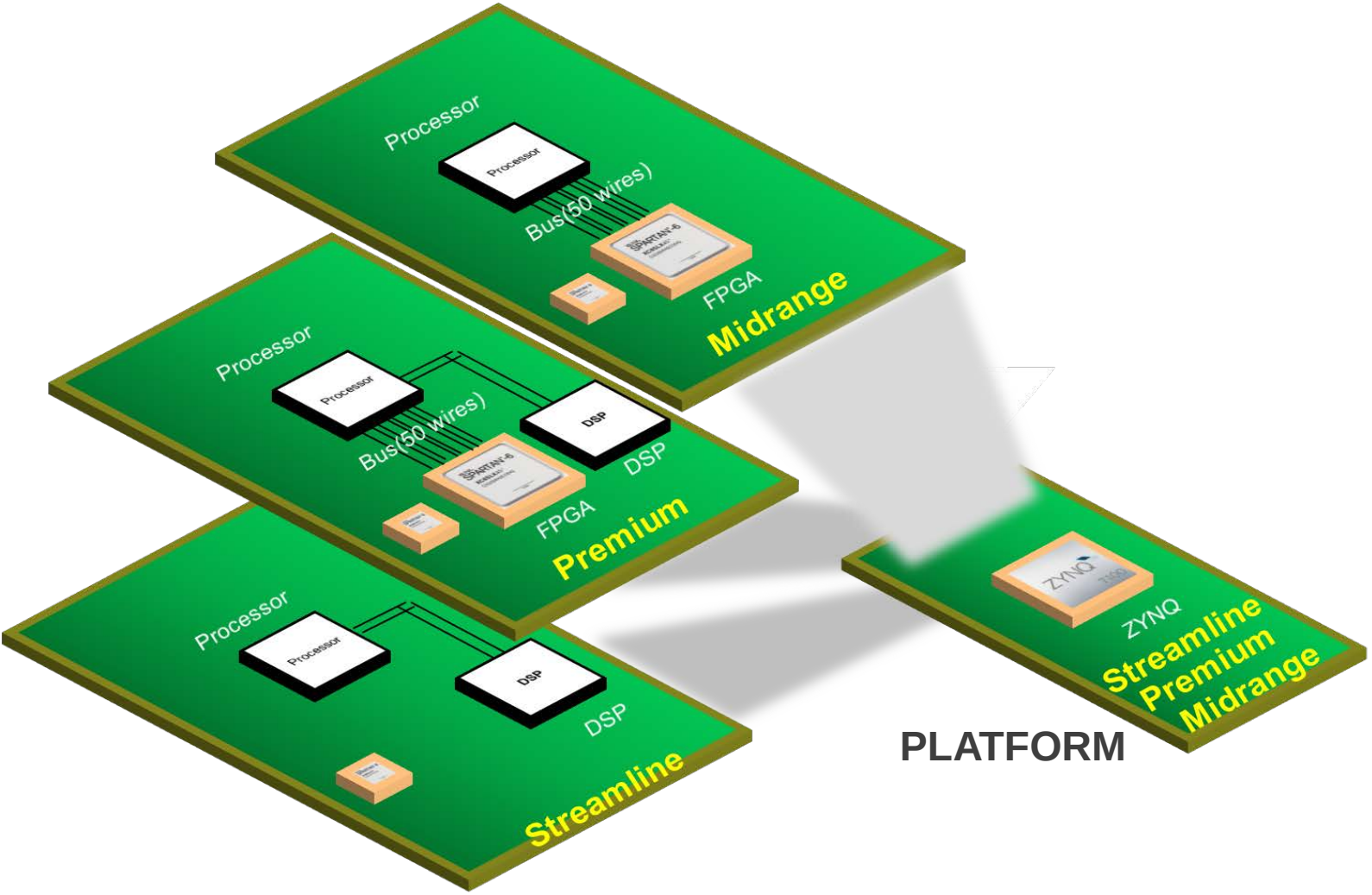
Use Case – internals of an Industry 4.0 Electric Drive



© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

Electric Drive in the next 3 to 5 years – The platform



- **1 PRODUCT PLATFORM**
- **1 PRODUCT TEMPLATE**
- **SMALLER SYSTEM**
  - Less weight
  - Less space
  - More Power Density
- **SCALABLE**
  - Streamline
  - Midrange
  - Premium
- **MULTI ARCHITECTURES**
- **SAFETY**
- **SECURITY**

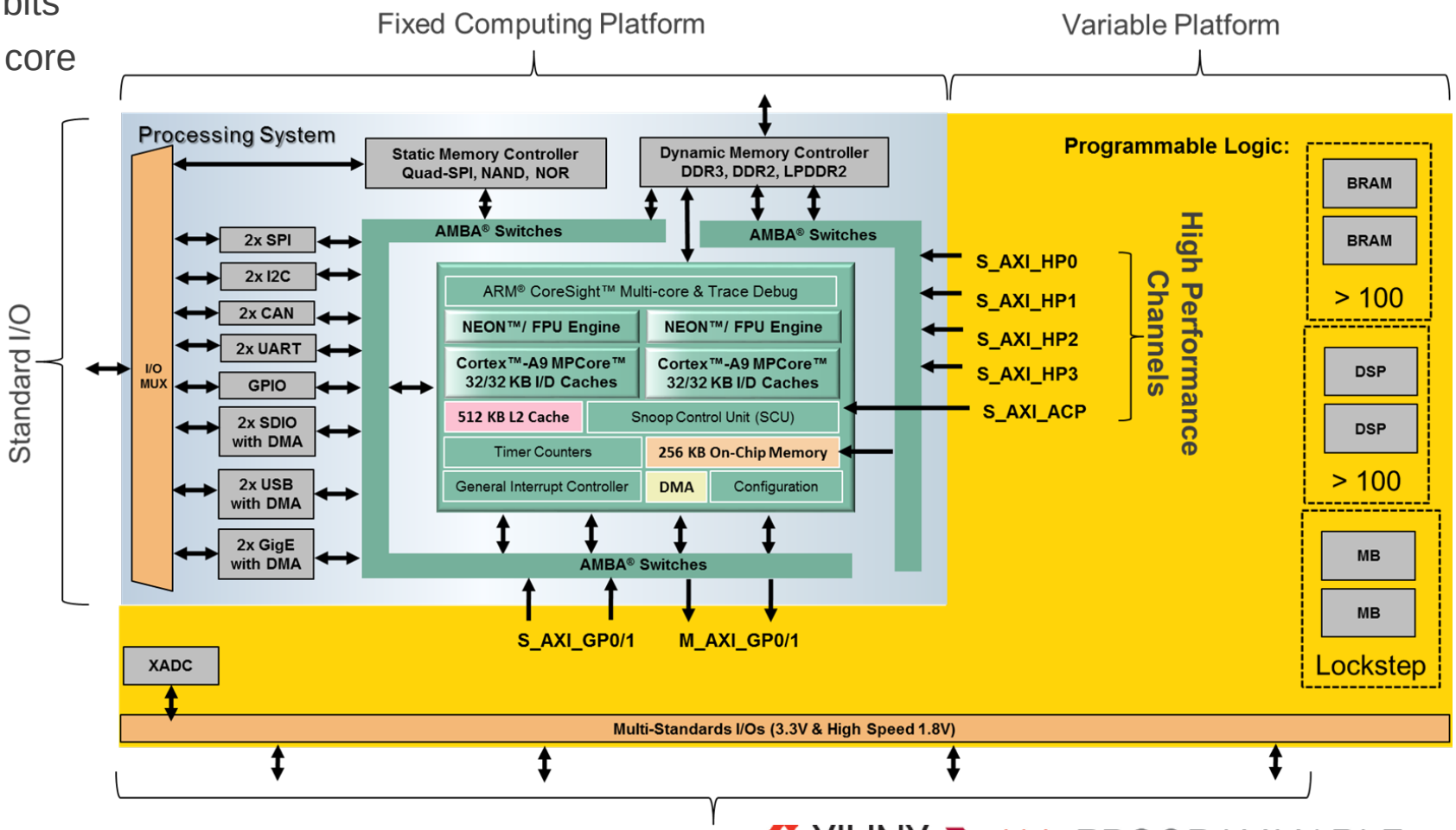
© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

The platform scalability

Platform

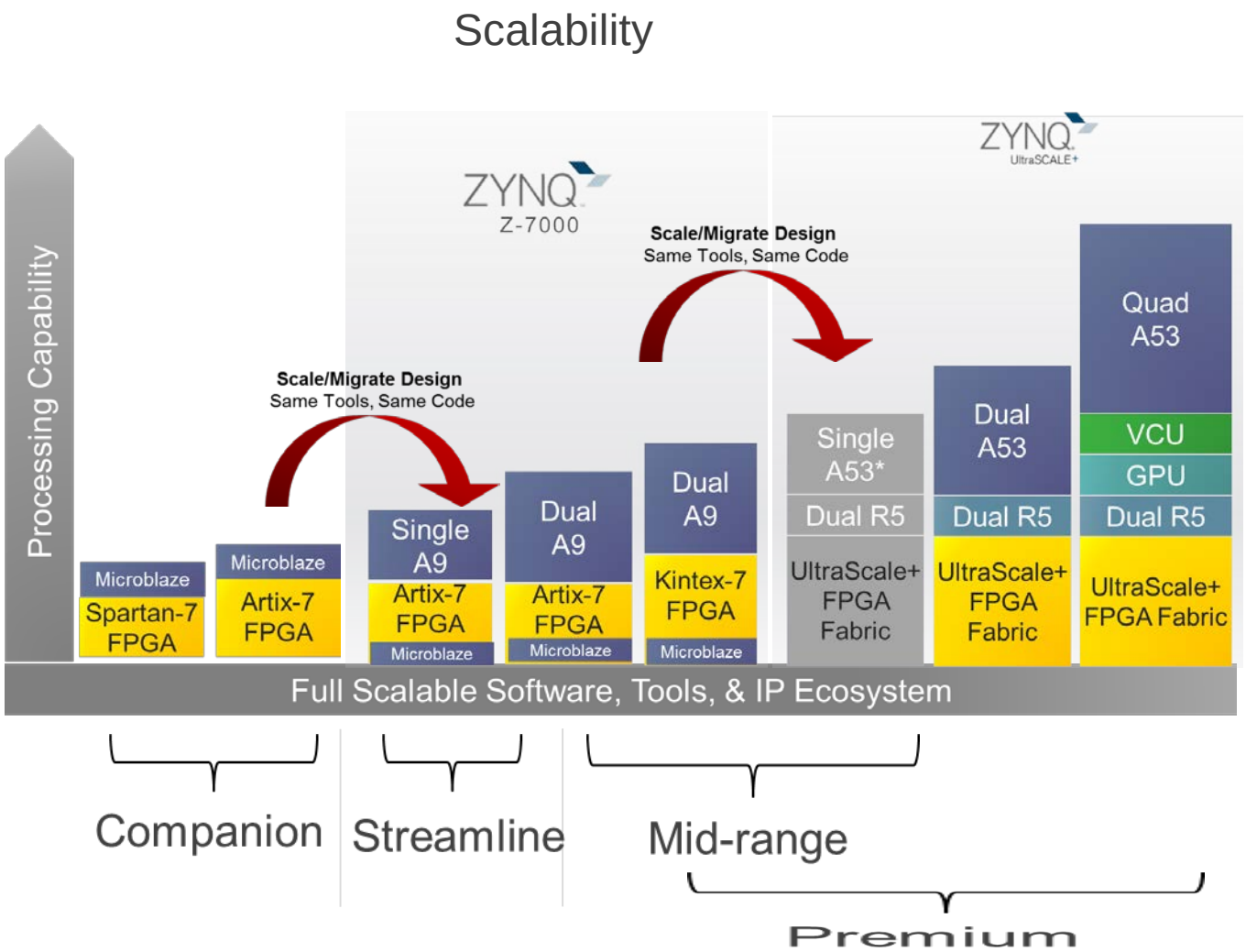
- 1-2 Cortex A9
- >100 DSP MAC
- > 100 True DPR 4 KB x 8 bits
- Dual Core Lock-step soft core 32bit like ARM M4



© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

The platform scalability



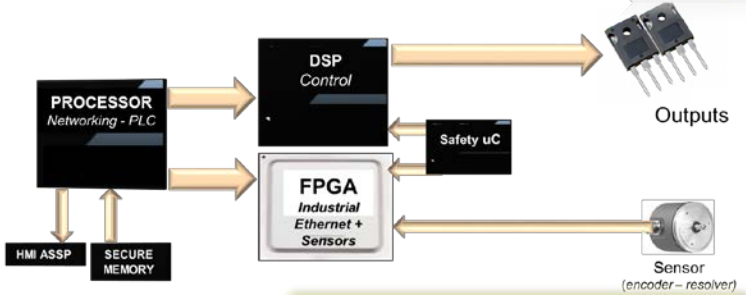
© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

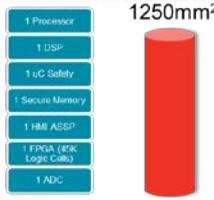
Challenges

1st CHALLENGE: *INTEGRATION*

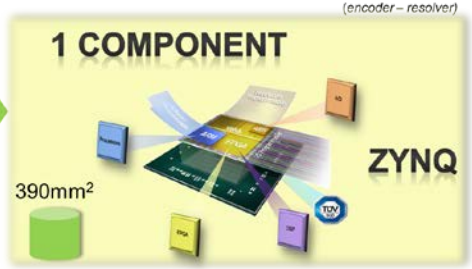
Solved ✓



6 COMPONENTS

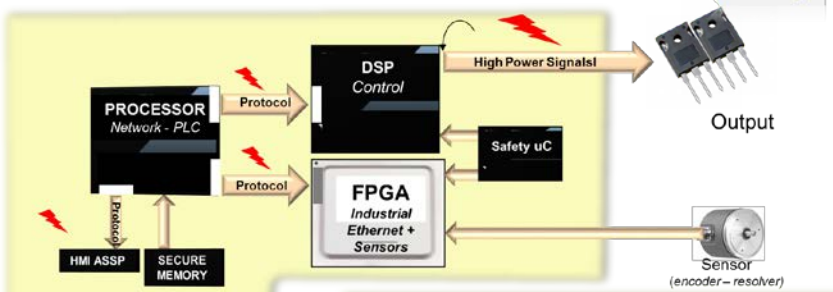


1 COMPONENT



2nd CHALLENGE: *COMPLEXITY & CHIP2CHIP RISKS*

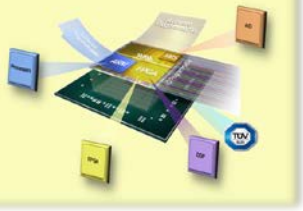
Solved ✓



3 PROTOCOLS
6 SW DRIVERS
3 EMI RISKS
4 SW COMPILERS
6 DEVELOPMENT

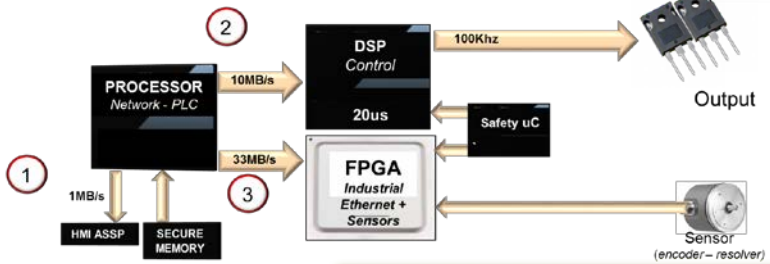
400% better

1 PROTOCOL
3 DRIVERS
0 EMI RISK
1 SW COMPILER
1 DEVELOPMENT



3rd CHALLENGE: *PERFORMANCES*

Solved ✓



200 uS DSP loop
33 MB/s to FPGA
10 MB/s to DSP
1 MB/s to HMI
200 us Ethernet Latency

~2 uS DSP control loop
800 MB/s all communications
0 uS Ethernet Latency
Accelerated Control

50x Faster

C/C++

4th CHALLENGE: *NETWORKING*

Solved ✓

TSN

- Networked units by IP Cores
- 8 industry standards on Industrial Ethernet
- 3 industry standard for high availability
- High variability of demand
- Need to tailor for specific world region

PROFINET | EtherNet/IP | EtherCAT | ETHERNET POWERLINK | Modbus | sercos the automation bus | CC-Link IE | MECHATROLINK

© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

Thank you for listening!!

Check out the next module to learn more

Getting Started with the Electric Drives Demo Platform

Module 2: Overview of the EDDP Hardware and Design Flows

© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

XILINX

Scalable IIoT Platforms
from Edge to Cloud

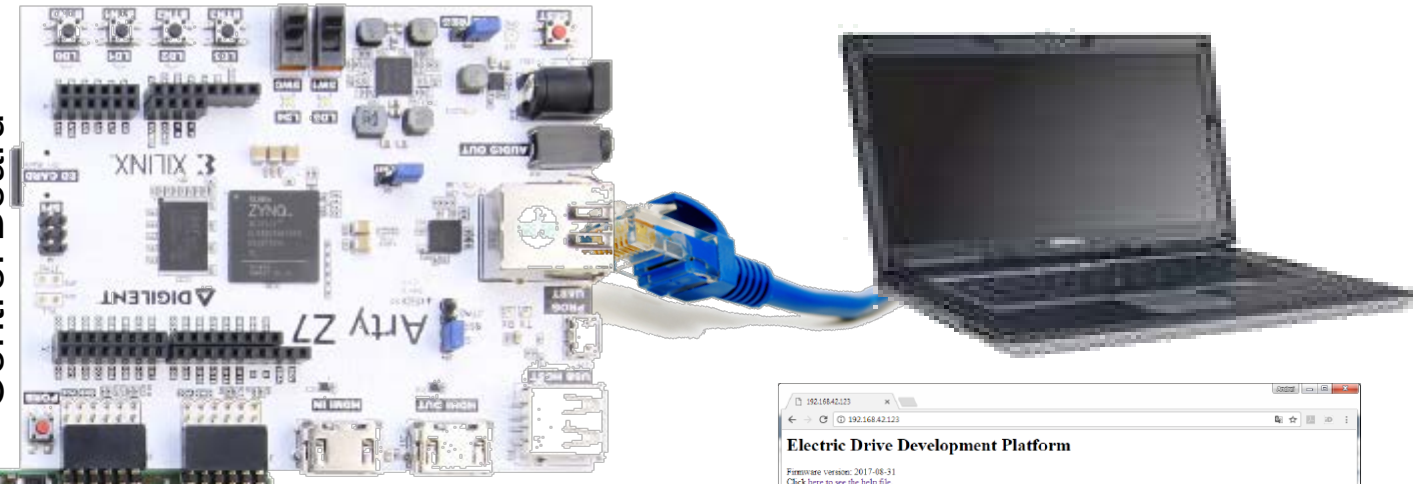
Getting Started with the Electric Drives Demo Platform

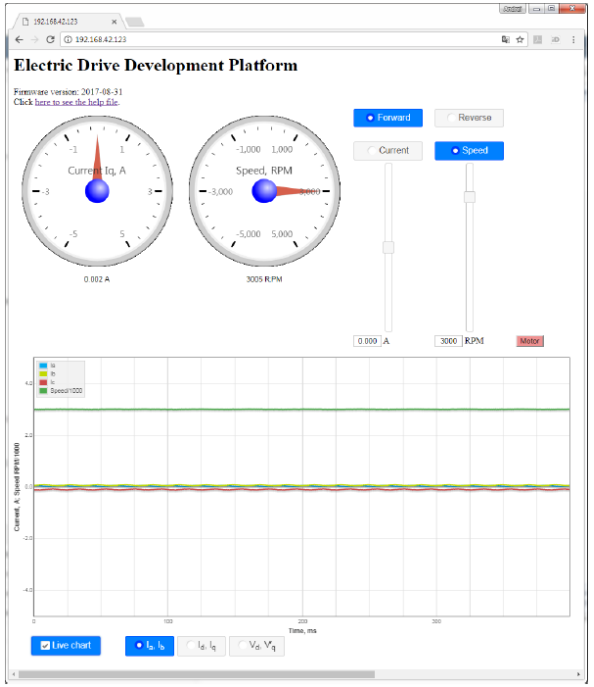
Module 2: Overview of the EDDP Hardware and Design Flows

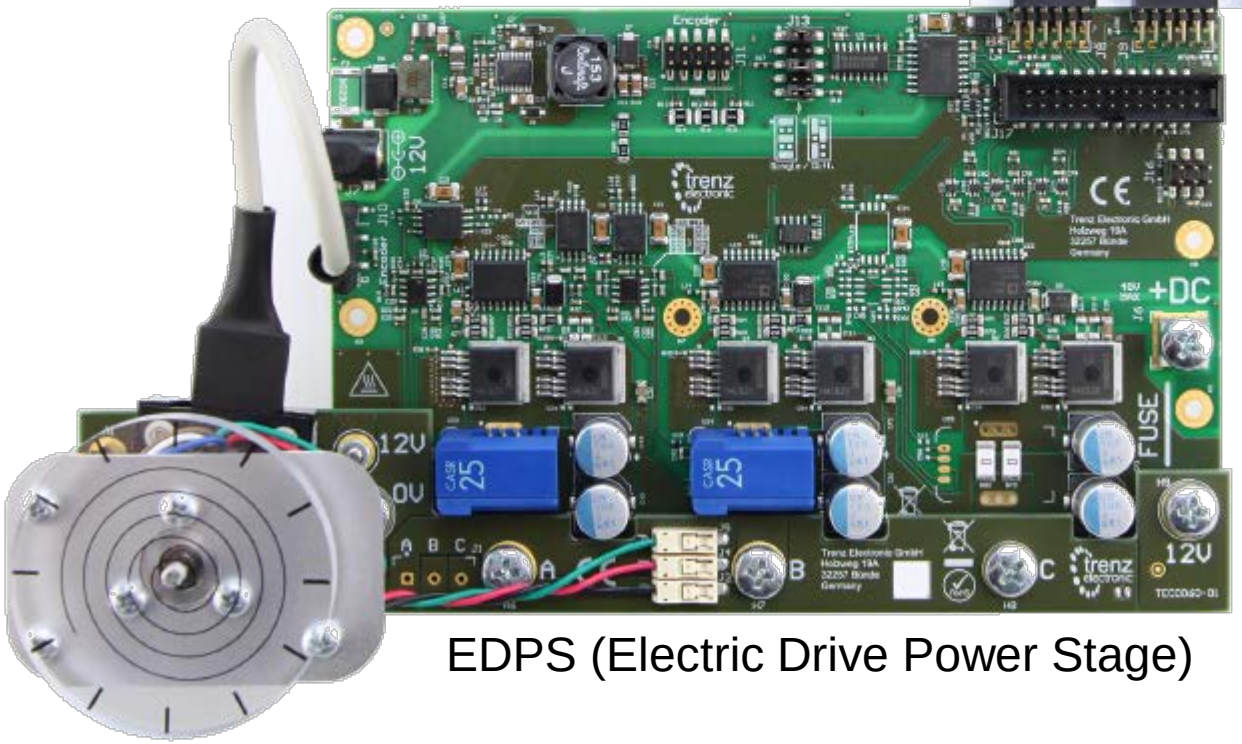
EDDP – Electric Drive Demonstration Platform

- PMOD connection between Control Board & EDPS
- Default motor – 15W BLDC
- Encoder included
- EDPS – Supports Upto 1KW

Control Board







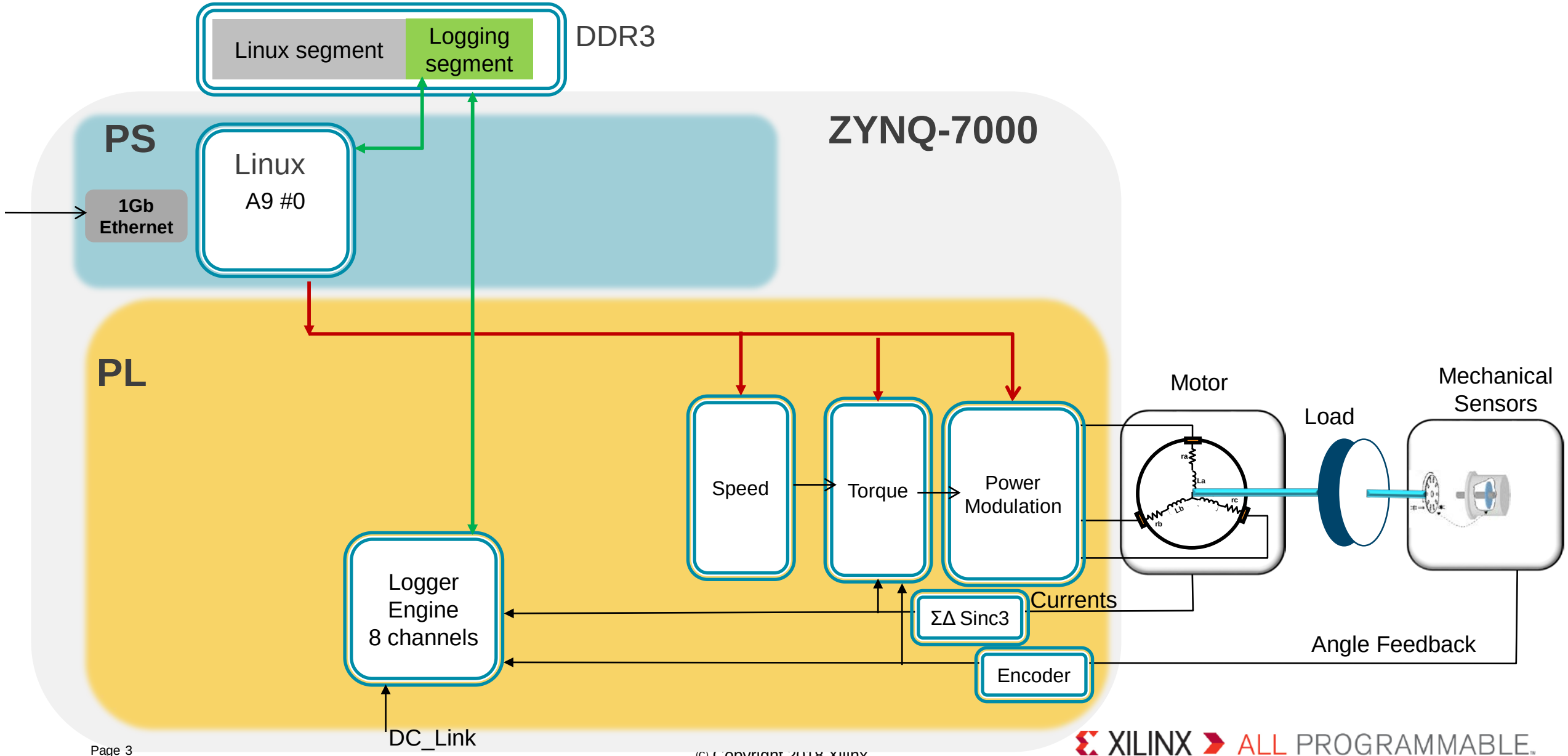
EDPS (Electric Drive Power Stage)

User ensures safety at all times & when operating the kit

© Copyright 2018 Xilinx

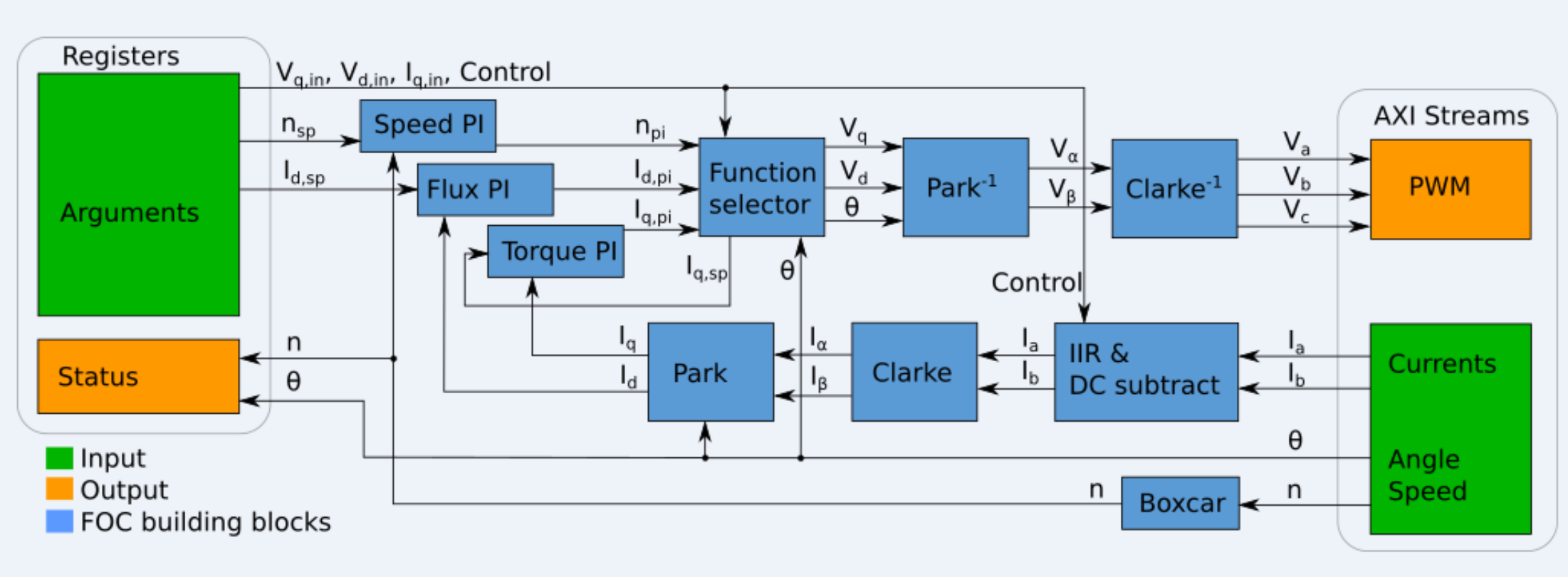
XILINX ALL PROGRAMMABLE.

Use case simple Linux + motor control single A9 CPU



© Copyright 2018 Xilinx
XILINX ALL PROGRAMMABLE.

Example Design – Architecture and Algorithm



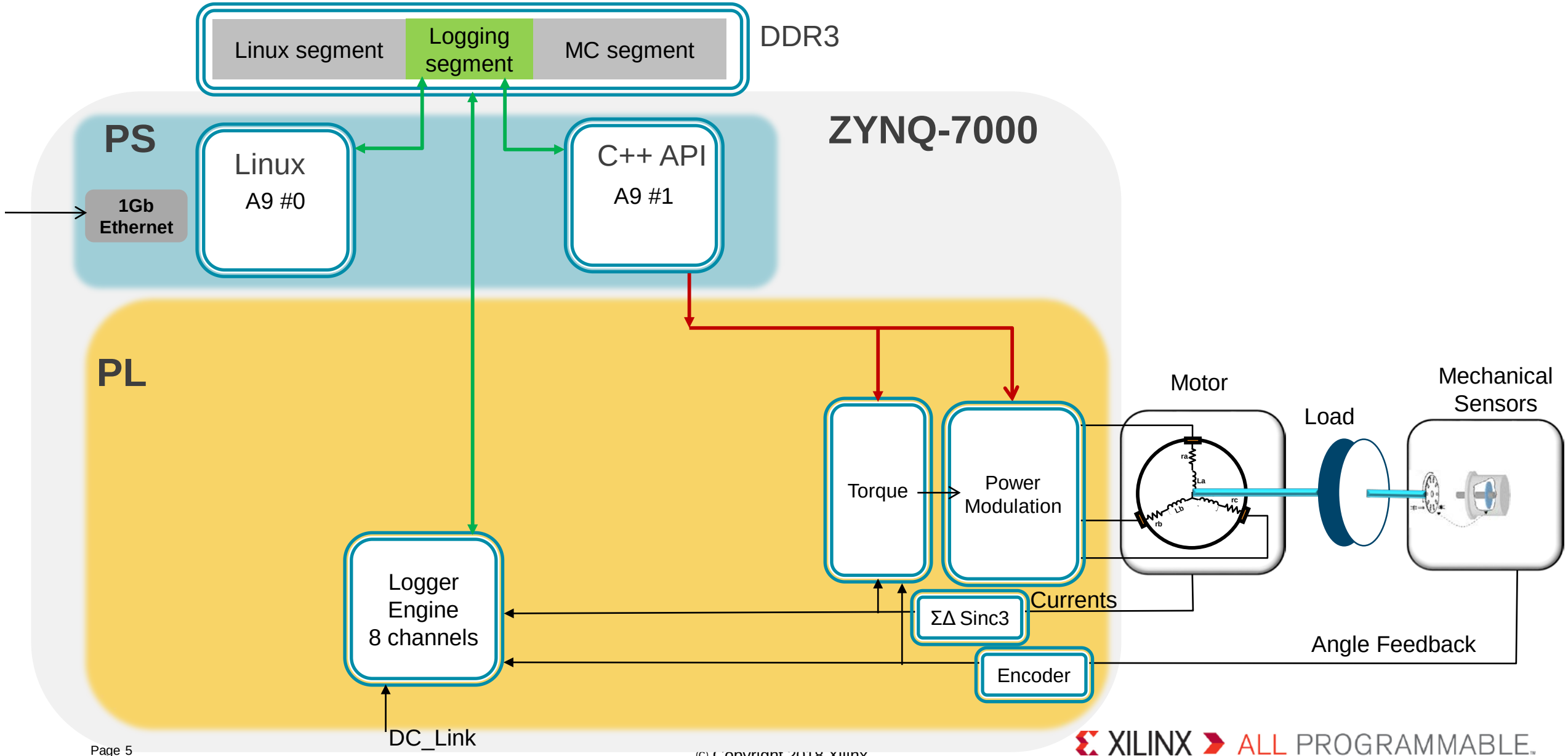
OPEN SOURCE

https://github.com/Xilinx/IIoT-EDDP

© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

Use case simple Linux + RTOS



© Copyright 2018 Xilinx

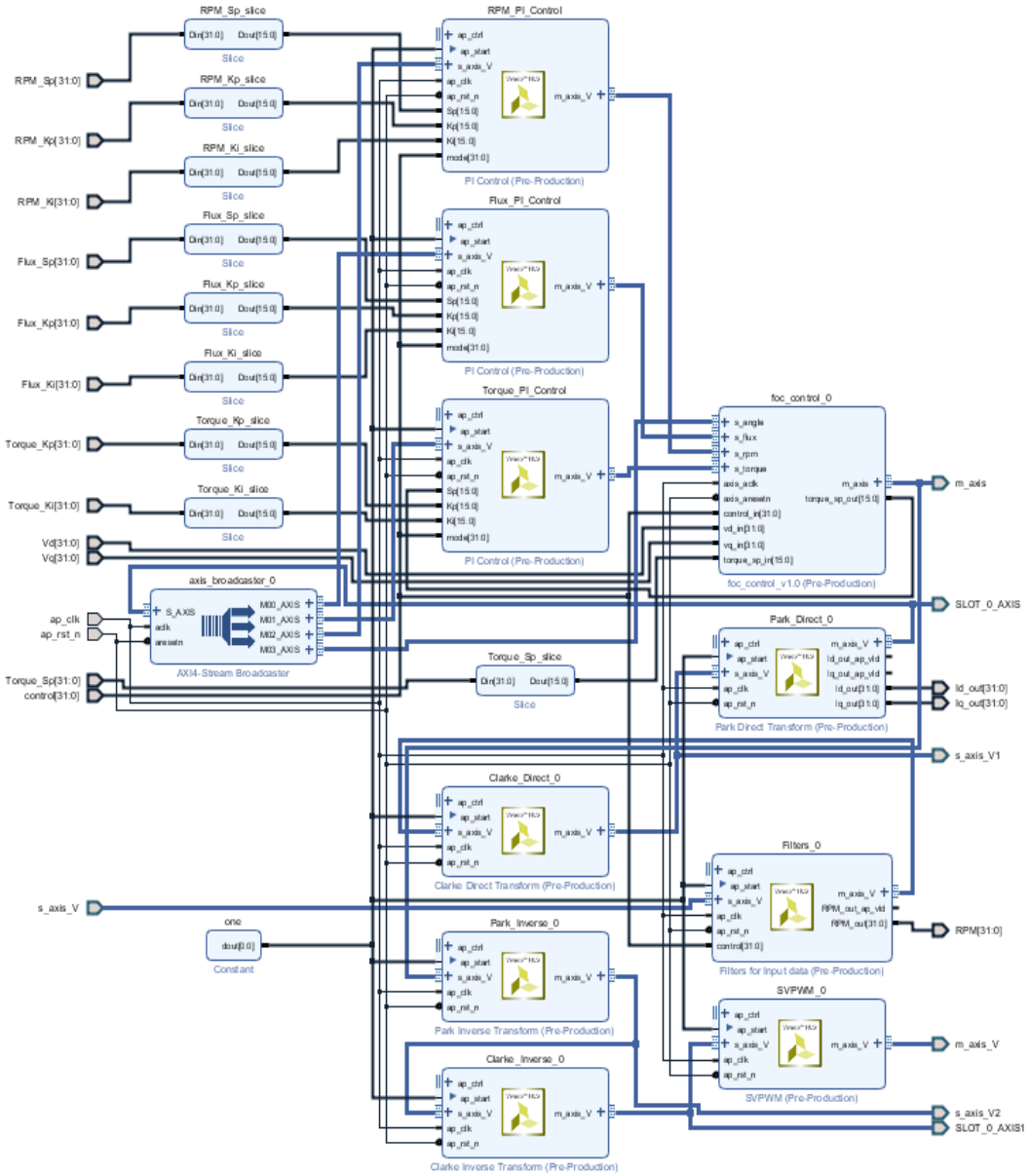
For every algorithm you have a C/C++ description

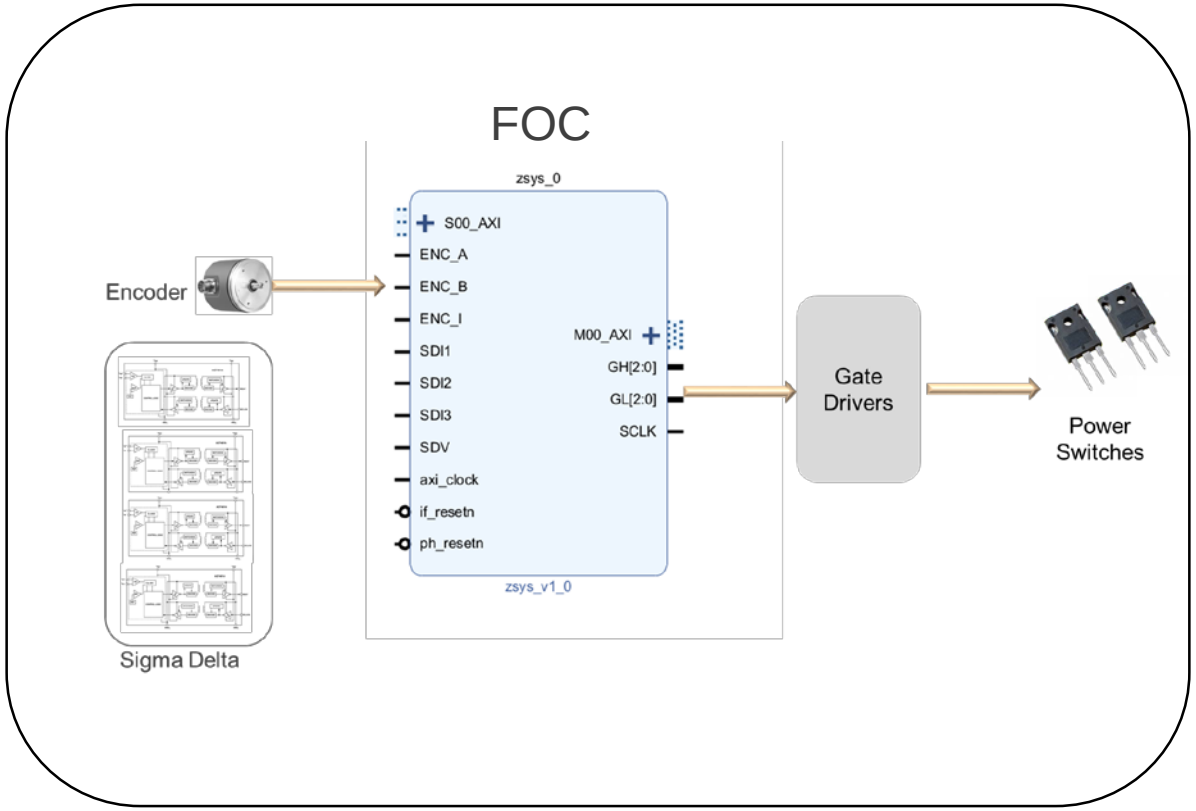
```
// See the header file for the documentation.
void Park_Direct(hls::stream<int64_t> &s_axis, hls::stream<int64_t> &m_axis, int32_t *Id_out, int32_t *Iq_out){

#pragma HLS interface axis port=m_axis
#pragma HLS interface axis port=s_axis
    int64_t in_data, res;
    int16_t Ialpha, Ibeta, Theta, RPM;
    int32_t Id, Iq;
    int32_t cos_theta, sin_theta;
    int32_t Ia_cos, Ib_sin, Ib_cos, Ia_sin;

    // Decode Input stream
    in_data = s_axis.read();                    // Read one value from AXI4-Stream
    Ialpha = int16_t(in_data & 0xFFFF);             // Extract Ialpha - bits[15..0] from input stream
    Ibeta = int16_t((in_data >> 16) & 0xFFFF);      // Extract Ibeta - bits[32..16] from input stream
    RPM = int16_t((in_data >> 32) & 0xFFFF);        // Extract RPM - bits[47..32] from input stream
    Theta = int16_t((in_data >> 48) & 0xFFFF);      // Extract Angle - bits[63..48] from input stream

    // Process data
    cos_theta = (int32_t)cos_table[Theta];
    sin_theta = (int32_t)sin_table[Theta];
    Ia_cos = (int32_t)Ialpha * cos_theta;
    Ib_sin = (int32_t)Ibeta * sin_theta;
    Ib_cos = (int32_t)Ibeta * cos_theta;
    Ia_sin = (int32_t)Ialpha * sin_theta;
    Id = (Ia_cos + Ib_sin) >> 15;
    Iq = (Ib_cos - Ia_sin) >> 15;
    Id = (Id > MAX_LIM) ? MAX_LIM : Id;             // Clip max
    Id = (Id < MIN_LIM) ? MIN_LIM : Id;             // Clip min
    Iq = (Iq > MAX_LIM) ? MAX_LIM : Iq;             // Clip max
    Iq = (Iq < MIN_LIM) ? MIN_LIM : Iq;             // Clip min

    *Id_out = Id;
    *Iq_out = Iq;
    // Write output stream
    res =   (((int64_t)Theta << 48) & 0xFFFF000000000000) | // Put Angle bits[63:48]
            (((int64_t)RPM << 32)   & 0x0000FFFF00000000) | // Put RPM bits[47:32]
            (((int64_t)Iq << 16)    & 0x00000000FFFF0000) | // Put Iq bits[31:16]
            ( (int64_t)Id           & 0x000000000000FFFF);  // Put Id bits[15:0]
    m_axis.write(res);                              // Write result to the output stream
}
```

© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

Whole FOC IP – it will be a **Hardware** library component





© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

IP integrator and Simulink flow

C-based IP Creation

C, C++, SystemC

Libraries

Arbitrary Precision
Video
Math
Linear algebra
IP: FFT and FIR

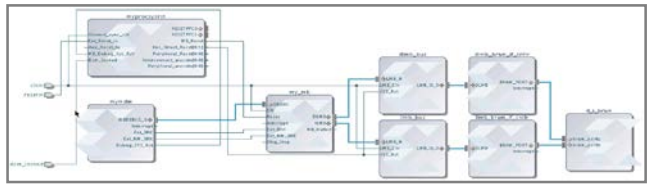
Vivado™ HLS

VHDL or Verilog
You do not see it

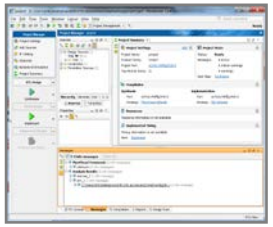
Matlab Simulink

System Integration

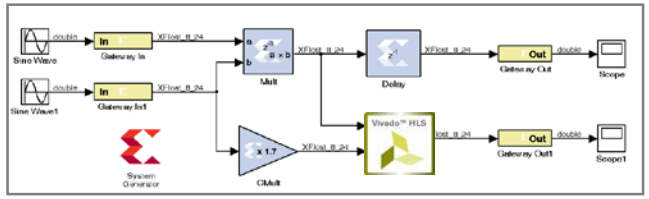
IP Catalog



Vivado IP Integrator



Vivado RTL

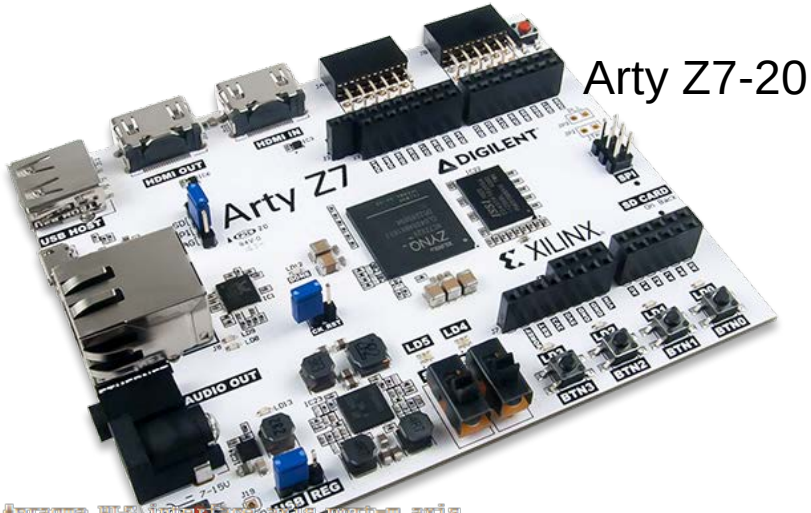


System Generator for DSP

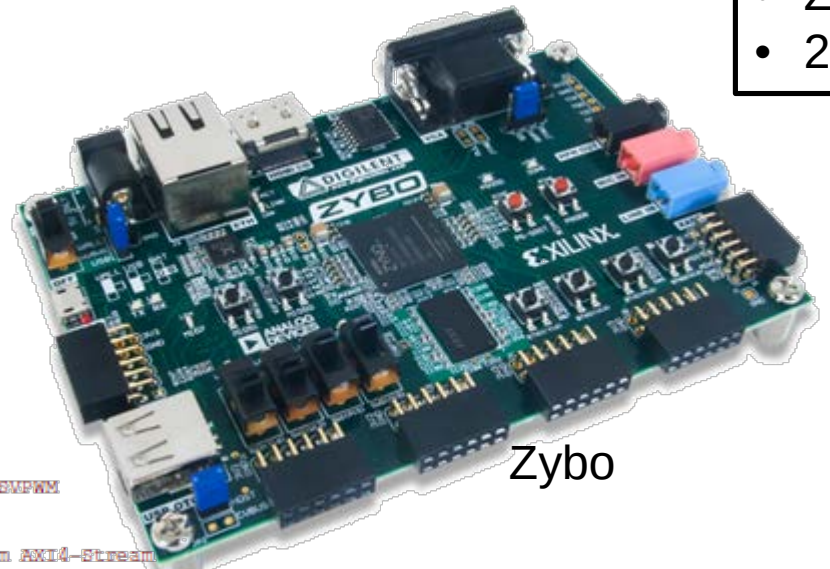
© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

EDPS Compatible with Multiple Control Boards



Arty Z7-20



Zybo

ZedBoard

Key compatibility requirements
- Zynq 7000 based board
- 2 PMOD connectors

```
19 #pragma HLS interface axis port=m_axis
20 #pragma HLS interface axis port=s_axis
21     int64_t in_data, res;
22     int16_t Valpha, Vbeta, Theta;
23     int32_t sZvb;                                   // Clarke Inverse
24     int32_t Va, Vb, Vc;                           // Clarke Inverse -> SVPWM
25
26     // Decode Input stream
27     in_data = s_axis.read();                      // Read one value from AXI4-Stream
28     Valpha = int16_t(in_data & 0xFFFF);             // Extract Valpha - bits[15..0] from input stream
29     Vbeta = int16_t((in_data >> 16) & 0xFFFF);      // Extract Vbeta - bits[31..16] from input stream
30     Theta = int16_t((in_data >> 32) & 0xFFFF);      // Extract Theta - bits[47..32] from input stream
31
32     // Process data
33     Va = Valpha;
34     sZvb = Vbeta * SQRT3C;                          // (sqrt(3)*(2^15))*Vbeta
35     Vb = ((sZvb >> 15) - Valpha) >> 1;            // (-Valpha + sqrt(3)*Vbeta)/2
36     Vc = (0 - Valpha - (sZvb >> 15)) >> 1;        // (-Valpha - sqrt(3)*Vbeta)/2
37     Vb = (Vb > MAX_LIM) ? MAX_LIM : Vb;             // Clip max
38     Vb = (Vb < MIN_LIM) ? MIN_LIM : Vb;             // Clip min
39     Vc = (Vc > MAX_LIM) ? MAX_LIM : Vc;             // Clip max
40     Vc = (Vc < MIN_LIM) ? MIN_LIM : Vc;             // Clip min
41
42     // Write output stream
43     res =    (((int64_t)Theta << 48) & 0xFFFF000000000000) | // Put Theta bits[63:48]
44              (((int64_t)Vc << 32)    & 0x0000FFFF00000000) | // Put Vc bits[47:32]
45              (((int64_t)Vb << 16)    & 0x00000000FFFF0000) | // Put Vb bits[31:16]
46              ( (int64_t)Va           & 0x000000000000FFFF);  // Put Va bits[15:0]
47     m_axis.write(res);                               // Write result to the output stream
48 }
```

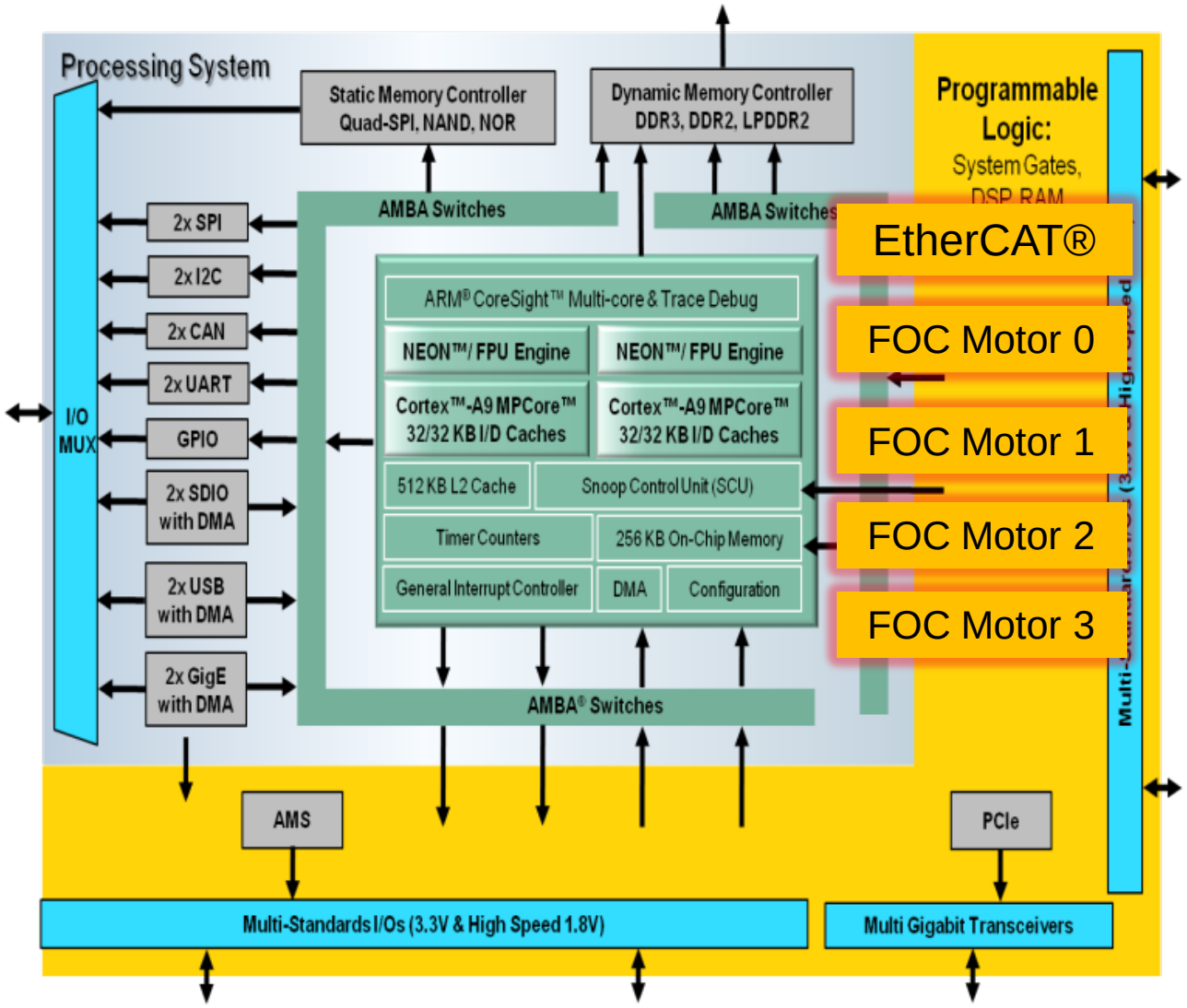
https://github.com/Xilinx/IIoT-EDDP

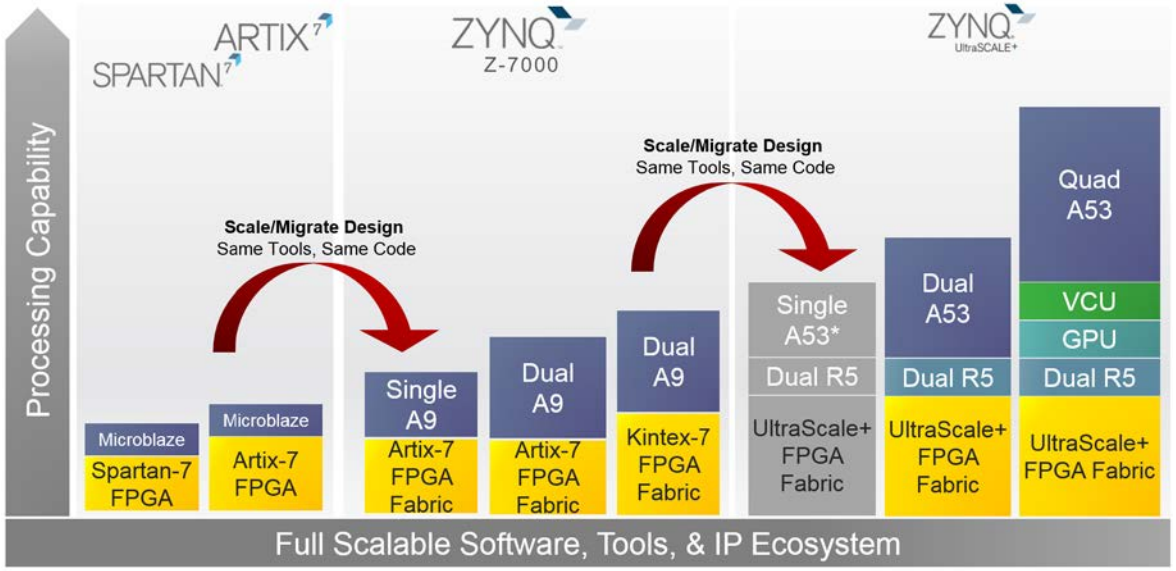
- Open Source Example Design
- Representative of an electric drive solution
- Usable with SDSoC

© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

Xilinx Solutions Deliver Lowest TCO Drives



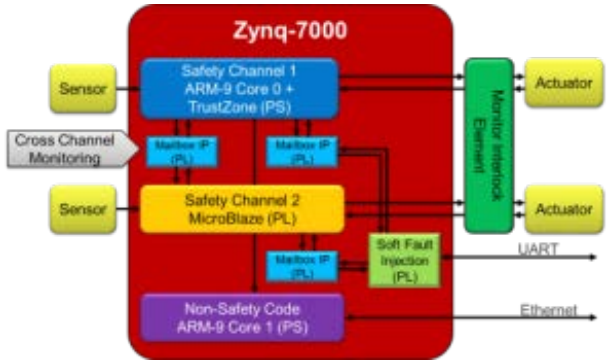


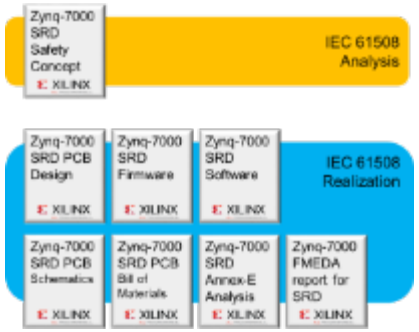
- Single chip, highly integrated
  - o Multiple-axis control
  - o Performance ↑
  - o Power ↓
  - o Overall system TCO ↓

© Copyright 2018 Xilinx

XILINX ALL PROGRAMMABLE.

ZYNQ-7000 forSafe Systems with High Reliability, Longevity





ZYNQ-7000 Reference Design & Safety Package
Single Chip SIL3 Certifiable, HFT = 1, 1oo2D
Full Documentation Package

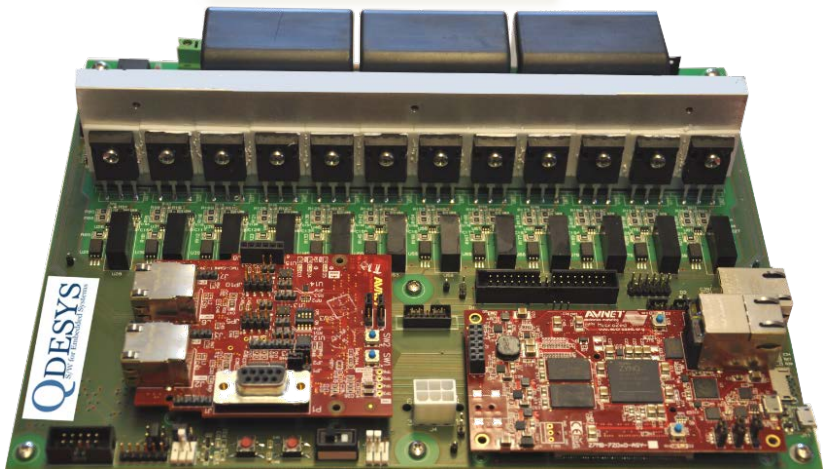
VIVADO Certified
Up to SIL3
HLS (C/C++)
SEM IP & IDF

Microblaze Compiler
Up To SIL4 & ASILD

- **Typical Product Life Cycle 15+ years + Last Time Buy Period**
- **Xilinx serves multiple markets with extended lifetime requirements**
  - Military, Industrial/Embedded, Aerospace, Medical, Wired

© Copyright 2018 Xilinx
XILINX ALL PROGRAMMABLE.

Multiple HW Platforms for Evaluation



www.qdesys.com

Standard & Advanced Controls

Zynq-7000 based Three Level Inverter (TLIMOT)

- Silicon-Carbide Technology
- Fast control loop through Zynq
- Optimal design for size, cost, EMI & more
- Complete with exhaustive IP, design services

© Copyright 2018 Xilinx

Thank you for listening!!

Check out the next module to learn more

- Getting Started with the Electric Drives Demo Platform

Module 3: Building Electric Drives with Xilinx SDSoC

© Copyright 2018 Xilinx