June 23, 2010

Using Enhanced Time Processing Unit (eTPU/eTPU2) for Combustion Engine Management and Electric Motor Control

FTF-AUT-F0447

Milan Brejl, Ph.D.

System Application Engineer

freescale™ semiconductor

Introduction

- Enhanced Time Processing Unit (eTPU) is
  - Freescale most advanced timer module
  - Competition differentiator
  - Highly questionable module
- After 6 years of experience with the eTPU, eTPU2 was introduced
- Main areas of eTPU usage
  - Engine control
  - Motor control
- Overview of Freescale support for eTPU/eTPU2
  - devices, tools, application notes and ref. designs
- Presenter: Milan Brejl Expertise: eTPU applications

Session Objectives

- This session should helped you to

- Get a good overview of the whole eTPU world

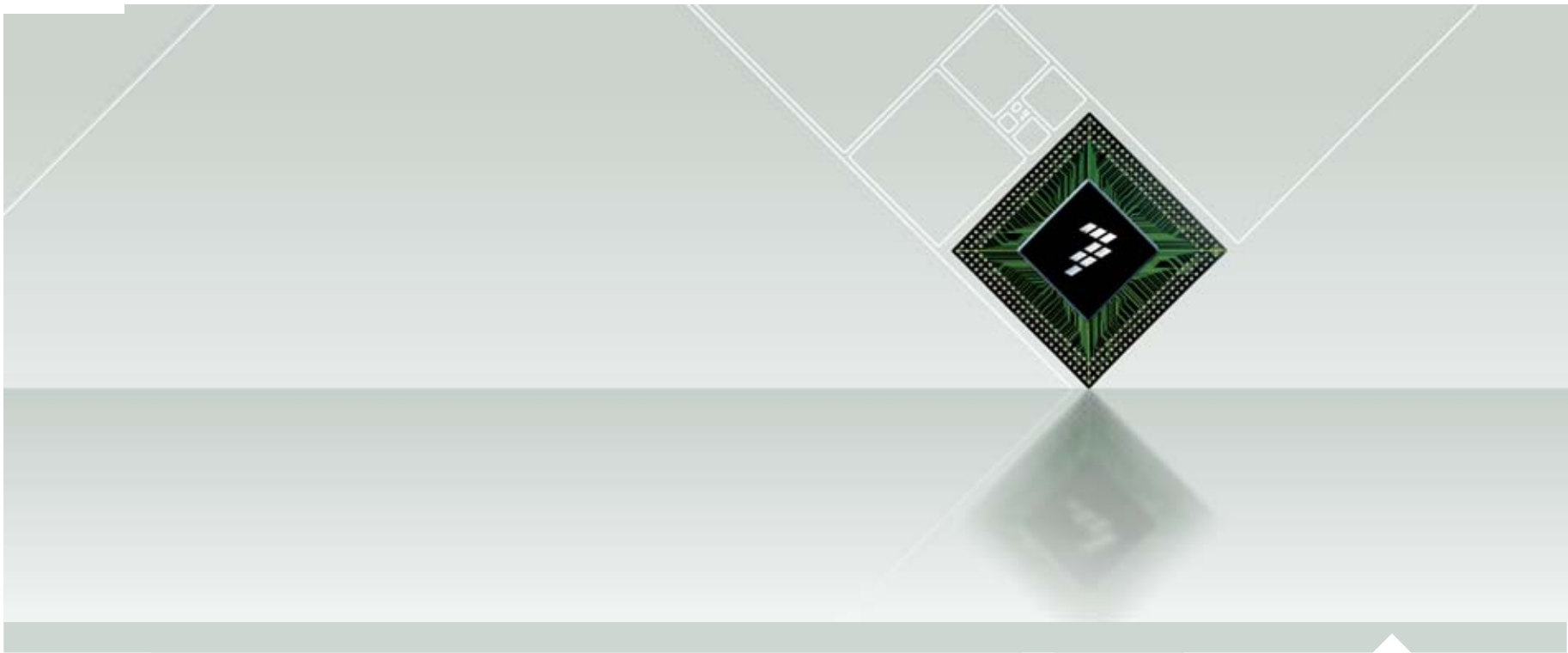
- Get familiar with eTPU usage in applications
  - performance, benefits
  - engine control and motor control applications

- Understand how to use the eTPU in your application
  - ready-to use library functions
  - development of customer functions

freescale semiconductor

Agenda

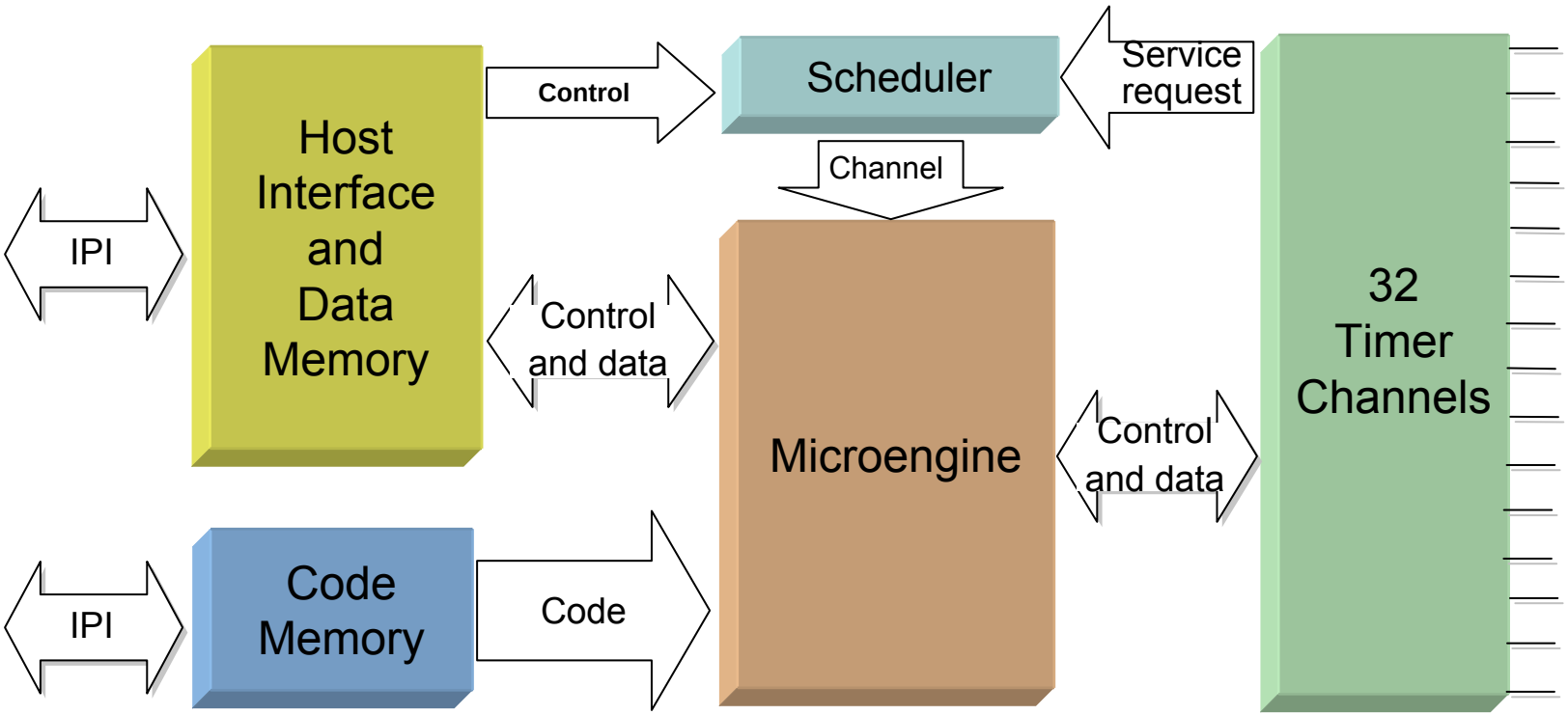
- eTPU Introduction
  - eTPU Usage
  - eTPU on Roadmaps
  - eTPU2 New Features
- eTPU for Engine Control
- eTPU as Motor Control Co-Processor
- Easy-to-use eTPU Tools
  - eTPU Function Selector web tool
  - eTPU Graphical Configuration Tool
  - eTPU Compilers, Simulators, Debuggers
- eTPU Software And Documentation Available from Freescale
- Summary



eTPU Introduction

freescale semiconductor ™

The *enhanced Time Processing Unit* is a programmable I/O controller with it's own core and memory system, allowing it to perform complex timing and I/O management *independently* of the CPU. The eTPU is essentially a microcontroller all by itself.



freescale semiconductor

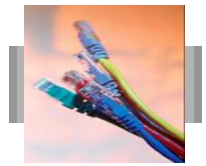
What the eTPU is good for

▶General Timing

- Input Capture, Output Compare, Pulse Accumulate, PWM

▶Serial Communication

- UARTs, I2C, ARINC, MOST, Proprietary Protocols



▶Combustion Engine Control

- Engine Position, Spark Timing, Fuel Injection

▶Electrical Motor Control

- Factory Automation, DC, AC and Stepper Motors

freescale semiconductor

32-bit Automotive Roadmap

Application Performance / Integration

MPC5500
130nm, 250DMIPS

MPC5561 Lance
1M Flash, 192K RAM
e200z6 core, up to 200MHz

MPC5566 Viper
3M Flash, 128K SRAM
e200z6 core, up to 144MHz

MPC5567 Tiger
2M Flash, 80K SRAM, FlexRay
e200z6 core, up to 144MHz

MPC5565 Taipan
2M Flash, 80K SRAM
e200z6 core, up to 144MHz

MPC5554 Copperhead
2M Flash, 64K SRAM
e200z6 core, up to 144MHz

MPC5553 Moccasin
1.5M Flash, 64K SRAM
e200z6 core, up to 144MHz

MPC5517x
1.5M Flash, 80 KB RAM,
up to 80MHz
e200z1+e200z0 (not on S version)

MPC5516x
1M Flash, 64KB RAM
Up to 80MHz
e200z1+e200z0 (not on S version)

MPC5515S
768KB Flash, 48 KB RAM
Up to 80MHz
e200z1+e200z0 (not on S version)

MPC5514E/G
512KB Flash
e200z0+e200z0

MPC5534 Coral
1M Flash, 64K SRAM
e200z3 core, 40-80 MHz

MPC5600
90nm, 600DMIPS

MPC5674F Mamba
4M Flash, 256K SRAM, FlexRay
e200z7 core, up to 264 MHz,

MPC5668G Fado
2M Flash, 592K SRAM, FlexRay
e200z6+e200z0, up to 128 MHz

MPC5644A Andorra
4M Flash, up to 192K SRAM
up to 145MHz, FlexRay

MPC5642A Andorra
2M Flash, 96K SRAM
up to 132MHz, FlexRay

MPC5634M Monaco
768K-1.5M Flash, 48K-94K SRAM
e200z3 core, up to 80MHz

MPC5604/3P Pictus
384K-512K Flash
36K-40K SRAM, FlexRay
e200z0 core, up to 64MHz

MPC5607/6/5B Bolero
768KK-1.5M Flash, 64KB Data Flas
64KB-96KB SRAM
e200z0 core, up to 64MHz

MPC5604/3/2B/C Bolero
256K-512K Flash,
24-32K(B), 32-48K(C) SRAM
e200z0 core, up to 64MHz

MPC5606/4/2S Spectrum
256K-1M Flash, 24K-48K SRAM
160K Graphics RAM, LCD
e200z0h core, up to 64MHz

MPC5700, 65nm, >1000DMIPS

180+ MHz

MPC5776C Cobra
6M Flash, 384K SRAM
2 x 200MHz, FlexRay

2M Flash, FlexRay
1.5M Flash, FlexRay

100+ MHz

4M Flash, FlexRay
2M Flash, FlexRay
1M Flash, FlexRay
512K Flash
2M Flash

50+ MHz

256K Flash
256K Flash
192K Flash
128K Flash

Time

Powertrain
- includes eTPU

Chassis, Safety

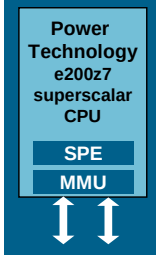
Body, Cluster

Known Good Die & High Temp Pkg options

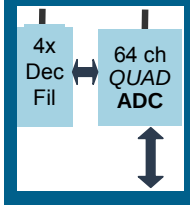
BUILT ON Power

freescale semiconductor

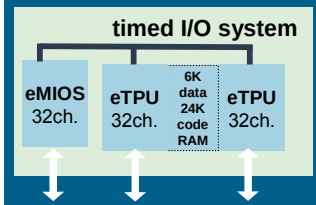
MPC5674F: 4 MB Engine Controller with FlexRay™



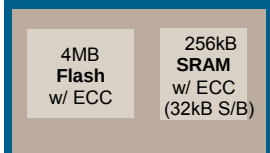
600 DMIPS from 264 MHz core, integrated DSP allowing users to create 'virtual sensors'



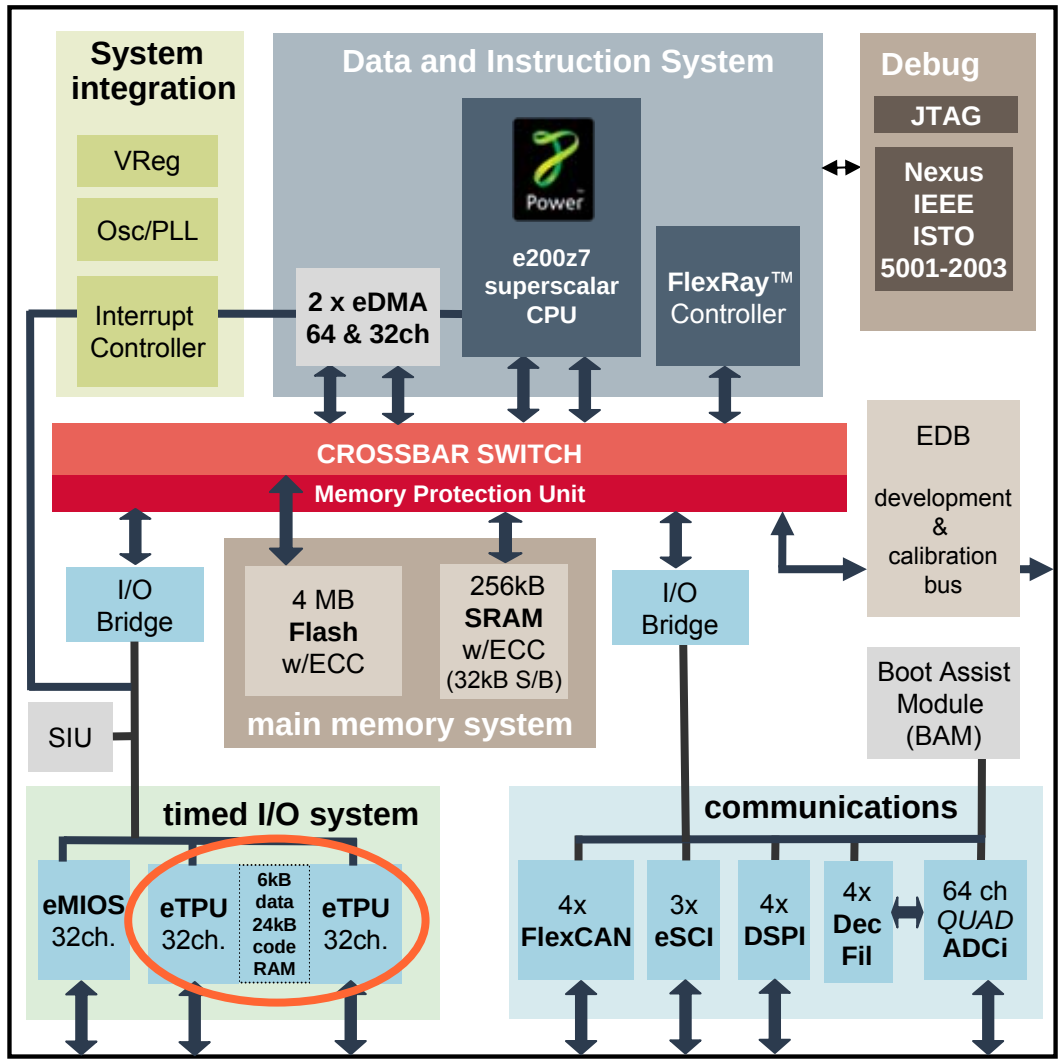
Only quadruple ADC on market, with built-in filtering system allows cost reduction of PCB



Most precise engine timers available, control fuel delivery & improve gas mileage



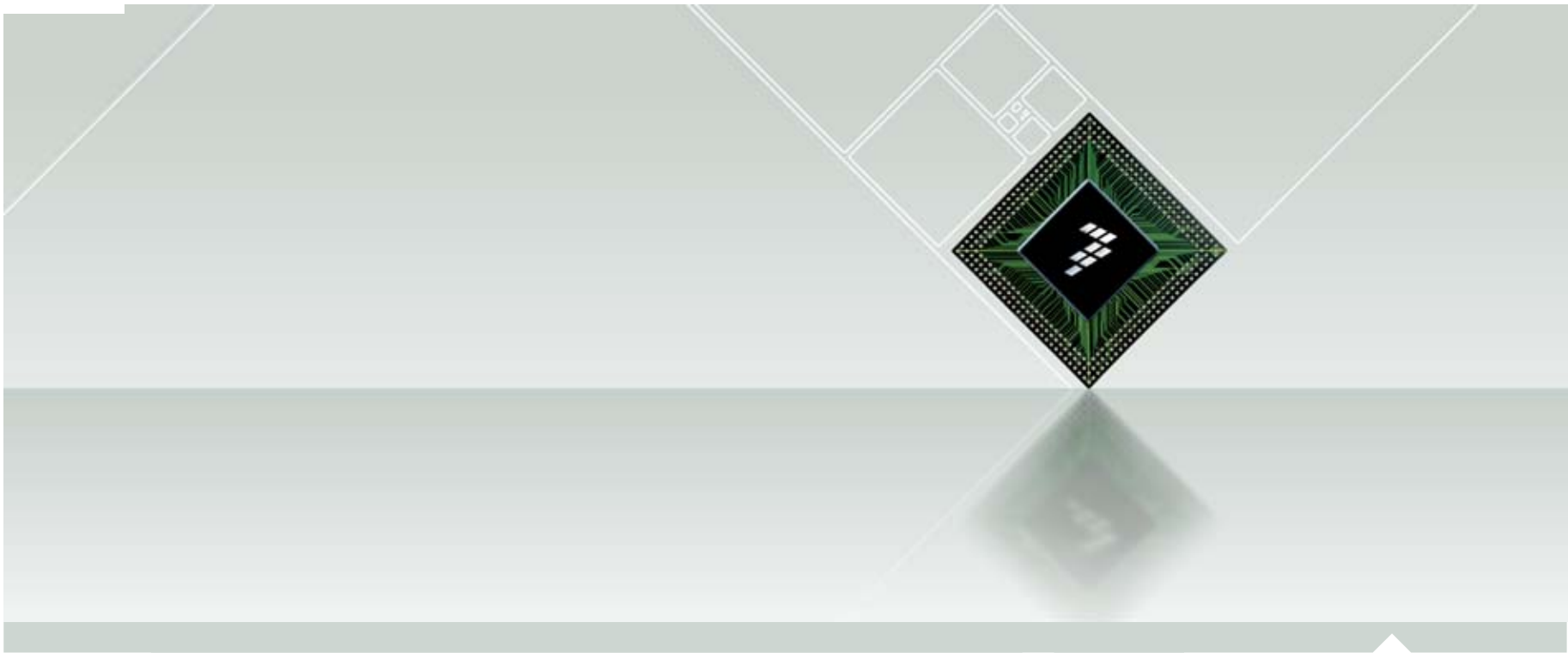
Largest program memory for market space helps with autocoding; zero defect technology on all memories



freescale semiconductor ™

eTPU2 Changes

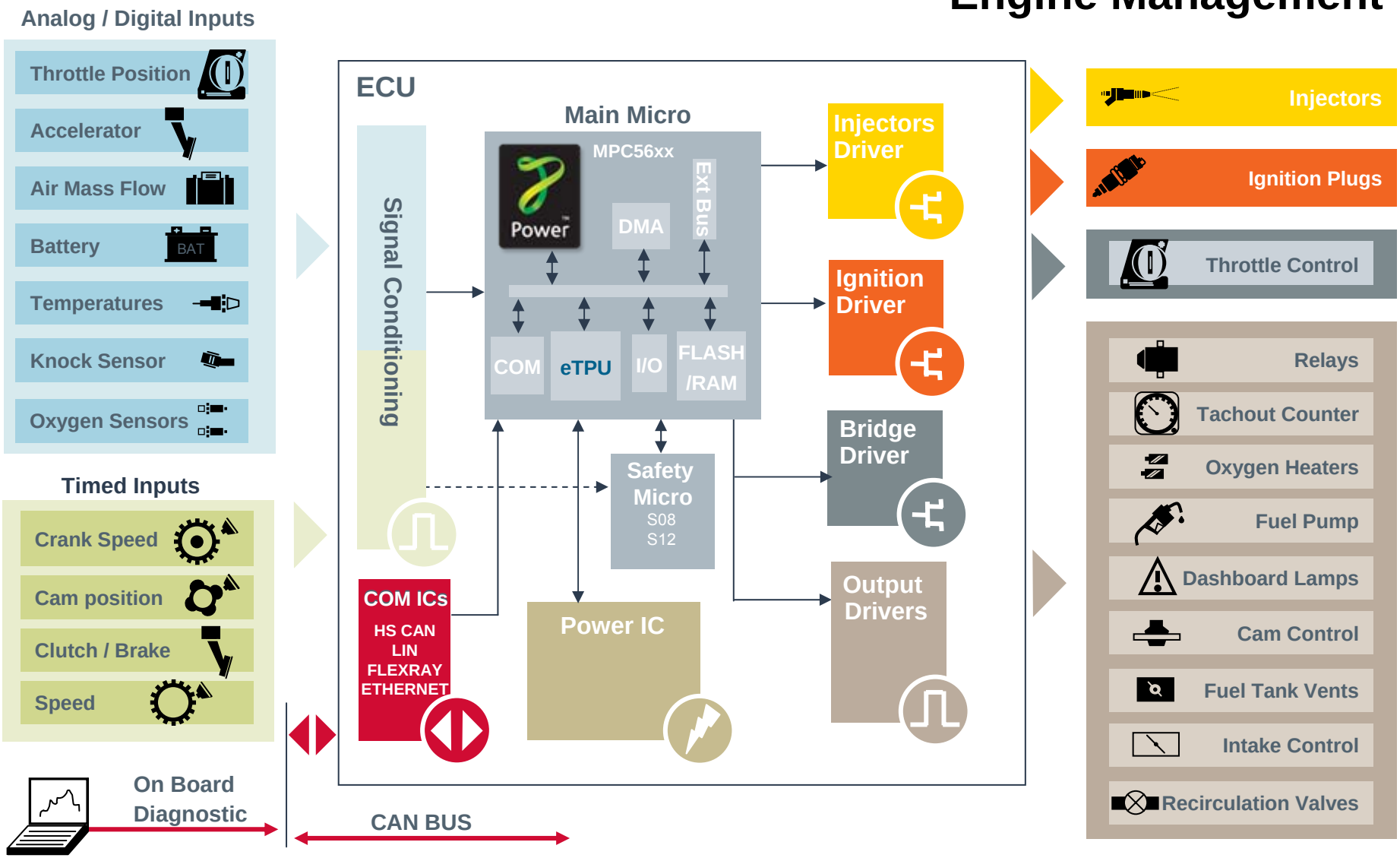
- 100% compatible with eTPU
  - No changes required to hardware or software if only eTPU features are used
- Supports a wider range to frequencies
  - Supports up to 200MHz operation and better resolution at lower frequencies
- New channel features
  - Main change is programmable channel mode
- New programming features
  - Biggest change is engine relative addressing mode
- Safety related enhancements
  - New software watchdog and error detection features
- Some devices have enhanced motor control features
  - This is not a change to the eTPU itself but a change in integration
  - More eTPU channels have separate input and output signals to allow an eTPU to control 4 BLDC motors



eTPU for Engine Control

freescale semiconductor ™

Engine Management

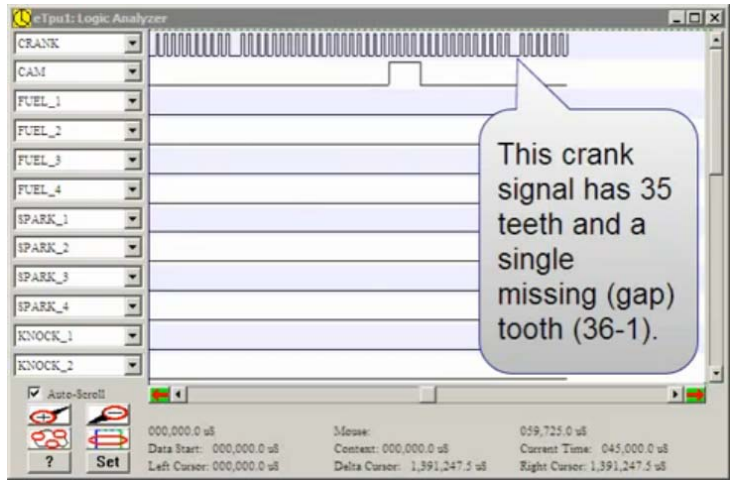


Engine Control eTPU Functions

- Engine Position (CRANK and CAM)
  - measure the speed and position of a rotating engine, generate an angle counter, synchronized to instantaneous angular position of the engine
- Fuel
  - generates one or more fuel injection pulses controlling the amount of fuel in the intake manifold
- Spark
  - generates pulses synchronized to specific angular positions, controlling the ignition timing
- Knock Window
  - outputs a number of angle-based knock windows to gate the ADC
- Tooth Generator
  - generation of CRANK and CAM signals for development purposes

Examples of Engine Control eTPU Functions

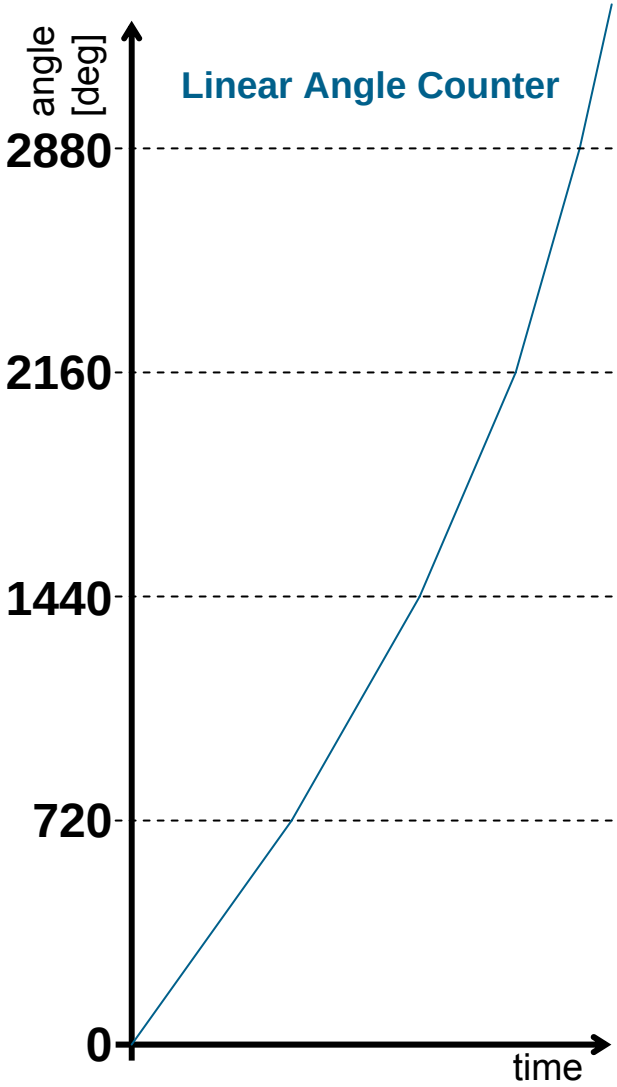
- **Simulator Example**:
  YouTube video eTPU set2 Simulator Demo
  http://www.youtube.com/watch?v=Mu_zBkoByVQ



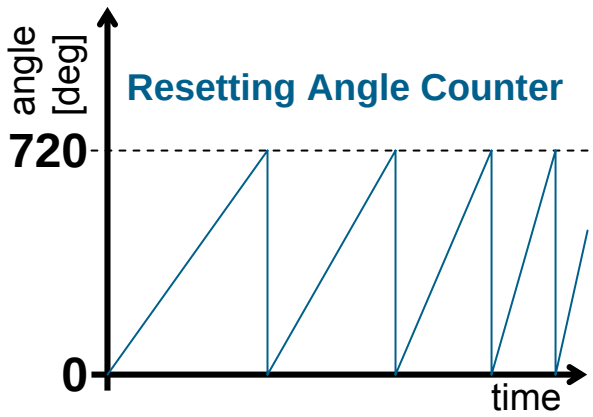
- **FreeMASTER Example**:
  YouTube video Using the Engine Position eTPU Functions - Freescale Application Note: 3769
  http://www.youtube.com/watch?v=a1NplNEAaww

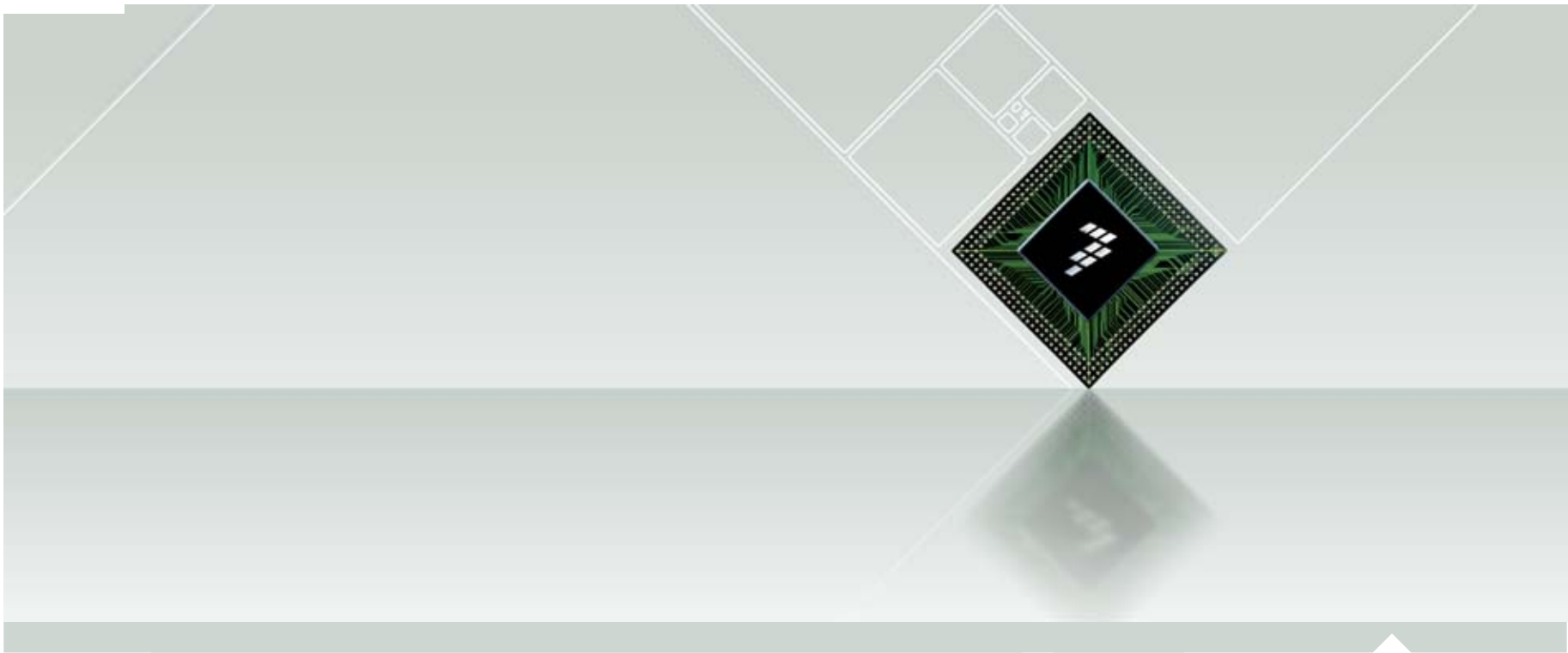
freescale semiconductor

Benefits of Linear vs. Resetting Angle Counter



- Angle as a liner free-running counter
  - Enables to schedule angle events several engine cycles in advance
  - Enables to calculate relative angle distances with no need to care about a discontinuation point
- Angle as a resetting counter 0 to 720 deg
  - Traditional approach



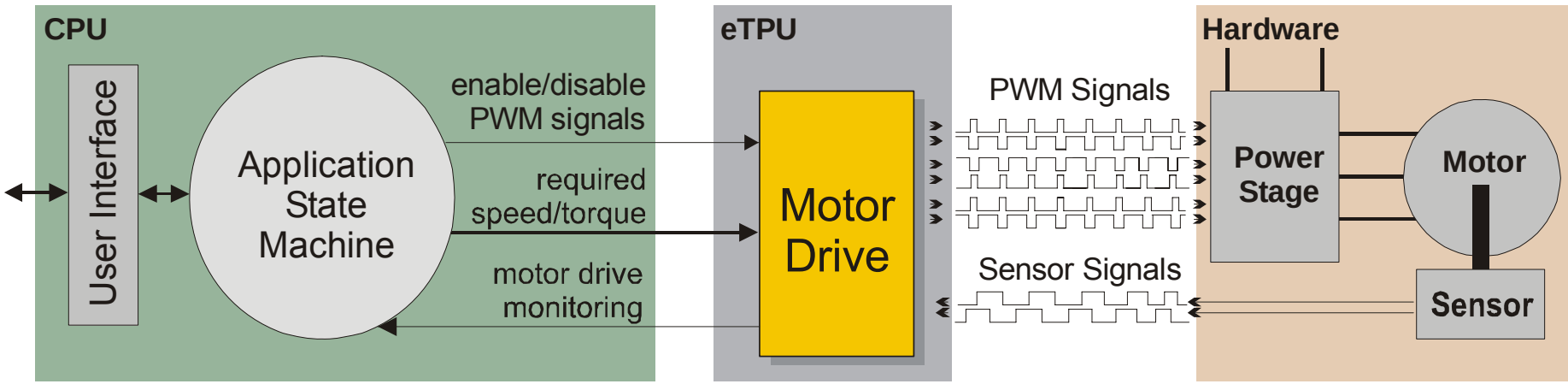


eTPU as Motor Control Co-Processor

freescale semiconductor

System Concept

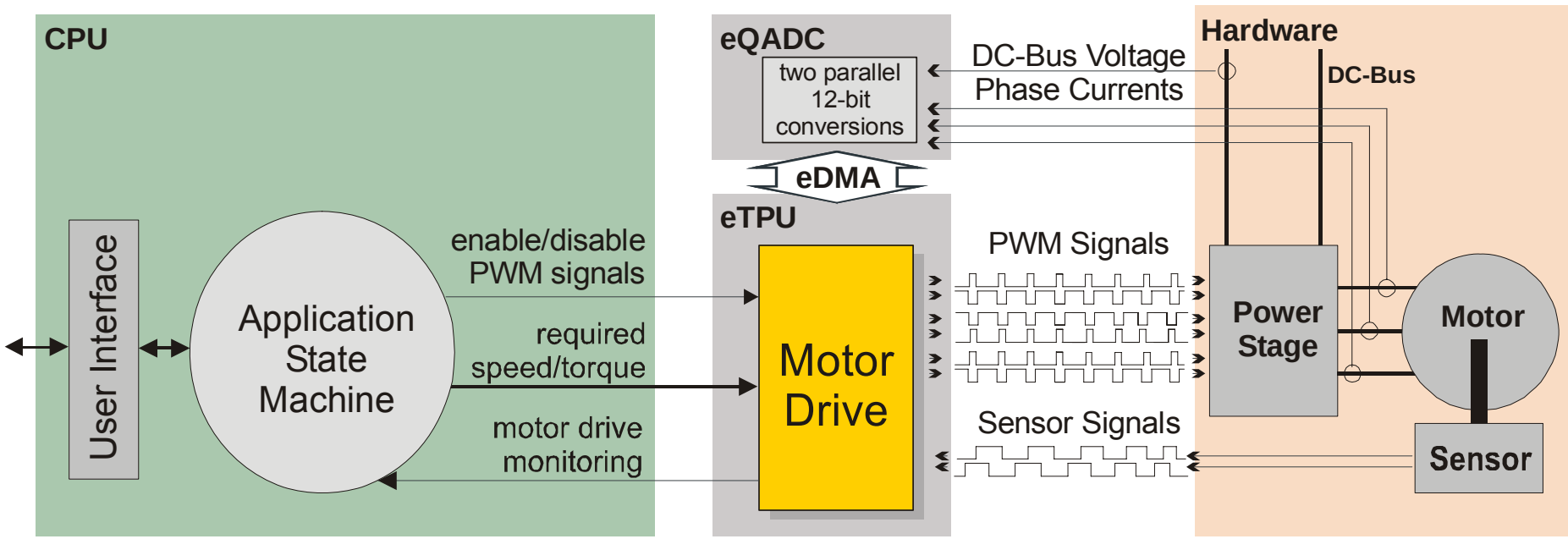
- eTPU drives a motor independently of the CPU
- CPU sets required quantity value (torque, speed, position)
- CPU handles higher level tasks



freescale semiconductor

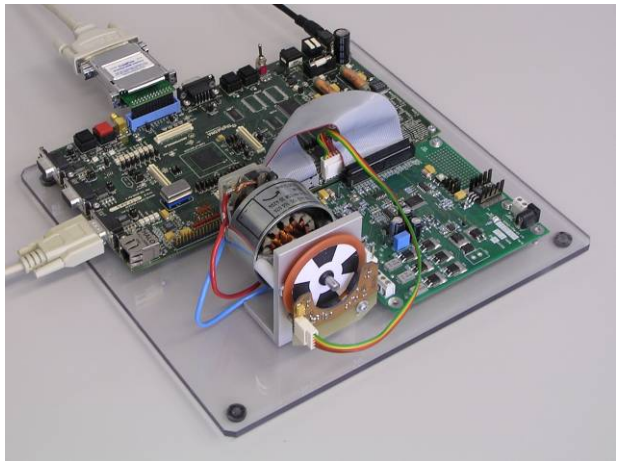
System Concept

- eTPU drives a motor independently of the CPU
- eQADC (triggered by eTPU) samples analog quantities
- eDMA transfers data between eQADC and eTPU
- CPU only sets required quantity value (speed or torque)
- CPU can handle higher level tasks

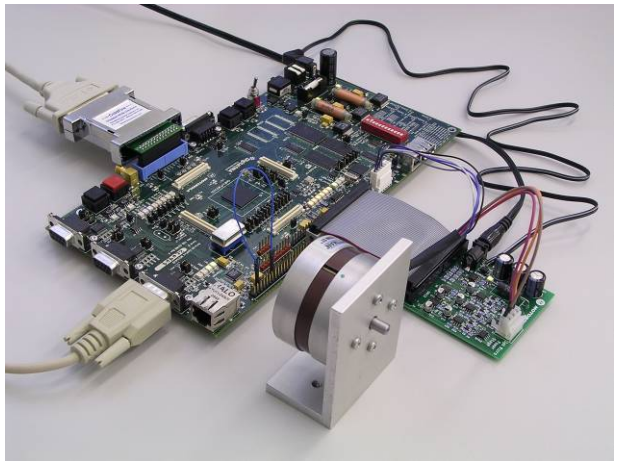


freescale semiconductor

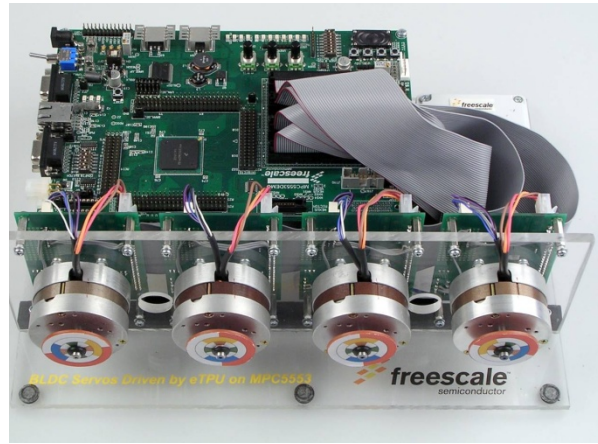
eTPU Motor Control Demo Applications



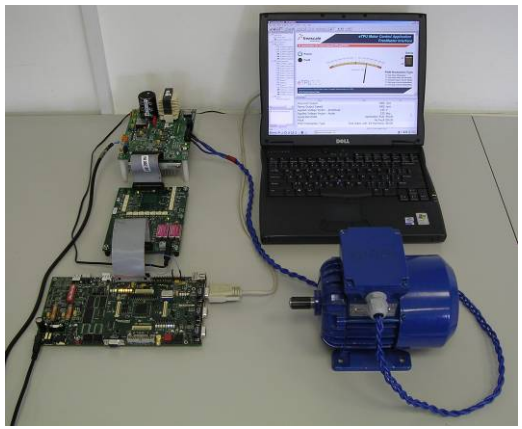
DC Motor



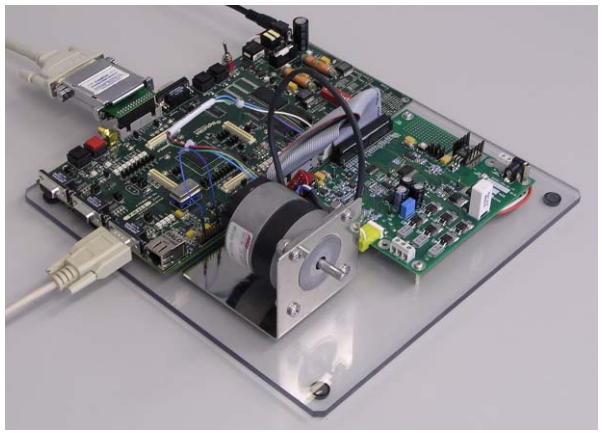
Brushless DC Motor



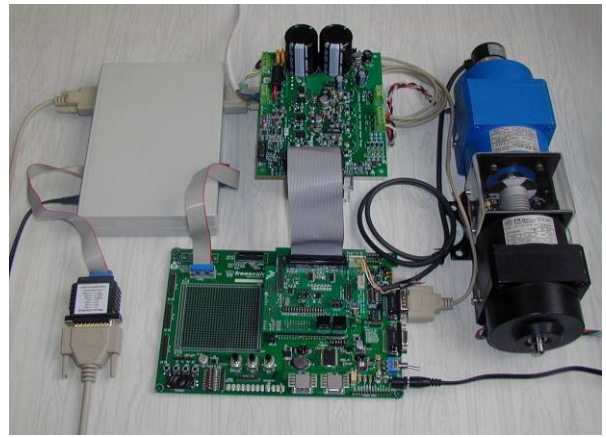
4 BLDC Motors



ACIM V/Hz



PMSM Vector Control



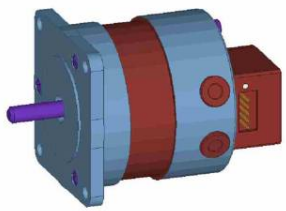
ACIM Vector Control

freescale semiconductor

eTPU Motor Control Library

DC Motors:

- PWMMDC - PWM Master For DC
- PWMF - PWM Full Range
- PWMC - PWM Commutated
- QD - Quadrature Decoder
- HD - Hall Decoder
- ASDC - Analog Sensing For DC
- SC - Speed Controller
- BC - Break Controller
- CC - Current Controller



AC Motors:

- PWMMAC - PWM Master For AC
- PWMF - PWM Full Range
- QD - Quadrature Decoder
- ASAC - Analog Sensing For AC
- SC - Speed Controller
- BC - Break Controller
- ACIMVHZ – ACIM V/Hz control
- PMSMVC – PMSM Vector Contr.
- ACIMVC - ACIM Vector Control

eTPU Performance – Vector Control Drives

	PMSM Vector Control		ACIM Vector Control	
	MPC5553/4	MPC5674F	MPC5553/4	MPC5674F
eTPU Clock	128MHz	200MHz	128MHz	200MHz
eTPU Engine Time Load				
average	45.6% @ 10RPM	29.2% @ 10RPM	53.1% @ 10RPM	34.0% @ 10RPM
	49.9% @ 1000RPM	31.9% @ 1000RPM	61.8% @ 1000RPM	39.6% @ 1000RPM
peak	50.3% @ 10RPM	32.2% @ 10RPM	58.8% @ 10RPM	37.6% @ 10RPM
	54.6% @ 1000RPM	34.9% @ 1000RPM	67.5% @ 1000RPM	43.2% @ 1000RPM
eTPU Memory Usage				
Code RAM	7.5kB (of 12kB)	7.5kB (of 24kB)	8.2kB (of 12kB)	8.2kB (of 24kB)
Data RAM	1.0kB (of 3kB)	1.0kB (of 6kB)	1.1kB (of 3kB)	1.1kB (of 6kB)

Application Parameters	
PWM frequency:	20kHz
Vector control update frequency:	20kHz
Speed controller update frequency:	1kHz
Shaft Encoder - increments per revolution:	4096

How to Drive 4 BLDC Motors using a Single eTPU?

- 9 signals to drive 1 BLDC motor with Hall sensor position feedback
  - 6 PWM outputs
  - 3 Hall sensor inputs

- 4x9 = 36 I/O signals needed but the eTPU has 32 channels only!

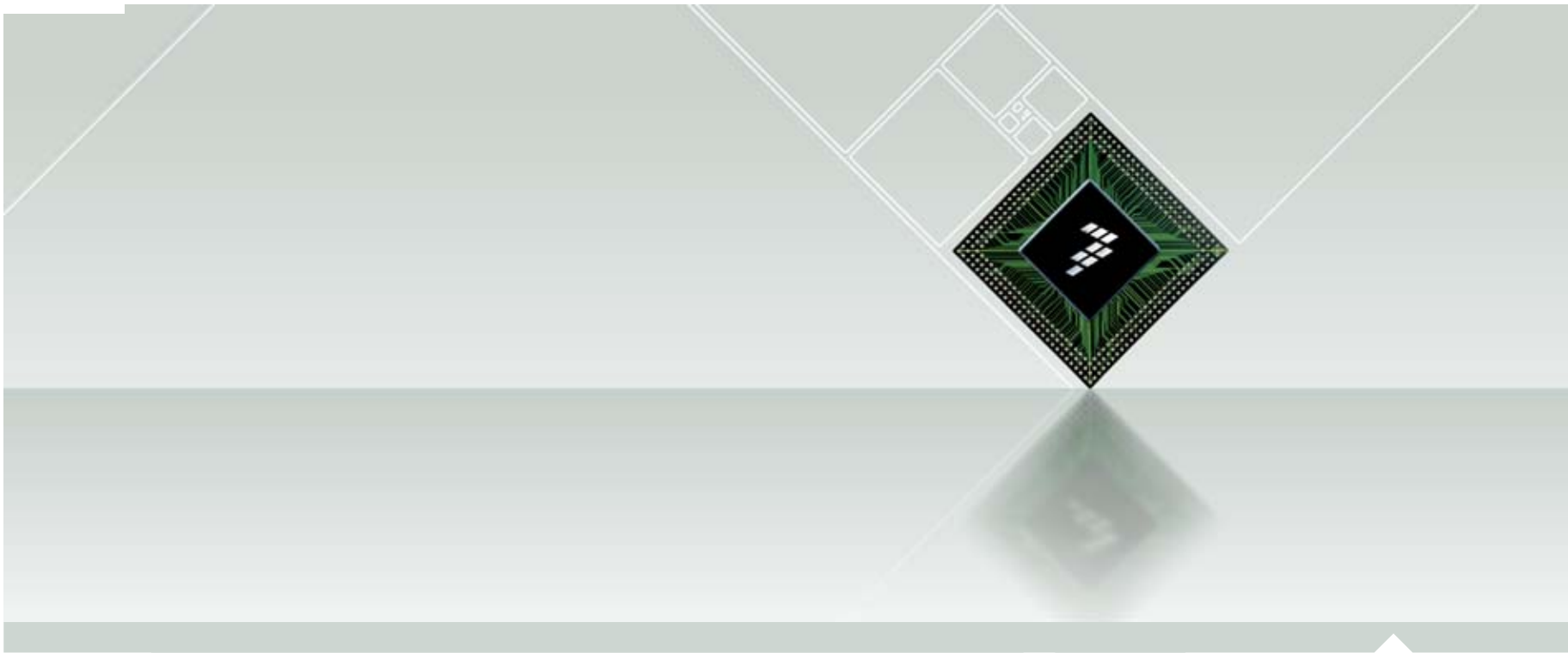
- Solution:
  - sharing an eTPU channel for Hall input and low-side PWM output
    - some eTPU channels have separate input and output pins

Example for ETPU_A channel 0 on MPC5534:
- channel input pin: GPIO114 (L3)
- channel output pin: GPIO179 (AB0)

eTPU channel assignment

Motor No	0	1	2	3
Hall A	0	3	6	9
Hall B	1	4	7	10
Hall C	2	5	8	11
PWM A Hi	12	15	18	24
PWM B Hi	13	16	19	25
PWM C Hi	14	17	20	26
PWM A Lo	0	3	6	9
PWM B Lo	1	4	7	22
PWM C Lo	2	5	8	27
Master CMPWX	28	29	30	31

freescale semiconductor



Easy-To-Use eTPU Tools

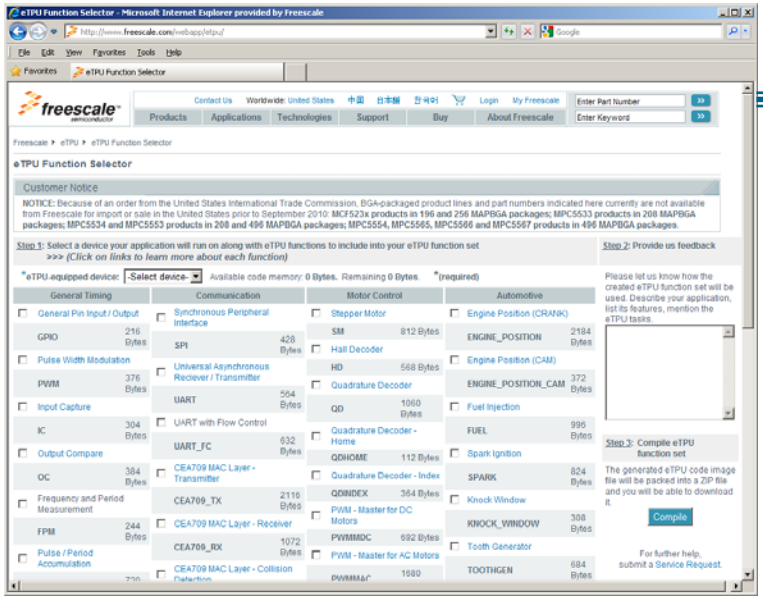
freescale semiconductor ™

Two Ways of Working with eTPU

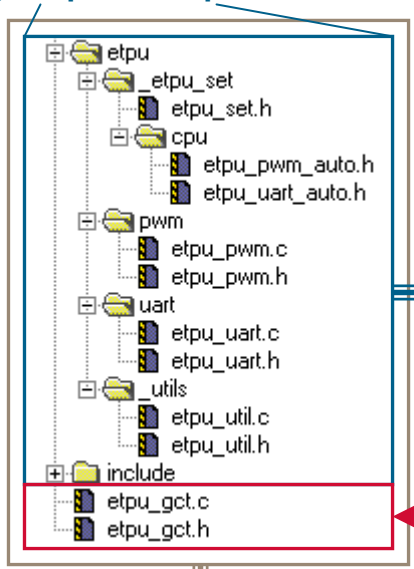
- eTPU User
  - uses ready-to-use eTPU functions
  - configures the eTPU module to run these functions

- eTPU Developer
  - writes eTPU code
    - modification of library functions
    - starting from scratch
  - debugs the developed code
  - creates eTPU API – CPU code to interface the eTPU function

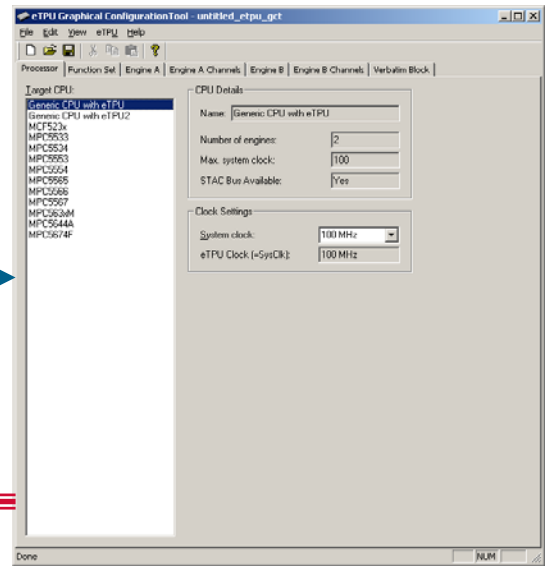
1. eTPU Function Selector web tool



etpu_set.zip

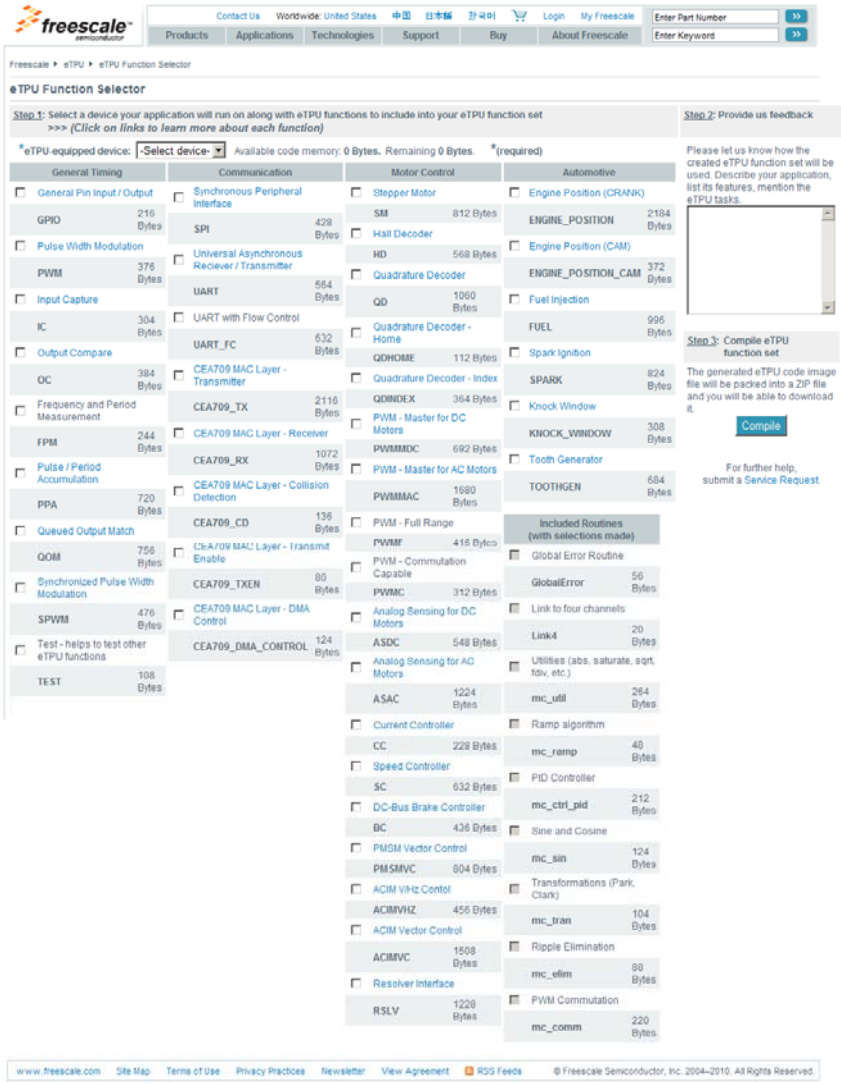


2. eTPU Graphical Configuration



3. Inclusion into application code
   - CodeWarrior or other application IDE

eTPU Function Selector – Web Tool



- Enables to create compiled binary image of eTPU function set – based on customer selection from all available eTPU functions
- Makes additional ready-to-use eTPU functions available
  - Resolver Interface
  - CEA709 functions
- Major update planned for Q4
  - updated eTPU functions
  - option to choose compiler:
    - ByteCraft
    - AshWare
    - Freescale

eTPU Function Selector
http://www.freescale.com/webapp/etpu/

freescale semiconductor

eTPU/eTPU2 Graphical Configuration Tool

User-friendly graphical environment to configure the eTPU

Supports all devices with eTPU - MCF5230, MPC5500, MPC5600

Generates Initialization Routines coded in C-language

Calculates eTPU engine load

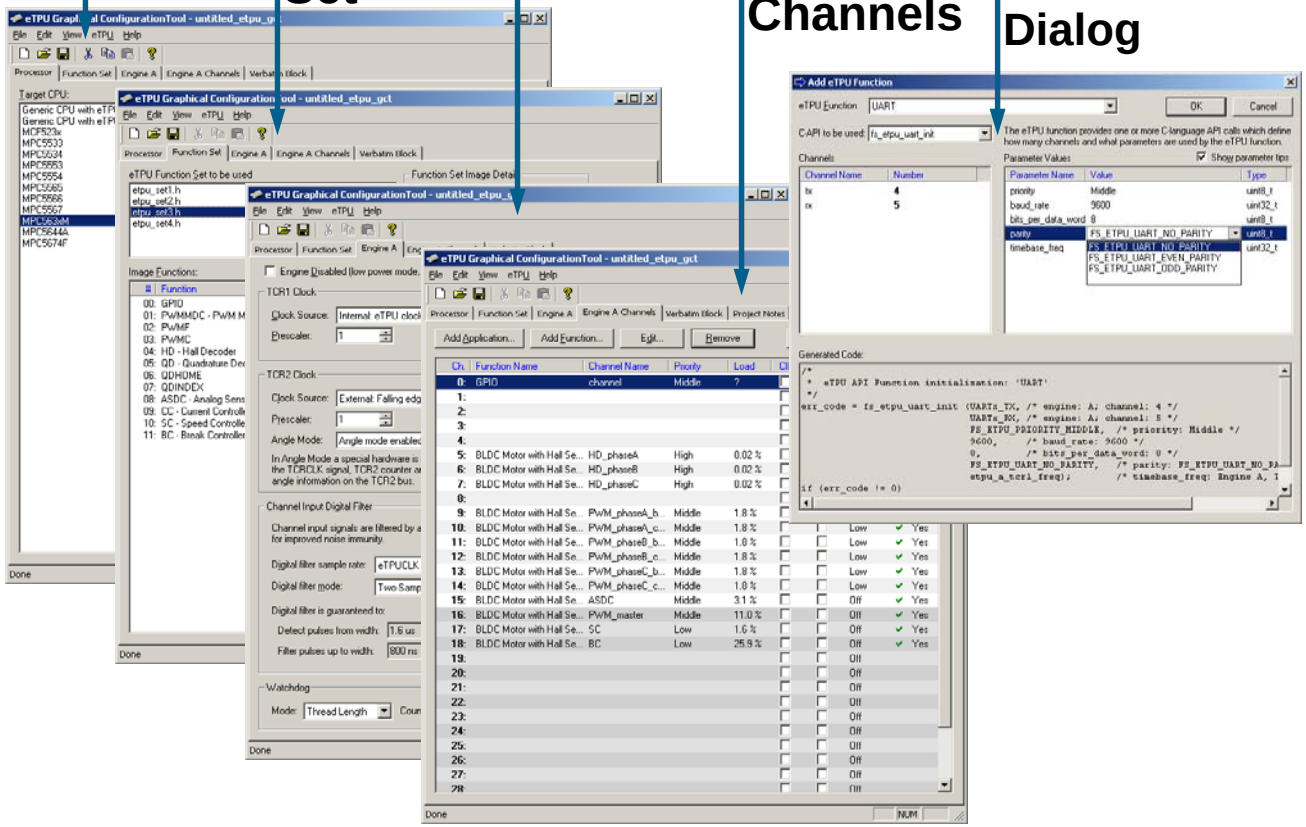
Select Device

Select eTPU Function Set

Setup Engine Settings

Assign eTPU Functions to Channels

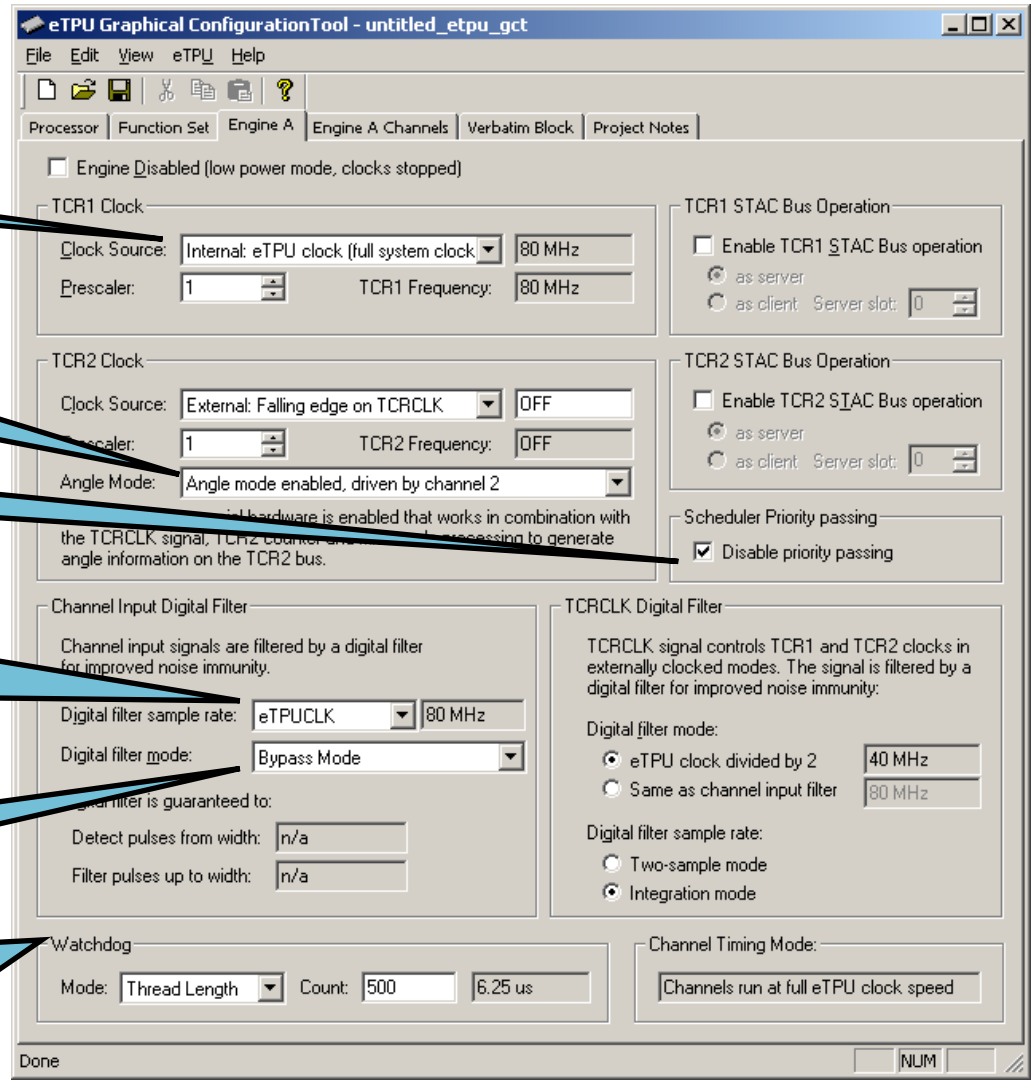
eTPU Function (UART) Configuration Dialog



freescale semiconductor

New eTPU2 Settings

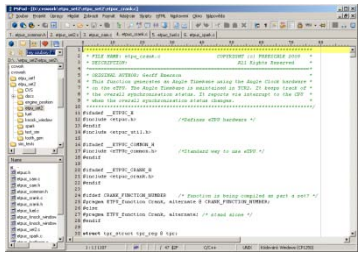
- TCR1 can run at full system clock speed
- More inputs to angle clock logic
- Priority passing can be disabled
- Digital filters on pins can run at full system clock speed
- Digital filters can be disabled
- Software watchdog with programmable timeout



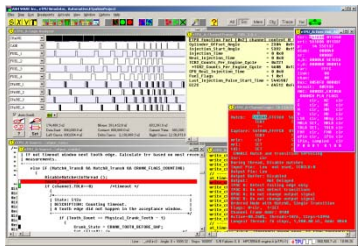
freescale semiconductor

eTPU Developer Tool Chain

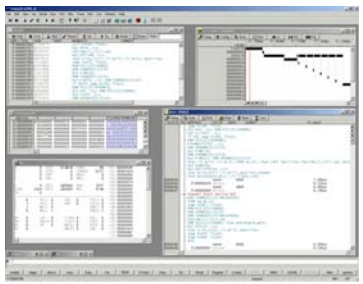
1. eTPU Compiler
   - ByteCraft
   - AshWare
   - Freescale



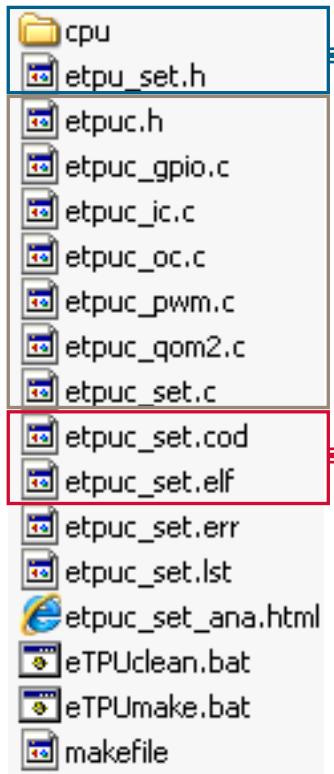
2. eTPU Simulator
   - AshWare
   - ByteCraft
   - Freescale



3. eTPU Debugger
   - Lauterbach
   - Freescale



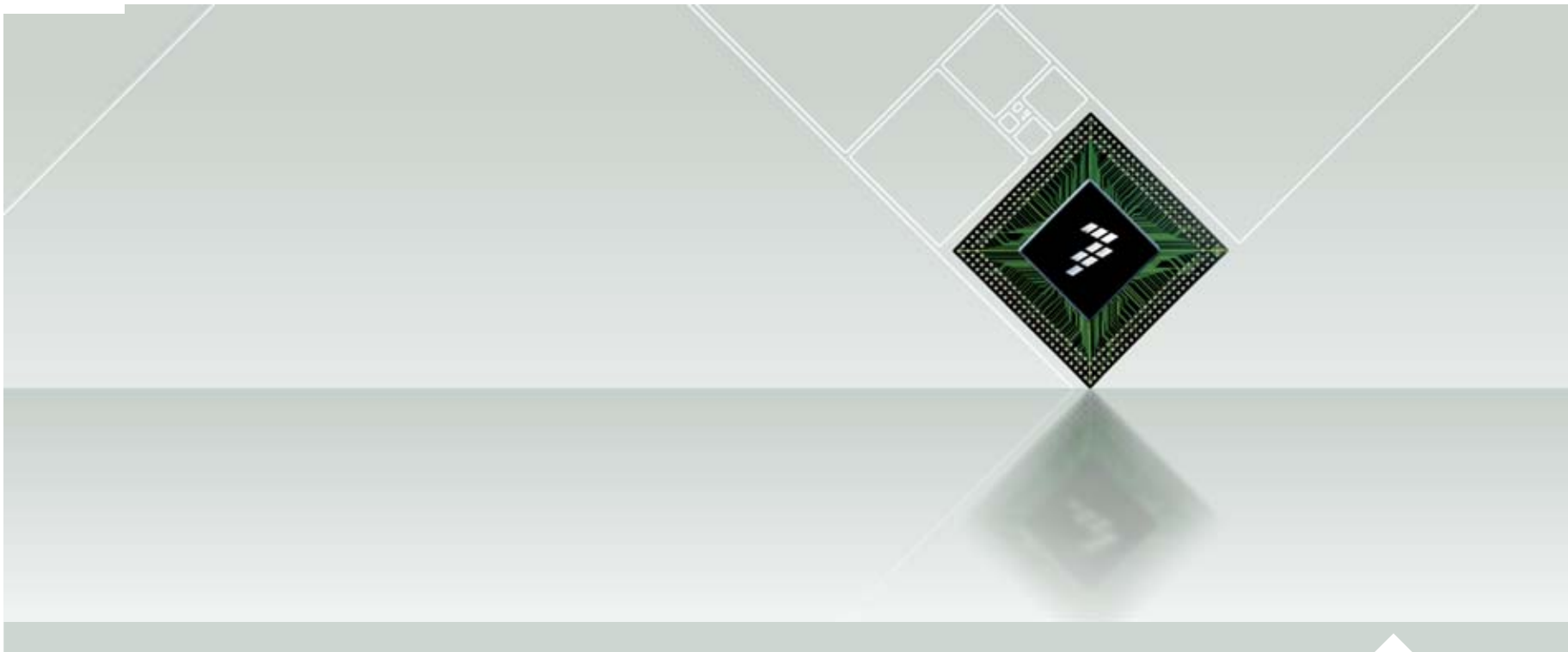
sTPU source code



CPU application

Simulator & Debugger

freescale semiconductor



eTPU Software and Documentation Available from Freescale

freescale semiconductor ™

eTPU Software and Documentation Available from Freescale

eTPU Product Summary Web Page
http://www.freescale.com/etpu

- General Documentation and Utilities
- eTPU Function Library and Application Interface (API)
  - General Timing eTPU Functions
  - Communication eTPU Functions
  - Automotive eTPU Functions
  - Motor Control eTPU Functions
- Example eTPU Applications

freescale semiconductor

General Documentation And Utilities

Item	Description	Download
ETPURM	Enhanced Time Processing Unit (eTPU) Reference Manual	-
ETPU2RMAD	eTPU Reference Manual Addendum	-
AN2353	The Essentials of Enhanced Time Processing Unit	-
AN2821	eTPU Host Interface	-
AN2848	Programming the eTPU	-
AN2864	General C Functions for the eTPU	AN2864SW
AN2897	Using the eTPU Angle Clock	-
AN2933	Understanding the eTPU Channel Hardware	-

eTPU Function Library and API – General Timing eTPU Functions

Item	Description	Download
AN2863	eTPU General Function Set (Set 1)	AN2863SW_GENERALSET
AN2849	Using the eTPU Pulse Width Modulation (PWM) Function	AN2849SW_PWM
AN2850	Using the General Purpose Input/Output (GPIO) eTPU Function	AN2850SW_GPIO
AN2851	Using the Input Capture (IC) eTPU Function	AN2851SW_IC
AN2852	Using the Output Compare (OC) eTPU Function	AN2852SW_OC
AN2854	Using the Synchronized Pulse-Width Modulation eTPU Function	AN2854SW_SPWM
AN2857	Using the Queued Output Match (QOM) eTPU Function	AN2857SW_QOM
AN2858	Using the Period and Pulse Accumulator (PPA) eTPU Function	AN2858SW_PPA
AN2859	Using the Frequency Pulse Measurement (FPM) eTPU Function API	AN2859SW_FPM
	set1 Test API	SET1_TEST_API

eTPU code **CPU code**

freescale semiconductor

eTPU Function Library and API – Communication eTPU Functions

Item	Description	Download
AN2863	eTPU General Function Set (Set 1)	AN2863SW_GENERALSET
AN2847	Using the Serial Peripheral Interface (SPI) eTPU Function	AN2847SW_SPI
AN2853	Using the Universal Asynchronous Receiver Transmitter (UART) eTPU Function	AN2853SW_UART
AN3379	Using the CEA709 eTPU Function	AN3379SW_CEA709SET

eTPU code **CPU code**

freescale semiconductor

eTPU Function Library and API – Automotive eTPU Functions

Item	Description	Download
AN3768	eTPU Automotive Function Set (Set 2)	AN3768SW_AUTOSET
AN3769	Using the Engine Position (CRANK and CAM) eTPU Functions	AN3769SW_ENGPOS
AN3770	Using the Fuel eTPU Function	AN3770SW_FUEL
AN3771	Using the Spark eTPU Function	AN3771SW_SPARK
AN3772	Using the Knock Window eTPU Function	AN3772SW_KNOCKWINDOW
AN3801	Using the Tooth Generator eTPU Function	AN3801SW_TOOTHGEN

eTPU code **CPU code**

freescale semiconductor

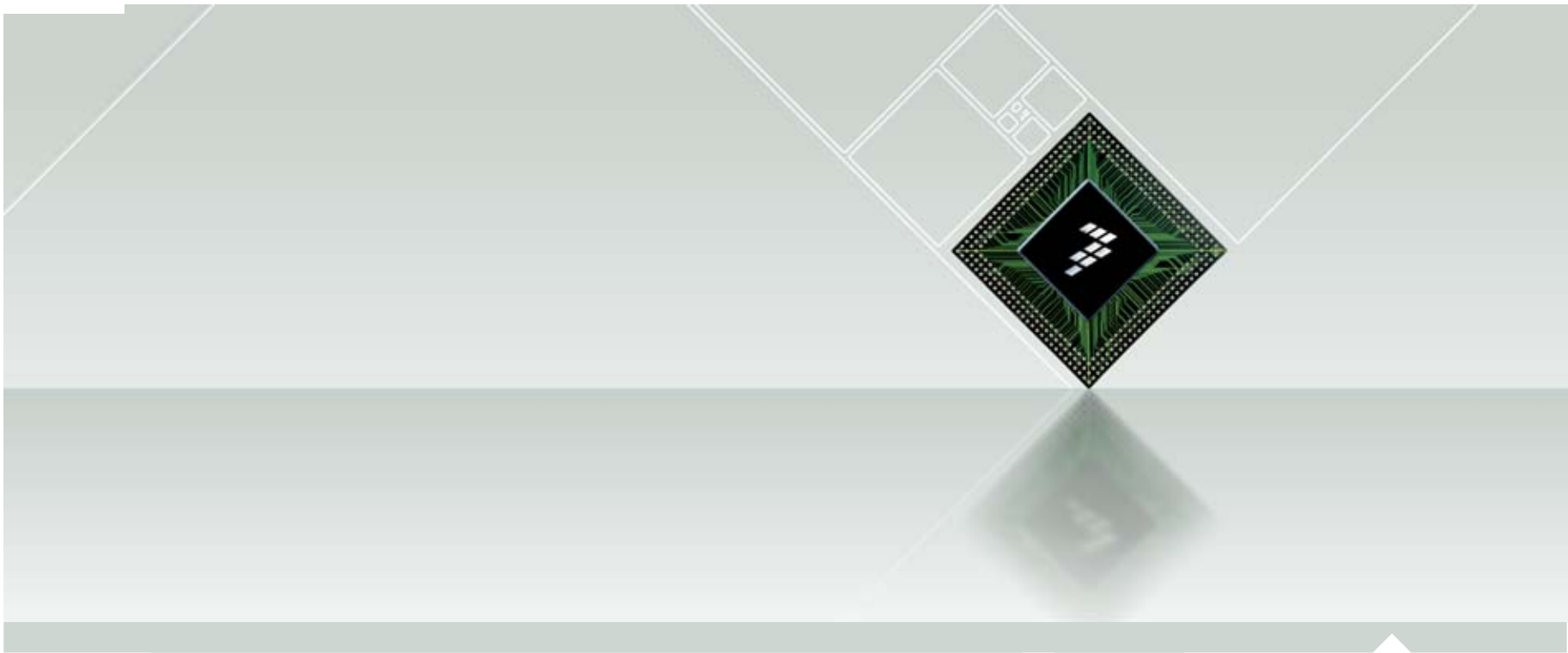
eTPU Function Library and API – Motor Control eTPU Functions

Item	Description	Download
AN2958	Using the DC Motor Control eTPU Function Set (set3)	AN2958SW_DCMOTORSET
AN2968	Using the AC Motor Control eTPU Function Set (set4)	AN2968SW_ACMOTORSET
AN2840	Using the DC Motor Control PWM eTPU Functions	AN2840SW_PWMMDC
AN2841	Using the Hall Decoder (HD) eTPU Function	AN2841SW_HD
AN2842	Using the Quadrature Decoder (QD) eTPU Function	AN2842SW_QD
AN2843	Using the Speed Controller (SC) eTPU Function	AN2843SW_SC
AN2844	Using the Current Controller (CC) eTPU Function	AN2844SW_CC
AN2845	Using the Break Controller (BC) eTPU Function	AN2845SW_BC
AN2846	Using the Analog Sensing for DC Motors (ASDC) eTPU Function	AN2846_ASDC
AN2869	Using the Stepper Motor (SM) eTPU Function	AN2869SW_SM
AN2969	Using the AC Motor Control PWM eTPU Functions	AN2969_PWMMAC
AN2970	Using the Analog Sensing for AC Motors (ASAC) eTPU Function	AN2970_ASAC
AN2971	Using the ACIM Volts per Hertz (ACIMVHZ) eTPU Function	AN2971_ACIMVHZ
AN2972	Using the PMSM Vector Control eTPU Function	AN2972_PMSMVC
AN2973	Using the ACIM Vector Control eTPU Function	AN2973_ACIMVC
AN3943	Using the ACIM Resolver Interface eTPU Function	AN3943SW

freescale semiconductor

Example eTPU Applications

Item	Description	Download
AN2892	3-Phase BLDC Motor with Speed Closed Loop, driven by eTPU on MCF523x	AN2892SW
AN2948	Three 3-Phase BLDC Motors with Speed Closed Loop, driven by eTPU on MCF523x	AN2948SW
AN2954	BLDC Motor with Speed Closed Loop and DC-Bus Break Controller, driven by eTPU on MCF523x	AN2954SW
AN2955	DC Motor with Speed and Current Closed Loops, driven by eTPU on MCF523x	AN2955SW
AN2957	BLDC Motor with Quadrature Encoder and Speed Closed Loop, Driven by eTPU on MCF523x	AN2957SW
AN3000	AC Induction Motor Volts per Hertz Control, Driven by eTPU on MCF523x	AN3000SW
AN3001	AC Induction Motor Vector Control, Driven by eTPU on MPC5500	AN3001SW
AN3002	Permanent Magnet Synchonous Motor Vector Control, Driven by eTPU on MCF523x	AN3002SW
AN3005	BLDC Motor with Quadrature Encoder and Speed Closed Loop, driven by eTPU on MPC5554	AN3005SW
AN3006	BLDC Motor with Hall Sensors and Speed Closed Loop, driven by eTPU on MPC5554	AN3006SW
AN3007	BLDC Motor with Speed Closed Loop and DC-Bus Break Controller, driven by eTPU on MPC5554	AN3007SW
AN3008	DC Motor with Speed and Current Closed Loops, Driven by eTPU on MPC5554	AN3008SW
AN3205	AC Induction Motor Volts per Hertz Control with Speed Closed Loop, Driven by eTPU on MPC5500	AN3205SW
AN3206	Permanent Magnet Synchronous Motor Vector Control, Driven by eTPU on MPC5500	AN3206SW
AN3769	Using the Engine Position eTPU Functions	AN3769SW



Summary

freescale semiconductor ™

Session Summary

- This session should have helped you to
- Get a good overview of the whole eTPU world
- Get familiar with eTPU usage in applications
  - performance, benefits
  - engine control and motor control applications
- Understand how to use the eTPU in your application
  - ready-to use library functions
  - development of customer functions


freescale™
semiconductor