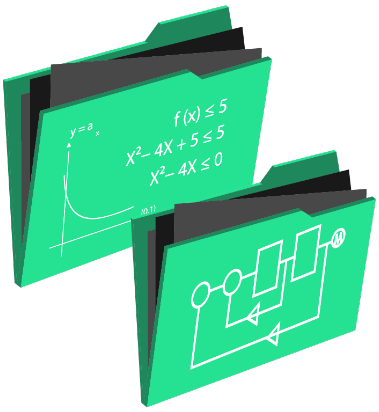
OLEA® LIB
Algorithms and function software libraries

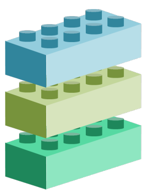


Rich collection of advanced software and automotive algorithms optimized for OLEA® FPCU, boosting application performance and enabling rapid time to market

- Configure and **customize** complex algorithms
- **Cutting edge** inverter, DC-DC converter and OBC control systems
- x40 computation **speed improvement** on advanced mathematical and computation functions
- Design and development **turnaround in minutes** with full integration into OLEA® COMPOSER

Customize and boost your control system

Boost performance: Software and algorithms included into OLEA® LIB have been optimized for OLEA® FPCU and take all the benefits of the hardware resources and accelerators available (mathematical units, DSP functions and standard peripherals) to deliver the highest achievable performance and integration.
Shorten development times : By using OLEA® LIB , developers drastically reduce the time required to develop, optimize, test and calibrate their algorithm's on OLEA® FPCU.
OLEA® LIB is packaged into three complementary levels of integration selectable upon the application needs:



OLEA® LIB System: Efficient System Functions
OLEA® LIB Algo: Specialized and Enhanced Algorithms
OLEA® LIB Math: Accelerated Mathematical Functions

Libraries comes as building blocks available as Reference and Target Models for MATLAB® Simulink, or as HDL pre-defined blocks, and tuned for best use of OLEA® FPCU. Models out of OLEA® LIB are directly usable within OLEA® COMPOSER for MiL simulations and automatic code generation.

OLEA® LIB Math

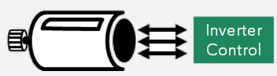
- Coordinate Rotation DIgital Computer
- Division operator
- Square Root operator
- Matrix Multiplier
- PID

OLEA® LIB Algo

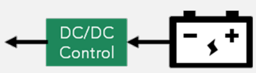
- Clarke and Park current transform
- Decoupling and Flux Weakening
- Inverse Park / Clarke voltage
- Space Vector Modulation PWM
- ID/IQ Regulations
- Motor Speed Regulation
- Tracking loop position estimator (Magnetic Resistive, Resolver...)
- Sensorless position estimation (Start-up/ Low-speed /High speed)
- Buck-Boost DC/DC regulation

OLEA® LIB System

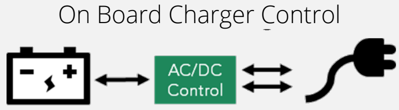
Inverter Control (FoC)



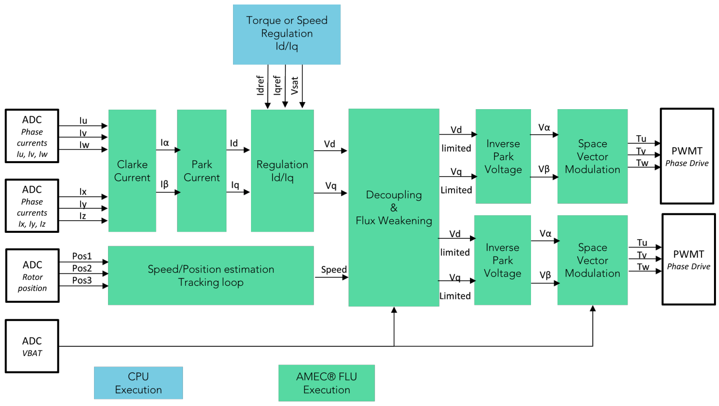
DC-DC Converter Control



On Board Charger Control



OLEA® LIB System Features

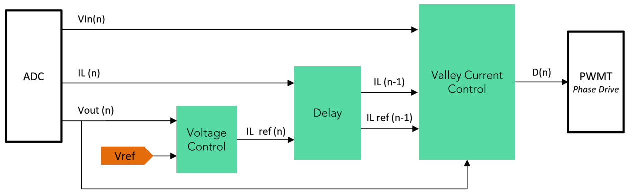


Inverter Control

Complete inverter control for PMSM or WRSM motors based on field oriented control and space vector modulation algorithms.

All system functions include:

- MATLAB / Simulink reference model
- MATLAB / Simuling target model ready for code generation



DC-DC Converter Control

Buck-Boost Valley Current control function supporting up to 6 DC-DC converters in parallel.

- Configurable parameters via GUI
- Diagnostic functions

OLEA® LIB Algorithm Features

- Speed regulation with D/Q-Axis control PI regulators with Anti Wind-up. DQ-axis Reference current computation
- Torque control with D/Q-Axis reference current computation
- Clarke Current: 3 to 2 phases or 6 to 2 phases current transformation
- Park Current: 2 phases current rotation from αβ to DQ framework
- Inverse Park Voltage: DQ framework reference voltage transform into αβ voltage space vector
- DQ-axis Decoupling and flux weakening
- IDQ regulation from torque set point
- Space Vector Modulation
- Position and speed estimation based on Tracking-loop algorithm
- Position and speed estimation: for standstill, low-speed and high-speed operating modes
- Buck-Boost valley current control
- Voltage control

OLEA® LIB Math Features

Operator	Description	Exec. Cycles	# of Operators*
CORDIC (COordinate Rotation DIgital Computer)	• $x \cdot \cos(\theta) - y \cdot \sin(\theta)$ • $y \cdot \cos(\theta) + x \cdot \sin(\theta)$ • $\text{atan}\left(\frac{y}{x}\right)$ • $\sqrt{x^2 + y^2}$ • $x \cdot \cosh(\theta) - y\ \sinh(\theta)$ • $y \cdot \cosh(\theta) + x\ \sinh(\theta)$ • $\text{atanh}\left(\frac{y}{x}\right)\ with\ \frac{y}{x} \in [-0{,}8; 0{,}8]$ • $\sqrt{x^2 - y^2}\ with\ \frac{y}{x} \in [-0{,}8; 0{,}8]$	Resolution in bit + 4	• 6 in parallel
Division	$A/B = Quotient\ with\ remainder$	26	• 3 in parallel
Square root	• $\sqrt{R}\ in\ unsigned\ mode$ • $\sqrt{\lvert R \rvert}\ in\ signed\ mode$	2	• 3 in parallel
Matrix Multiplier	• $\begin{bmatrix} r_0 \\ r_1 \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 & a_4 & a_5 \\ a_6 & a_7 & a_8 & a_9 & a_{10} & a_{11} \end{bmatrix} \times \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$ • $r_0 = \sum_{i=0}^{iter}(a_i \times b_i \gg Q_f)$ • $r_1 = \sum_{i=0}^{iter}(a_{i+6} \times b_i \gg Q_f)$	Iter + 4	• 3 in parallel
PID (Proportional Integral Derivative controller)	Saturation with Anti-windup: ▪ Back calculation : $if\ saturation\ then\ integral_n = K_i \times e_n - K_b\ (pid_{n-1} - pid\_sat_{n-1}) + integral_{n-1}$ ▪ Integral clamping : $if\ saturation\ and\ sign(pid_{n-1}) = sign(e_{n-1})\ then\ integral_n = integral_{n-1}$	8	• 6 in parallel

*OLEA LIB Math is using hardware dedicated resources available in OLEA

Silicon Mobility

www.silicon-mobility.com
sales@silicon-mobility.com
535 Route des Lucioles
Les Aqueducs – Bâtiment 2
06560 Sophia Antipolis
France

Legal Disclaimer: The information given in this Brief shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Silicon Mobility hereby disclaims any and all warranties and liabilities of any kind, including without limitation, warranties of non-infringement of intellectual property rights of any third party.
©2021 Silicon Mobility. All trademarks are property of their respective owner.