Advanced Features of MLX/SLX Series Modules

Applicable to MLX160, MLX120, MLX080, MLX040, SLX 160 and SLX040

The MLX/SLX series of Digital DLynxIII™ power module provide advanced features which can be used to configure modules for atypical applications and optimize performance in routine conditions. These modules use an advanced PID based adjustable digital control loop which ensures loop stability, provides fast transient response and reduces amount of required output capacitance. The module also provides control and adjustment of the availability of the internal power stages.

This document will provided instructions on how to achieve the following:

- Phase Addition / Shedding, Diode Emulation mode
- Security Setting
- Adaptive Transient Algorithm
- Loop Tuning

Some of the tools that are used to facilitate these features are:

**Digital Power Insight (DPI)**
OmniOn offers a software tool that helps users evaluate and simulate the PMBus performance of the MLX series modules without the need to write software. The software can be downloaded for free at omnionpower.com.

An OmniOn USB to I2C adapter and associated cable set are required for proper functioning of the software suite. For first time users, we recommend using the OmniOn’s DPI Evaluation Kit, which can be purchase from any of the leading distributors. Please ensure the OmniOn USB to I2C adapter being used/purchased is Version 2.2 or higher.

**Power Module Wizard**
OmniOn offers a free web based easy to use tool that helps users simulate the Tunable Loop performance of the DJT090A0X43-SRPZ. Go to omnionpower.com and sign up for a free account and use the module selector tool. The tool also offers Simplis models that can be used to assess transient performance, module stability, etc.

**NOTE** — For clarity OmniOn DPI GUI screenshots have been used to demonstrate the advanced features. Users can use their own I2C/PMBus tools to program the MLX modules to achieve the same results.
1) Phase Addition / Shedding

Phase Addition / Shedding can be set through a combination of the Power Mode (0x34) and Dynamic Phase Control Commands available through the D0 command series.

**POWER_MODE [0x34] :** Sets power state of the Module as follows:

<table>
<thead>
<tr>
<th>Format</th>
<th>16-bit unsigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit Position</td>
<td>15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</td>
</tr>
<tr>
<td>Access</td>
<td>R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W R/ W</td>
</tr>
<tr>
<td>Max Efficiency (automatically enables Diode emulation when current drops below threshold)</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>Max Power – Max configured phases operate (Default)</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1</td>
</tr>
<tr>
<td>Power State1-Commands phases to drop to 1 or 2 phases</td>
<td>0 0 0 0 0 0 0 0 0 0 0 1 0 0</td>
</tr>
<tr>
<td>Power State2-Commands phases to drop to 1 phase diode emulation mode</td>
<td>0 0 0 0 0 0 0 0 0 0 0 1 0 1</td>
</tr>
</tbody>
</table>

Apart from [0x34] there are other options in the [0xD0] register setting that allow for:

**DPC**—Dynamic Phase Control

**Auto_PS**—Automatic Power State Mode

**DE**—Diode Emulation Mode
1) Phase Addition / Shedding (continued)

**POWER_MODE [0x34]**: Setting options using OmniOn DPI software are as follows:

OmniOn DPI GUI screenshot shown above. Similar screenshots appear elsewhere in the document.

- **Power Mode Selection**:
  0 ➔ Max Efficiency
  3 ➔ PS0 Max Power
  4 ➔ PSI, 1 or 2 phase running
  5 ➔ PS2, Diode Emulation
1) Phase Addition / Shedding (continued)

Dynamic Phase Control (DPC): How does it work?

DPC sets the current thresholds at which different Power Phases/Stages are enabled/disabled. This allows the controller to run the module at higher efficiency levels even when the module is lightly loaded. For an 8 Phase module which corresponds to a MLX160 + SLX160 on a single output, the settings could be as shown below to achieve a good match of higher efficiency and spare available capacity as the loading changes.

In the table below the register setting value translates to Amps by multiplying the register value by a factor of 2. So 10 corresponds to a value of 20A and 15 corresponds to a value of 30A. The L**_P**_delta value is added to the previous threshold to arrive at the threshold for the next phase to be turned on.

<table>
<thead>
<tr>
<th>Command</th>
<th>Register Setting</th>
<th>Action</th>
<th>Current Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loop1_Phase1_thresh</td>
<td>10</td>
<td>Phase 1 → 2</td>
<td>20A</td>
</tr>
<tr>
<td>Loop1_Phase2_delta</td>
<td>15</td>
<td>Phase 2 → 3</td>
<td>50A</td>
</tr>
<tr>
<td>Loop1_Phase3_delta</td>
<td>15</td>
<td>Phase 3 → 4</td>
<td>80A</td>
</tr>
<tr>
<td>Loop1_Phase4_delta</td>
<td>15</td>
<td>Phase 4 → 5</td>
<td>110A</td>
</tr>
<tr>
<td>Loop1_Phase5_delta</td>
<td>15</td>
<td>Phase 5 → 6</td>
<td>140A</td>
</tr>
<tr>
<td>Loop1_Phase6_delta</td>
<td>15</td>
<td>Phase 6 → 7,8</td>
<td>170A</td>
</tr>
</tbody>
</table>

![Phase Number vs Output Current Graph](image)
1a) PS0 Max Power mode and Phase Addition/Shedding

Using an example of a 4-Phase Module (MLX160), the module can be set for all phases ON/no Phase Shedding operation as follows:

Since all the turn-on thresholds are 0A, all the phases will be ON regardless of load. Scope shot below shows the PWM signals as the load is increased. All Phases are enabled regardless of load current.
1a) PS0 Max Power mode and Phase Addition/Shedding (continued)

Using MLX160 module in Max Power mode, the phase transitions can be set at 24A, 24+18, 24+18+18A as shown.

Scope shot below shows the corresponding increase in PWM signal activity as the load crosses above setpoints.
1a) PS0 Max Power mode and Phase Addition/Shedding (continued)

Scope shot below the reduction in PWM signal activity as the load decreases past load shedding setpoints
1a) PS0 Max Power mode and Phase Addition/Shedding (continued)

Efficiency improvement with Phase Addition and Shedding on MLX160 using the 24-18-18A thresholds for turning on and off phases. There is a noticeable improvement in efficiency in the <30% rating of the MLX160.
1b) PS1 1 Phase Power mode and Phase Addition / Shedding

Using an example of a 4-Phase Module (MLX160) + 2–single phase (SLX040), the modules can be set to trigger from single phase operation to all 6 phases operating. These 2 parameters—1phase and 6 total phases are set using the GUI as shown below:

![GUI screenshot showing power mode settings]

PWM signals below show that as the load (Iout signal) is increased beyond 20A, all Phases are enabled.

![Graph showing PWM signals with load increase]

```
Advanced DO Commands

LOOP1_PHASE_ACT1
VE_PS1: The number of active operating phases in PS1 mode (when Load current less than 20A)

LOOP1_PHASE_ACT1
VE_MAX: The maximum number of phases that can be active on loop 1 = LOOP1_PHASE_ACT1
VE_MAX + 1
```
1b) PS1 - 2 Phase Power mode and Phase Addition/Shedding

Using an example of a 4-Phase Module (MLX160) + 2–single phase (SLX040), the modules can be set to trigger from **dual** phase operation to all 6 phases operating. These 2 parameters—2 phases and 6 total phases are set using the GUI as shown below:

Initially 2 phases active, PWM signals below show that as the load is increased beyond 20A, all Phases are enabled.
1c) PS2 - Diode Emulation(DE) and Phase Addition/Shedding

The module provides different options for the Diode Emulation Mode. Depending on the application, the module can be set up from a simple skip mode under no load to setting inductor negative current threshold, pulse on time, etc.

First option is setting the Inductor current setting to operate the module in diode emulation mode

![Image of current and time graph](image)

After setting Loop1_Inductor_ni_thresh= 16 which corresponds to 5A, Loop1_auto_PS_mode set to 1 and Power_mode set to 5 then when the total current is below 5A the power module will operate in PS2 diode emulation mode. When total module is over 5A, module will run in PS1/PS0 mode.

Next options are to set Pulse On Time, Pulse OFF time or the Error Threshold turn-on

![Image of pulse on time and error threshold](image)

These setting options are explained through the GUI tool in the following few pages.
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

Using an example of a 2 Phase Module (MLX080), the module can be set to operate in low duty cycle skip mode for no load conditions. Apart from the Power_Mode setting, 2 more sub settings are required as shown below:

1. Inductor current negative component threshold has to be set above 0
2. Automatic Power State Mode has to be disabled

Next choose an NI Threshold >0A from the many options.

Set NI_THRESH > 0 to enable PS2. Note that actual values of NI_THRESH are disabled until AUTO_PS_MODE is enabled.

Next Disable Automatic Power State Mode so that the above value of NI_THRESH is not enabled.
1c) PS2 - Diode Emulation and Phase Addition/Shedding

Finally we enter the number of available power phases based on the module being used. Since MLX080 is being used for the scope capture the D0 register is set for max 2 phases.

Scope capture when the module is powered with no load
1c) PS2 - Diode Emulation and Phase Addition/Shedding

Scope capture showing PWM activity with varying load—large/slow 5ms timescale

In the above setting once the load cycle occurs module stays in PS0 mode unless DEM is again manually set. Next we can look at setting the module using an automatic mode where the module goes back to DEM once the load is removed.
1c) PS2 - Diode Emulation and Phase Addition/Shedding

The module can also use the inductor negative current threshold to drive the diode emulation mode beyond no load. For that AUTO_PS_MODE register has to be enabled. Once the AUTO mode is enabled it has the additional advantage that module goes back to PS2 mode operation after every load cycle, or else it would have remained in PS0 mode.

Also now the INDUCTOR_NI_THRESH value comes into play.

Scope capture when the module is powered with NI Threshold set to 0.25A
1c) PS2 - Diode Emulation and Phase Addition/Shedding

Scope capture when the module is powered with NI Threshold set to max value of 15.75A

The benefit of using the automatic power state mode is that the module goes back to PS2 (DE) once load is removed
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

Using an example of a single Phase Module (MLX040), the module can be set to operate in low duty cycle skip as shown before.

Next the ON/OFF times is doubled.

**Advanced D0 Command**

Double DE Pulse Width
1 = doubles ON/OFF times for diode emulation. Used when using large L & C.
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

The following scope capture shows how the Double DE Pulse Width can affect the output voltage waveform.
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

Diode Emulation can also be programmed to set the error threshold in the output voltage for triggering a pulse. This controls the amount of deviation permitted on the bus during DE mode.

A pulse is generated as low threshold is broken.

Next Threshold is raised to 12mV

New threshold = 1 - 0.012 ~ 0.988 Volts out
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

Another option is to reduce the Off time of pulses when module is in DE mode.

Next Threshold is raised to 375.3ns.
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

Option to run module in skip mode and turn on all phases once voltage dips below error threshold. Set as follows:

- Disable AUTO_PS_MODE for test so that PS2 exit is not triggered by current.
- Must be set > 0 to activate PS2 mode.

**Note:** Error threshold to go from discontinuous to continuous mode. Date:

<table>
<thead>
<tr>
<th>Index</th>
<th>Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6 mV</td>
</tr>
<tr>
<td>2</td>
<td>12 mV</td>
</tr>
<tr>
<td>3</td>
<td>18 mV</td>
</tr>
<tr>
<td>4</td>
<td>24 mV</td>
</tr>
</tbody>
</table>

+/- 20mV threshold
1c) PS2 - Diode Emulation (DE) and Phase Addition/Shedding

If threshold is not triggered module runs in discontinuous model.
2) Logic level settings for Enable Pin

The MLX series modules offer the option to switch between TTL and LVT logic levels for the enable pin. This can be done through the D0 register setting as shown below:

<table>
<thead>
<tr>
<th>Command Name and explanation in parenthesis</th>
<th>Application: Common, Loop1 or Loop2</th>
<th>Description, Range</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>d2p_enable_LVT_Thresh (Sets the input threshold level)</td>
<td>D0 0048 [15:15] COMMON</td>
<td>0 (Sets the input threshold level TTL for the EN input pads.) 1 (Sets the input threshold level LVT for the EN input pads.)</td>
<td>0</td>
</tr>
</tbody>
</table>

MLX modules also offer the option of sequencing the outputs when Enable Pins are used to control outputs

<table>
<thead>
<tr>
<th>En_delay_mode</th>
<th>Description</th>
<th>Loop1 starts</th>
<th>Loop2 starts</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Independent ENs</td>
<td>After Loop1 Enable pin</td>
<td>After Loop2 Enable pin</td>
</tr>
<tr>
<td>1</td>
<td>Shared EN</td>
<td>After (Loop1 Enable pin + En_delay_time)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>L1 EN ➤L2</td>
<td>After Loop1 Enable pin</td>
<td>After (Loop1 Enable pin + En_delay_time)</td>
</tr>
<tr>
<td>3</td>
<td>L2 EN ➤L1</td>
<td>After (Loop2 Enable pin + En_delay_time)</td>
<td>After Loop2 Enable pin</td>
</tr>
<tr>
<td>4</td>
<td>L1 PG ➤L2</td>
<td>After Loop1 Enable pin</td>
<td>After (Loop1 PowerGood + EN_delay_time)</td>
</tr>
<tr>
<td>5</td>
<td>L2 PG ➤L1</td>
<td>After (Loop2 PowerGood + EN_delay_time)</td>
<td>After Loop2 Enable pin</td>
</tr>
<tr>
<td>6,7</td>
<td>OFF</td>
<td>Reserved</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EN_delay_time</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delay</td>
<td>No delay</td>
<td>0.25ms</td>
<td>0.5ms</td>
<td>1.0ms</td>
<td>2.5ms</td>
<td>5.0ms</td>
<td>10ms</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
3) Security Settings—Write and Read Protection

- The MLX series modules provide multiple register read and write access protection mechanisms.
- All protection mechanisms must be disabled in order to access a protected register.
- Password-based access protection is enabled by setting the protection mode and PASSWORD.
- The password values cannot be read, will always return 0xFFFF when these registers are read.
- Once a password is programmed to a non-zero value, the user must program matching password value in the user_try_password register.
- User can attempt to program the correct value into user_try_password up to 4 times, after which they will be locked out. However, the password attempt count is cleared when power is cycled.

- Password-based access protection is enabled by setting the protection mode and PASSWORD
- The password values cannot be read, will always return 0xFFFF when these registers are read
- Once a password is programmed to a non-zero value, the user must program matching password value in the user_try_password
- The operator can attempt to program the correct value into user_try_password up to 4 times, after which they will be locked out, the register

**password attempt count is cleared when power is cycled**

**Common Security register settings:**

**COMMON_WRITE_PROTECT_MODE:**
- 0—Password
- 1—Lock Forever

**COMMON_READ_PROTECT_MODE**
- 0- Password
- 1- Lock Forever

**COMMON_WRITE_PROTECT_SELECTION**
- 0 - No Protection
- 1- Protect configuration, OTP_CNFG, OTP_TRIM,OTP_USR, PMBus registers*
- 2- Reserved
- 3- Protect all, For all USER register

**COMMON_READ_PROTECT_SELECTION**
- 0- No Protection
- 1- Protect configuration, OTP_CNFG, OTP_TRIM,OTP_USR, PMBus registers*
- 2- Protect all but telemetry,
- 3- Protect all, all CNFG, TRIM, and USER register

**COMMON_USER_PASSWORD**
- A 16-bit password that provides read/write protection for the User registers*

**COMMON_USER_TRY_PASSWORD**
- Used to access user registers when the password matches to user_try_password values

* Note : User Registres listed on next page
### 3) Security Settings—Register Listing

#### User Register Categories

<table>
<thead>
<tr>
<th>REG Section</th>
<th>Start</th>
<th>End</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common - OTP_CNFG Registers</td>
<td>0x0000</td>
<td>0x0002</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Loop1 - OTP_CNFG Registers</td>
<td>0x0400</td>
<td>0x0402</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Loop2 - OTP_CNFG Registers</td>
<td>0x0800</td>
<td>0x0802</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Common - OTP_Trim Registers</td>
<td>0x0008</td>
<td>0x001C</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Loop1 - OTP_Trim Registers</td>
<td>0x0408</td>
<td>0x041C</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Loop2 - OTP_Trim Registers</td>
<td>0x0808</td>
<td>0x081C</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Common - Read_Write_Registers</td>
<td>0x0080</td>
<td>0x00A6</td>
<td>COMMON_PHASE_GATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COMMON_LOOP1_SELECT_PHASE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COMMON_LOOP2_SELECT_PHASE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COMMON_DEBUG_LOCK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COMMON_PHASE_GATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COMMON_IOUT_CALIBRATION_EN</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>COMMON_USER_TRY_PASSWORD</td>
</tr>
<tr>
<td>Loop1 - Read_Write_Registers</td>
<td>0x0480</td>
<td>0x04A6</td>
<td>ABB used Only</td>
</tr>
<tr>
<td>Loop2 - Read_Write_Registers</td>
<td>0x0880</td>
<td>0x08A6</td>
<td>ABB used Only</td>
</tr>
</tbody>
</table>
### 3) Security Settings—Register Listing

**User Register Categories (continued)**

<table>
<thead>
<tr>
<th>REG Section</th>
<th>Start</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common OTP_USER Registers</td>
<td>0x0020</td>
<td>0x005C</td>
</tr>
<tr>
<td>COMMON_I2C_DEVICE_ADDR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PMB_DEVICE_ADDR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_IMON_MAX_CODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_TELEMETRY_BW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_READ_INOUT_SCALE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE_ACTIVE_PS1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE_ACTIVE_MAX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP2_PHASE_ACTIVE_PS1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP2_PHASE_ACTIVE_MAX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE_ACTIVE_PS1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE_ACTIVE_MAX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP2_PHASE_ACTIVE_PS1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP2_PHASE_ACTIVE_MAX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE1_THRESH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE2_DELTA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE3_DELTA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE4_DELTA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE5_DELTA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP1_PHASE6_DELTA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP2_PHASE1_THRESH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_LOOP2_PHASE2_DELTA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_WRITE_PROTECT_MODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_READ_PROTECT_MODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_WRITE_PROTECT_SELECTION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_READ_PROTECT_SELECTION</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_FIXED_MEASURED_IIN_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_DISABLE_OUTPUT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_EN_DELAY_MODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_EN_DELAY_TIME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY2_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY3_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY4_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY5_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY6_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY7_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_PHY8_CURRENT_OFFSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_ISNS_USER_GAIN_PHASE_8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_D2P_ENABLE_LVT_THRESHOLD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMON_USER_PASSWORD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### 3) Security Settings—Register Listing

**User Register Categories** (continued)

<table>
<thead>
<tr>
<th>REG Section</th>
<th>Start</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loop1</td>
<td>0x0420</td>
<td>0x045C</td>
</tr>
</tbody>
</table>

- LOOP1_RELATIVE_OVP_THRESH_EN
- LOOP1_RELATIVE_OVP_THRESH
- LOOP1_RELATIVE_UVP_THRESH_EN
- LOOP1_RELATIVE_UVP_THRESH
- LOOP1_TSEN_FAULT_EN
- LOOP1_TSEN_FAULT_SHUTDOWN
- LOOP1_PID_KP
- LOOP1_PID_KI
- LOOP1_PID_KD
- LOOP1_PID_KPOLE1
- LOOP1_PID_KPOLE2
- LOOP1_FC_D
- LOOP1_FC_HTH
- LOOP1_FC_SHAPE
- LOOP1_FC_P
- LOOP1_V_LIFT
- LOOP1_DB_DURATION
- LOOP1_ERR_I_TH
- LOOP1_FC_SLOPE_TH
- LOOP1_DIODE_BRAKE
- LOOP1_BBRK_FREQ_TH
- LOOP1_LOADLINE_BW
- LOOP1_PSI_OC_EN
- LOOP1_PI_FAULT_EN
- LOOP1_DIODE_EMU_X2
- LOOP1_DIODE_EMU_PW
- LOOP1_DIODE_EMU_THRESH
- LOOP1_DE_OFF_TIME_ADJ
- LOOP1_LE_TH
- LOOP1_AUTO_PS_MODE
- LOOP1_INDUCTOR_NI_THRESH
- LOOP1_TEMPERATURE_OFFSET
- LOOP1_IIN_PER_PHASE_OFFSET
- LOOP1_FIXED_IIN_OFFSET

Category above also applies to Loop 2 0x0820 to 0x085C
3) Security Settings—Register Listing

User Register Categories (continued) -

<table>
<thead>
<tr>
<th>REG Section</th>
<th>Start</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loop1 Pmbus</td>
<td>0x00</td>
<td>0xD6</td>
</tr>
</tbody>
</table>

- PAGE
- OPERATION
- VOUT_COMMAND
- VOUT_MODE
- VOUT_TRIM
- POWER_MODE
- VOUT_MAX
- VOUT_MIN
- VOUT_MARGIN_HIGH
- VOUT_MARGIN_LOW
- VOUT_TRANSITION_RATE
- VOUT_DROOP
- VOUT_RESET
- RESET_TRANSITION_RATE
- WRITE_PROTECT
- FREQUENCY_SWITCH
- IOUT_CAL_OFFSET
- IOUT_CAL_GAIN
- ON_OFF_CONFIG
- VIN_ON
- VIN_OFF
- POWER_GOOD_ON
- POWER_GOOD_OFF
- TON_DELAY
- TON_RISE
- TOFF_DELAY
- TOFF_FALL
- TON_MAX_FAULT_LIMIT
- TON_MAX_FAULT_RESPONSE
- VOUT_OV_FAULT_LIMIT
- VOUT_OV_FAULT_RESPONSE
- VOUT_OV_WARN_LIMIT
- VOUT_OV_FAULT_RESPONSE
- VOUT_OV_FAULT_RESPONSE
- VOUT_OV_WARN_LIMIT
- IOUT_OC_FAULT_LIMIT
- IOUT_OC_FAULT_RESPONSE
- IOUT_OC_WARN_LIMIT
- OT_FAULT_LIMIT
- OT_FAULT_RESPONSE
- OT_WARN_LIMIT
- VIN_OV_FAULT_LIMIT
- VIN_OV_FAULT_RESPONSE
- VIN_OV_WARN_LIMIT
- IN_OC_WARN_LIMIT
- POUT_OP_WARN_LIMIT
- FIN_OP_WARN_LIMIT
- SMBALERT_MASK_STATUS_VOUT
- SMBALERT_MASK_STATUS_IOUT
- SMBALERT_MASK_STATUS_INPUT
- SMBALERT_MASK_STATUS_TEMPERATURE
- SMBALERT_MASK_STATUS_CML
- SMBALERT_MASK_STATUS_MFR_SPECIFIC

Category above also applies to Loop 2 0x00 to 0xD6
3) Security Settings—Write and Read Protection

Process to set a password for write/read protection:

Step 1: set common_write_protection_mode=0 as password protection mode

Step 2: select common_write_protection_selection=3 to protect all user registers

Step 3: set common_read_protection_mode=0 as password protection mode

Step 4: select common_write_protection_selection=2 to protect all user registers but telemetry

Step 5: create a password (range 0-65535), default password is 62235 (please keep your password safe)

Process to access register value under write/read protection:

User Defined - COMMON_USERTRY_PASSWORD - Type the password created before

- Password-based access protection is enabled by setting the protection mode and PASSWORD
- The operator can attempt to program the correct value into user_try_password up to 4 times, after which they will be locked out, the register password attempt count is cleared when power is cycled
4) Control Loop—ATA and PID Tuning

The MLX series modules have 2 control loops:

- Adaptive Transient Algorithm (ATA) is a wideband non-linear control loop which can react faster to load transients and ensures that the output voltage is within the regulation limits even during fast dynamic load and voltage change events.

- A linear Proportional-Integral-Derivative (PID) digital controller on the Dylinx III family provides loop compensation for the system regulation. The Digital compensator processes the digitized error voltage coming from the high-speed voltage error ADC. The MLX has 2 identical and independent loops to control 2 independent outputs if configured that way. The PID loop operates slower than the ATA Loop. The transfer function of the compensator is:

\[
(Kp + \frac{Ki}{s} + Kd \cdot s) \cdot \left(\frac{1}{1 + \frac{s}{\omega p_1}}\right) \cdot \left(\frac{1}{1 + \frac{s}{\omega p_2}}\right)
\]

- \(Kp\) = Proportional Coefficient
- \(Ki\) = Integral Coefficient
- \(Kd\) = Derivative Coefficient
- \(P1\) = Configurable filter 1 pole
- \(P2\) = Configurable filter 2 pole

The 2 poles are designed to filter and roll off the high frequency gain that \(Kd\) coefficient generates.
4a) Control Loop—ATA Tuning

- The ATA Loop is triggered once the magnitude (/FC_HTH) and Slope Thresholds (/FC_Slope_TH) are exceeded.
- Next the shape term (/FC_Shape) kicks in and once the slope falls below threshold the shape term is disengaged.
- The P Term (/FC_P) multiplies the magnitude of V_error.
- The D Term (/FC_D) multiples the slope of V_error.
- FC_Shape is additional gain applies to P and D terms.
- Once the voltage error slope goes to zero the contribution from D-Term goes to zero.
- Offset (/V_Lift) is temporary offset added to VOUT following a load add event.
- Enable Diode Break (/Diode_brake) helps reduce V_OUT overshoot following a load release event and Diode Break duration (DB_duration) specifies max length of time the Diode Break function will operate to limit overshoot.
- Load Oscillation Frequency (/BBRK_FREQ_TH) is the frequency below which body braking is allowed.
- Once overshoot threshold (/ERR_IITH) is exceeded all pulses are terminated.
- When Voltage error goes to zero (or positive) ATA is disengaged and PID takes over.
- A value of 0F(h)/15(d) in the /FC_HTH register and 0 in /FC_P registers DISABLES ATA.

![Diagram of Control Loop and ATA Tuning](image)
4a) Control Loop—ATA Tuning—Simulation Example

ATA functionality can be evaluated through the Simplis models available on the Power Module Wizard website. The following configuration was run to demonstrate improvement using ATA algorithm:

- Step up load current from 20A to 100A in 1us, then step down from 100A to 20A in 1us, Vout=1.0V and Loadline=1mOhm
- Red dotted line — ATA disabled  Red solid line— ATA enabled with proper tuned parameters

ATA Parameters:
- FC_HTH=5
- FC_P=15
- FC_slope_TH=2
- FC_D=1
- FC_shape=0
- V_lift=0
- Err_ITH=5
- BBRK_EN=1
- Boost_Duration=4
- BBRK_Duration=1
- Loadline=1 mohm
- AC_EN=0

From graph of Simplis simulation, we can see that Vout has smaller undershoot and overshoot with enabled ATA.
4a) Control Loop—ATA Tuning—Simulation Example

The mechanism by which ATA improves the Transient performance is as follows:

ATA reduces undershoot by increasing Freq of switching

ATA reduces overshoot by turn off both high/low side FETs to enable Tri-state PWM
4a) Control Loop—ATA Tuning—Evaluation Board Example

The following scenario was created on an evaluation board to demonstrate an example where ATA can make a dramatic improvement over the conventional Loop.

For the above example, ATA brought some nice improvements.

<table>
<thead>
<tr>
<th>Test Condition</th>
<th>Vin</th>
<th>9.0V</th>
<th>Vout</th>
<th>1.5V</th>
<th>Iout</th>
<th>0A-&gt;80A in 1us</th>
<th>slew Rate</th>
<th>80A/us</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>Kp</td>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ki</td>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kd</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pole1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pole2</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATA</td>
<td>FC_D</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FC_HTH</td>
<td>15/disabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FC_Shape</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FC_P</td>
<td>0/ATA disabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>V_Lift</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DB_duration</td>
<td>0/666ns</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Err_lth</td>
<td>15/60mV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FC_slope</td>
<td>7.84mV/us</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Diodo Brake</td>
<td>0/disabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ATA disabled</th>
<th>9.0V</th>
<th>1.5V</th>
<th>80A-&gt;0A in 1us</th>
<th>80A/us</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vout Undershoot=30.5mV</td>
<td>Vout Overshoot=30.0mV</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATA Enabled</td>
<td>9.0V</td>
<td>1.5V</td>
<td>3/12mV</td>
<td>0</td>
</tr>
<tr>
<td>Vout Undershoot=15.0mV</td>
<td>Vout Overshoot=15.5mV</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Remember modules ship with ATA disabled! From graph of Simplis simulation, we can see that Vout has smaller undershoot and overshoot with enabled ATA.
4b) Control Loop—PID Tuning

- A linear Proportional-Integral-Derivative (PID) digital controller on the MLX family provides loop compensation for the system regulation. The Digital compensator process the digitized error voltage coming from the high-speed voltage error ADC.
- The pulse-width for each of the active phase is determined from the outputs of the PID and phase current balance control signals and feed into the PWM generator.
- The MLX family have 2 identical and independent loops to control 2 independent output if configured as such.
- The loop compensator coefficients are user configurable to optimize the system response, the PID algorithm has 2 additional programable poles that serve as an equivalent type III analog compensator.

To Increase Zero1 Frequency
  a) Increase Ki
  b) Decrease Kp

To decrease Zero2 frequency
  a) Increase Kd
  b) Decrease Kp
4b) Control Loop—PID Tuning

Through the DPI tool, the PID values for each loop can be adjusted through the following registers.

The values set through the registers are then scaled and translated to gain or frequency adjustment. Scaling is first determined based on the phases. The scaling helps keep a consistent loop gain response across different phases (see next page).

<table>
<thead>
<tr>
<th>Compensation Scaling Factors based on Phases</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phases</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>Masters 160A-40A-160A Sat 40A-160A</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
</tbody>
</table>

This is then translated from the register values to change in gain:

\[
k_p[5:0] = (4+k_p[1:0]) \times 2^{(k_p[5:2]-9)}; \quad \text{Ex.} \ 24h = \theta_{100100} = 4+\theta 2^{(9-9)} = 4
\]

\[
k_i[5:0] = (4+k_i[1:0]) \times 2^{(k_i[5:2]-21)}; \quad \text{Ex.} \ 1Fh = \theta_{011111} = 4+\theta 2^{(7-21)} = 0.00427
\]

\[
k_d[5:0] = (4+k_d[1:0]) \times 2^{(k_d[5:2]-10)}; \quad \text{Ex.} \ 21h = \theta_{000001} = 4+\theta 2^{(8-10)} = 1.25
\]

Or change in Frequency:

\[
Kp[3:0] = (4+Kp[1:0]) \times 2^{(Kp[3:2]-8)}; \quad \text{Ex.} \ 0Eh = 0\theta 1100 = 4+\theta 2^{(3-9)} = 0.094
\]

\[
\text{Band Width} = (C\times 486)/(\pi\times 4\times C\times C^{0.5}) = 753.22 \text{ kHz}
\]

\[
Kp[2:3] = (4+Kp[1:0]) \times 2^{(Kp[2:2]-8)}; \quad \text{Ex.} \ 0Eh = 0\theta 1000 = 4+\theta 2^{(2-9)} = 0.063
\]

\[
\text{Band Width} = (C\times 24e6)/(\pi\times 4\times C\times C^{0.5}) = 246.69 \text{ kHz}
\]
4b) Control Loop—PID Tuning

Scaling done by the module ensures a consistent loop response by the module as phases are added or dropped. Example of a 3 Phase module (MLX120) shown below. As Phases are added from 1 to 3 there is very small change in crossover frequency, Phase Margin and Gain Margin.

Setting used—shows register value and corresponding translated value. Solid line below is single phase. Dotted line shows added phases.
4b) Control Loop—Kp—Proportional Coefficient

The proportional coefficient affects the mid-band frequencies of the Loop gain Plot. As Kp is increased, the crossover frequency increases. However in the Phase Margin curves an increase in Kp causes a reduction in the Phase Margin. So caution should be used that Kp is not increased to a level to drop below the desired 45° of Phase Margin.

Curve below generated from Simplis model, Kp =24, 28, 40

Measured Data, Kp = 28, Kp = 28+4, Kp = 28-4 for a MLX120, 3 phase module. Baseline curve is 6.3PH-Gain/Phase shown as a solid line.
4b) Control Loop—Kp—Proportional Coefficient


- Larger P term increase overshoot and reduce settling time


- Smaller P term decrease overshoot and increase settling time
4b) Control Loop—Ki—Integrative Coefficient

The integrative coefficient affects the low-band frequencies of the Loop gain Plot. As Ki is increased, the gain of lower band frequencies increased and the Phase angle reduced. Curve below generated from Simplis model, Ki = 6, 14, 24.

Measured Data, Ki = 14, Ki = 14+10, Ki = 14-10 for a MLX120, 3 phase module. Baseline curve is 6_3PH-Gain/Phase in solid line.
4b) Control Loop—Ki—Integrative Coefficient

**INCREASING Ki**—Measured Data, Ki = 14 vs Ki = 24 for transient response behavior

Larger I term increase overshoot and reduce steady-state error

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=7

Kp=28, Ki=24, Kd=47, Pole1=5, Pole2=7

**DECREASING Ki**—Measured Data, Ki = 14 vs Ki = 4 for transient response behavior

Smaller I term reduce overshoot and increase steady-state error

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=7

Kp=28, Ki=4, Kd=47, Pole1=5, Pole2=7
4b) Control Loop—Kd—Differentiation Coefficient

The Differentiation coefficient affects the low-band frequencies of the Loop gain Plot. As Kd is increased, the gain of higher band frequencies increased causing an increase in crossover frequency and the Phase angle also increased leading to a higher Phase margin. Curve below generated from Simplis model, Kd =37, 47 , 57

Measured Data, Kd = 47, Kd = 47+5, Kd = 47-5 for a MLX120, 3 phase module. Baseline curve is 6_3PH-Gain/Phase in solid line.
4b) Control Loop—Kd—Differentiation Coefficient

**INCREASING Kd**—Measured Data, Kd = 47 vs Ki = 52 for transient response behavior

K<sub>p</sub>=28, K<sub>i</sub>=14, K<sub>d</sub>=47, Pole1=5, Pole2=7

K<sub>p</sub>=28, K<sub>i</sub>=14, K<sub>d</sub>=52, Pole1=5, Pole2=7

Larger D term reduce overshoot and increase settling time

**Decreasing Kd**—Measured Data, Kd = 47 vs Kd = 42 for transient response behavior

K<sub>p</sub>=28, K<sub>i</sub>=14, K<sub>d</sub>=47, Pole1=5, Pole2=7

K<sub>p</sub>=28, K<sub>i</sub>=14, K<sub>d</sub>=42, Pole1=5, Pole2=7

Smaller I term increase overshoot and reduce settling time
4b) Control Loop—Kpole1-Adjustable Pole 1

The adjustable Pole 1 helps increase High Frequency Gain and filter Noise. Curve below generated from Simplis model, Kpole1 = 2, 5, 10

Measured Data, LPF1=5, LPF1=5+2, LPF1=5-2, 3 phase module. Baseline curve is 6_3PH-Gain/Phase in solid line
4b) Control Loop—Kpole1-Adjustable Pole 1

**INCREASING Kpole1**—Measured Data, Pole1=5, Pole1=7 for transient response behavior

Typically, Pole is to filter noise and reduce high Frequency gain created by D term

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=7

**DECREASING Kpole1**—Measured Data, Pole1=5, Pole1=3 for transient response behavior

Typically, Pole is to filter noise and reduce high Frequency gain created by D term

Kp=28, Ki=14, Kd=47, Pole1=3, Pole2=7
4b) Control Loop—Kpole2-Adjustable Pole 2

The adjustable Pole 2 helps increase High Frequency Gain and filter Noise. Curve below generated from Simplis model, Kpole2 = 2, 7, 12

Measured Data, LPF2=7, LPF2=7+2, LPF2=7-2, 3 phase module. Baseline curve is 6_3PH-Gain/Phase in solid line
4b) Control Loop—Kpole2-Adjustable Pole 2

**INCREASING KPole2**—Measured Data, Pole2=7, Pole2=9 for transient response behavior

Typically, Pole is to filter noise and reduce high frequency gain created by D term

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=7

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=9

**DECREASING KPole2**—Measured Data, Pole2=7, Pole2=5 for transient response behavior

Typically, Pole is to filter noise and reduce high frequency gain created by D term

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=7

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=5
4b) Control Loop— Balanced PID

Effect of optimum loop on transient response behavior

Well tuned PID setting makes transient response better
Smaller overshoot/undershoot
Faster rising time
Shorter settling time

Kp=28, Ki=14, Kd=47, Pole1=5, Pole2=7
Kp=33, Ki=35, Kd=58, Pole1=3, Pole2=5
### Change History (excludes grammar & clarifications)

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Description of the change</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>2/28/2023</td>
<td>Updated Page 24</td>
</tr>
<tr>
<td>1.2</td>
<td>11/07/2023</td>
<td>Updated as per OmniOn template</td>
</tr>
</tbody>
</table>