High-Speed Multi-Phase Driver Interfaces with Per-Phase Sensing, Interlocks, and Fail-Safe Detuning for Solid-State Virtual Rotors

Docket: NTL-PROV-008
Inventors: [Inventor Names]
Assignee: [Assignee / Organization]
Date: February 1, 2026

Abstract

Disclosed are apparatus, systems, methods, and non-transitory computer-readable media for driving multi-phase electromagnetic arrays (including solid-state virtual rotors) using a high-speed driver interface that integrates per-phase sensing, fault interlocks, and fail-safe behaviors. In various embodiments, a driver subsystem comprises C independently addressable channels configured to energize electromagnetic elements according to a commutation schedule. Each channel includes (i) a controllable switching stage, (ii) per-phase sensing of at least one electrical quantity (e.g., current, voltage, temperature proxy), and (iii) interlocks that enforce constraints including overcurrent, overvoltage, overtemperature, timing/jitter violations, and loss-of-load conditions.

The disclosure further provides a multi-layer control surface separating high-level schedule intent from low-level waveform realization, and provides a hardware/firmware architecture that supports deterministic replay and tamper-evident configuration. In one embodiment, when a fault is detected, the driver transitions to a safe state that includes at least one of: detuning/phase-slip injection, switching to a neutral safe kernel, insertion of a dump load/crowbar, or full shutdown with a latched fault state. The disclosed driver interface enables reproducible operation at high switching rates while bounding risk and enabling rigorous metrology.

Technical Field

The present disclosure relates to power electronics and control interfaces for multi-phase electromagnetic systems, and more particularly to high-speed multi-channel driver interfaces with per-phase sensing, safety interlocks, deterministic replay, and fail-safe detuning or shutdown behaviors for commutated coil arrays and virtual rotor systems.

Background

Multi-channel commutation systems can synthesize rotating magnetic fields by energizing multiple electromagnetic elements according to a schedule. At high switching rates and power levels, these systems face failure modes including overcurrent, overtemperature, EMI-induced mis-triggering, timing skew, controller faults, and unsafe behavior under load disconnect.

Conventional drivers often provide only coarse aggregate sensing (e.g., total supply current) and may not detect per-phase faults, resulting in damage or unsafe behavior. Additionally, experiments requiring high reproducibility benefit from deterministic replay and from configuration integrity guarantees.

Accordingly, there is a need for an integrated driver interface that provides per-phase sensing, fast interlocks, deterministic replay, and fail-safe mitigation actions appropriate for high-speed virtual rotor operation.

Summary

This disclosure provides a driver interface for multi-phase commutation systems. In one aspect, each channel includes a switching stage and per-phase sensing and is controlled by a controller that executes commutation schedules. In another aspect, the driver provides a structured control surface separating schedule-level parameters (phase order, frequency, duty) from hardware-level parameters (gate timing, rise/fall shaping, current limits).

In another aspect, the driver includes interlocks that monitor per-phase current, voltage, and temperature (or proxies) and trigger mitigation actions including detuning, safe neutral patterns, dump-load insertion, or shutdown.

In another aspect, the driver stores configuration artifacts and supports deterministic replay and integrity checks.

Brief Description of the Drawings

Drawings may be provided later. For purposes of this specification:

- **FIG. 1** depicts a multi-channel driver subsystem connected to a virtual rotor array.
- **FIG. 2** depicts a per-channel block including switching stage, current sensing, temperature sensing, and interlocks.
- **FIG. 3** depicts a control surface separating schedule intent from waveform realization.
- **FIG. 4** depicts a safety state machine including nominal, detune, and shutdown states.
- **FIG. 5** depicts deterministic replay packaging and configuration integrity checks.

Definitions and Notation

Unless otherwise indicated:

- $C \in \mathbb{N}$ is the number of driver channels/phases.
- A *channel* refers to a controllable drive path from a power stage to an electromagnetic element or phase group.
- A *schedule* refers to a discrete-time commutation intent, such as phase order and dwell timing.
- A *waveform realization* refers to hardware-level signals that implement the schedule (e.g., gate drive signals).
- An *interlock* refers to a hardware and/or firmware condition that prevents unsafe operation by forcing a safe state.
- A *safe state* refers to a state in which drive is disabled or detuned and risk is bounded.
- A *detune action* refers to deliberate destruction of resonance/ordering by modifying phase timing or sequence.

Detailed Description

1. System Architecture

In one embodiment, a virtual rotor system comprises:

- a multi-phase electromagnetic array (coil elements or phase groups);
- a driver subsystem comprising C channels;
- a controller producing commutation schedules;
- sensors and interlocks providing safety enforcement.

2. Driver Channel Block

In one embodiment, each channel comprises:

- a switching stage (e.g., MOSFET half-bridge, full-bridge, or other topology);
- a gate driver with controlled rise/fall time and dead-time insertion;
- a current sense element (e.g., shunt, Hall, or current transformer);
- a voltage sense element (e.g., divider, isolated measurement);
- a temperature measurement element (sensor or proxy);
- a fast interlock path that can disable the channel independent of the main controller.

3. Per-Phase Sensing

3.1 Current sensing. In one embodiment, per-channel current $I_c(t)$ is measured and compared to thresholds:

$$|I_c(t)| \leq I_{\max,c}.$$

Thresholds may be static or schedule-dependent (e.g., different phases may permit different peaks).

3.2 Voltage sensing. In one embodiment, per-channel voltage $V_c(t)$ is measured and compared to thresholds:

$$|V_c(t)| \leq V_{\max,c}.$$

3.3 Temperature sensing. In one embodiment, per-channel temperature $T_c(t)$ (or proxy) is measured and compared to thresholds:

$$T_c(t) \leq T_{\max,c}.$$

3.4 Derived metrics. In one embodiment, the system computes derived metrics such as:

- instantaneous power proxy $P_c(t) = V_c(t)I_c(t)$,
- RMS current over a window,
- duty-cycle and switching-loss proxies.

4. Timing Integrity and Interlocks

In one embodiment, the driver integrates timing integrity monitoring, including:

- detection of missing pulses, extra pulses, or out-of-order phases,
- detection of jitter exceeding a configured bound,
- detection of skew between channels exceeding a bound.

If timing integrity is violated, the interlock transitions the system to a safe state.

5. Control Surface (Schedule Intent vs Waveform Realization)

In one embodiment, the driver exposes two layers:

- **Schedule intent layer:** phase order, dwell time, direction, amplitude profile.
- **Waveform realization layer:** gate timing, dead-time, edge shaping, current limiting, PWM carrier details.

Separating these layers allows consistent experiments across hardware revisions while protecting the safety and integrity of low-level implementations.

6. Fault Detection and Safe-State Machine

6.1 Fault classes. Faults include (non-limiting):

- overcurrent, overvoltage, overtemperature,
- timing faults (jitter/skew/out-of-order),
- supply rail faults (undervoltage/overvoltage),
- load disconnect / open circuit on a phase,
- sensor faults (stuck-at, out-of-range),
- watchdog timeout.

6.2 Safety state machine. In one embodiment, the driver implements a state machine:

- **NOMINAL:** schedule executes as commanded.
- **DETUNE:** driver injects detune timing/sequence or switches to a safe neutral kernel.
- **SHUTDOWN:** all switching disabled; fault latched until reset.

6.3 Mitigation actions. Mitigation actions include:

- per-channel disable (selective shedding),
- global disable (hard stop),
- detune/phase slip injection (soft stop),
- dump-load insertion or crowbar on output bus,
- amplitude/duty reduction and ramp-down.

7. Deterministic Replay and Integrity

In one embodiment, a run is recorded as a bundle containing:

- schedule intent data (phase order, dwell, direction, setpoints),
- waveform realization parameters (gate timing, dead-time, limits),
- sensor logs (per-phase I/V/T),
- interlock events and state transitions,
- hashes and optional signatures for tamper evidence.

8. Example Embodiments (Non-Limiting)

Embodiment A: per-phase current-limited driver. Each phase has independent current sensing and a fast comparator that disables the phase if $I_c > I_{\max,c}$ within a configured blanking time to avoid false triggers.

Embodiment B: jitter-gated high-speed commutation. The driver measures commutation event timing and triggers DETUNE if RMS jitter exceeds a threshold for more than a persistence interval.

Embodiment C: load disconnect protection. The driver detects a phase open circuit (e.g., current below threshold while commanded on) and triggers a dump load and global detune to prevent overshoot.

Claims (Draft)

Note: The following claims are an initial, non-limiting claim set intended to preserve multiple fallback positions. Final claim strategy should be reviewed by counsel.

Independent Claims

1. **(System)** A multi-channel driver system for a phased electromagnetic array, the system comprising: a plurality of driver channels configured to energize electromagnetic elements; per-channel sensing circuitry configured to measure at least one of current, voltage, or temperature for each driver channel; and an interlock subsystem configured to transition the driver system to a safe state when a sensed quantity violates a threshold.

2. **(Method)** A method of operating a solid-state virtual rotor, the method comprising: energizing a plurality of electromagnetic elements according to a commutation schedule; sensing per-phase electrical quantities during energization; detecting at least one fault condition based on the sensed quantities; and, responsive to detecting the fault condition, performing a mitigation action comprising at least one of detuning the commutation schedule, switching to a neutral safe schedule, inserting a dump load, or disabling switching.

3. **(Non-transitory medium)** A non-transitory computer-readable medium storing instructions that, when executed by one or more processors, cause the one or more processors to: receive schedule intent parameters; translate the schedule intent parameters into waveform realization

parameters including at least one of gate timing or dead-time; and enforce per-channel interlock thresholds based on measured per-channel current, voltage, or temperature.

Dependent Claims (Examples; Non-Limiting)

4. The system of claim 1, wherein the per-channel sensing circuitry comprises a current shunt and a comparator configured to disable a channel within a bounded response time.

5. The system of claim 1, wherein the interlock subsystem is configured to detect a timing integrity fault comprising at least one of excessive jitter, excessive skew, missing pulses, or out-of-order phases.

6. The method of claim 2, wherein detuning comprises injecting a phase slip in the commutation schedule.

7. The method of claim 2, wherein inserting the dump load comprises switching a resistive load onto a DC bus to absorb energy.

8. The non-transitory medium of claim 3, further comprising storing a deterministic replay bundle including schedule intent, waveform realization parameters, and sensor logs.

9. The system of claim 1, wherein the safe state comprises a shutdown state that latches until manual reset.

10. The system of claim 1, further comprising a watchdog configured to force the safe state upon controller timeout.

11. The system of claim 1, wherein the driver channels comprise half-bridge or full-bridge stages.

Additional Embodiments and Fallback Positions (Non-Limiting)

- Sensing may be analog, digital, or mixed-signal and may include isolated sensing for high-voltage stages.
- Interlocks may be implemented purely in hardware, purely in firmware, or as a hybrid with a hardware fast path and firmware supervisory path.
- Safe states may include selective shedding of channels, reduced duty operation, or substitution of a verified safe commutation kernel.
- Thresholds may be schedule-dependent and may be parameterized per phase group.
- The driver may include EMI mitigation features such as slew-rate control, spread-spectrum modulation, and shield/return-path enforcement.

End of Specification (Draft)