

BESIII 快亮度测量系统
软件设计方案书

项目名称：快亮度监控软件

版本：V2.0（设计评审稿）

日期：2026 年 9 月

目录

1 引言 1
1.1 编写目的 1
1.2 项目背景 1
1.3 现状与旧系统分块介绍 1
1.3.1 亮度显示 1
1.3.2 高压监控 2
1.3.3 阈值配置 2
1.3.4 新旧系统对比 3
1.4 设计依据与参考资料 3
1.5 术语与缩略语 4
1.6 适用范围 4
2 需求分析 4
2.1 功能需求 4
2.2 性能需求 5
2.3 约束与假设 5
3 系统总体设计 6
3.1 设计目标 6
3.2 设计原则 6
3.3 系统规模 6
3.4 业务功能框架 7
3.5. 总体架构 8
4.系统模块详细设计 9
4.1 高压控制模块 9
4.1.1 模块概述 9
4.1.2 设计目标 9
4.1.3 软件方案 10
4.2 电子学监控模块 10
4.2.1 模块概述 10
4.2.2 设计目标 10

4.2.3 软件方案 10
4.2.4 关键设计要点 11
4.3 亮度数据处理与发布模块 11
4.3.1 模块概述 11
4.3.2 硬件工作机制 12
4.3.3 设计目标 13
4.3.4 软件方案 13
4.3.5 关键设计要点 13
4.4 数据采集与存储模块 14
4.4.1 模块职责 14
4.4.2 软件方案 14
4.4.4 存储设计 15
4.4.5 关键设计 15
4.4.6 与其他模块关系 16
4.5 网页监控模块 16
4.5.1 模块概述 16
4.5.2 软件结构 16
4.5.3 页面规划 17
4.5.4 关键设计要点 17
4.6 系统管理与通用服务 18
5. 接口设计 18
5.1 内部接口 18
5.2 外部接口 18
5.3 PV 命名规范（示例） 19
6. 数据库设计（概要） 19
6.1 设计目标与原则 19
6.2 MySQL 表设计 20
6.3 Redis 实时缓存设计 20
6.4 Kafka 主题设计（可选） 20
6.5 数据一致性与降级 20
7. 系统关键技术 21
8. 性能与可靠性设计（对应指标） 21

9. 部署方案 21
10. 技术选型汇总 22
11. 实施计划 22
11.1 阶段划分 22
11.2 进度风险与应对 23
12. 测试与验收 23
12.1 测试环境 23
12.2 功能测试 23
12.3 性能测试 24
12.4 集成与验收 24
13. 风险与对策 24

1 引言

1.1 编写目的

本文档为 BESIII 快亮度测量系统数字化改造项目的软件设计方案书，在《快亮度测量系统软件需求规格说明书》（V1.0）的基础上，描述系统的总体架构、功能模块详细设计、数据与接口设计、关键流程设计以及性能、安全、可靠性等非功能设计，并给出实施计划与测试验收方案，作为软件开发、集成联调、测试验收和后续维护的设计依据。

预期读者包括：项目管理人员、软件开发人员、测试人员、硬件设计人员、加速器运行与调束人员及验收评审专家。

1.2 项目背景

北京正负电子对撞机重大改造工程（BEPCII）是一台多束团、高亮度双环对撞机，设计的峰值亮度为 $10^{33}cm^{-2}s^{-1}$@1.89GeV。逐束团亮度监测是对撞机束流调试的重要手段。BEPCII 亮度监测器分为西端的小角度亮度探测器（LUM）和东端的零角度亮度探测器（ZDD），长期以来一直是束流调试不可缺少的手段。

加速器升级后对撞周期由 8 ns 缩短为 6 ns，原有亮度探测器及电子学系统老化严重（光输出降低、成形时间变宽），且原有数据获取系统针对 8 ns 对撞周期设计，在 6 ns 对撞周期下丢数频繁，造成柱间本底增大（可达柱高的 30%）、数据精度变差，影响调束。此外，亮度显示与加速器调试目前依靠人工处理、反馈慢，亟需将亮度显示集成到加速器调束系统实现智能反馈。为此，决定研发基于 LYSO 晶体+硅光电倍增管（SiPM）的新一代快亮度探测器及配套电子学与监控软件，替代现有 LUM 和 ZDD。

快亮度监控软件是新快亮度测量系统的组成部分，负责控制系统硬件运行、监视其运行状态、处理亮度数据并实时显示与发布，为加速器调束提供实时反馈。

1.3 现状与旧系统分块介绍

原系统显示控制软件采用 C++Builder 开发，为单机桌面程序；高压监控部分采用 LabVIEW 页面。按照旧系统的功能分块，现状如下：

1.3.1 亮度显示

旧系统亮度监视界面（如图 1-1）提供 E 端束团亮度柱状图、总亮度曲线及 E_max 数值显示，是调束人员观察束团亮度分布的主要手段。但该界面数据依赖人工处理与本地专用总线，无法远程访问，也不能向调束系统自动发布数据。

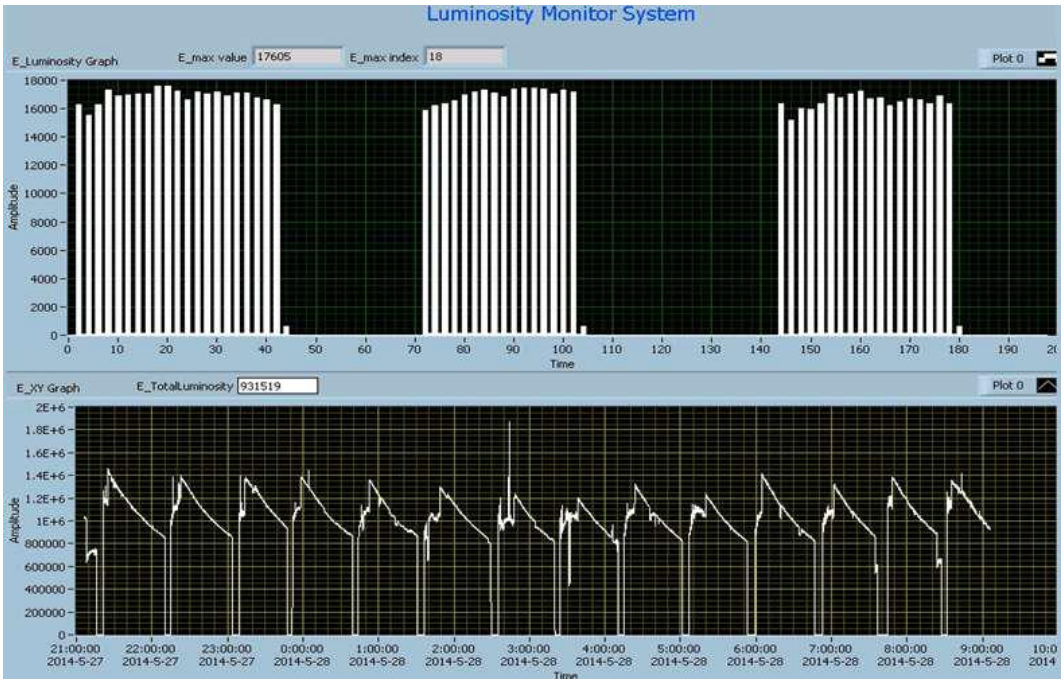


图 1-1 旧系统亮度监视界面（C++Builder）

1.3.2 高压监控

旧系统高压监控基于 LabVIEW 页面（见图 1-2），可对高压通道电压、电流进行显示与控制，界面功能较为简单，缺乏历史数据查询、曲线对比、权限管理与远程访问能力，且与亮度数据无关联分析手段。

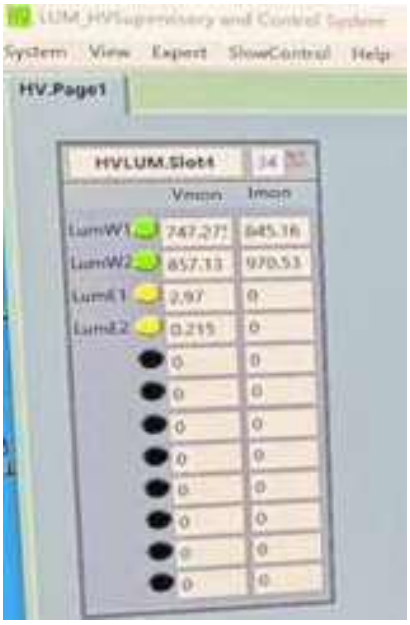


图 1-2 旧系统高压监控界面（LabVIEW）

1.3.3 阈值配置

旧系统阈值等参数通过 LabVIEW 配置面板设置（见图 1-3），参数分散、缺乏回读校验与操作审计机制，不支持远程批量配置与版本化管理。

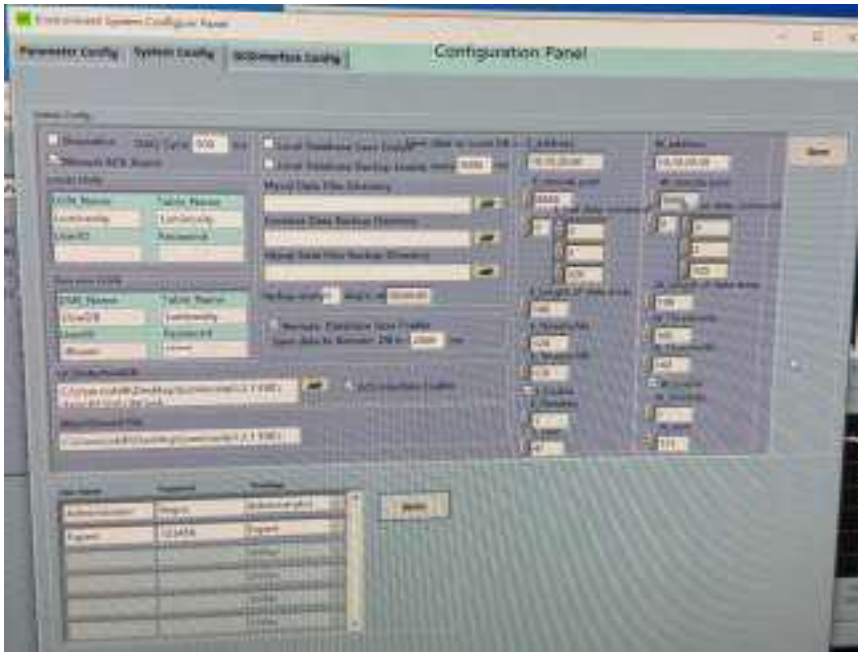


图 1-3 旧系统阈值配置面板（LabVIEW）

1.3.4 新旧系统对比

表 1-1 新旧系统对比

对比项	当前系统	改造后系统
探测器	钨转换体+熔融石英+PMT（LUM）；铅-闪烁光纤 ZDD	LYSO 晶体+SiPM 阵列（东西两端各 8 个探测单元，共 16 单元、240 路 SiPM）
前端电子学	OPA846/AD8000 带宽：300MHz/1.75GHz	OPA847（950MHz）/AD8009（1.6GHz）+ADCMP562 甄别+脉冲成形
后端电子学	MC100 系列分立器件+FPGA 计数	ZYNQ FPGA（ARM Cortex-A9+PL）嵌入式系统
显示控制软件	C++Builder 桌面程序/LabVIEW 页面，单机界面	分层架构网页化监控（设备接入—采集—处理—服务—Web 展示）
数据交互	本地/专用总线	千兆以太网 IPbus+EPICS Channel Access/PVAccess
数据能力	仅实时显示，无历史与发布	历史归档、实时发布（最短 0.1 s）、告警与审计

1.4 设计依据与参考资料

- 《快亮度测量系统软件需求规格说明书》V1.0（2025 年 9 月）；
- 《快亮度测量系统数字化改造》方案书；
- BESIII 快亮度监控讨论材料（2026 年 9 月）；
- EPICS Base 及相关模块版本文档；
- CAEN SY4527 高压机箱及 A2254/A2554 高压插件手册；
- 前端/后端电子学硬件接口文档。

1.5 术语与缩略语

表 1-2 术语与缩略语

术语/缩略语	说明
BEPCII	北京正负电子对撞机重大改造工程
BESIII	北京谱仪 III
LUM/ZDD	小角度亮度探测器/零角度亮度探测器
LYSO	硅酸钇镥晶体
SiPM/MPPC	硅光电倍增管
EPICS	实验物理与工业控制系统框架
PV / CA / PVA	EPICS 过程变量 / Channel Access 协议 / PVAccess 协议
IOC	EPICS 输入输出控制器（设备驱动与 PV 服务进程）
IPbus	基于 IP 网络的轻量级通信协议，用于 ZYNQ 与前端设备间控制与数据传输
束团	加速器束流中的粒子团；BEPCII 每环存储约 180 余个束团（对撞周期 6 ns）
相对亮度	原始亮度按参考束流强度归一化后的亮度值
前放	前置放大器
OPI	操作员界面（网页监控界面）
HV	高压（High Voltage）
DAQ	数据获取（Data Acquisition）

1.6 适用范围

本方案适用于快亮度监控软件的设计、开发、部署、运维和验收，覆盖以下范围：

240 路高压电源及高压分配板监控；

东西两端前放阈值远程设置；

亮度数据高速采集、快速积分与归一化；

在线相对亮度柱状图、总相对亮度曲线显示；

束团相对亮度数据发布，供加速器调束系统集成；

2 需求分析

快亮度监控软件需实现高压控制、电子学监控、亮度数据处理、在线相对亮度显示与发布等功能。

2.1 功能需求

快亮度监控软件面向快亮度探测器及其配套电子学系统，为加速器调束提

供稳定、实时、可靠的亮度监测与反馈数据。

软件系统主要完成以下需求：

- 实现 240 路 SiPM 高压电源的远程监控与控制；
- 实现东西两端前置放大器阈值的远程配置与监控；
- 实现快亮度原始数据的高速采集、解析及亮度计算；
- 实现束团相对亮度的实时计算、显示和发布；
- 实现实时运行数据和历史数据的统一管理；
- 通过标准化接口向加速器总控及其他子系统提供实时亮度数据；
- 建立模块化、可扩展的软件架构，为后续硬件升级及系统扩展提供基础。

根据现有需求，系统可以按优先级划分为以下几个业务需求：

表 2-1 主要业务需求

编号	需求描述	优先级
FR-01	240 路高压远程连接、输出控制、监测（电压/电流曲线）	高
FR-02	高压历史数据查询与图表显示	高
FR-03	东西两端前放阈值远程设置，0–2.5V，精度 0.01V，带设定值回读	高
FR-04	电子学状态参数实时显示（高压、电流曲线）与历史查询	高
FR-05	亮度数据高速采集与快速积分算法计算瞬时亮度	高
FR-06	亮度数据按参考束流强度归一化，生成相对亮度	高
FR-07	实时束团相对亮度柱状图显示	高
FR-08	束团相对亮度数据对外发布，供加速器调束/束团调试界面集成	高
FR-09	总相对亮度实时曲线显示与历史查询	中
FR-10	实时亮度发布间隔可调（0.1s–10s）	高

2.2 性能需求

主要的性能指标如表 2-2 所示。

表 2-2 主要性能指标

编号	指标	设计响应
NFR-01	阈值设置范围 0–2.5V，精度 0.01V	设定值以整型标度（0–250，单位 10mV）传输，避免浮点误差
NFR-02	实时亮度发布间隔可调，最短 0.1s	Kafka 发布＋WebSocket 推送，节拍器可调
NFR-03	可用性＞95%	进程看门狗自动重启、关键服务冗余、降级运行模式（第 9 节）
NFR-04	维护周期＞1 年	全部采用成熟开源组件，远程诊断与日志体系（第 9 节）

2.3 约束与假设

- 硬件接口协议、束团数量、排列方式、参考束流强度来源需在联调前确

认；

- 控制网与数据网隔离，跨区仅开放必要端口；
- 高压控制可复用现有 LabVIEW 逻辑，也可由 EPICS IOC 直接驱动，需现场确定；
- 全系统以控制网 NTP 对时。
- Kafka 是否启用取决于对外集成需求，Grafana 是否启用取决于历史曲线展示需求。

3 系统总体设计

3.1 设计目标

- 在 6 ns 新对撞周期下稳定完成快亮度系统的硬件控制、状态监视、亮度数据读取、统计与直方图显示；
- 实现 240 路 SiPM 高压、东西两端前放阈值（0~2.5 V，精度 0.01 V）的远程监控与配置，回读校验；
- 实现亮度数据高速采集、快速积分与归一化处理，束团相对亮度经 EPICS 发布，发布间隔可调、最短 0.1 s；
- 提供网页化监控界面，覆盖并扩展原 C++ Builder/LabVIEW 系统的全部功能，具备历史数据查询、告警、权限管理与操作审计能力；
- 通过修改配置即可复用到其他同类型设备系统，无需修改核心代码。

3.2 设计原则

- **标准化**：设备接入遵循 EPICS 标准接口（CA/PVA），对外发布面向加速器调束系统采用 EPICS，便于第三方工具（MATLAB、Python 等）接入；
- **分层解耦**：按“设备接入—数据采集—数据处理—数据服务—Web 展示—外部接口”分层，层间通过标准协议交互；
- **高可靠**：关键服务具备故障检测、告警与自动恢复能力；
- **可配置**：通道数、设备地址、PV 命名规则、界面布局、归一化参数等均通过配置文件管理；
- **安全性**：高压控制、阈值写入等关键操作具备权限管理与操作确认机制，符合加速器控制网安全接入要求

3.3 系统规模

新快亮度探测器东西两端各 8 个探测单元，每单元由 LYSO 晶体耦合 15 个 SiPM 阵列，全系统共 16 个探测单元、共 240 路 SiPM。15 路 SiPM 信号合并为 5 组输出，共 5×4×2=40 路信号；沿束流方向上下两个单元为 1 组共 4 组，每组信号经前端电子学求和（sum 信号）后甄别成形，送后端读出电子学处理。后

端软件可根据 4 组信号的噪声情况选择最优信号处理。涉及的主要设备规格可以参考表 3-1。

表 3-1 系统设备规模

设备	型号/规格	数量	作用
高压机箱	CAEN SY4527（16 插槽）	1	快亮度高压机箱
高压插件	A2554（0~64 V 输出板，8 通道）	5	SiPM 高压输出
高压分配板	定制 1 分 32	8	高压分配转接，支持每通道独立微调
高压转接板	定制	3	高压分配转接
读出电子学板(后端)	自研 ZYNQ 电子学板	3	阈值配置、亮度采集与计算
电子学快前放	自研（东西两端）	5	亮度信号与符合信号放大、甄别
PDU	科创 8 路智能 PDU	2	给电子学板供电
串口服务器	485 串口服务器	2	PDU 转 Modbus/TCP

快亮度探测器整体示意图如图 3-1 所示：

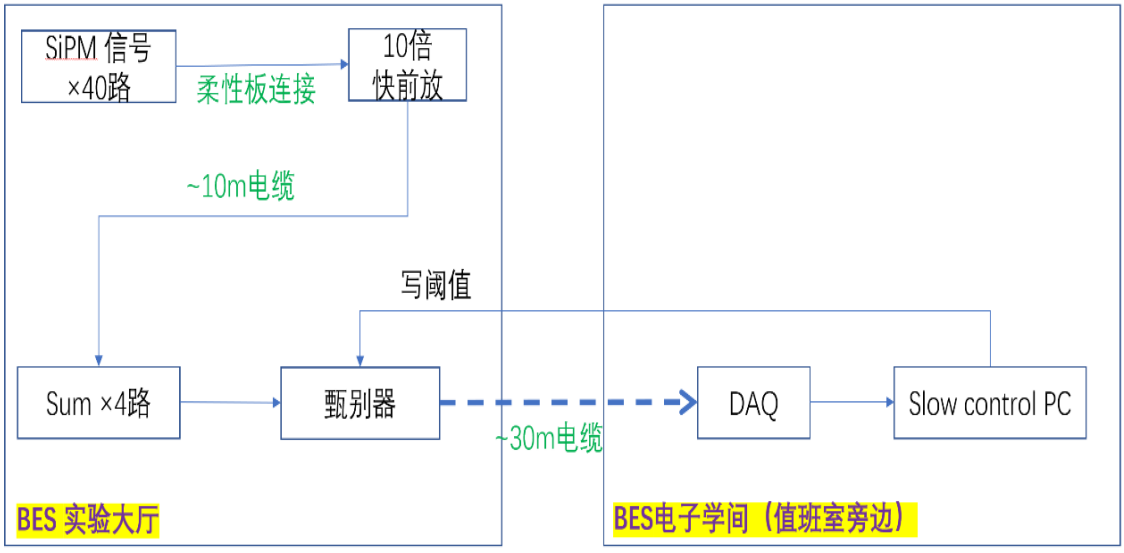


图 3-1 快亮度探测器电子学示意图

3.4 业务功能框架

快亮度监控软件业务功能框架如图 3-2，按“角色驱动、业务分层、设备解耦、数据贯通”设计。面向调束、探测器维护、管理员三类角色；含监控测量校准、电子学监控、亮度数据处理、显示与发布、系统控制五类业务，底层为高压系统、前放、Zynq 高速电子学、快亮度探测器。该框架与《3.5 总体架构》逐层对应：



图 3-2 系统功能框架图

3.5. 总体架构

系统采用分层架构设计，自下而上分为设备接入层、数据采集层、数据处理层、数据服务层、Web 监控层和外部系统接口层，总体技术架构如图 3-3 所示。

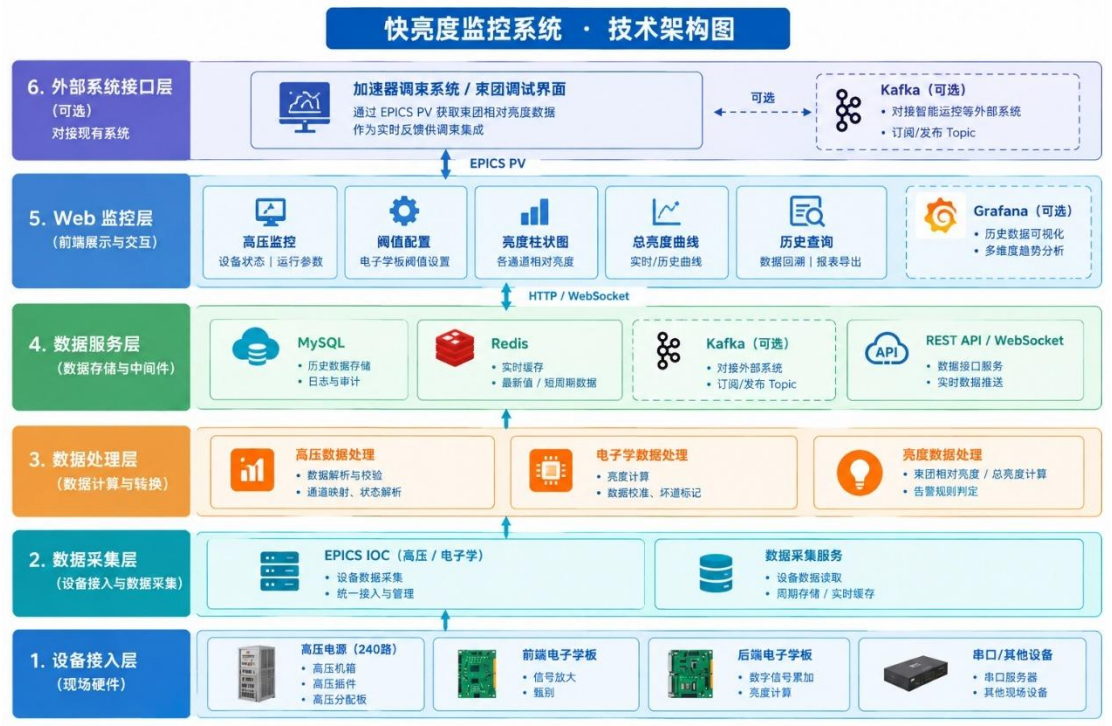


图 3-3 快亮度监控软件技术架构

每层职责划分如下表:

表 3-2 分层架构职责划分

层级	组成	职责
设备接入层	高压电源×240 路、高压分配板、前放（东/西）、数据读出系统、串口服务器、PDU 等	现场硬件；软件经以太网/串口等标准接口访问本层设备
数据采集层	EPICS IOC（东）、EPICS IOC（西）、高压桥接适配器、读出系统适配器	设备驱动与 PV 服务；屏蔽硬件协议差异，对外提供标准 PV/CA 接口
数据处理层	亮度算法引擎（积分/归一化）、监测数据规范化（通道映射、校准系数、坏道标记、状态字解析）与告警规则判定	亮度数据处理与其他监测数据的规范化处理
数据服务层	实时缓存（秒级响应）、消息总线、采集归档服务、发布服务（0.1 s 节拍）、告警服务	实时数据缓冲、历史存储、节拍发布与告警
Web 监控层	Web 前端（React + ECharts + grafana）、后端 API 服务（查询聚合/权限/日志）	高压总览、单路详情、阈值设置、亮度实时柱状图、历史查询、系统管理
外部系统接口层	EPICS CA/PVA（面向加速器调束系统/总控）	发布束团相对亮度数据，作为调束实时反馈

数据流向：

- **上行数据流：**设备 → EPICS IOC → Python 数据采集服务（通过 CA 订阅 PV）→ Redis/MySQL/Kafka → Web/外部接口。
- **亮度数据流：**ZYNQ 乒乓缓存 → 读出适配器 → 亮度算法引擎 → EPICS PV + Redis + MySQL + Kafka。
- **下行控制流：**Web 操作 → FastAPI 鉴权/限幅 → EPICS CA 写 IOC → 设备执行 → 回读比对 → 审计日志。

4.系统模块详细设计

4.1 高压控制模块

4.1.1 模块概述

高压控制模块是软件系统与高压硬件之间的接口层，负责管理由 CAEN SY4527 高压机箱、A2554 高压插件（0~64 V 输出，8 通道/块）及 8 块定制 1 分 32 高压分配板组成的高压供电系统，覆盖全系统 240 路 SiPM 通道的高压输出控制、通断控制、单通道微调及电压/电流实时监测。

4.1.2 设计目标

- 支持远程连接并访问高压机箱，实现对 240 路 SiPM 高压的逐通道控制与监测；
- 远程控制高压输出及各路通断，支持按通道独立微调；
- 关键操作需权限校验与二次确认，全程审计；

- 实时显示各路高压、电流，支持曲线查看与多路对比；
- 支持历史高压、电流查询，图表显示、缩放、平移及导出（CSV）。

4.1.3 软件方案

高压控制模块由 EPICS 高压 IOC、采集归档服务和高压控制接口三部分组成。软件结构与数据流如下：

- **EPICS 高压 IOC**：基于 EPICS 开发 SY4527 IOC 应用，实现机箱与插件的访问及通道读出控制；基于 asynDriver 开发端口驱动，调用 CAENHVWrapper 与 SY4527 通信；按通道模板实例化 240 组记录，包含 VSet、VMon、IMon、Pw、Status 等 PV。
- **采集归档服务**：Python 采集程序通过 CA 订阅高压 PV，将实时值写入 Redis 缓存，并周期归档至 MySQL 数据库。
- **高压控制接口**：前后端采用 React + FastAPI（PyAPI），实现高压监控。（控制流程为 Web → FastAPI 权限/范围校验 → CA 写参数 PV → IOC 下发 → 回读校验 → 刷新 Redis → audit_log；超差置通道异常并告警。）

4.2 电子学监控模块

4.2.1 模块概述

电子学监控模块负责前端电子学板与后端电子学板的远程监控与配置。前端电子学板对单端探测器输出的 120 路信号进行放大和甄别；后端电子学板（ZYNQ 平台）对亮度数字信号进行累加计数，得到多通道计数数据。本模块实现前放甄别阈值的远程设置与回读校验、电子学运行状态实时监控，并将读出的亮度原始数据转交亮度数据处理模块。

4.2.2 设计目标

- 远程设置东西两端前放甄别阈值：0~2.5 V，精度 0.01 V；亮度信号与符合信号分别配置；
- 设置后自动回读校验，界面显示生效值；回读偏差超过 0.01 V 判定失败并告警；
- 实时显示电子学运行状态及高压、电流等参数曲线；支持历史数据查询与图表显示；
- 将原始亮度计数数据可靠移交亮度数据处理模块（M3），作为相对亮度计算的数据源。

4.2.3 软件方案

电子学监控部分的软件结构如图 4-1 所示，主要是以下四部分：

1. 基于 EPICS 开发电子学 IOC 应用，实现对电子学板的读出与阈值设置；
2. 读出的亮度原始数据交给亮度处理模块进行数据处理；

3. 采集程序周期采集电子学数据存档 MySQL，同时实时数据通过 Redis 缓存；
4. 前后端采用 React + FastAPI（PyAPI），实现电子学监控与阈值设置。

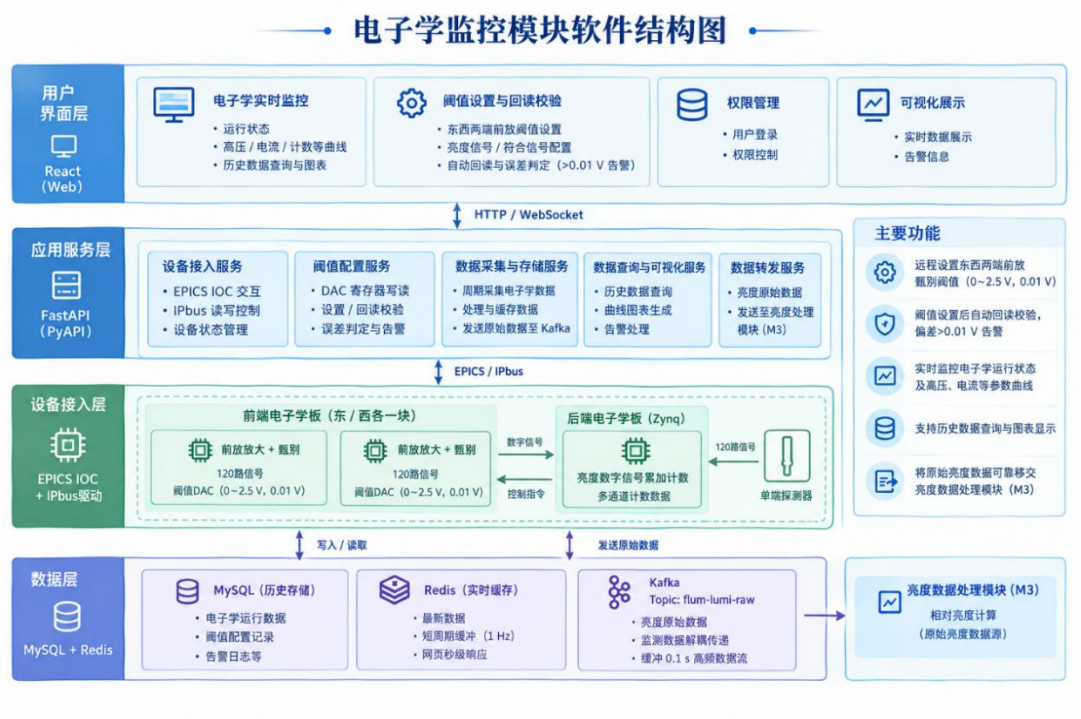


图 4-1 电子学监控模块业务功能图

数据流：IOC 周期采集电子学遥测 → Python 采集服务通过 CA 订阅 → Redis + MySQL；

原始亮度数据流：读出适配器从 ZYNQ 乒乓缓存批处理读取计数帧 → 交给亮度处理模块处理。

4.2.4 关键设计要点

- 东西两端对称建模：端别作为 PV/Kafka Key 的一级维度，页面按端别分组渲染，新增端别仅需配置扩展；
- 回读闭环校验：阈值设置不以"写入成功"为判据，以"回读在容差内"为判据，杜绝 DAC 未稳定造成的假生效；
- 阈值操作留痕：threshold_log 完整记录目标值/生效值/操作人，支持历史追溯与误操作定位；
- 乒乓缓存协同：批处理连续多帧读取完整数据帧并复原束团顺序；
- 硬件复位与安全态：远程复位需二次确认；复位期间置数据质量标志，复位完成后自动回读状态确认。

4.3 亮度数据处理与发布模块

4.3.1 模块概述

亮度数据处理模块是系统的算法核心，承担三类处理任务：①亮度数据处

理——从数据读出系统高速采集原始亮度数据，采用快速积分算法计算各束团瞬时亮度，按参考束流强度归一化，得到各束团相对亮度与总相对亮度；②将束团相对亮度、总相对亮度、时间戳、质量标志发布到 EPICS PV，提供给加速器调束系统/束团调试界面调用；③其他数据处理——监测数据规范化（通道映射、校准系数、坏道标记、状态字解析）与告警规则判定。

4.3.2 硬件工作机制

亮度信号经前端电子学放大、甄别、成形为宽度约 2 ns 的标准脉冲（零死时间成形电路，见图 4-2），以 250 MHz 速率串行产生，经 8 位串并转换后成为 62.5 MHz 并行数据送 FPGA。

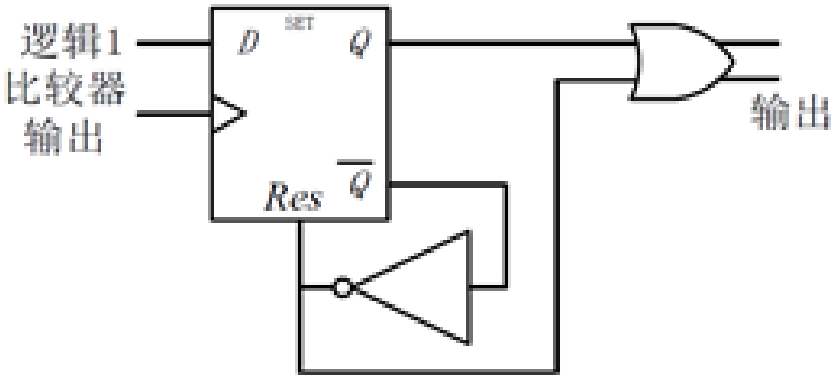


图 4-2 脉冲成形电路与输出信号

FPGA 内设置 A/B 两组存储模块构成乒乓结构：一组累加记录亮度计数，另一组响应系统命令、经 IPbus 将数据传送上位机，实现无丢数读写。一圈束流的亮度信号产生时间约 1 μs，模块记录的束流圈数与预设值比较确定乒乓切换时机。该机制是满足 6 ns 对撞周期实时性要求的基础（见图 4-3）。

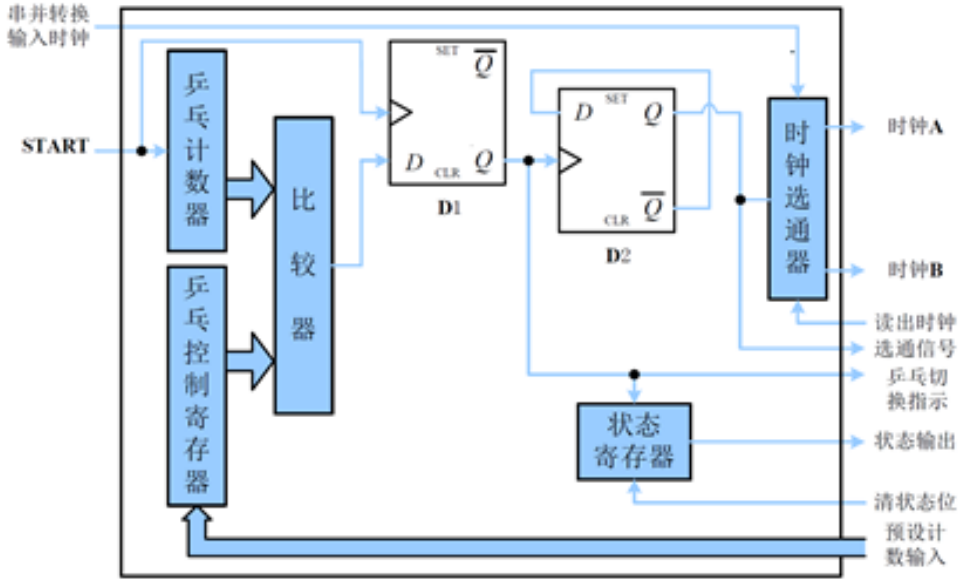


图 4-3 亮度读出板乒乓操作逻辑

4.3.3 设计目标

- 经 IPbus 从 ZYNQ 后端高速读取亮度数据，满足 6 ns 对撞周期实时性要求；
- 采用快速积分算法计算瞬时亮度，批处理连续多帧读取完整数据帧并复原数据顺序；
- 按参考束流强度归一化生成各束团相对亮度；参考源可配置，并可根据 4 组 sum 信号噪声自动或手动选择最优信号；
- 对溢出、丢数等异常置数据质量标志，供发布与显示层使用。

4.3.4 软件方案

亮度处理模块的主要软件业务分以下三类：

- 采用 C++ 或 Python 等开发亮度数据处理算法；
- 原始亮度数据经读出适配器从 ZYNQ 乒乓缓存批处理读取，不经过 EPICS 扫描路径，保证高速与低延迟；
- 计算结果周期存档 MySQL，实时值缓存 Redis；
- 原始亮度数据与相对亮度数据通过网页显示；
- 可选通过 Kafka 对外发布，用于对接智能运控等外部系统。
- **计算得到的束团相对亮度数据发送 EPICS PV 中，提供给加速器调用。**

算法引擎以固定节拍（默认 0.1 s，可调）循环执行，软件处理流程如下：

1. **数据接入**：从数据读出系统读取原始亮度计数帧；
2. **帧完整性校验**：解析帧头、束团数量、帧序号、时间戳；
3. **顺序复原**：依据乒乓缓冲帧结构复原束团顺序，缺失帧标记并计数；
4. **快速积分**：对有效时间窗内各束团计数快速积分，扣除本底，得到各束团瞬时亮度；
5. **信号选择**：评估 4 组 sum 信号噪声，自动或手动选择最优信号组；
6. **归一化**：按参考束流强度（注入电流或标准亮度值，参考源类型、系数可配置）计算各束团相对亮度与总相对亮度；
7. **质量判定**：溢出、丢数、参考源异常、超量程等置数据质量标志位；
8. **结果发布与分发**：
    - 写 EPICS PV，供加速器系统调用；
    - 刷新 Redis，供网页秒级显示；
    - 归档 MySQL，供历史查询；
    - 可选发送 Kafka，供外部系统集成。

4.3.5 关键设计要点

- **节拍保障**：整条链路设超时监控，单次循环超节拍即告警并降级；

- **本底校正**：针对 6 ns 下柱间本底可达柱高 30% 的问题，本底校正系数配置化，与硬件方联合标定；
- **坏道处理**：坏道束团标记后从总亮度统计中剔除（策略可配），柱状图特殊颜色显示；
- **数据质量标志贯穿全链**：算法产出的质量标志随数据同帧传递，显示层按标志着色、发布层供调束系统过滤使用。
- **时间同步**：全系统以控制网 NTP 对时，PV 时间戳与网页时间轴一致，便于与束流历史事件对齐分析；
- **界面延续性**：柱状图+曲线布局延续原系统习惯，调束人员平滑过渡；
- **对外接口清晰**：加速器系统通过 EPICS PV 订阅束团相对亮度；总控系统可经 EPICS CA 或其他约定接口互联。

4.4 数据采集与存储模块

4.4.1 模块职责

数据采集与存储模块统一负责全系统的数据采集、实时缓存、历史归档、消息发布与报警处理，是连接 EPICS IOC、亮度处理模块、网页监控模块与外部系统的数据枢纽。

主要职责如下：

1. **数据采集**：通过 EPICS CA 订阅高压 IOC、电子学 IOC 的 PV，获取高压、电子学等实时数据；
2. **实时缓存**：将所有 PV 实时值写入 Redis，供网页秒级响应与亮度处理模块快速读取；
3. **历史存储**：周期从 Redis 获取数据，批量归档到 MySQL；
4. **报警处理**：执行报警规则判定，产生报警记录并写入 alarm_log；
5. **消息发布**：可选通过 Kafka 对接现有接口，如智能运控等，也可用于报警；
6. **数据保留与清理**：按策略管理 MySQL 分区、聚合与清理，控制数据规模；
7. **可选可视化**：Grafana 是否使用待定，若启用则作为历史曲线展示的可选方案。

4.4.2 软件方案

数据采集与存储模块采用 Python 服务实现，以 EPICS CA 作为统一数据入口。高压 IOC 和电子学 IOC 对外提供标准 PV，Python 服务通过 CA 订阅 PV 后，将最新值写入 Redis；存储服务再按周期从 Redis 读取数据，批量归档到 MySQL。报警规则在 Python 服务内执行，产生报警记录写入 alarm_log，并可通过网页推送；Kafka 仅作为可选对外通道，用于对接智能运控等外部系统或

进行报警消息分发。网页监控模块通过 REST/WebSocket 从 Redis 和 MySQL 获取实时值、历史值和告警信息。上述数据流如图 4-4 所示。

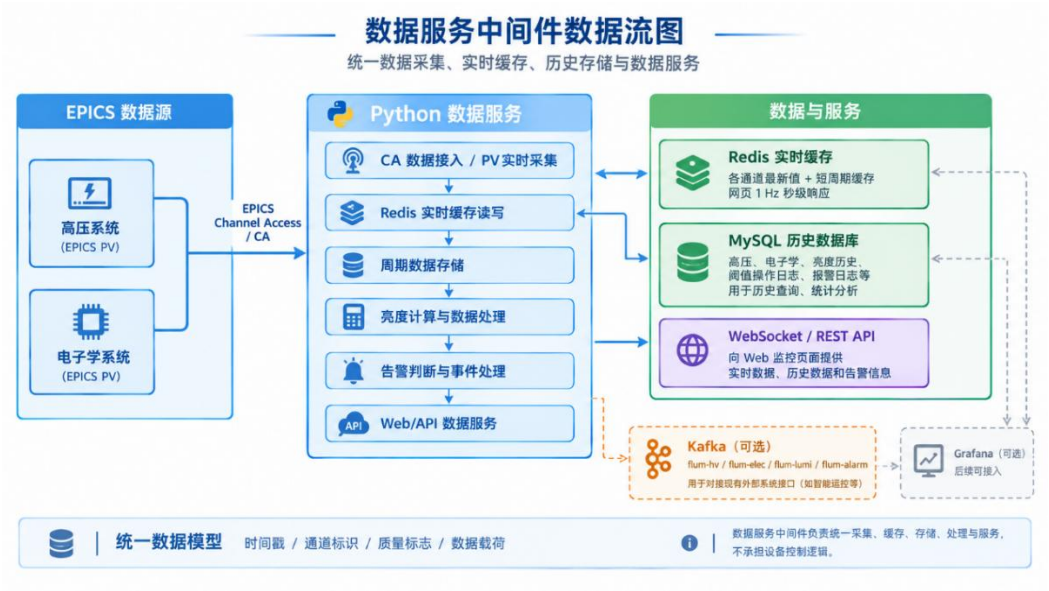


图 4-4 数据服务流程图

4.4.4 存储设计

本模块负责全系统数据采集、实时缓存、历史归档、消息发布与报警处理。数据库表、字段、索引、分区和保留策略统一见第 6 章“数据库设计”。本节仅说明存储分工和实现要点。

系统存储采用“MySQL 历史库 + Redis 实时缓存 + Kafka 可选消息”的组合。MySQL 用于保存高压、电子学、亮度、阈值、告警、审计、配置和用户等历史数据；Redis 用于保存各 PV 最新值和短周期环形缓冲，供网页秒级显示与亮度处理模块快速读取；Kafka 仅在需要对接智能运控等外部系统或进行告警消息分发时启用。

详细表结构、字段定义、索引和分区策略见第 6 章。

4.4.5 关键设计

- **采集与存储统一**：由同一个 Python 服务或一组协作服务完成，减少 CA 重复订阅与写库冲突；
- **Redis 作为实时数据枢纽**：所有 PV 实时值先入 Redis，网页和存储均从 Redis 获取，降低 CA 订阅压力；
- **数据保留策略**：MySQL 分区 + 聚合表 + 清理策略，控制数据规模；
- **与亮度处理模块分工**：原始亮度数据不经过普通 PV 采集路径，由亮度处理模块直接读取 ZYNQ 乒乓缓存；亮度处理模块计算得到的相对亮度结果写 Redis/MySQL，供本模块统一缓存与归档。

4.4.6 与其他模块关系

该模块与其他模块的关系如表 4-2 所示。通过模块分工，系统可以避免重复订阅 PV、重复写库和职责交叉。采集与存储统一由 Python 数据服务完成，亮度处理模块只负责算法与发布，网页模块只负责展示与交互。

表 4-2 与其他模块关系

相关模块	关系
M1 高压控制模块	高压 IOC 提供 PV，本模块通过 CA 订阅并缓存 Redis、归档 MySQL
M2 电子学监控模块	电子学 IOC 提供 PV，本模块通过 CA 订阅并缓存 Redis、归档 MySQL
M3 亮度数据处理与发布模块	本模块接收其相对亮度结果写 Redis/MySQL；原始亮度数据由 M3 直接读取，不走普通 PV 采集
M5 网页监控模块	本模块提供 Redis 实时值与 MySQL 历史数据，供网页 WebSocket/REST 显示
M7 系统管理与通用服务	本模块的报警、日志、配置纳入统一管理与健康监控

4.5 网页监控模块

4.5.1 模块概述

网页监控模块是系统的操作入口，拟采用 React + FastAPI 架构（WebSocket 实时推送），实现高压设备实时监控、设备状态实时监测、电子学板阈值远程配置、高压/亮度/电子学参数的实时曲线与历史图表查看、告警提示与系统管理（用户权限、参数配置、操作审计）。

4.5.2 软件结构

网页监控模块采用 React + FastAPI 分层架构，整体交互关系如图 4-5 所示。

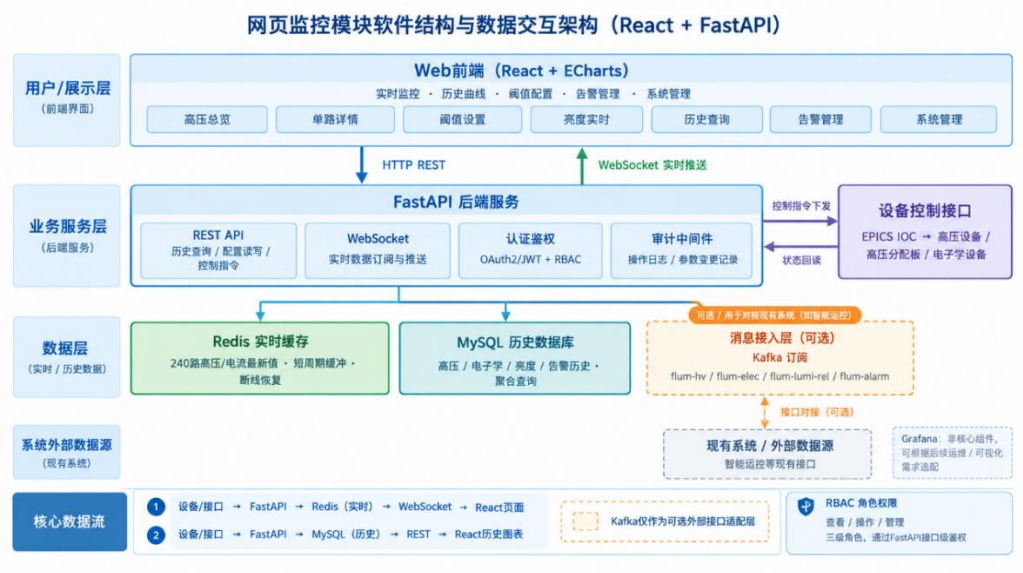


图 4-5 网页监控模块交互图

其中各个模块的构成组件如表 4-3 所示。

表 4-3 Web 模块组件构成

组件	技术	职责
Web 前端	React + ECharts	页面组件与图表渲染；WebSocket 客户端实时更新；用户交互与表单校验
后端 API 服务	FastAPI（Python）	RESTful 接口：历史查询聚合、配置读写、控制指令下发；WebSocket 推送：订阅 Kafka/Redis 转发实时数据；认证鉴权（OAuth2/JWT）；审计中间件
推送服务	WebSocket 网关	将 flum-hv/flum-elec/flum-lumi-rel/flum-alarm 四类消息按订阅关系推送到浏览器
权限管理	RBAC 角色模型	查看/操作/管理三级角色，FastAPI 依赖注入逐接口校验

4.5.3 页面规划

网页监控页面按运行监视、控制配置、历史分析、系统管理四类规划，覆盖原 C++Builder/LabVIEW 系统的全部功能，并增加历史查询、告警管理和权限审计能力。表 4-4 给出主要页面、功能和数据来源。

表 4-4 Web 页面规划

页面	主要功能	数据来源
高压总览	240 路高压/电流一览表、异常高亮、通道状态总览	Redis 最新值（1 Hz）
单路详情	单通道实时曲线、多路对比、历史查询、通断与微调控制	Redis 短周期缓冲 + MySQL 历史
阈值设置	东西两端亮度/符合阈值设置、回读校验、生效值显示	REST 写操作 + Redis 回读值
亮度实时	束团相对亮度柱状图（可放大）、总相对亮度曲线、质量标志着色	WebSocket 推送（flum-lumi-rel）
历史查询	高压/电子学/亮度历史图表、缩放、平移、CSV 导出	REST 查询 MySQL
告警管理	实时告警列表、历史检索、规则配置	flum-alarm + alarm_log
系统管理	用户与权限、参数配置、操作审计日志	REST + user_info/audit_log

4.5.4 关键设计要点

- 实时与历史分离：实时页面走 WebSocket 推送，历史/配置走 REST 查询，两类通道互不争用；
- 断线恢复：WebSocket 断线自动重连，重连后先拉取 Redis 最新值补全断线期间数据，再恢复推送；

- 图表性能：约 180 柱柱状图采用增量更新与数据采样下钻，长时间曲线查询走 MySQL 聚合（降采样）；
- 控制防误操作：所有控制表单带范围校验与二次确认，按钮状态与回读值联动（未生效显示"校验中/失败"）；
- 审计内建：审计中间件统一拦截控制类接口，自动记录操作人/参数/结果，业务代码零侵入。

4.6 系统管理与通用服务

本服务对应方案书中的监控软件硬件控制职责，负责系统级硬件控制与健康管理：远程复位亮度监测系统硬件并确认复位状态；配置 FPGA 计数缓冲区乒乓切换时间等运行参数及前端甄别阈值；周期性（约 1 s）读取硬件状态判断是否允许读数据；检测溢出等异常并告警。

- **配置中心**：通道表、数据库连接、发布节拍、告警阈值集中存放，支持热更新；
- **告警服务**：规则引擎（阈值越限、通信断开、数据超时、回读不一致）→ 网页弹窗 + 可选短信/邮件；
- **日志体系**：操作审计日志（数据库）+ 运行日志（文件滚动，保留 90 天）；
- **健康监控**：各进程心跳上报，Web 首页服务状态总览；
- **硬件复位管理**：远程复位需二次确认；复位期间置全局质量标志，完成后自动回读状态并记录审计；
- **参数管理**：FPGA 运行参数（乒乓切换时间、束团数量、帧长度）统一管理，支持查看/修改/下发/回读；
- **告警闭环**：告警产生 → 网页/通知推送 → 人工确认处置 → 状态归档，全程可追溯。

5. 接口设计

5.1 内部接口

内部接口围绕 EPICS IOC、数据服务、亮度处理、发布服务和 Web 后端展开，各接口任务如下:

- IOC ↔ 数据服务：PV 数据订阅/写入，数据采集 ↔ 存储：Python 服务写 Redis/MySQL/Kafka；
- 数据采集 ↔ 存储：采集服务写 Redis/MySQL/Kafka；
- 数据处理 ↔ 发布服务：亮度统计结果按 0.1 s 节拍经内部消息传递，发布服务统一写 EPICS PV 并刷新缓存；
- 后端 API ↔ 数据库/缓存：历史查询聚合、配置读写、审计记录。

5.2 外部接口

外部接口分为设备控制接口、数据发布接口、历史查询接口和用户界面接口。设备控制接口面向高压源、高压分配板、前放和 ZYNQ 后端电子学；数据

发布接口面向加速器调束系统和总控系统；历史查询接口面向历史数据展示；用户界面接口面向 B/S 网页监控。表 5-1 给出主要外部接口。

表 5-1 外部接口

编号	接口	类型	说明
IF-1	高压源/高压分配板通信接口	RS-232/以太网等（协议待硬件方确认）	远程控制高压输出，获取高压、电流数据，240 路
IF-2	前放控制接口	IPbus/LVDS	东西两端前放甄别阈值的远程读写（0~2.5 V）
IF-3	ZYNQ 后端电子学接口	千兆以太网 IPbus	亮度数据读取、乒乓缓存控制、硬件复位、状态监视
IF-4	EPICS 数据发布接口	Channel Access / PVAccess	发布束团相对亮度数据，面向加速器调束系统集成
IF-5	历史数据接口	EPICS Archiver/数据库	历史数据查询与图表显示
IF-6	用户界面接口	B/S 网页（WebSocket+Pyapi）	监控网页
IF-7	总控系统接口	EPICS CA 或其他,	按总控要求实现

5.3 PV 命名规范（示例）

EPICS PV 命名遵循"系统:子系统:端别:通道:属性"规则，具体前缀与加速器控制网命名规范对齐。示例：

- **高压**：FLUM:HV:E01:CH001:VMon（电压监测）、FLUM:HV:E01:CH001:IMon（电流监测）、FLUM:HV:E01:CH001:VSet（设定值）
- **阈值**：FLUM:THR:EAST:LUMI:Set（东端亮度阈值设定）、FLUM:THR:EAST:LUMI:Rdb（回读值）、FLUM:THR:WEST:COIN:Set（西端符合阈值）；
- **亮度**：FLUM:LUMI:BUNCH:REL（各束团相对亮度数组）、FLUM:LUMI:TOTAL:REL（总相对亮度）、FLUM:LUMI:QUALITY（数据质量标志）、FLUM:LUMI:PERIOD（发布节拍）。

6. 数据库设计（概要）

6.1 设计目标与原则

数据库设计面向快亮度监控系统的实时数据、历史数据、事件日志和配置管理，遵循以下原则：

- **统一数据模型**：所有监测与亮度数据统一包含时间戳、端别、通道/束团标识、质量标志和数据载荷，便于跨模块检索与对齐。
- **冷热分离**：实时值放 Redis，历史值放 MySQL，高频消息可选放 Kafka。
- **分区与聚合**：大表按时间分区，必要时生成 hour 级、day 级聚合表，控制查询与存储规模。
- **配置化保留**：原始数据保留不少于 3 个月，聚合数据保留不少于 1 年，具体周期可配置。

- **全系统对时**：以控制网 NTP 为准，数据库时间戳、PV 时间戳和网页时间轴保持一致。

6.2 MySQL 表设计

历史数据采用 MySQL 关系数据库存储，根据数据用途，MySQL 表划分为实时归档类、事件日志类和配置管理类。表 6-1 给出主要表规划。

表 5-2 数据库表规划

表名	内容	写入频率
hv_monitor	各路高压电压、电流、状态	实时（1 Hz）
elec_monitor	电子学状态、阈值生效值	实时（1HZ）
threshold_log	阈值设置/回读记录、操作人、时间	事件触发
lumi_history	束团相对亮度统计、总相对亮度、质量标志	0.1~1 Hz（可配）
alarm_log	告警事件（类型、等级、内容、处置状态）	事件触发
audit_log	关键操作审计（高压控制、参数修改、用户登录）	事件触发
device_config	设备地址、通道映射、校准系数、归一化参数等	事件触发
user_info	用户、角色、权限	事件触发

表 6-1 说明：lumi_history、hv_monitor、elec_monitor 为大表，应按时间范围分区。建议 lumi_history 按天或月分区，hv_monitor 和 elec_monitor 按天分区。常用索引包括 ts、end + channel、bunch_id、status 和 quality。如需提升长周期查询性能，可增加 hv_hourly、elec_hourly、lumi_hourly 等聚合表

6.3 Redis 实时缓存设计

Redis 作为实时数据枢纽，所有 PV 实时值先入 Redis。Key 规范为 flum:{类别}:{端别}:{通道}:{字段}。最新值使用 String，TTL 为 2 倍采集周期；短周期环形缓冲默认保留 10 分钟，刷新频率 1 Hz，与网页推送节拍对齐。Redis 异常时，网页降级读取 MySQL 近期数据并提示，EPICS 发布链路不受影响。

6.4 Kafka 主题设计（可选）

Kafka 仅用于对外集成、告警分发和模块解耦，不强制启用。建议主题包括 flum-hv、flum-elec、flum-lumi-rel、flum-alarm。消息 Key 建议按端别、通道或束团标识设计，便于外部系统按维度订阅，如对接智能运控系统。

6.5 数据一致性与降级

亮度处理模块计算得到的相对亮度结果写入 Redis 和 MySQL；原始亮度数

据不经过普通 PV 采集路径，由亮度处理模块直接读取 ZYNQ 乒乓缓存。数据质量标志随数据同帧传递，显示层按标志着色，发布层供调束系统过滤。系统统一以控制网 NTP 对时，保证 PV、数据库和网页时间轴一致。

7. 系统关键技术

1. 0.1s 发布节拍的端到端延迟：算法引擎独立进程＋Kafka 异步解耦＋前端增量渲染，全链路目标＜0.5s。
2. 240 路高压轮询效率：分批流水线轮询＋异常通道降频，避免单路超时拖垮整体。
3. EPICS IOC 统一接入：高压和电子学均基于 EPICS IOC，PV 统一管理，便于与加速器控制网集成。
4. 阈值精度保障：整型标度传输＋回读比对，从机制上保证 0.01V 分辨率与设置正确性。
5. 归一化参考源可靠性：参考源丢失时保持上一参考值并置标志位，帧内携带 ref_mode/ref_value，消费方可自行判断数据有效性。
6. 乒乓缓存无丢数：读出适配器与 M3 处理节奏对齐，批处理连续多帧读取完整数据帧并复原束团顺序。

8. 性能与可靠性设计（对应指标）

本章将需求分析中的非功能指标转化为可落地的设计措施，表 8-1 给出主要指标与设计措施。

表 8-1 主要性能指标

指标	设计措施
可用性 >95%	关键服务由看门狗守护，异常退出 5 s 内自动拉起；IOC 独立部署；网页/数据库故障切换
维护周期 >1 年	成熟开源组件（EPICS、Redis、MySQL、React、FastAPI）；日志、健康监控、远程诊断完备
阈值精度 0.01 V	整型标度传输 + 回读闭环
发布间隔 ≥0.1 s	EPICS PV 发布 + WebSocket 推送
安全性	RBAC、OAuth2/JWT、控制二次确认、审计日志、控制网/数据网隔离
可观测性	进程心跳、MySQL 写入延迟、Redis 命中率、服务状态总览
可扩展性	通道、端别、算法、存储策略插件化/配置化

9. 部署方案

- **操作系统：**Linux（EPICS、Redis、MySQL、Web、Python 采集服务）；
- **网络：**控制网与数据网隔离，必要安全策略；

- **部署节点：**
  - EPICS IOC 服务器：高压 IOC、电子学 IOC；
  - 数据采集服务器：Python 采集服务、Redis、MySQL；
  - 亮度计算服务器：快速积分与归一化；
  - Web 服务器：React 前端 + FastAPI 后端；
  - Kafka 服务器：可选；
  - Grafana 服务器：可选；

  注：服务资源足够的情况下，可以集成部署在一台单节点服务器上。

网络按控制网/数据网分区，IOC 位于控制网，Web 服务位于数据网，跨区经防火墙仅开放所需端口。

10. 技术选型汇总

技术选型遵循成熟开源、与 EPICS 兼容、便于维护、可替换和避免厂商锁定的原则。控制平台优先采用 EPICS，数据服务优先采用 Python，前端统一采用 React 技术栈，数据库采用 MySQL，缓存采用 Redis，消息队列 Kafka 按需启用。表 10-1 给出主要选型。

表 10-1 主要技术选型

层次	选型	说明
控制平台	EPICS Base + 自研设备 IOC	高压、电子学统一接入
高压控制	EPICS 设备集成 IOC；备选复用 LabVIEW 桥接	推荐 IOC 直接控制
消息队列	Kafka（可选）	发布/解耦/外部集成
实时缓存	Redis	最新值、短周期缓冲
历史数据库	MySQL 8.0	分区+聚合表
算法引擎	Python NumPy 或 C++	插件化加载
后端服务	Python FastAPI	REST+WebSocket
前端	React + ECharts + Ant Design	与 4.5 节统一
部署	Linux + Docker 可选 + systemd 守护	控制网/数据网隔离

11. 实施计划

按照项目总体安排，距验收剩余约 3 个月，实施划分为软件开发、联调运行优化、测试验收三个阶段。阶段之间允许交叉，联调前移，实验室阶段尽量完成模拟数据的初期处理。

11.1 阶段划分

表 11-1 给出阶段任务和交付物。

表 11-1 实施计划

阶段	时间	主要任务	交付物

一、软件开发	第 1~2 个月（约第 1~6 周）	①开发环境搭建与配置框架； ②EPICS IOC 开发（高压、电子学、读出系统适配）； ③亮度数据处理与发布服务开发； ④Web 前端各页面开发； ⑤数据库与缓存/消息服务部署	IOC 联调通过；Web 页面原型可用；数据处理链路打通
二、联调运行优化	第 2~3 个月（约第 7~10 周）	①实验室硬件对接联调（SY4527、电子学板、PDU）； ②6 ns 对撞周期现场试运行； ③发布节拍、归一化参数、告警阈值优化； ④与调束系统 EPICS 接口联合调试	现场稳定运行；调束系统集成验证通过；参数固化
三、测试验收	第 3 个月末（约第 11~13 周）	①功能测试（FR-1~FR-5 逐项）； ②性能测试（PR-1~PR-7 逐项）； ③复用性演示（配置修改后用于同类型系统）； ④文档整理与验收评审	测试报告；需求跟踪矩阵；用户操作手册；验收通过

11.2 进度风险与应对

进度风险主要来自高压通信协议未最终确认、束团数量/排列方式待确认、参考束流强度来源待定、现场联调时间紧张。应对策略是：先按假设协议开发适配层，协议确定后只改适配器；帧结构参数配置化；预留多参考源接口；实验室阶段完成模拟数据前期处理。详细风险及对策见第 13 章。

12. 测试与验收

12.1 测试环境

- 实验室环境：SY4527 高压机箱+A2554 插件、电子学板、ZYNQ 开发板、PDU 及串口服务器，配合模拟信号源完成全链路测试；
- 现场环境：BEPCII 6 ns 对撞周期运行期间，接入真实束流信号，与调束系统联合验证。

12.2 功能测试

功能测试覆盖高压控制、电子学监控、亮度数据处理、在线显示与发布、系统控制、历史与审计六类功能。测试采用“实验室模拟信号 + 现场真实束流”结合方式，先验证单模块，再进行集成联调。表 12-1 给出功能测试项。

表 12-1 功能测试项

编号	测试项	测试方法	通过准则
FT-1	高压控制	远程通断、升降压、单通道微调；越限/无权限操作验证	240 路均可控，生效值正确，非法操作被拒绝并告警

FT-2	电子学监控	东西两端亮度/符合阈值设置与回读；超差注入	0~2.5 V 连续可调，回读精度优于 0.01 V，超差正确告警
FT-3	亮度数据处理	模拟束流信号注入，校验积分与归一化结果	各束团相对亮度计算正确，数据顺序复原无误
FT-4	在线显示与发布	柱状图/曲线显示核对；EPICS 客户端订阅 PV	显示与原始数据一致；发布内容与质量标志完整
FT-5	系统控制	远程复位、参数配置、状态巡检、溢出告警	复位状态确认；异常检测与告警准确
FT-6	历史与审计	历史查询、导出；关键操作审计核对	查询/导出正确；审计记录完整

12.3 性能测试

性能测试在实验室和现场两个环境进行。实验室用模拟信号源验证容量、节拍和稳定性；现场在 6 ns 对撞周期下验证真实束流条件下的端到端性能。表 12-2 给出性能测试项。

表 12-2 性能测试项

编号	测试项	通过准则
PT-1	高压监控容量	240 路实时监控，刷新周期达标，无丢帧
PT-2	发布节拍	0.1 s 间隔连续发布稳定，节拍抖动在允许范围内
PT-3	端到端延迟	IPbus 读取→处理→发布链路延迟满足 0.1 s 要求
PT-4	相对亮度精度	系统级测试精度优于 5%（软件侧不引入附加误差）
PT-5	可用性	试运行期间正常运行时间/总运行时间>95%（统计口径待确认）
PT-6	长时间稳定性	连续 72 小时运行无内存泄漏、无服务异常退出

12.4 集成与验收

- 集成测试：与加速器调束系统通过 EPICS CA/PVA 完成束团相对亮度数据订阅，调束界面验证数据显示正确、刷新及时；
- 验收测试：FR-1~FR-5 全部功能通过测试；PR-1~PR-7 逐项测试合格；
- 6 ns 对撞频率下稳定完成硬件控制、状态监视、数据读取、统计与直方图显示；
- 通过配置修改演示软件在同类型设备系统上的复用能力；
- 交付物：测试报告、用户操作手册、配置文件包及全部源代码。

13. 风险与对策

系统风险分为接口与协议风险、束流参数风险、参考源风险-网页渲染风险和数据质量风险等等。风险应对遵循“先适配、后固化；先只读、后控制”的原

则。表 13-1 给出主要风险与对策。

表 13-1 风险与对策

风险	影响	对策
高压通信协议未最终确认	IOC 开发阻塞	先按假设协议开发适配层，协议确定后仅改适配器；尽快与硬件方锁定接口文档
束团数量/排列方式待确认	柱状图与帧解析不确定	帧结构参数配置化，支持现场快速调整
参考束流强度来源待定	归一化链路不完整	预留多参考源接口（手动/自动），联调阶段确认
现场联调时间紧张	验收延期	实验室阶段完成模拟数据全链路预演；关键人员现场值守
0.1 s 节拍下网页渲染压力	页面卡顿	前端增量更新 + 可调降频显示，显示与发布节拍解耦