

УДК 004.02

## МЕТОДИКА МОНИТОРИНГА ПАРАМЕТРОВ БУСТЕРА ДЛЯ ИСТОЧНИКА СИНХРОТРОННОГО ИЗЛУЧЕНИЯ NSLS-II

А. А. Дербенев, С. Е. Карнаев, Е. А. Симонов, П. Б. Чеблаков

*Институт ядерной физики им. Г. И. Будкера СО РАН,  
630090, г. Новосибирск, просп. Академика Лаврентьева, 11  
E-mail: aaderbenev@gmail.com*

Предложена методика мониторинга параметров электронного синхротрона. Проанализирована специфика настройки и управления синхротроном. Рассмотрены архитектура, логика работы и программные средства системы мониторинга. Описано программное обеспечение системы мониторинга параметров бустера для источника синхротронного излучения NSLS-II, разработанное на основе программной среды EPICS.

*Ключевые слова:* бустерный синхротрон, система мониторинга, синхронизация данных, сохранение/восстановление режимов работы.

**Введение.** Ускорительный комплекс NSLS-II, построенный в Брукхейвенской национальной лаборатории США, является источником синхротронного излучения третьего поколения и рассчитан на максимальную энергию электронов 3 ГэВ. В качестве инжектора источника синхротронного излучения используется бустерный синхротрон (бустер). На вход бустера из линейного ускорителя с частотой 1 или 2 Гц подаётся электронный пучок с током 20 мА, который в течение 300 мс ускоряется до энергии 3 ГэВ и перепускается в основное кольцо NSLS-II.

В течение 2010–2014 гг. Институт ядерной физики (ИЯФ) СО РАН выполнил контракт на изготовление и запуск в эксплуатацию бустера [2]. Помимо создания самого бустера (магнитов, вакуумной камеры, источников питания и т. п.) была разработана система управления на основе программной среды EPICS (Experimental Physics and Industrial Control System) [3], начиная от выбора аппаратуры и заканчивая созданием программного обеспечения (ПО).

Ускорение пучка в бустере осуществляется путём синхронного увеличения токов в элементах магнитной системы, включающей более сотни магнитов, и напряжения на структуре ускоряющих резонаторов, обеспечивающих рост энергии частиц. Магнитная система бустера состоит из трёх групп поворотных магнитов с функцией фокусировки пучка и трёх групп квадрупольных линз. Для питания этих магнитов используется шесть источников тока, которые в процессе ускорения синхронно обрабатывают заданные значения в диапазоне от десятков ампер до килоампера с относительной погрешностью  $10^{-3}$ . Отклонение от заданного значения в любой момент ускорения на величину, превышающую допустимую погрешность, приводит к гибели пучка. Для впуска и выпуска пучка применяются несколько импульсных магнитов, кратковременно изменяющих траекторию частиц с помощью полей величиной от сотен Гс до нескольких кГс. Отклонение полей в этих магнитах на относительную величину более  $10^{-4}$  также приводит к потере частиц. Кроме того, для регулировки положения орбиты и качества оптики бустера в процессе ускорения задействованы более полусотни корректирующих магнитов, управляемых в диапазоне  $\pm 6$  А с погрешностью  $10^{-2}$ . Всего на бустере насчитывается более тысячи параметров, меняющихся в процессе рабочего цикла и влияющих на эффективность его работы. В их число входят как основные характеристики работы устройств (например, выходной ток и напряжение), так и внутренние показатели (ток сигнала ошибки, температура, состояние

блокировок и т. п.). При стабильной работе бустера значениям его параметров свойственна повторяемость в определённом диапазоне. Но в процессе работы значения параметров могут изменяться как из-за воздействия различных внешних факторов (изменений температуры и характеристик элементов, уровня стабильности внешней сети и т. п.), так и из-за возникновения неисправностей в работе устройств. Без мониторинга значений параметров устройств появление отклонений и неисправностей в работе оборудования приводит к значительным потерям времени на диагностику и настройку бустера и привлечению специалистов по различным системам [4].

Существующие в мире системы управления аналогичными синхротронами [5] не обеспечивают возможности проводить измерения в процессе выполнения цикла и сопоставлять значения различных изменяющихся параметров. Мониторинг скалярных значений — это часть базовой функциональности EPICS 3.14, используемой для построения системы управления бустера. Но в этой среде отсутствует поддержка мониторинга для массивов данных, которые характеризуют большую часть параметров. Кроме того, важной функцией системы мониторинга является сохранение согласованных по времени значений параметров в виде фиксированного режима работы для последующего сравнения. Ряд существующих приложений для сохранения и восстановления режимов работы установок в системах управления на базе EPICS 3.14, таких как SynApps AutoSave [6], JLAB Operations Save/Restore Tool [7], SNS SCORE [8] и BNL MASAR [9], имеют ограниченную поддержку восстановления значений (без возможности сравнения), а также не имеют механизма согласования значений параметров.

Целью данной работы было создание автоматизированной системы мониторинга на основе программной среды EPICS 3.14, обеспечивающей оперативное выявление и наглядную визуализацию отклонений параметров устройств, влияющих на процесс ускорения пучка. В предлагаемой работе анализируется специфика настройки и управления бустером, описывается программная среда и формулируются требования к системе мониторинга. Затем представляется разработанная система мониторинга и делаются выводы о результатах и перспективах её использования.

**Анализ специфики управления и настройки бустера.** *Основные проблемы.* Бустер является системой, параметры которой тесно связаны между собой, и изменение одного из них требует соответствующей подстройки других. Эти взаимосвязи не всегда поддаются точному моделированию, поэтому настройка оборудования бустера представляет собой итеративный процесс, выполнение которого предполагает подбор значений параметров устройств. При недопустимом отклонении параметров устройств от нормальных рабочих значений эти отклонения подлежат выявлению с целью восстановления значений. Изменение параметров может быть обусловлено медленным их отклонением от изначальных значений в процессе работы бустера или в результате его длительного простоя, а также следствием неисправности или намеренной перенастройки (например, при регулировке энергии выпуска пучка). При этом из-за сложной взаимосвязи элементов системы поиск конкретного устройства, нуждающегося в настройке, может быть нетривиальной задачей. Кроме того, для выявления отклонений, не охваченных системой мониторинга, привлекаются специалисты и используется специальное оборудование.

Работа бустера имеет три характерные особенности. Первая — это цикличность, что приводит к периодическому обновлению данных и их доступности для обработки только в течение цикла. Вторая особенность заключается в том, что поведение многих параметров описывается массивами значений, т. е. система управления оперирует большим количеством массивов данных. Третья особенность состоит в том, что данные в системе управления обновляются с разной периодичностью и в разные моменты цикла и требуют согласования друг с другом. Например, измеренные данные о состоянии источников питания появляются только после завершения цикла, данные о токе пучка доступны в

процессе выполнения цикла, а обновление информации о состоянии вакуума вообще не связано с циклом работы бустера.

В итоге основными проблемами при настройке и эксплуатации бустера являются трудность или невозможность выявления отклонившегося параметра, быстро изменяющегося в ходе выполнения цикла, и отсутствие средств удалённой диагностики параметров.

*Программная среда.* Система управления бустером NSLS-II построена на основе среды EPICS 3.14. Главной структурной единицей в этой среде является так называемый программный контроллер ввода—вывода (Input-Output Controller, IOC) — интерфейс EPICS, который либо осуществляет взаимодействие с оборудованием, либо выполняет дополнительную обработку данных и другие функции, не связанные ни с каким реальным устройством, например набор статистики или управление другими IOC.

Большая часть параметров установки в системе управления выражается в виде численных данных определённого типа. Представление данных в EPICS базируется на специальных структурах (рекордах), описываемых в файлах конфигурации IOC и определяющих с помощью свойств своих полей тип и характеристики данных. Каждый параметр системы должен быть описан своим рекордом с уникальным именем. После инициализации IOC параметр будет доступен в системе управления в качестве PV (Process Variable) — переменной, содержащей текущее значение параметра. Для передачи данных используется общий для всех элементов системы протокол EPICS Channel Access, позволяющий получать значения PV по уникальному имени и предоставляющий программной среде преимущества одноранговой распределённой архитектуры [10], в частности гибкость и масштабируемость структуры системы управления. Программное обеспечение использует PV для единообразного хранения, обработки и доступа к данным.

Система управления имеет трёхуровневую архитектуру программного обеспечения: уровень встроенного ПО (программы, работающие внутри контроллеров оборудования), средний уровень для взаимодействия с оборудованием (IOC) и приложения верхнего уровня (пользовательские графические интерфейсы и скрипты). В такой модели каждый из программных уровней предоставляет «интерфейс» только вышележащему уровню в целях обеспечения масштабируемости (для включения в систему новых элементов), расширяемости (для простоты реализации новой функциональности) и простоты поддержки системы.

Уровень встроенного ПО является наименее изменяемым. Это связано с тем, что разработчики низкоуровневых интерфейсов часто не имеют возможности адаптировать их к конкретным потребностям специализированных решений. Тем не менее встроенное ПО должно удовлетворять потребностям системы в мониторинге основных параметров оборудования, т. е. предоставлять значения всех необходимых сигналов и состояний.

Роль ПО среднего уровня в системе управления бустера выполняют IOC, которые обычно соответствуют конкретному физическому устройству (или множеству устройств одного вида), настраиваются нужным образом с помощью файлов конфигурации и предоставляют системе управления набор PV. Контроллеры ввода—вывода могут быть запущены на множестве платформ, в частности как на обычном компьютере, так и в контроллерах CompactPCI- или VME-крейтов под управлением RTEMS или Linux. Для работы с оборудованием сами IOC могут использовать драйверы, поставляемые производителем оборудования, или иные подходы (прямая работа с памятью с помощью контроллеров системных шин CompactPCI и т. п.). Реализация взаимодействия между встроенным и средним уровнями ПО в EPICS осуществляется посредством IOC и не рассматривается в данной работе.

Приложения верхнего уровня представляют собой набор пользовательских графических интерфейсов и инструментов-скриптов, которые созданы для обеспечения возможности работы пользователя с системой. Типичными задачами для верхнего уровня ПО являются управление работой устройств и алгоритмов, визуализация состояния устройств,

отображение параметров, просмотр и сравнение актуальных и архивных данных измерений, сохранение и восстановление режимов работы установки, обработка данных, которую нецелесообразно выполнять на уровне ИОС.

**Формулировка требований к системе мониторинга.** По результатам анализа специфики работы и настройки бустерного синхротрона и возможностей программной инфраструктуры системы управления сформулированы требования к системе мониторинга состояния бустера, которая должна:

- обеспечивать формирование наборов значений параметров, относящихся к одному циклу работы бустера, т. е. выполнять согласование данных между собой;
- предоставлять актуальную информацию о статусе подсистем бустерного синхротрона, формируя индикаторы состояния параметров в каждом цикле;
- давать возможность отслеживать изменения рабочих параметров во время всего цикла ускорения;
- работать как со скалярными данными, так и с массивами;
- обеспечивать инструменты для оперативного выявления неисправностей;
- быть реализована на основе программной среды EPICS.

Общую задачу формирования индикаторов состояния для целей мониторинга рационально решить на уровне ИОС, так чтобы все приложения более высокого уровня могли использовать результаты без дублирования той же функциональности.

Для поддержки и настройки системы мониторинга инструмент для сохранения и восстановления режимов работы бустера должен при сохранении запоминать, а при восстановлении задавать не только значения уставок (прямое восстановление), но и опорные значения для всех уставок и измерений в системе. Ввиду циклического характера работы бустера требуется сохранение измерений и уставок, согласованных по времени, т. е. относящихся к одному и тому же циклу работы.

**Разработанные и реализованные решения.** Схема мониторинга является частью системы управления и представляет собой сочетание программных решений, реализованных на уровнях ИОС и приложений высокого уровня (рис. 1). Пользовательские приложения с графическим интерфейсом осуществляют сохранение и восстановление опорных значений и режимов работы бустера (Save/Restore), автоматическую подстройку меняющихся параметров (Parameter Tuner), визуализацию и диагностику состояния различных элементов установки, включённых в систему мониторинга (Status Monitor). Они используют протокол Channel Access для связи с различными ИОС, которые взаимодействуют с оборудованием, осуществляют согласование поступающих данных и реализуют алгоритм мониторинга. При этом приложение для сохранения и восстановления режимов работы

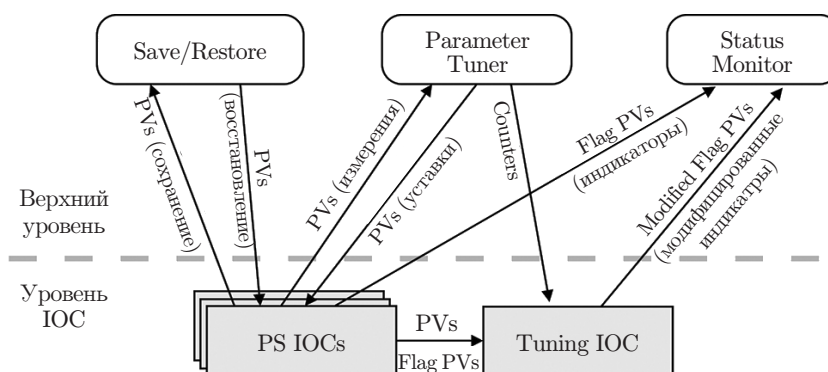


Рис. 1. Взаимодействие между уровнями ПО

производит конфигурирование системы мониторинга посредством задания опорных значений параметров, а функционирование механизма автоматической подстройки учитывается с помощью специальной схемы модификации индикаторов состояния.

Алгоритм мониторинга, изображённый на рис. 2, реализуется на уровне ИОС и может предоставлять соответствующую функциональность для любых типов данных. Для каждого отслеживаемого параметра (PV) дополнительно создаётся набор вспомогательных PV, необходимых для работы алгоритма: Reference, Difference, Tolerance, Flag, Coeff.

Алгоритм отслеживает изменения какого-либо значения, сравнивая текущие уставки и измерения в системе с соответствующими опорными (Reference) значениями. В случае нескалярных данных операция сравнения выполняется точка за точкой. Для установления величины отклонения отслеживаемого параметра системы (уставки или измерения) от опорного значения последнее вычитается из текущего значения. Если абсолютное значение этой разности (Difference) превышает определённый для данного параметра системы порог (Tolerance), то выставляется соответствующее значение предупредительного сигнала (индикатора состояния Flag) для этой уставки или измерения. Различаются небольшое (допустимое) отклонение от опорного значения (значение индикатора равно 1) и существенное отклонение (значение индикатора равно 2), разница между которыми определяется с помощью значения коэффициента Coeff (Coeff по умолчанию равен 10 и в дальнейшем может быть настроен отдельно для каждого параметра системы). Значение предупредительного сигнала (индикатора состояния), выставленное системой мониторинга, может быть использовано приложением высокого уровня для визуализации состояния установки и связано со специальной системой оповещения о нежелательных отклонениях, доступной в инфраструктуре EPICS (Alarm Handler) [11].

В ИОС для источников питания бустера поддерживаются два режима работы схемы мониторинга.

В режиме SP (сравнение с уставкой) в качестве опорного значения для определённых измерительных каналов служит значение связанной с ними уставки. Режим используется для контроля за правильностью отработки устройствами уставок в процессе настройки бустера.

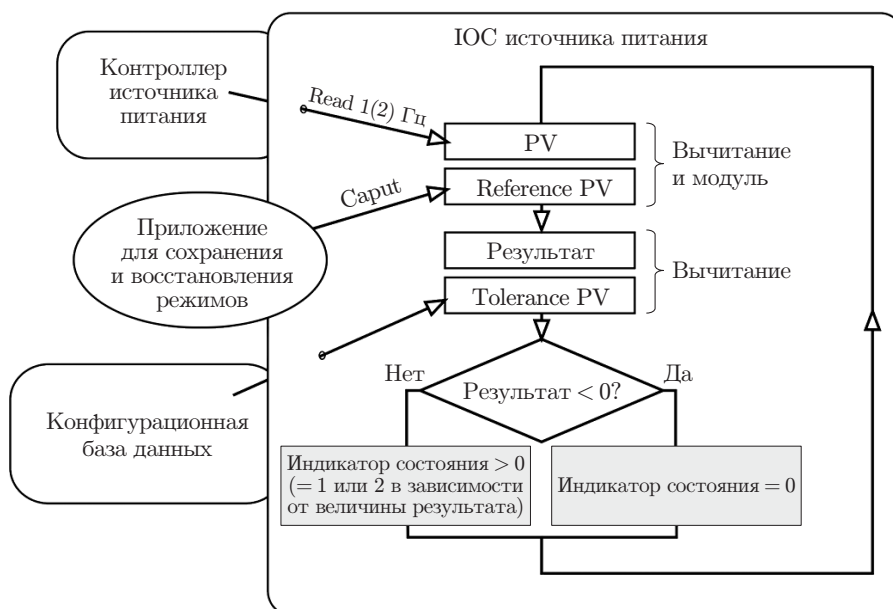


Рис. 2. Алгоритм поддержки мониторинга и архивирования на уровне ИОС

В режиме Reference (сравнение с опорным значением) опорное значение задаётся при выполнении процедуры восстановления состояния установки. Режим позволяет контролировать соответствие состояния установки некоторому состоянию, сохранённому ранее.

Разработанный подход является более мощным и гибким по сравнению со стандартным механизмом, имеющимся в EPICS [11], так как предоставляет возможность единообразно отслеживать состояние как скалярных значений параметров, так и массивов данных. Кроме того, имеется возможность прямого доступа к значению разности между текущим и опорным значениями, по которой формируется индикатор состояния, а для переопределения опорного значения достаточно выполнения только одной записи в Reference PV (против четырёх в стандартном механизме EPICS: HIGH, LOW, LOLO).

*Модификация индикаторов состояния.* В некоторых случаях значение индикатора состояния не должно приниматься во внимание. Примером может служить работа программы автоподстройки, которая автоматически модифицирует уставки для приближения значений измерений определённых параметров оборудования к их опорным значениям (в частности, подстройка уставки магнита для получения нужного значения поля). Алгоритм мониторинга может быть усложнён, для того чтобы учитывать подобные случаи, но такой подход не является целесообразным, поскольку создаст зависимость между приложением высокого уровня и уровнем ИОС. В целях обеспечения гибкого взаимодействия между высокоуровневыми приложениями и алгоритмом мониторинга разработана и внедрена схема модификации индикаторов состояния на уровне ИОС. Алгоритм работы схемы модификации изображён на рис. 3. Данный механизм вычисляет значения модифицированных индикаторов состояния на основе значений индикаторов, формируемых алгоритмом мониторинга, гибко учитывая при этом наличие в системе работающих механизмов автоподстройки параметров, особенности поведения аппаратуры и т. д. Механизм работает следующим образом. Активное приложение автоподстройки по мере обновления данных уведомляет о своей работе систему модификации индикаторов состояния, записывая значение Counter в специальные PV (см. рис. 1), инициализированные для тех параметров, которые охвачены данным процессом автоподстройки. Проверка значения Counter выполняется в ИОС в каждом цикле бустера. Если его значение положительно (что означает активность приложения автоподстройки), то модифицированный индикатор состояния зануляется, а значение Counter уменьшается на единицу. В случае выключения или неисправности приложения

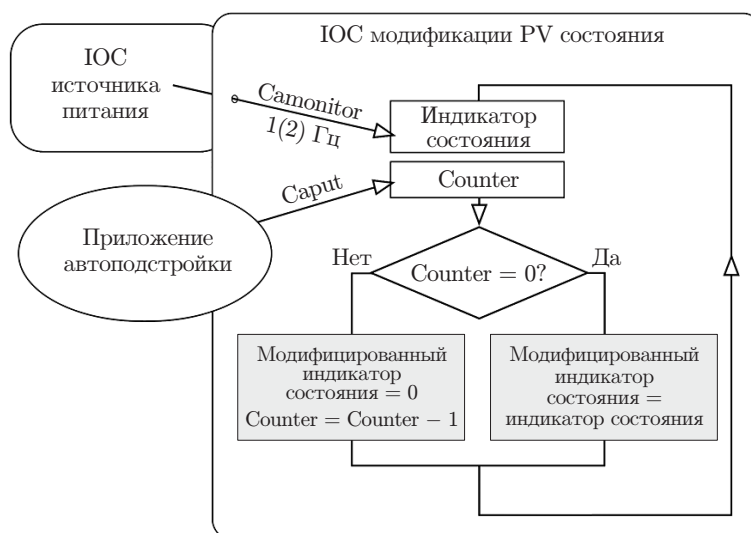


Рис. 3. Схема взаимодействия программы автоподстройки с алгоритмом мониторинга

автоподстройки значение Counter со временем занулится и значению модифицированного индикатора будет присвоено значение соответствующего индикатора состояния. В такой схеме не возникает никакого конфликта с работой самой схемы мониторинга, поскольку в неё не требуется вносить дополнительных изменений, а приложения высокого уровня имеют в своём распоряжении как обычные, так и модифицированные индикаторы для решения любых задач, связанных с мониторингом.

*Согласование данных.* Для того чтобы учесть циклический характер процесса работы бустера, на уровне ИОС реализована схема обеспечения согласования значений по меткам времени. Внедрение этой схемы в систему управления позволяет назначать одну и ту же EPICS-метку времени всем записанным в течение цикла уставкам и всем полученным в этом цикле измерениям. Метка времени при этом формируется общей системой синхронизации и не зависит от фактического (в пределах цикла) момента записи или измерения. Такой подход устраняет затруднения с синхронизацией данных, связанные с разной скоростью и режимом работы различных устройств, и с задержками при передаче и обработке данных. Как следствие, все данные, поступающие в систему, естественным образом синхронизированы по значению временных меток и могут быть при необходимости легко разбиты по циклам для дальнейшего анализа и визуализации.

*Монитор состояния устройств бустера.* Набор экранов для визуализации состояния установки Booster Status Monitor (BSM) создан на основе приложения Control System Studio [12] и предназначен для наблюдения за работой всего бустера, за состоянием его элементов в целом и за состоянием их отдельных параметров с возможностью быстрого просмотра значений. Приложение отображает на экране данные о состоянии установки в виде разноцветной подсвеченной схемы бустера (рис. 4).

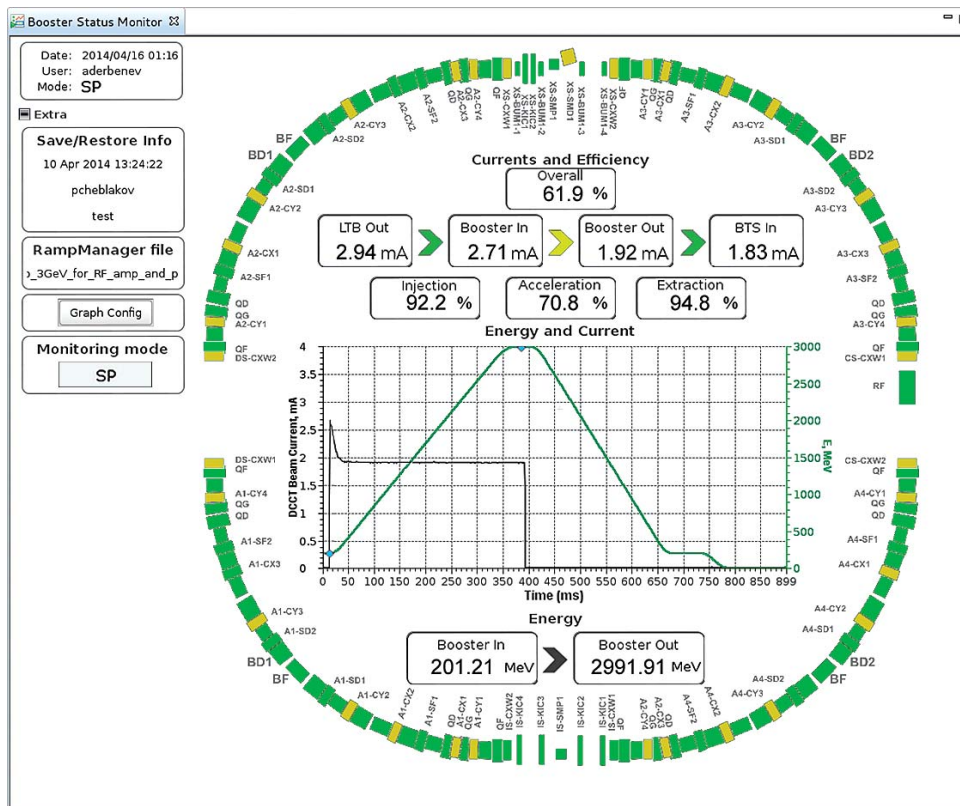


Рис. 4. Главное окно приложения Booster Status Monitor

Состояние каждого из параметров системы может характеризоваться как «хорошее» (зелёный цвет — нормальный режим работы), «предупреждение» (жёлтый цвет — снижение эффективности работы) и «неисправность» (красный цвет — устройство не работает должным образом). Для того чтобы узнать состояние произвольного элемента системы, требуется учесть все его индикаторы состояния. Это реализовано с помощью приложения на уровне ИОС, в котором индикаторы состояния для отдельных параметров системы группируются поэлементно и логически суммируются в интегральные состояния элементов. Полученные интегральные индикаторы отображаются на главном экране BSM. Пользовательский интерфейс позволяет оператору в одно действие получать как детальную информацию обо всех параметрах выбранного элемента, так и посмотреть значение любого параметра.

*Сохранение и восстановление режимов.* Графическое Python-приложение для сохранения и восстановления режимов работы бустера Application for Save and Restore предоставляет оператору набор инструментов для полного или частичного сохранения и восстановления согласованного (относящегося к одному циклу) состояния элементов установки. Каждое сохранённое состояние называется «режимом». Согласованность значений между собой обеспечивается схемой согласования временных меток (описано выше). Процесс сохранения состояния бустера в согласованном режиме требует отслеживания изменений в течение нескольких циклов работы, чтобы все сохраняемые параметры успели обновиться (с учётом задержек обработки и передачи данных).

При сохранении состояния в файл режима записываются значения уставок и измерений. Однако при выполнении восстановления только уставки заносятся в оборудование (восстановление значений для измерений бессмысленно), но при этом все сохранённые значения задаются так же, как опорные (Reference) для всех параметров, содержащихся в данном режиме, чем обеспечивается конфигурирование вышеописанного алгоритма мониторинга.

**Заключение.** Для системы управления бустерным синхротроном NSLS-II разработана и введена в эксплуатацию целостная схема мониторинга данных на уровне ИОС, представлен набор пользовательских графических интерфейсов и инструментов [13]. В предлагаемой работе впервые в практике управления подобными ускорительными установками был реализован механизм непрерывного согласованного мониторинга множества параметров с предоставлением всех данных на уровень, доступный пользовательским приложениям. Такой механизм позволяет сравнивать текущие уставки и измерения в системе с соответствующими опорными значениями и при наличии отклонений формирует предупредительный сигнал, что облегчает локализацию отклонений в системе и диагностику неполадок. Это дало разработчикам возможность создавать приложения для анализа состояния установки и внедрять в систему управления сложные автоматизированные процессы по настройке режима функционирования синхротрона: коррекцию орбиты пучка в процессе ускорения, стабилизацию систем инжекции и выпуска частиц. Архитектура программного инструментария обеспечивает расширяемость системы для добавления новой функциональности как на уровне ИОС, так и в приложениях высокого уровня. Созданная структура ПО поддерживает масштабируемость системы и делает предельно простым добавление в систему нового мониторируемого оборудования. Поскольку взаимодействие программного обеспечения с оборудованием осуществляется стандартным образом, приложения могут быть с минимальными модификациями развёрнуты помимо EPICS в какой-либо другой программной среде, предоставляющей унифицированный доступ к данным в системе управления. Разработанный подход может быть применён для мониторинга других систем с большим количеством параметров. По результатам работы представителями сообщества разработчиков EPICS принято решение о внедрении предложенных подходов в следующих версиях программного инструментария.



## СПИСОК ЛИТЕРАТУРЫ

1. **Willeke F.** Status of the NSLS-II project // Proc. of IPAC'11. N. Y., USA. P. 732–736.
2. **Gurov S., Akimov A., Anchugov O. et al.** Status of NSLS-II booster // Proc. of IPAC'13. P. 1864–1866.
3. **Проект EPICS.** URL: <http://www.aps.anl.gov/epics/> (дата обращения: 03.06.2014).
4. **Gros P., Bobault S., Loulergue A.** The 3Hz power supplies of the soleil booster // Proc. of EPAC 2006. Edinburgh, Scotland. P. 2652–2654. URL: <https://accelconf.web.cern.ch/accelconf/e06/PAPERS/WEPLS118.PDF> (дата обращения: 03.06.2014).
5. **Ludeke A.** Application of Digital Regulated Power Supplies for Magnet Control at the Swiss Light Source. Cornell University Library. URL: <http://arxiv.org/pdf/cs.SE/0111019> (дата обращения: 03.06.2014).
6. **Mooney T. M.** synApps: EPICS application software for synchrotron beamlines and laboratories // Proc. of PCaPAC 2010. Saskatoon, Canada. THCOMA02. P. 106–108. URL: <https://accelconf.web.cern.ch/accelconf/pcapac2010/papers/thcoma02.pdf> (дата обращения: 03.06.2014).
7. **Проект BURT: Back Up and Restore Tool.** URL: <http://www.aps.anl.gov/epics/extensions/burt/> (дата обращения: 03.06.2014).
8. **Galambos J., Patton J., Pelalía T.** SCORE — A save, compare, and restore application for snapshotting machine settings // Proc. of ICALEPCS07. Knoxville, USA. P. 656–658. URL: <https://accelconf.web.cern.ch/accelconf/ica07/PAPERS/RPPB23.PDF> (дата обращения: 03.06.2014).
9. **Yang L., Choi J., Hidaka Y. et al.** Development progress of NSLS-II accelerator physics high level applications // Proc. of IPAC-2012. New Orleans, USA. P. 4005–4007. URL: <http://accelconf.web.cern.ch/AccelConf/IPAC2012/papers/thppr018.pdf> (дата обращения: 03.06.2014).
10. **Нестуля Р. В., Сердюков О. В., Скворцов А. Н.** Масштабируемая отказоустойчивая среда управления сложными технологическими объектами на основе одноранговой распределённой архитектуры // Автометрия. 2013. **49**, № 6. С. 70–78.
11. **Проект EPICS Alarm Handler.** URL: <http://www.aps.anl.gov/epics/extensions/alh/> (дата обращения: 03.06.2014).
12. **Проект Control System Studio.** URL: <http://controlsystemstudio.org/> (дата обращения: 03.06.2014).
13. **Karnaev S., Cheblakov P., Derbenev A. et al.** Status of the NSLS-II booster control system // Proc. of ICALEPCS-2013. San Francisco, USA. P. 362–365. URL: <http://accelconf.web.cern.ch/AccelConf/ICALPCS2013/papers/morpc108.pdf> (дата обращения: 03.06.2014).

*Поступила в редакцию 3 июня 2014 г.*