УДК 621.865.8

Гибадулин Руслан Рашитович,
студент, 4 курс, Институт цифровых интеллектуальных систем,
направления подготовки 15.03.04 «Автоматизация технологических процессов и производств»
ФГАОУ ВО «МГТУ «СТАНКИН»
Россия, г. Москва
Сидоренко Максим Сергеевич,
студент, 4 курс, Институт цифровых интеллектуальных систем
направления подготовки 15.03.06 «Мехатроника и робототехника»
ФГАОУ ВО «МГТУ «СТАНКИН»
Россия, г. Москва
Колесников Евгений Александрович,
студент, 4 курс, Институт цифровых интеллектуальных систем
направления подготовки 15.03.06 «Мехатроника и робототехника»
ФГАОУ ВО «МГТУ «СТАНКИН»
Россия, г. Москва

ИНТЕГРАЦИЯ РОБОТИЗИРОВАННОГО МАНИПУЛЯТОРА В АВТОМАТИЗИРОВАННУЮ ТЕХНОЛОГИЧЕСКУЮ ЛИНИЮ

Аннотация: *В статье рассмотрена задача интеграции роботизированного манипулятора в состав автоматизированной технологической линии. Предложена функциональная архитектура, предусматривающая разделение задач между контроллером робота, программируемым логическим контроллером линии, подсистемой функциональной безопасности и средствами диспетчеризации. Сформирован алгоритм межмашинной синхронизации на основе состояний готовности, разрешения запуска, подтверждения завершения операции и обработки отказов. Показано влияние задержек координации на расчетную*

производительность линии и сформулированы рекомендации по диагностике, восстановлению после сбоев и масштабированию роботизированной ячейки. Практическая значимость подхода состоит в создании типовой структуры интеграции, которую можно адаптировать к различным производственным операциям и типам оборудования.

Ключевые слова: *роботизированный манипулятор, технологическая линия, программируемый логический контроллер, промышленная автоматизация, синхронизация, OPC UA, функциональная безопасность.*

Annotation: *The article considers the integration of a robotic manipulator into an automated technological line. A functional architecture is proposed in which motion control, line coordination, functional safety and supervisory functions are separated between dedicated subsystems. A state-based handshake algorithm is formed using readiness, start permission, cycle completion confirmation and fault processing signals. The influence of coordination delays on the calculated line throughput is demonstrated, and recommendations for diagnostics, recovery after failures and scaling of the robotic cell are formulated. The proposed approach can be adapted to various manufacturing operations and equipment configurations.*

Key words: *robotic manipulator, technological line, programmable logic controller, industrial automation, synchronization, OPC UA, functional safety.*

Современная автоматизированная технологическая линия представляет собой совокупность взаимосвязанных машин, транспортных устройств, датчиков, исполнительных механизмов и средств управления. Включение в такую систему роботизированного манипулятора позволяет автоматизировать загрузку и выгрузку деталей, межоперационное перемещение, сборку, сортировку, укладку, обслуживание станочного оборудования и другие повторяющиеся операции. Однако установка промышленного робота сама по себе не обеспечивает требуемого результата: манипулятор должен быть согласован с логикой линии, технологическим оборудованием, системой безопасности и средствами диагностики. Поэтому ключевой задачей

становится не выбор отдельного робота, а построение корректной архитектуры его интеграции в производственный цикл.

Требования к безопасности промышленных роботов и робототехнических систем в российской нормативной базе установлены ГОСТ Р 60.1.2.1-2016 и ГОСТ Р 60.1.2.2-2016. Первый документ распространяется на промышленные роботы, второй - на робототехнические системы, роботизированные модули и процессы их интеграции [1; 2]. Для программируемых контроллеров применяется комплекс стандартов IEC 61131, принятый в России в виде национальных и межгосударственных стандартов; ГОСТ Р МЭК 61131-3-2016 устанавливает синтаксис и семантику языков программирования контроллеров, включая Structured Text, Ladder Diagram и Function Block Diagram [3]. Это создает нормативную основу для разделения функций локального управления движением робота и координации всей технологической линии.

Целью статьи является разработка типовой структуры и алгоритма интеграции роботизированного манипулятора в автоматизированную технологическую линию с учетом обмена сигналами, межмашинной синхронизации, диагностики и требований безопасности. В рамках работы предлагается двухуровневый принцип управления: контроллер робота отвечает за траекторию и работу исполнительных осей, а ПЛК линии - за последовательность технологических операций, межблокировки и взаимодействие с остальным оборудованием.

Функциональное разделение задач управления

При интеграции манипулятора целесообразно исключать дублирование функций между контроллером робота и ПЛК. Роботизированная система должна сохранять автономность в части кинематики: расчет траектории, интерполяция, управление сервоприводами, контроль допустимых положений и выполнение локальной программы относятся к контроллеру робота. ПЛК технологической линии, напротив, должен оперировать более крупными состояниями: «робот готов», «деталь обнаружена», «станок готов к

обслуживанию», «цикл разрешен», «операция завершена», «ошибка». Такое распределение снижает связанность программного обеспечения и упрощает последующее обслуживание.

Таблица 1.

Функциональные уровни интеграции

Подсистема	Основные функции	Типовые данные обмена
Контроллер робота	Траектория, сервоприводы, локальная программа, управление захватом	Ready, Busy, CycleDone, RobotFault, позиция/режим
ПЛК линии	Последовательность операций, межблокировки, координация оборудования	StartEnable, MachineReady, PartPresent, Recipe, Reset
Контроллер безопасности	Аварийный останов, ограждения, безопасные зоны, разрешение движения	SafetyOK, E-stop, GuardClosed, SafeStop
HMI/SCADA	Индикация, журнал событий, счетчики, диагностика и сервис	Состояния, коды ошибок, время цикла, счетчики продукции

Предлагаемое разделение позволяет менять модель манипулятора или отдельные элементы технологического оборудования без полного переписывания логики линии. Для этого интерфейс между ПЛК и роботом должен быть представлен ограниченным набором стандартизированных команд и состояний. В отечественной учебной литературе по промышленным роботам и роботизированным комплексам подчеркивается необходимость согласования программного управления, технологической последовательности и взаимодействия исполнительных подсистем [7; 8]. Для современных систем целесообразно дополнительно использовать промышленную сеть, обеспечивающую передачу расширенных диагностических данных.

Структура интегрированной роботизированной ячейки

На рисунке 1 представлена предлагаемая функциональная структура. Центральным координатором технологического цикла выступает ПЛК линии. Он получает информацию о наличии детали, готовности рабочего места и состоянии робота, после чего формирует разрешение на запуск роботизированной операции. Контроллер робота выполняет заранее подготовленную программу перемещения и возвращает в ПЛК состояние

выполнения. Контроллер безопасности функционирует независимо от обычной логики технологического цикла и формирует разрешение SafetyOK только при выполнении заданных условий безопасности.

Рисунок 1. Функциональная структура интеграции роботизированного манипулятора

Для обмена оперативными командами могут применяться дискретные входы/выходы либо промышленная сеть, поддерживаемая конкретными контроллерами. На верхнем информационном уровне целесообразно использовать унифицированное представление состояния оборудования. Спецификация OPC UA for Robotics Part 1 описывает информационную модель для вертикальной интеграции робототехнических систем и передачи данных о состоянии в более высокие уровни производства, включая ПЛК линии, MES и облачные системы [5]. Такой подход особенно полезен при построении единого мониторинга для нескольких роботизированных ячеек.

При проектировании интерфейса необходимо разделять циклические сигналы управления и диагностические данные. К первым относятся разрешение запуска, готовность, занятость, завершение цикла и аварийное состояние. Ко вторым - номер выполняемой программы, текущий режим, счетчик циклов, код последней ошибки, показатели нагрузки и сервисная информация. Это уменьшает объем критичного обмена и делает работу линии предсказуемой при потере некритичных диагностических сообщений.

Алгоритм межмашинной синхронизации

Для согласованной работы ПЛК и робота предлагается использовать конечный автомат состояний. После включения выполняется инициализация и сверка исходных состояний. Далее ПЛК проверяет отсутствие аварий, наличие разрешения системы безопасности и готовность связанных механизмов. Запуск роботизированной операции разрешается только при одновременном выполнении всех необходимых условий. Логическое условие запуска можно представить выражением:

$$Start = SafetyOK\ AND\ RobotReady\ AND\ MachineReady\ AND\ PartPresent\ AND\ NoFault.$$

После формирования команды Start робот должен подтвердить переход в состояние Busy. Если подтверждение не получено за установленное время, ПЛК фиксирует ошибку связи или запуска и переводит ячейку в безопасное диагностическое состояние. По завершении движения контроллер робота формирует CycleDone. ПЛК подтверждает прием результата, обновляет состояние технологической линии и только после этого разрешает следующую операцию. Такой двухсторонний обмен исключает ситуацию, при которой короткий импульс или потеря одного сообщения ошибочно воспринимаются как завершенный цикл.

Таблица 2.

Рекомендуемый минимальный набор сигналов обмена

Сигнал	Направление	Смысл	Контроль
RobotReady	Робот → ПЛК	Робот готов к внешнему запуску	Должен быть снят при аварии/неподходящем режиме
StartEnable	ПЛК → Робот	Разрешение выполнить цикл	Выдается только после проверки межблокировок
Busy	Робот → ПЛК	Цикл принят и выполняется	Контролируется тайм-аут перехода
CycleDone	Робот → ПЛК	Цикл завершен штатно	Подтверждается ПЛК перед новым запуском
RobotFault	Робот → ПЛК	Ошибка робота или программы	Блокирует дальнейшие команды до сброса
SafetyOK	Safety → ПЛК/робот	Условия безопасной работы выполнены	Формируется независимой системой безопасности

Дополнительно рекомендуется применять контроль последовательности состояний. Например, переход CycleDone без предварительного Busy следует считать некорректным. Аналогично повторная команда запуска не должна формироваться, пока не завершена процедура подтверждения предыдущего цикла. Для сетевого обмена полезны счетчик транзакции или номер задания: они позволяют отличать актуальное подтверждение от задержанного сообщения прошлого цикла. В программной реализации ПЛК эти функции удобно оформлять в виде отдельного функционального блока, что соответствует модульному подходу, предусмотренному ГОСТ Р МЭК 61131-3-2016 [3].

Надежность, диагностика и безопасность

Особое значение имеет поведение системы при отказах. После аварийного останова, потери связи или перезапуска одного из контроллеров автоматическое продолжение движения недопустимо без сверки фактического состояния оборудования. ПЛК должен определить положение технологического процесса: находится ли деталь в захвате, свободна ли рабочая зона, завершена ли операция станка, какой этап выполнял робот до остановки. На основе этой информации выбирается либо безопасная процедура восстановления, либо перевод оборудования в ручной сервисный режим.

Требования ГОСТ Р 60.1.2.2-2016 относятся непосредственно к интеграции промышленного робота в робототехническую систему и должны учитываться при проектировании ограждений, зон доступа, аварийных остановов и других мер снижения риска [2]. В архитектуре линии функции функциональной безопасности следует отделять от обычной последовательностной логики. Сигнал SafetyOK может использоваться ПЛК как разрешающее условие, однако формирование безопасного останова и контроль защитных устройств должны выполняться средствами, предназначенными для соответствующих функций безопасности.

Для диагностики рекомендуется регистрировать не только итоговый код «авария робота», но и источник нарушения: отсутствие готовности, тайм-аут запуска, потеря детали захватом, несоответствие датчиков, недоступность рабочей позиции, срабатывание защиты, отказ коммуникации. На HMI целесообразно отображать текущее состояние конечного автомата и последнее ожидаемое условие. Такой подход сокращает время поиска причины простоя и облегчает ввод линии в эксплуатацию.

Проверка программной интеграции должна включать сценарии штатного цикла и преднамеренно созданных нарушений: задержку сигнала готовности, пропуск подтверждения CycleDone, открытие защитного ограждения, аварийный останов, потерю связи, перезапуск ПЛК и отказ датчика наличия детали. Практика построения промышленных роботизированных комплексов показывает, что устойчивость совместной работы в значительной степени определяется формализованным протоколом состояний, межблокировок и подтверждений между участниками технологического цикла [6; 8]. Поэтому правила обмена должны быть явно заданы независимо от конкретного производителя оборудования.

При расширении линии до нескольких манипуляторов возникает необходимость унифицировать представление каждой роботизированной ячейки. Для этого можно использовать типовую структуру данных, содержащую режим работы, готовность, занятость, завершение операции, код ошибки, номер задания и счетчики. Локальный ПЛК или центральный контроллер линии взаимодействует с каждым роботом через одинаковый программный интерфейс, а различия конкретного оборудования скрываются внутри адаптационного слоя.

OPC UA for Robotics ориентирован на вертикальную интеграцию и передачу состояния робототехнических систем в производственные информационные уровни [5]. Практический интерес такого подхода состоит в возможности централизованно собирать состояние манипуляторов, статистику циклов и диагностические сведения без переноса функций

реального времени в SCADA или MES. В результате контур технологического управления остается локальным и детерминированным, а верхний уровень получает унифицированные данные для мониторинга и анализа.

Современная отечественная учебная литература по автоматизации и промышленной робототехнике рассматривает роботизированный комплекс как часть общей системы технологического управления, в которой функции манипулирования, контроля, программного управления и взаимодействия с технологическим оборудованием должны проектироваться совместно [6; 8]. Для базовой технологической линии применение сложных интеллектуальных методов не является обязательным; более существенный эффект на первом этапе дает корректная структуризация состояний, отказоустойчивая синхронизация и прозрачная диагностика. Интеллектуальные функции могут добавляться как следующий уровень развития системы, не меняя основной протокол безопасного взаимодействия.

Интеграция роботизированного манипулятора в автоматизированную технологическую линию требует согласования управления движением, последовательности операций, обмена данными, диагностики и функциональной безопасности. Предложена архитектура с распределением функций между контроллером робота, ПЛК линии, подсистемой безопасности и HMI/SCADA.

Для взаимодействия ПЛК и робота сформирован алгоритм конечного автомата с проверкой готовности, подтверждением запуска и завершения, тайм-аутами и контролем последовательности состояний. Расчетный пример показал заметное влияние задержек синхронизации на производительность короткоцикловой линии, поэтому оптимизации подлежит не только траектория манипулятора, но и логика межмашинного обмена.

Практическая значимость подхода определяется модульностью: стандартизированный набор команд и состояний упрощает замену и расширение компонентов. Перспективным развитием является применение

OPC UA, предиктивной диагностики и цифрового моделирования роботизированной ячейки.

Использованные источники:

1. ГОСТ Р 60.1.2.1-2016/ИСО 10218-1:2011. Роботы и робототехнические устройства. Требования по безопасности для промышленных роботов. Часть 1. Роботы. — М.: Стандартинформ, 2020.

2. ГОСТ Р 60.1.2.2-2016/ИСО 10218-2:2011. Роботы и робототехнические устройства. Требования по безопасности для промышленных роботов. Часть 2. Робототехнические системы и их интеграция. — М.: Стандартинформ, 2020.

3. ГОСТ Р МЭК 61131-3-2016. Контроллеры программируемые. Часть 3. Языки программирования. — М.: Стандартинформ, 2016.

4. ГОСТ IEC 61131-2-2012. Контроллеры программируемые. Часть 2. Требования к оборудованию и испытания. — М.: Стандартинформ, 2014.

5. OPC Foundation. OPC UA for Robotics. Part 1: Vertical Integration. OPC 40010-1, Version 1.02 [Электронный ресурс]. — URL: https://reference.opcfoundation.org/Robotics/v102/docs/ (дата обращения: 16.08.2026).

6. Аббясов, В. М. Автоматизация технологических процессов. Промышленные роботизированные комплексы : учебник для вузов / В. М. Аббясов, С. Л. Петухов. — Москва : Юрайт, 2026. — 89 с. — ISBN 978-5-534-21971-5.

7. Архипов, М. В. Промышленные роботы: управление манипуляционными роботами : учебник для вузов / М. В. Архипов, М. В. Вартанов, Р. С. Мищенко. — 2-е изд., испр. и доп. — Москва : Юрайт, 2026. — 170 с. — ISBN 978-5-534-11992-3.