

От описания детали — до реза на вашем станке

Одна вертикаль: браузер → двойник станка → конфигурация стойки → LinuxCNC → серво. Без цепочки CAD → CAM → постпроцессор и без освоения тяжёлых пакетов технологом.

cncai.ru — программы и двойник станка

cncctl.ru — наладка приводов

для ПРОИЗВОДСТВА • РУКОВОДИТЕЛЮ ПРОИЗВОДСТВА • ГЛАВНОМУ ТЕХНОЛОГУ • ГЛАВНОМУ МЕХАНИКУ

От программы до сервопривода — и обратно одним кадром

- **Программа** описывает деталь или логику узла, а не станок.
- **Двойник** — источник правды: оси как требования, механизм — данные URDF.
- **«Сделать текущим»** — выбор станда и профилей осей по ролям; станок к станду не привязан.
- **Станд · cncctl** — профиль оси настраивается один раз **ГОТОВИТСЯ**.
- **Конфигурация LinuxCNC / HAL** порождается из требований и железа; **EtherCAT** ведёт приводы разных вендоров.

Один кадр обратной связи — в монитор cncctl, наладку cncctl и панель оператора у станка.



Одно значение · одно имя · один владелец записи на каждой ступени

Знание о станке живёт в четырёх местах

Ходы, пределы, параметры привода и инструмент записаны в разных местах и в разных версиях. Когда они расходятся, правым оказывается тот, кто громче.

САМ-станция

Постпроцессор под конкретный станок, у технолога.

копия 1 • ходы, стол



Стойка ЧПУ

Файлы конфигурации и параметры внутри контроллера.

копия 2 • пределы, скорости



Утилита привода

Windows-программа вендора, своя на каждый бренд.

копия 3 • ток, инерция, энкодер



Наладчик

Блокнот и память человека, который «знает, как на самом деле».

копия 4 • всё остальное

Цепочка ожиданий

Технолог ждёт САМ-специалиста, тот — наладчика. Каждое звено со своей лицензией и своим человеком.

Один маршрут для всего

Простая деталь проходит тот же путь, что сложная, — потому что маршрут единственный.

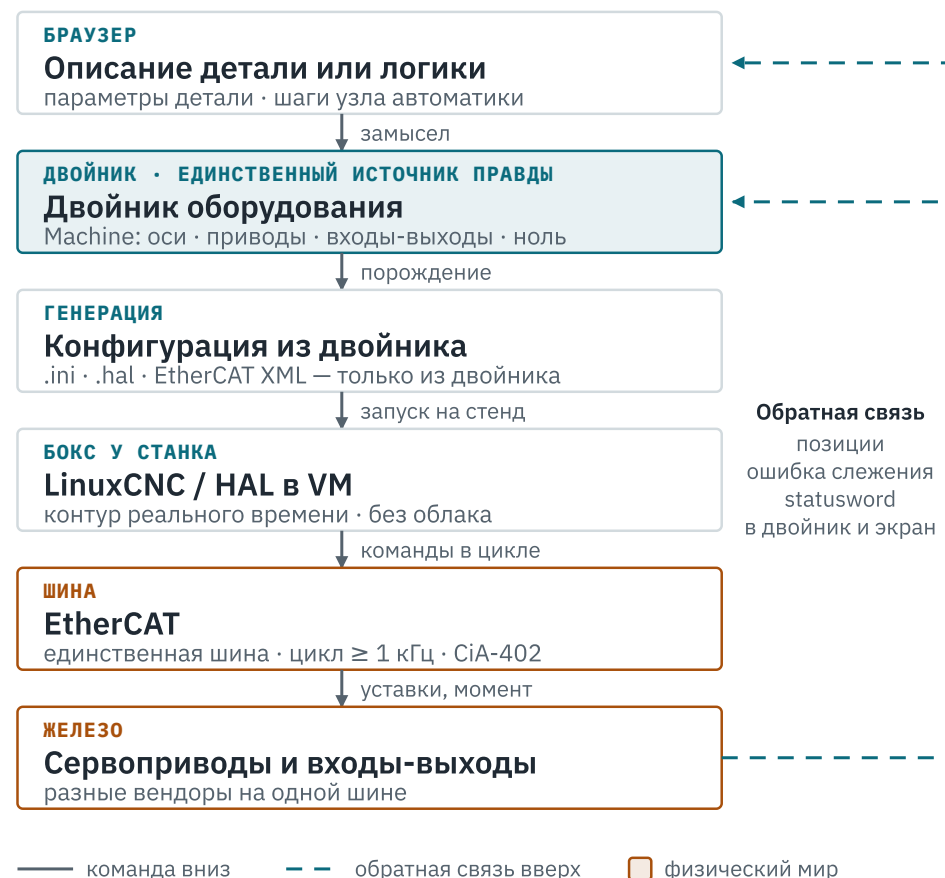
Ручной постпроцессор

Замена привода или стойки означает переписывание постпроцессора руками.

Двойник станка — единственный источник правды

- **Всё, что знает станок, живёт в двойнике:** оси, приводы, ходы, пределы, инструмент, геометрия рабочего органа.
- **Конфигурация стойки всегда порождается из двойника,** а не правится рядом с ним. Нет «правьте ini руками, но осторожно».
- **Импорт с железа только засекает двойник** — второй правды не возникает.
- **Новое свойство станка — правкой двойника.** Не правкой постпроцессора, не файлами на стойке, не заявкой интегратору.

Обратная связь идёт тем же путём назад: факт с пина возвращается в двойник и на экран.



Три роли, один источник данных

ТЕХНОЛОГ

Описывает деталь параметрами. Плита с отверстиями, фланец, диск, кольцо, вал со ступенями, карман, окно в панели, раскрой листа. Съём материала виден в 3D до металла.

НАЛАДЧИК

Видит оси, приводы и пины. Настройка привода и наблюдение за ним — там же, где паспорт оборудования, а не в отдельной утилите.

ОПЕРАТОР

Работает без облака. Локальная панель у станка: пуск, стоп, состояние. Связь с облаком для работы не требуется.

ЕДИНСТВЕННЫЙ ИСТОЧНИК ПРАВДЫ Двойник оборудования

РОЛЬ 1

Производство

ДЕЛАЕТ

описывает деталь или логику узла своими параметрами

ПОЛУЧАЕТ

программу под свой станок и результат на железе

ГРАНИЦА

без САМ-барьера и без постпроцессора

РОЛЬ 2

Интегратор

ДЕЛАЕТ

описывает оборудование данными: URDF, паспорта, привязки

ПОЛУЧАЕТ

порождённый конфиг LinuxCNC / HAL и шины без своего кода

ГРАНИЦА

оборудование — его, платформа — наша: стык назван

РОЛЬ 3 · OEM

Станкостроитель

ДЕЛАЕТ

двойник породы один раз на серию своих станков

ПОЛУЧАЕТ

каждый экземпляр — данные, а не код, платформа одна

ГРАНИЦА

порода — его код, экземпляр — данные заказчика

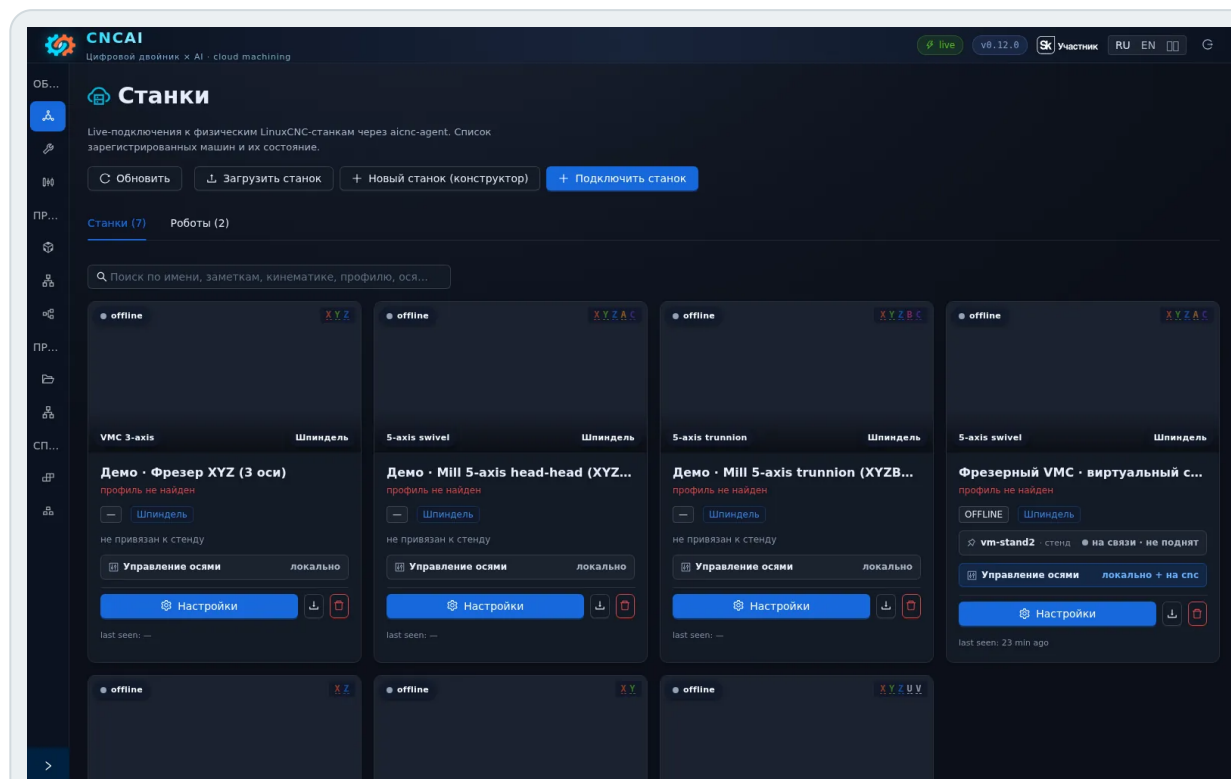
Речь об отсутствии CAD-барьера, а не о простоте вместо возможностей: сложную трёхмерную обработку поверхностей мы не заменяем и не заявляем.

От паспорта станка до панели у станка

1 • ПАРК

Все станки предприятия — в одном списке

Каждый станок со своим двойником: кинематика, оси, приводы, инструмент. Отсюда же — запуск и состояние подключения к контуру реального времени.



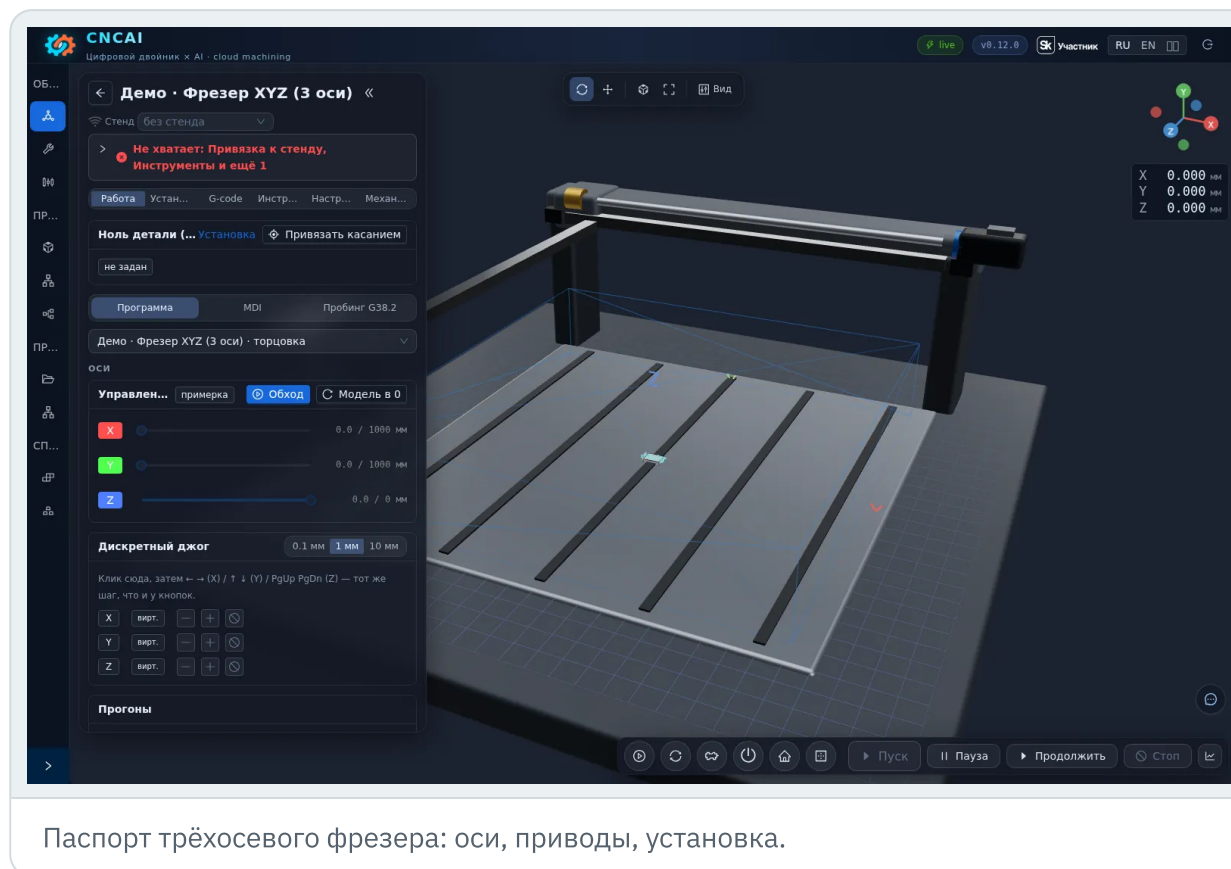
Каталог станков двойника — все породы одного пользователя на одном экране.

От паспорта станка до панели у станка

2 • ПАСПОРТ

Станок — это паспорт, а не набор файлов

Оси и их ходы, приводы и энкодеры, установка детали и ноль — всё в одном месте, у числа назван источник: замер, паспорт или «не измерено». Конфигурация стойки порождается отсюда.

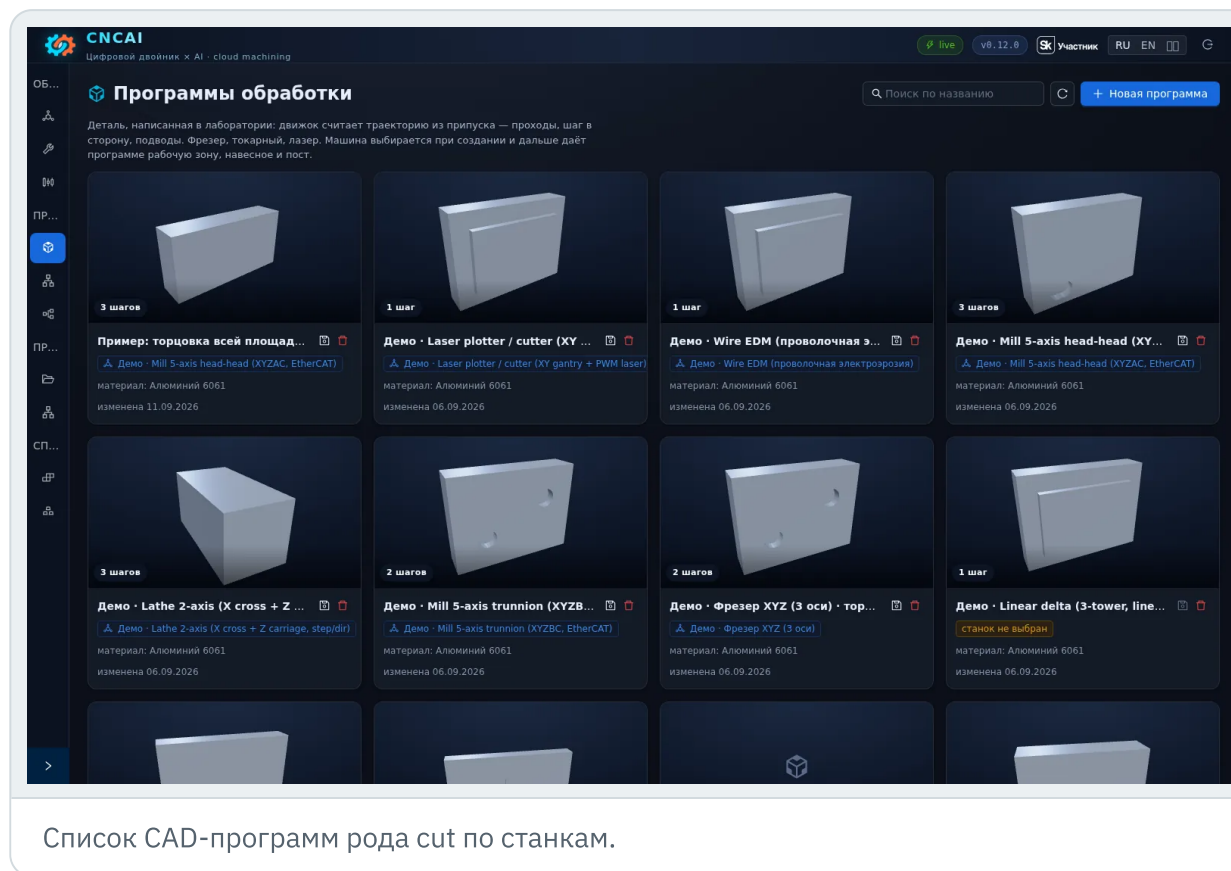


От паспорта станка до панели у станка

3 • ТЕХНОЛОГ

Деталь описывается параметрами

Технолог задаёт деталь и операции — карман, отверстия, контур — и видит съём материала в 3D до того, как включится шпиндель. Программы хранятся по станкам, с превью детали; та же программа выполняется на другом станке без правок.



От паспорта станка до панели у станка

4 • НАЛАДЧИК

Привод привязан к оси там же, где паспорт

Какой серво-слейв на какой оси, люфт, привод шпинделя — наладчик задаёт и проверяет сопоставление в одном окне, а не в утилите вендора и не в файле на стойке.

Скриншот интерфейса cncctl, раздел «Сопоставление» (Mapping). В заголовке страницы указано: «Демо · KUKA KR16 welder (6R arm + MIG/MAG torch)». В меню слева: «Файлы конфигов», «Сопоставление», «Оверлеи HAL», «Монитор · LinuxCNC», «Прогон · джог», «Осциллограф пинов», «HAL-пины (все)», «Шина · EtherCAT», «Мастер, слейвы, ср...», «СЛЕЙВ S1», «Наладка приводов ...».

В центре экрана — таблица сопоставления осей и сервоприводов:

Ось	Серво-слейв	Люфт, мм
X	поз 0 · нет на шине	0
Y	поз 1 · нет на шине	0
Z	поз 2 · нет на шине	0
A	поз 3 · нет на шине	0
B	поз 4 · нет на шине	0
C	— не привязано	0

Внизу таблицы — кнопка «Сохранить оси».

Ниже — раздел «Шпиндель»:

Привод головы: не на шине (пин / инвертор 0-10 В)

Шпиндель не на шине: роли «Шпиндель — вкл» и «Задание оборотов» ждут физических пинов на вкладке «Датчики и IO».

Внизу экрана — текст: «Привязка физических IO к ролям станка (доезд факта до пина).»

От паспорта станка до панели у станка

5 • ОПЕРАТОР

Панель у станка без облака

Локальная панель бокса: состояние бокса, станков и станда, привязка к платформе, удалённая поддержка по кнопке. Работает внутри периметра предприятия; связь с облаком для работы не нужна.

SyncTwin 85630fc4

● онлайн

Статус

Станки (VM)

Стенд

Сеть

Бокс

ХОСТ	ВЕРСИЯ	ЯДРА	RT-ИЗОЛИР.	RAM, МБ	LOAD 1M
спс	1.2.13	16	2-13	12745	3.76

VM ЗАПУЩЕНО
2

Привязка к платформе

СТАТУС	ПЛАТФОРМА	ПОСЛЕДНИЙ HEARTBEAT
привязан	https://cncctl.ru	ok (200)

Удалённая поддержка

Открывает временный защищённый канал к инженеру SyncTwin. Работает только пока вы держите его включённым — нажмёте «Отключить», доступ закроется.

● выключено

Подключиться к поддержке

Локальная панель бокса у станка — состояние, привязка к платформе, поддержка; работает без облака

Два рода программ — один исполнитель

РЕЗ

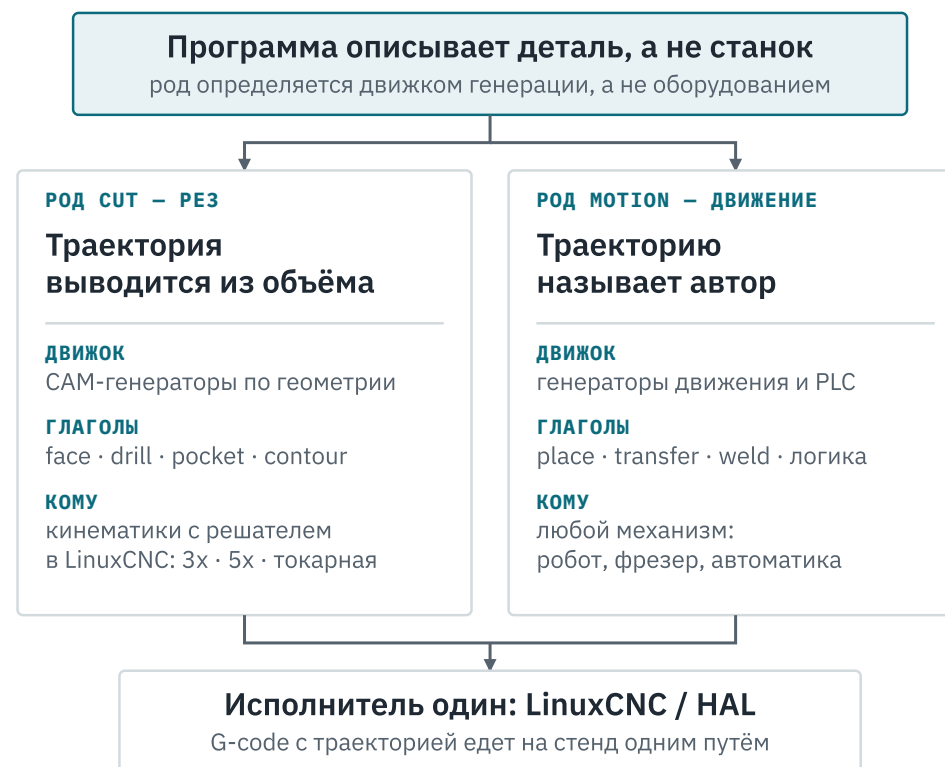
Траектория выводится из объёма детали. Технолог описывает деталь, генератор строит траекторию под конкретный станок: 15 генераторов — 2.5D, формообразующие, не-фрезерные.

ДВИЖЕНИЕ

Траекторию называет автор. Перенос, переворот, постановка, сварка — программа для роботов и узлов автоматики поверх того же двойника.

Проверка простая: меняет ли смена рабочего органа саму траекторию. Да — рез, нет — движение. Сварка — не отдельный род, а рабочий орган.

Программа описывает **деталь**, а не станок: координаты от угла заготовки, ноль детали принадлежит установке. Поэтому та же программа выполняется на другом станке без правок.



Проверка до пуска, факт после

- **Preflight.** Инструмент найден в библиотеке, вылет фрезы измерен, ноль детали задан. Незаданное — отказ с причиной, а не «успех».
- **Запуск.** Конфигурация стойки порождается из двойника и уходит на бокс одной командой; режим — виртуальные оси или серво.
- **Прогон.** Хоминг, отправка программы, исполнение LinuxCNC в реальном времени.
- **Обратная связь.** Обороты, заданные технологом в операции, найдены на пине шпинделя; оси прошли заданный путь; журнал без ошибок.

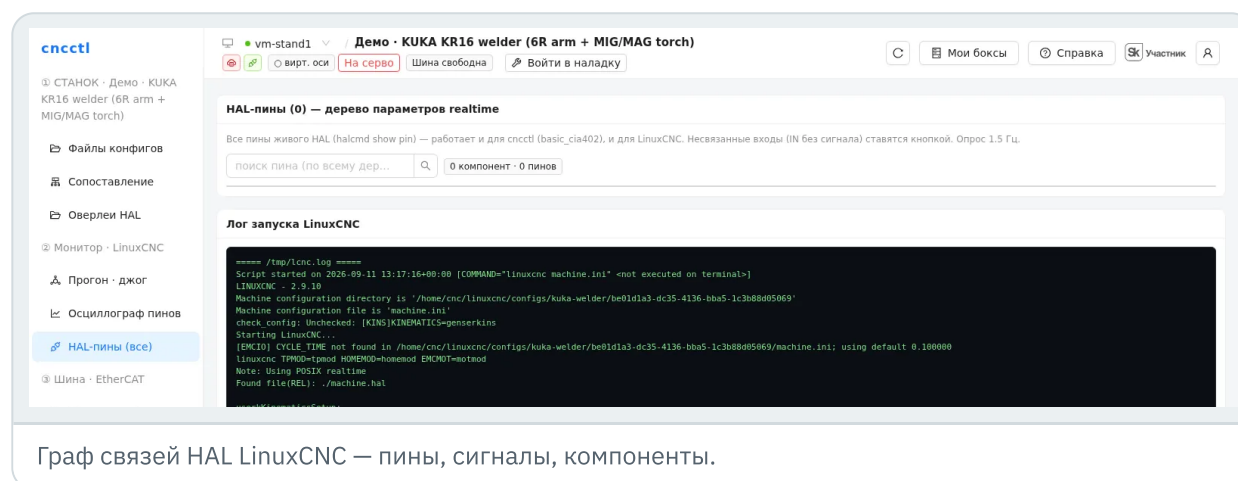
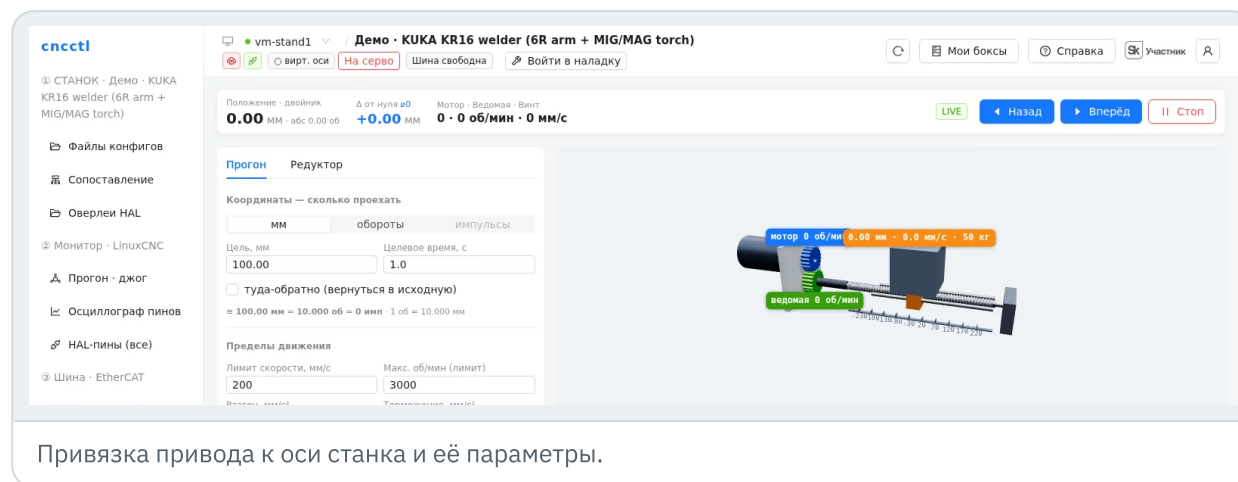
«Команда принята» и «ось поехала» различаются намеренно: факт берётся с пина, а не из ответа кнопки.



Наладка привода в браузере, в реальном времени

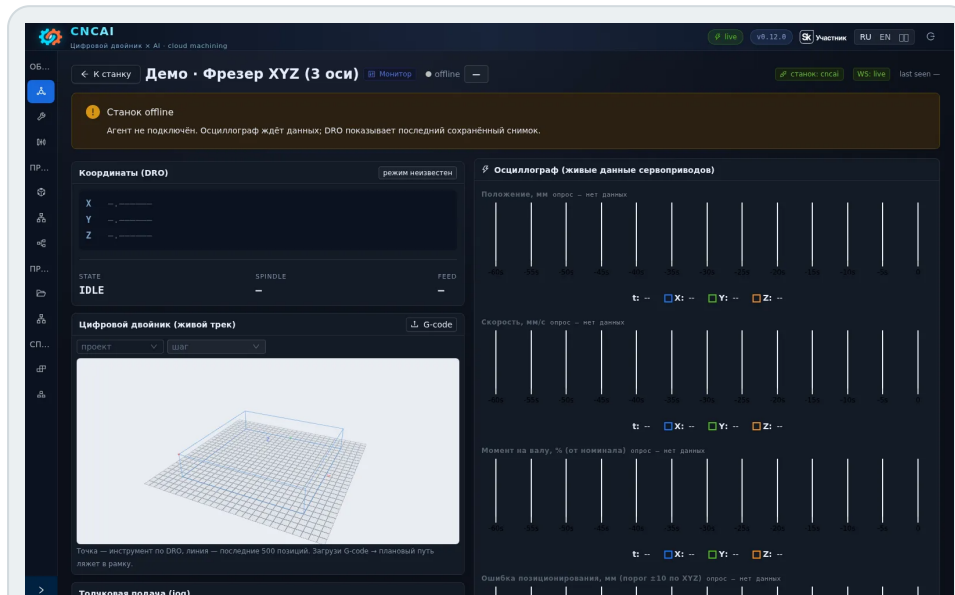
- **Осциллограф** — ток, момент, ошибка слежения, задание против факта.
- **Прямое управление CiA-402** (CSP / CSV / CST) без ПЛК-посредника.
- **Каталог параметров приводов** — словарь объектов снимается с привода автоматически.
- **Расчёт инерции и подбор редуктора** поверх 3D-двойника механизма.

Контур реального времени остаётся на боксе у станка. Облако наблюдает и настраивает, но не держит петлю управления.

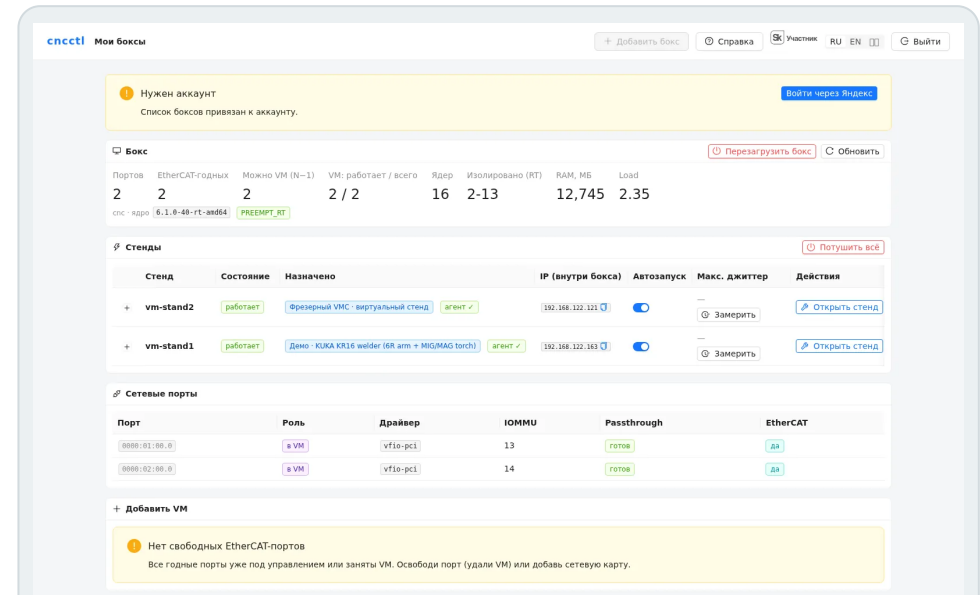


Факт берётся с железа, а не из ответа кнопки

«Команда принята» и «ось поехала» — разные события. Приёмка у нас — это значение, заданное человеком и найденное на пине: обороты шпинделя, путь осей, состояние привода.



Обратная связь в реальном времени: позиции осей, состояние.



Супервизор Бокс → VM → Станок: какие стенды подняты и чем заняты.

Доработка считается доказанной разницей ответа железа до и после правки — а не отчётом о проделанной работе.

Разные бренды приводов — на одной шине

ETHERCAT · ОДНА ШИНА, ЛИНЕЙНАЯ ТОПОЛОГИЯ, ЦИКЛ ≥ 1 КГц

МАСТЕР ШИНЫ

LinuxCNC / HAL

один мастер · драйвер lces · цикл ≥ 1 КГц

Inovance IS620N

сервопривод

Inovance SV660

сервопривод

Wecon VD3E

сервопривод

Schneider LXM28E

сервопривод

Mitsubishi MR-J4-TM

сервопривод

Omron NX-ECC202

входы-выходы

ОДИН ПРОФИЛЬ

CiA-402

одно включение,
один хоминг,
одно движение —
у любого вендора

паспорт слева:
vid, pid, PDO-карта

ВХОДЫ-ВЫХОДЫ

датчики и реле



Стенд: коуплер Omron, приводы Mitsubishi, Inovance и Wecon на одной шине EtherCAT

Проверено на железе: три Inovance IS620N выведены в ОР и проехали оси фрезерного станка; SV660 — в ОР, 30 оборотов вперёд-назад в CSP; Wecon VD3E — от онбординга до движения оси. Обмен по шине — 1 КГц.

Принято в открытое ядро: драйверы IS620N / SV660 — в linuxcnc-ethercat (PR #512, 21.08.2026), драйвер Wecon VD3E — в master. Ставится штатным пакетом, без форка.

Бокс у станка, контур реального времени — локально

- **Промышленный мини-ПК у станка** — в шкафу или рядом. Ставится одной командой, обновляется штатным пакетом.
- **Каждый станок — своя VM LinuxCNC** с ядром реального времени и своим портом EtherCAT. Один бокс — несколько станков в изоляции друг от друга.
- **Двойник, программы и наладка** — на сервере предприятия или в облаке. Петля управления туда не уходит никогда.
- **Канал пропал — станок работает.** Панель у станка и исполнение программы не зависят от связи.



Один бокс — несколько станков, каждый в своей VM и на своей шине

Одно значение — в четырёх окнах

Всё, что умеет станок, проброшено от железа до экрана и видно одинаково везде, откуда на это смотрят.

ДВОЙНИК	Где факт живёт и откуда берётся — владелец записи.
СТЕНД	Пин HAL, строка конфигурации, объект на шине — то, что исполняет.
НАЛАДКА	Через snccctl факт настраивается и проверяется осциллографом.
ПАНЕЛЬ	Оператор у станка видит и управляет без облака.

Возможность не считается готовой, пока её не видят все четыре окна: одно имя, одно значение, один владелец записи.

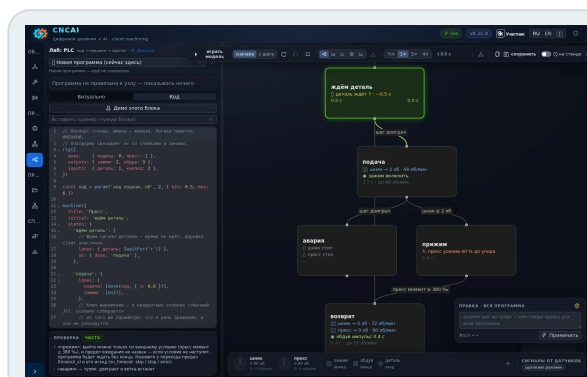


Готово, когда все четыре окна показывают одно и то же

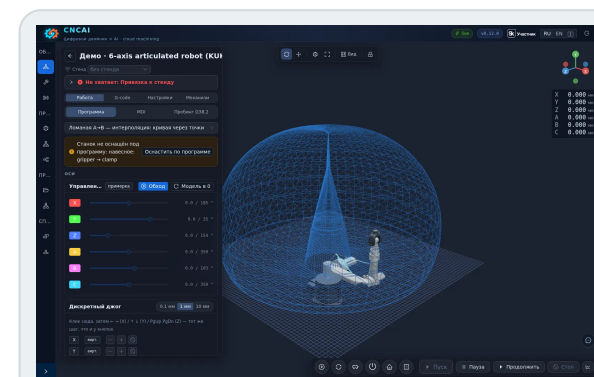
Станок — частный случай автоматки

- **Та же среда собирает логику узла:**
подача, зажим, переворот —
программы движения поверх того
же двойника, без отдельного ПЛК.
- **Механизм — данные, а не код.**
Робот, стол, узел описываются
файлом URDF: звенья, шарниры,
пределы. Новое оборудование
заводится файлом.
- **Роботы и параллельные
кинематики** — прямая и обратная
кинематика считаются в двойнике;
на сегодня отработаны виртуально.

R&D • ВИРТУАЛЬНО



Логика узла автоматки шагами —
блоки, входы-выходы.



Паспорт 6-осевого робота: звенья,
суставы, пределы.

Что мы не делаем — это часть предложения

По границе планируется работа, по обещанию «умеем всё» — только разочарование на первой технической встрече.

Не заменяем тяжёлый CAM

Сложная трёхмерная обработка поверхностей и тяжёлая пятиосевая — не наш предмет.

Не строим САПР

Своего геометрического ядра не пишем и не собираемся: деталь описывается параметрами.

Не тянем ROS

Многозвенные системы с управлением всем телом живут в другом стеке.

Не форкаем LinuxCNC

Работаем со штатными механизмами открытого ядра. Обученный наладчик остаётся нужен — и остаётся в том же LinuxCNC.

Наш предмет — **вертикаль от браузера до серво плюс двойник как единственный источник правды**. Остальное берём готовым либо не берём вовсе.

Что работает сегодня, что в работе

РАБОТАЕТ

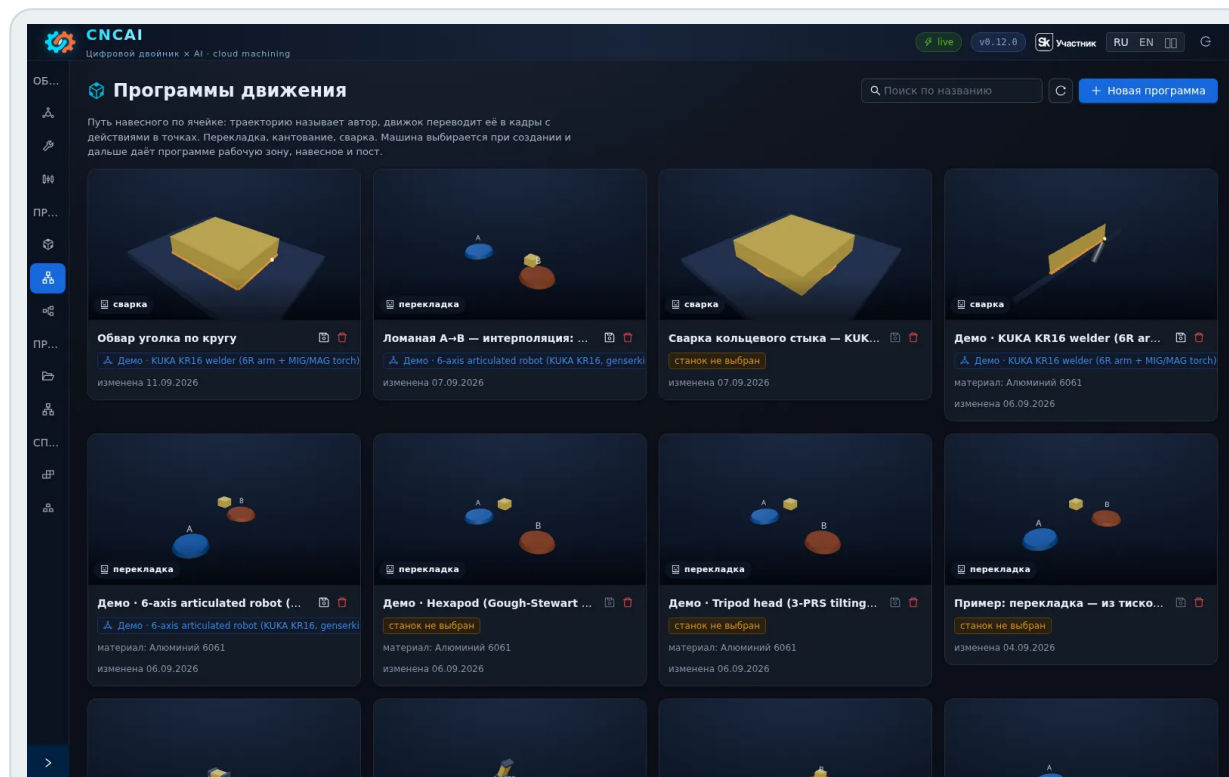
Вертикаль пройдена целиком

Программа создана в браузере, порождён G-code, конфигурация ушла на стойку, LinuxCNC отработал. Три породы: трёхосевой фрезерный, пятиосевой с поворотным столом, токарный. Приводы четырёх брендов на одной шине EtherCAT, 1 кГц. Локальная панель без облака.

R&D

Что пока виртуально

Серво на стенде работают без механики осей; роботы, параллельные кинематики и автоматика отработаны виртуально. Остальные породы проводим по одной: провести целиком → починить всплывшее → закрепить сценарием.



Одна ваша деталь на одном вашем станке

Провести её по вертикали целиком — тем же циклом, которым мы закрываем породы у себя. Результат проверяем, а не презентация.

1 · ПАСПОРТ Станок → двойник Оси, приводы, ходы, инструмент. Импорт со стойки засекает двойник.	2 · ДЕТАЛЬ Параметры → программа Технолог описывает деталь, видит съём материала в 3D, сохраняет программу.	3 · ПРОГОН На вашем железе Preflight, запуск, хоминг, исполнение. Факт снимается с пинов: обороты, путь осей, журнал.	4 · РАЗБОР Расхождения по факту Что разошлось между описанием и живым станком — чиним там, откуда утекло.
---	---	--	---

О чём стоит спросить нас прямо: какие породы прошли вертикаль и чем это доказано · что будет с приводом другого производителя · где живёт число, если экран и стойка разошлись · что работает без связи с облаком.

КОНТАКТЫ

ООО «СинкТвин Технологии»

info@synctwin.ru

+7 925 353-56-35

synctwin.ru · cncai.ru — программы и двойник станка · cncctl.ru — наладка приводов

Москва

Участник «Сколково»

Открытое ядро: LinuxCNC · EtherCAT