

5 дракониевых реакторов в одном ПК

Этот гайд поможет вам обуздать мощь сразу пяти дракониевых реакторов. Система на базе OpenComputers возьмет на себя все расчеты, предотвращая взрывы и обеспечивая стабильный приток энергии на долгие дни.

1. Список компонентов

Для стабильной работы системы управления пятью реакторами нам понадобится мощный сервер и специфическое оборудование Draconic Evolution.

Контрольный узел (Сервер)

Компонент	Кол-во	Примечание
Серверная стойка	1 шт	Основа системы
Сервер (Уровень 3)	1 шт	Максимальная мощность
Монитор (Уровень 3)	6 шт	Для сетки 3x2
Клавиатура	1 шт	Ставится на монитор
Видеокарта (Уровень 3)	1 шт	Вставляется в сервер
Процессор (ЦП) (Уровень 3)	1 шт	Вставляется в сервер
Память (Уровень 3.5)	4 шт	Для работы с 5-ю реакторами
Компонентная шина (Ур. 3)	3 шт	Для подключения всех адаптеров
Жесткий диск (4MB)	1 шт	Уровень 3
Дисковод для стойки	1 шт	Для установки ОС
EEPROM (Lua BIOS)	1 шт	Прошивка
Преобразователь энергии	1 шт	Питание сервера
Кабель	6 шт	Соединение компонентов
Адаптер	5 шт	По одному на каждый реактор

Реакторная установка (на 5 реакторов)

1. **Ядро энергии:** уже настроенное (см. [предыдущий гайд](#)).

2. **Энергетический пилон:** 2 шт.
3. **Флаксовый гейт:** 10 шт.
4. **Стабилизатор реактора:** 20 шт.
5. **Реакторный инжектор энергии:** 5 шт.
6. **Драконье ядро реактора:** 5 шт.
7. **Криостабилизированная труба:** 20 шт.
8. **Пробужденный дракониевый блок:** 40 шт (топливо).

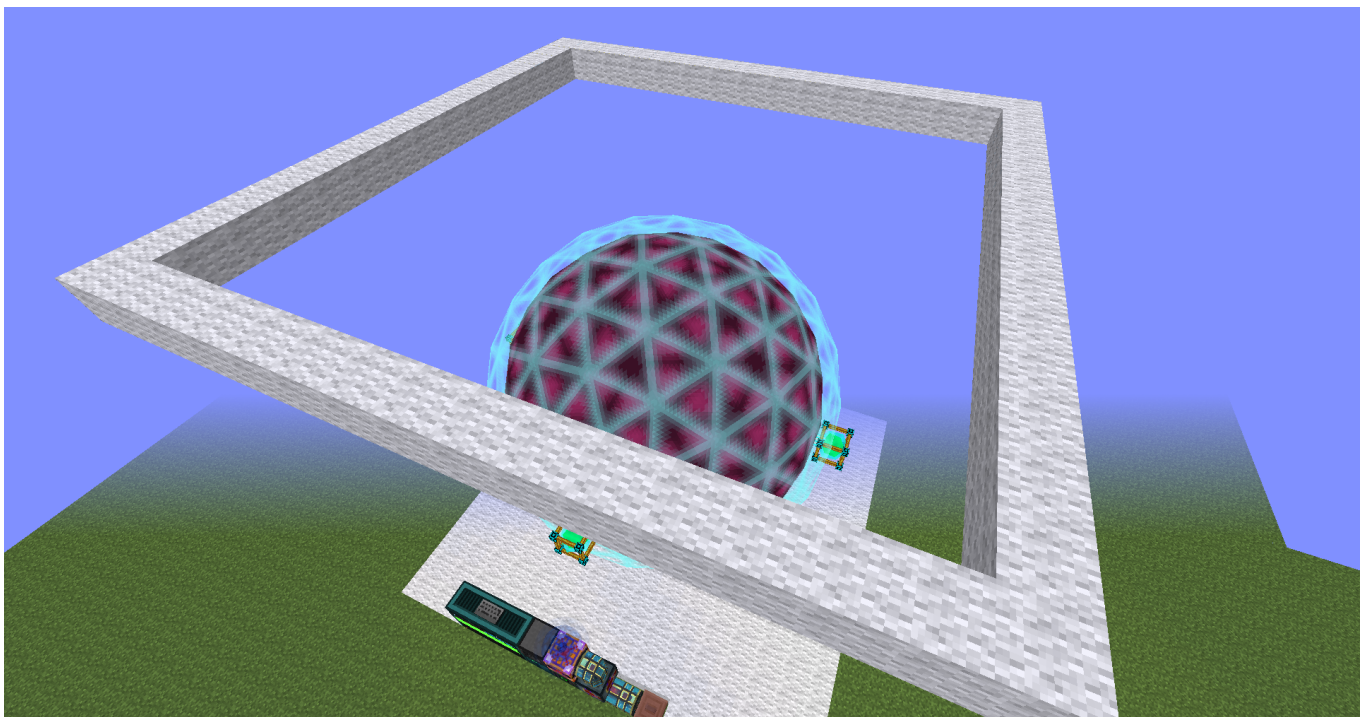
2. Установка реакторов

Физическая сборка

Первым делом нам нужно ядро энергии с программой стабилизации ядра - [гайд тут](#).

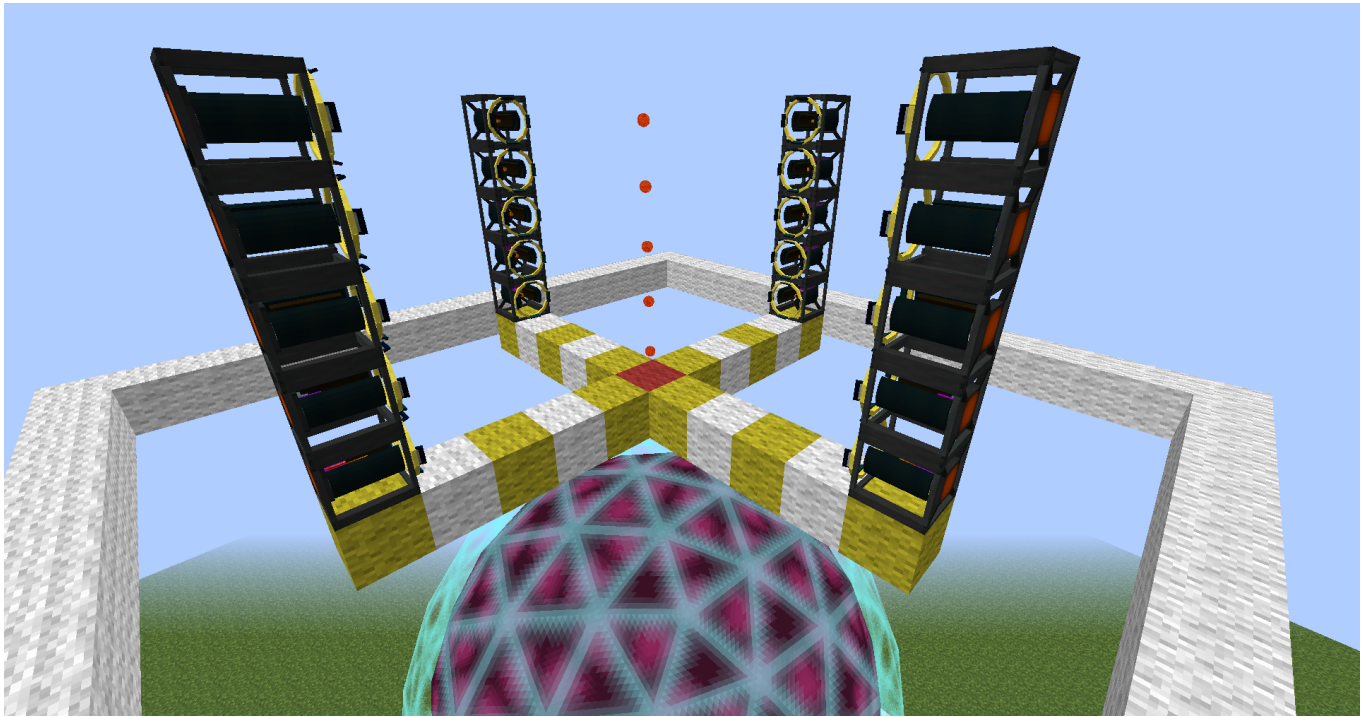
- **Локация:** Стройте реакторы прямо над вашим Энергетическим ядром. Важно делать это в пределах того же чанка, чтобы избежать ошибок десинхронизации.

Место под реакторы



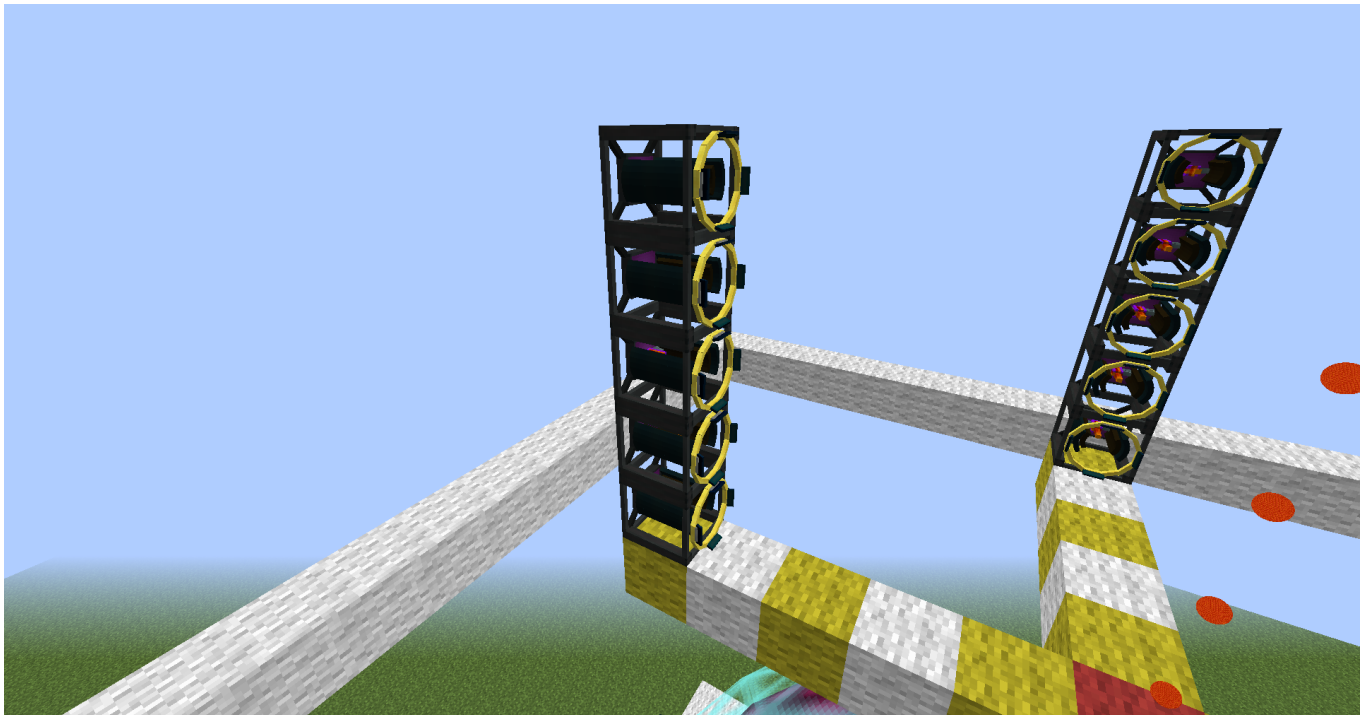
Ставим реакторы примерно так (не обязательно ставить все 5 сразу, программа работает одинаково и с одним реактором и с пятью)

Установка реакторов

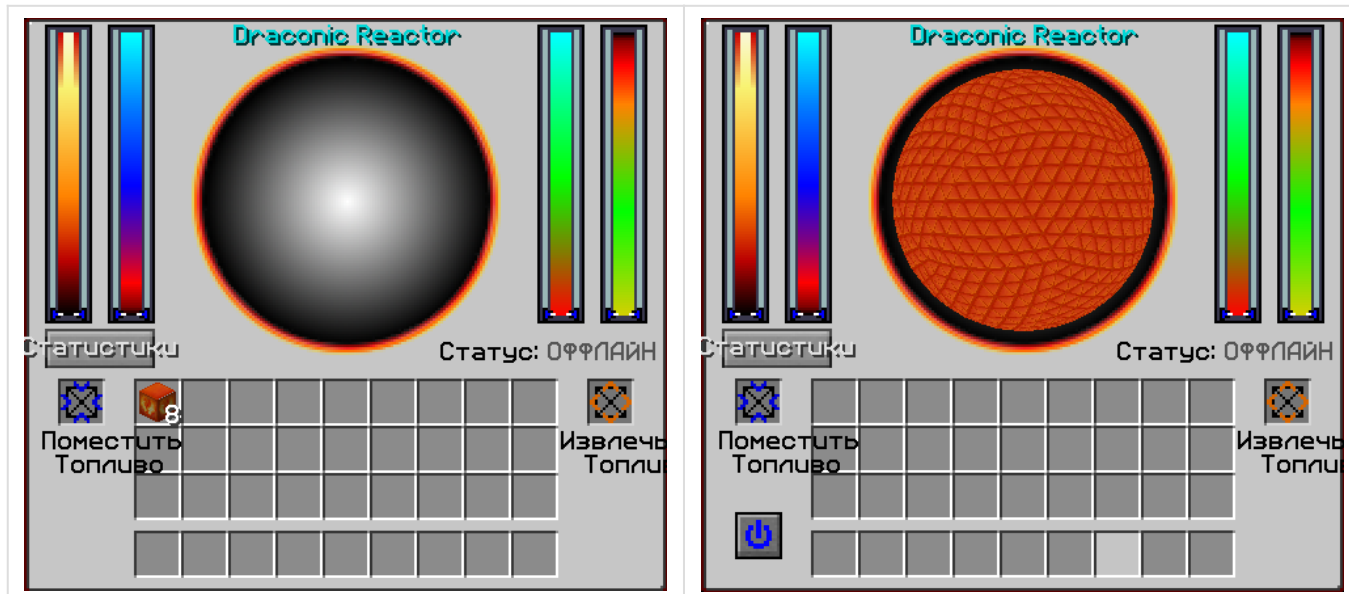


- **Порядок:** Устанавливайте компоненты реактора строго по инструкции мода, иначе структура не распознается.
- **Топливо:** Подойдите к столбу из Стабилизаторов, нажмите **ПКМ** и вставьте по 8 Пробужденных дракониевых блоков в каждый реактор.

Стабилизаторы

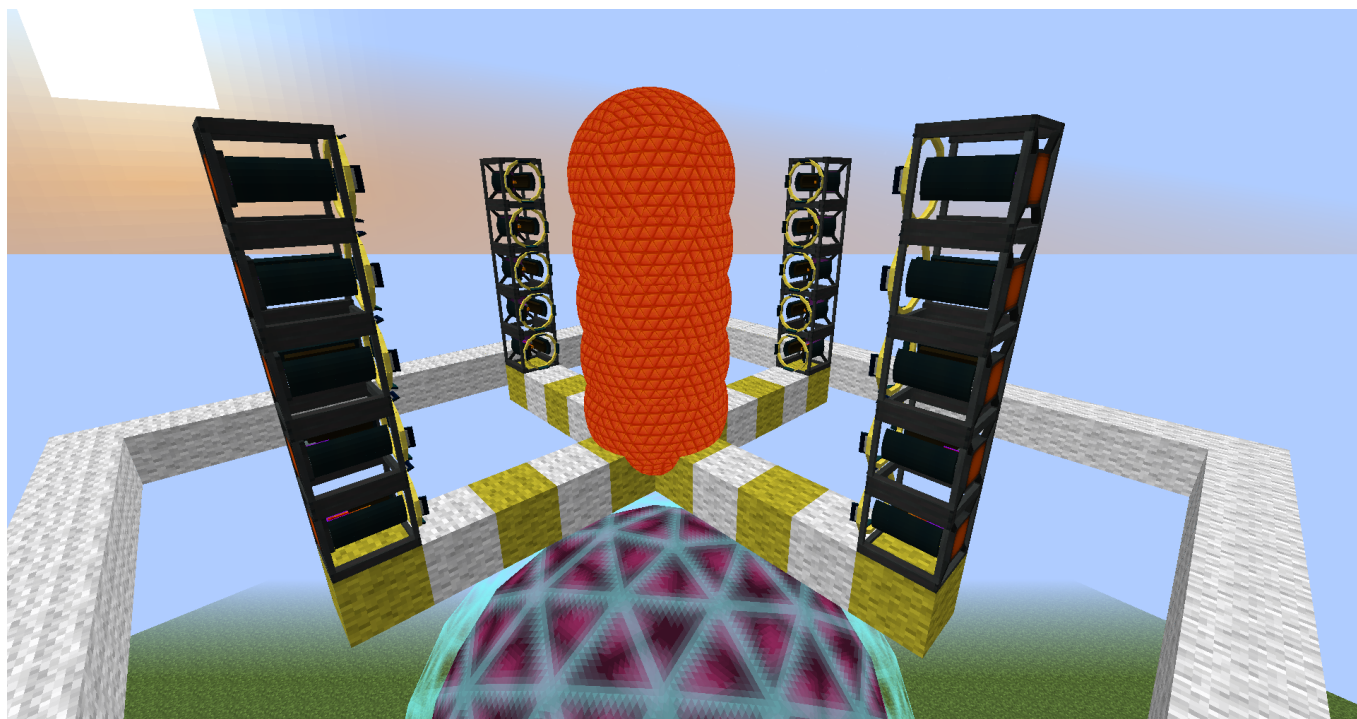


Топливо



Имеем такую картину

Реакторы с топливом



3. Настройка сервера

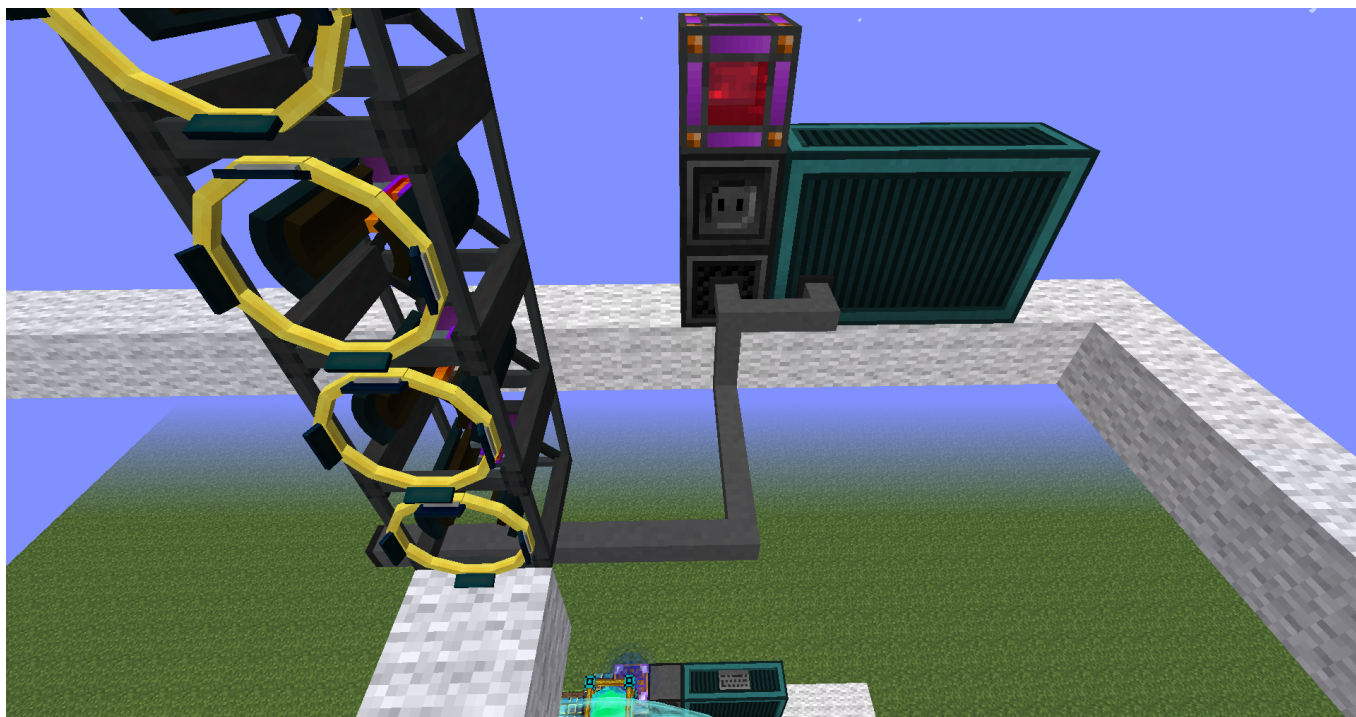
Сборка и подключение

- Установите серверную стойку, мониторы и преобразователь энергии. Проложите кабели к мониторам. Запитайте энергией преобразователь энергии.

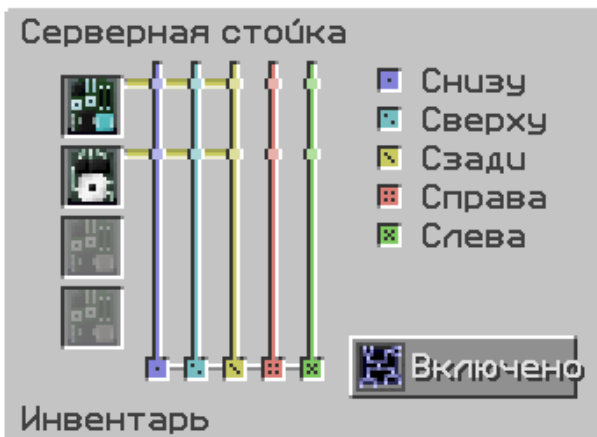
Вид спереди



Вид сзади



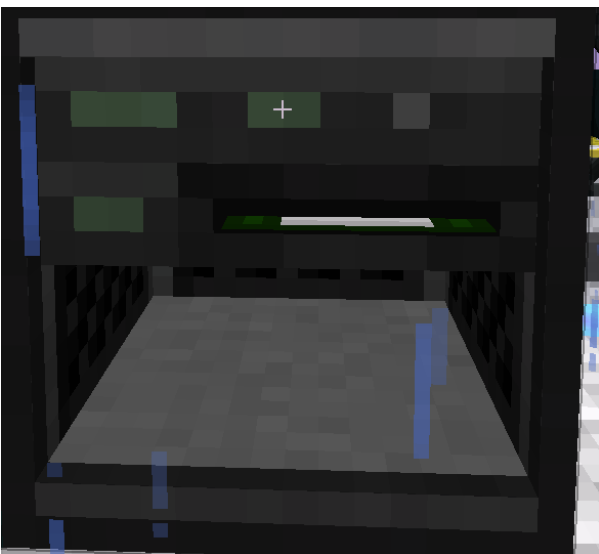
- **Настройка стойки:** Нажмите **ПКМ** по стойке. Вставьте сервер и дисковод. Кликните по желтым точкам на корпусе, чтобы задать направление сторон для подключения кабелей.

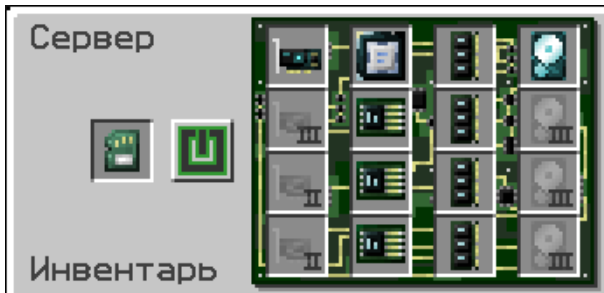


- **Установка ОС:** Вставьте дискету OpenOS в дисковод сервера через **Shift + ПКМ**.



- **Установка компонентов:** Откройте интерфейс сервера и вставьте видеокарту, процессор, память и шины.





Программная часть

Включите сервер и выполните следующие шаги:

```
install
```

```
/home # install  
Install OpenOS? [Y/n] █
```

Нажимайте **Enter**, подтверждая установку. В конце система предложит перезагрузку - снова нажмите **Enter**.

```
Installation complete!  
Reboot now? [Y/n] █
```

Безопасность и автозапуск

Добавьте себя в список администраторов (команда чувствительна к регистру):

```
useradd ник
```

```
/home # useradd DesDope  
/home # █
```

Это мы делали для того чтобы ПК могли пользоваться только вписанные в него пользователи.

Настройте автозапуск программы при включении ПК:

```
edit .shrc
```

и дописываем в самом конце такую строку

```
reactor.lua
```

```
alias l="ls -lh"
alias ..="cd .."
alias df="df -h"

alias grep="grep --color"
reactor.lua
```

И нажимаем `Ctrl + S` (сохранить) и `Ctrl + W` (закреть).

Создание кода

Создайте файл программы:

```
edit reactor.lua
```

Вставьте код. Код очень длинный, вставляйте его частями (по 250 строк) из-за ограничений буфера обмена.

Код

```
local component = require("component")
local event = require("event")
local serialization = require("serialization")
local filesystem = require("filesystem")
local computer = require("computer")
local unicode = require("unicode")
local term = require("term")
local gpu = component.gpu

-- === АВТОМАТИЧЕСКАЯ НАСТРОЙКА РАЗРЕШЕНИЯ ===
local maxW, maxH = gpu.maxResolution()
gpu.setResolution(maxW, maxH)
local W, H = gpu.getResolution()

-----

-- [КОНФИГУРАЦИЯ]
-----

local CFG = {
    filename = "/home/reactors.cfg",
    maxReactors = 5,
    updateInterval = 1.0,
```

```

-- Настройки по умолчанию
defaultTargetTemp = 8000,

targetShield = 25,
startShield = 50.0,
startSat = 50.0,
warmupTemp = 2000,

minFuelPct = 10.0,
initialFlow = 300000,
chargeFlow = 1000000,
maxDrainFlow = 2000000
}

local COLORS = {
  bg = 0x111111, tileBg = 0x222222, tileHeader = 0x333333,
  text = 0xEEEEEE, label = 0x999999,
  good = 0x55FF55, warn = 0xFFAA00, bad = 0xFF5555,
  energy = 0x00AAFF, sat = 0xFFAA00, fuel = 0xFFFF55,
  btn = 0x444444, btnActive = 0x006699
}

-----
-- [ПЕРЕМЕННЫЕ]
-----

local reactors = {}
local buttons = {}
local running = true
local wizard = { active = false, step = 1, coreAddr = nil, inAddr = nil, outAddr = nil,
  candidates = {} }
local needFullRedraw = true

-----
-- [ЛОГИКА]
-----

local function saveConfig()
  local data = {}
  for _, r in ipairs(reactors) do
    table.insert(data, { r=r.proxy.address, i=r.gateIn.address, o=r.gateOut.address,

```

```

auto=r.auto, t=r.targetTemp })
end
local f = io.open(CFG.filename, "w")
f:write(serialization.serialize(data))
f:close()
end

local function safeCall(func, ...)
    local ok, res = pcall(func, ...)
    return ok and res or nil
end

local function getFlow(proxy)
    if not proxy then return 0 end
    return tonumber(safeCall(proxy.getSignalLowFlow)) or 0
end

local function setFlow(proxy, val)
    if proxy then safeCall(proxy.setSignalLowFlow, math.floor(val)) end
end

local function tickReactors()
    for _, r in ipairs(reactors) do
        local info = safeCall(r.proxy.getReactorInfo)
        if info then
            r.isOnline = true
            r.info = info

            if not r.targetTemp then r.targetTemp = CFG.defaultTargetTemp end

            local status = tostring(info.status or "off"):lower()
            local temp = tonumber(info.temperature) or 0
            local drain = tonumber(info.fieldDrainRate) or 0
            local gen = tonumber(info.generationRate) or 0
            local field = tonumber(info.fieldStrength) or 0
            local maxField = tonumber(info.maxFieldStrength) or 1
            local fuel = tonumber(info.fuelConversion) or 0
            local maxFuel = tonumber(info.maxFuelConversion) or 1
            local sat = tonumber(info.energySaturation) or 0

```

```

local maxSat = tonumber(info.maxEnergySaturation) or 1

r.temp = temp
r.gen = gen
r.drain = drain
r.fieldPct = (maxField > 0) and (field/maxField)*100 or 0
r.fuelPct = (maxFuel > 0) and (1 - (fuel/maxFuel)) * 100 or 0
r.satPct = (maxSat > 0) and (sat/maxSat) * 100 or 0
r.statusStr = status:upper()
r.maxFuel = maxFuel

-- === ЛОГИКА ЗАПУСКА ===
if r.isStarting then
    r.auto = false
    setFlow(r.gateOut, 0)

    if maxFuel == 0 or r.fuelPct <= 0 then
        r.modeStr = "НЕТ ТОПЛИВА"
    else
        if status == "cold" or status == "cooling" or status == "offline" or status
== "stopping" then
            r.modeStr = "ИНИЦИАЛИЗАЦИЯ"
            safeCall(r.proxy.chargeReactor)
        elseif status == "charging" then
            if r.fieldPct < CFG.startShield then r.modeStr = "ЗАРЯДКА ПОЛЯ"
            else r.modeStr = "ЗАРЯДКА НАСЫЩ." end
            if r.fieldPct >= CFG.startShield and r.satPct >= CFG.startSat then
                safeCall(r.proxy.activateReactor); r.modeStr = "АКТИВАЦИЯ..."
            end
        elseif status == "charged" then
            r.modeStr = "ГОТОВ К ПУСКУ"
            safeCall(r.proxy.activateReactor)
        elseif status == "warming_up" or status == "warming up" then
            r.modeStr = "ПРОГРЕВ > " .. CFG.warmupTemp
            if temp >= CFG.warmupTemp then r.isStarting = false; r.auto = true end
        elseif status == "running" or status == "online" then
            r.isStarting = false; r.auto = true
        end
    end
end
end

```

```

end

-- === УПРАВЛЕНИЕ ЩИТОМ ===
if status == "charging" or status == "warming_up" or status == "warming up" then
    setFlow(r.gateIn, CFG.chargeFlow); r.shieldCost = CFG.chargeFlow
elseif status ~= "off" and status ~= "cold" then
    if r.fieldPct < 15 then setFlow(r.gateIn, CFG.chargeFlow); r.shieldCost =
CFG.chargeFlow
    else
        local targetIn = math.ceil(drain / (1 - (CFG.targetShield / 100)))
        local safeIn = math.max(10000, targetIn)
        setFlow(r.gateIn, safeIn); r.shieldCost = safeIn
    end
else setFlow(r.gateIn, 0); r.shieldCost = 0 end

-- === УПРАВЛЕНИЕ ВЫХОДОМ ===
if (status == "stopping" or status == "cooling") and not r.isStarting then
    r.auto = false
    r.modeStr = "ОСТЫВАНИЕ"
    if r.satPct > 10 then setFlow(r.gateOut, CFG.maxDrainFlow)
    else setFlow(r.gateOut, CFG.initialFlow) end

elseif r.auto then
    if r.fuelPct <= CFG.minFuelPct and (status == "running" or status == "online")
then
        safeCall(r.proxy.stopReactor); r.auto = false; r.isStarting = false
        r.modeStr = "ТОПЛИВО < 10%"; saveConfig()
    elseif status == "running" or status == "online" then
        local target = r.targetTemp
        local crit = target + 150

        if temp > crit then
            r.modeStr = "!!! ПЕРЕГРЕВ !!!"
            setFlow(r.gateOut, CFG.initialFlow)
        elseif temp > target then
            local overheat = temp - target
            local range = crit - target
            local penalty = (overheat / range) * 0.8
            local multiplier = 1.0 - penalty

```

```

        if multiplier < 0.1 then multiplier = 0.1 end
        r.modeStr = string.format("УДЕРЖАНИЕ %.0f", temp)
        setFlow(r.gateOut, math.max(CFG.initialFlow, gen * multiplier))
    else
        local underheat = target - temp
        local aggressive = 1.0 + (underheat * 0.0005)
        if aggressive > 2.0 then aggressive = 2.0 end
        r.modeStr = string.format("НАГРЕВ -> %.0f", target)
        setFlow(r.gateOut, math.max(CFG.initialFlow, gen * aggressive))
    end
else if not r.isStarting then r.modeStr = "ОЖИДАНИЕ" end end
else
    if not r.isStarting and status ~= "stopping" and status ~= "cooling" then
        if r.fuelPct <= CFG.minFuelPct then r.modeStr = "ТОПЛИВО < MIN"
        elseif status == "cold" or status == "offline" then r.modeStr = "СТОП"
        else r.modeStr = "РУЧНОЙ" end
        if getFlow(r.gateOut) > CFG.initialFlow then setFlow(r.gateOut,
CFG.initialFlow) end
            end
        end
    else r.isOnline = false end
end
end
end

-----
-- [ГРАФИКА]
-----
local function rect(x, y, w, h, col) gpu.setBackground(col); gpu.fill(x, y, w, h, " ") end
local function text(x, y, str, fg, bg) if bg then gpu.setBackground(bg) end;
gpu.setForeground(fg); gpu.set(x, y, str) end
local function center(x, y, w, str, fg, bg) if bg then gpu.setBackground(bg) end;
gpu.fill(x, y, w, 1, " "); local px = x + math.floor((w - unicode.len(str))/2); if fg then
gpu.setForeground(fg) end; gpu.set(px, y, str) end
local function drawVal(x, y, w, str, fg, bg) if bg then gpu.setBackground(bg) end;
gpu.fill(x, y, w, 1, " "); if fg then gpu.setForeground(fg) end; gpu.set(x, y, str) end
local function fmt(num) if num > 1000000000 then return string.format("%.2f G",
num/1000000000) elseif num > 1000000 then return string.format("%.2f M", num/1000000)
elseif num > 1000 then return string.format("%.0f K", num/1000) else return
tostring(math.floor(num)) end end

```

```

local function progressBar(x, y, w, pct, colBar, colBg) rect(x, y, w, 1, colBg); local
barW = math.floor(w * (pct/100)); if barW > 0 then rect(x, y, barW, 1, colBar) end end
local function btn(id, x, y, w, h, str, bg, fg) rect(x, y, w, h, bg); center(x, y +
math.floor(h/2), w, str, fg or COLORS.text, bg); buttons[id] = {x=x, y=y, w=w, h=h} end

local function drawStaticInterface()
  gpu.setBackground(COLORS.bg); term.clear()
  local margin = 1; local areaH = math.floor(H * 0.66); local tileW = math.floor(W / 5) -
margin; local tileH = areaH - 2
  for i = 1, 5 do
    local x = 1 + (i-1) * (tileW + margin); local y = 2
    rect(x, y, tileW, tileH, COLORS.tileBg); rect(x, y, tileW, 3, COLORS.tileHeader)
    center(x, y+1, tileW, "РЕАКТОР #" .. i, COLORS.text, COLORS.tileHeader)
    local r = reactors[i]
    if r then
      local tBg = COLORS.tileBg
      text(x+2, y+9, "ТОПЛИВО:", COLORS.label, tBg); text(x+2, y+12, "ЩИТ ПОЛЯ:",
COLORS.label, tBg)
      text(x+2, y+15, "НАСЫЩЕНИЕ:", COLORS.label, tBg)
      local curY = y+18
      text(x+2, curY, "ВЫРАБОТКА:", COLORS.label, tBg); text(x+2, curY+1, "ЗАТРАТЫ:",
COLORS.label, tBg)
      text(x+2, curY+2, "ИТОГ:", COLORS.label, tBg); text(x+2, curY+4, "ВЫХОД:",
COLORS.label, tBg)
    else center(x, y+10, tileW, "СЛОТ СВОБОДЕН", COLORS.label, COLORS.tileBg) end
  end
  local areaY = math.floor(H * 0.66) + 2; local areaH = H - areaY + 1
  rect(1, areaY, W, areaH, 0x181818); rect(1, areaY, W, 1, 0x444444)
  local col2 = 40; local col3 = 80; local line1 = areaY + 2; local line2 = areaY + 4
  text(col2, line1, "ОБЩАЯ ГЕНЕРАЦИЯ:", COLORS.label, 0x181818); text(col2, line2, "РАСХОД
НА ЩИТЫ:", COLORS.label, 0x181818); text(col3, line1, "ЧИСТАЯ ПРИБЫЛЬ:", COLORS.energy,
0x181818)
  local crY = H - 3; local crX = W - 40
  text(crX, crY, "Разработчик: DesOore", COLORS.label, 0x181818); text(crX, crY+1,
"Проект: McSkill", COLORS.label, 0x181818); text(crX, crY+2, "Сервер: HitechClassic
1.7.10", COLORS.label, 0x181818)
  btn("exit", 3, H-2, 10, 1, "ВЫХОД", COLORS.bad)
end

```

```

local function updateLiveDisplay()
  -- ИСПРАВЛЕНИЕ: Очищаем список кнопок, чтобы удалить фантомные старые кнопки
  buttons = {}
  -- Сразу возвращаем кнопку выхода, так как мы стерли всё
  buttons["exit"] = {x=3, y=H-2, w=10, h=1}

  local margin = 1; local areaH = math.floor(H * 0.66); local tileW = math.floor(W / 5) -
margin; local tileH = areaH - 2
  local totalGen = 0; local totalCost = 0; local totalProfit = 0; local activeCount = 0
  for _, r in ipairs(reactors) do
    if r and r.isOnline and (r.statusStr == "ONLINE" or r.statusStr == "RUNNING") then
      activeCount = activeCount + 1; local p = (r.gen or 0) - (r.shieldCost or 0)
      totalGen = totalGen + (r.gen or 0); totalCost = totalCost + (r.shieldCost or 0);
totalProfit = totalProfit + p
    end
  end
  for i = 1, 5 do
    local x = 1 + (i-1) * (tileW + margin); local y = 2; local r = reactors[i]
    if r then
      if r.isOnline then
        local curProfit = (r.gen or 0) - (r.shieldCost or 0)
        local stCol = (r.statusStr == "ONLINE" or r.statusStr == "RUNNING") and
COLORS.good or COLORS.warn
        if r.statusStr == "OFFLINE" then stCol = COLORS.bad end
        if r.statusStr == "COOLING" or r.statusStr == "STOPPING" then stCol =
COLORS.energy end
        center(x, y+4, tileW, r.statusStr, stCol, COLORS.tileBg)
        local modeCol = (r.auto and COLORS.energy or COLORS.warn)
        if r.modeStr == "!!! АВАРИЯ !!!" then modeCol = COLORS.bad end
        if r.isStarting then modeCol = COLORS.good end
        center(x, y+5, tileW, "[" .. (r.modeStr or "-") .. "]", modeCol, COLORS.tileBg)

        local tCol = r.temp > (r.targetTemp + 50) and COLORS.warn or COLORS.good
        if r.temp > (r.targetTemp + 150) then tCol = COLORS.bad end

        local tempStr = string.format("%.0f C", r.temp)
        local targetStr = string.format("(Цель: %d)", r.targetTemp or 8000)

        center(x, y+7, tileW, tempStr, tCol, COLORS.tileBg)

```

```

center(x, y+8, tileW, targetStr, COLORS.label, COLORS.tileBg)

local cx = x + math.floor(tileW/2)
btn("tdn_"..i, cx - 9, y+7, 3, 1, "-", COLORS.btn, COLORS.text)
btn("tup_"..i, cx + 6, y+7, 3, 1, "+", COLORS.btn, COLORS.text)

local fuelCol = r.fuelPct <= CFG.minFuelPct and COLORS.bad or COLORS.fuel
drawVal(x+tileW-6, y+9, 5, string.format("%d%%", r.fuelPct), fuelCol,
COLORS.tileBg)
progressBar(x+2, y+10, tileW-4, r.fuelPct, fuelCol, 0x444400)
drawVal(x+tileW-6, y+12, 5, string.format("%d%%", r.fieldPct), COLORS.energy,
COLORS.tileBg)
progressBar(x+2, y+13, tileW-4, r.fieldPct, COLORS.energy, 0x002244)
local satCol = r.satPct > 50 and COLORS.warn or COLORS.sat
drawVal(x+tileW-6, y+15, 5, string.format("%d%%", r.satPct), satCol,
COLORS.tileBg)
progressBar(x+2, y+16, tileW-4, r.satPct, satCol, 0x332200)
local curY = y+18
drawVal(x+tileW-11, curY, 9, fmt(r.gen), COLORS.good, COLORS.tileBg)
drawVal(x+tileW-11, curY+1, 9, fmt(r.shieldCost), COLORS.warn, COLORS.tileBg)
drawVal(x+tileW-11, curY+2, 9, fmt(curProfit), COLORS.energy, COLORS.tileBg)
drawVal(x+tileW-11, curY+4, 9, fmt(getFlow(r.gateOut)), COLORS.text,
COLORS.tileBg)
local by = y + tileH - 4
local btnText, btnColor, btnAct
if r.auto or r.isStarting or (r.statusStr == "RUNNING" or r.statusStr == "ONLINE")
then
    btnText = "СТОП"; btnColor = COLORS.bad; btnAct = "stop"; if r.isStarting then
btnText = "ОТМЕНА" end
    else btnText = "ПУСК"; btnColor = COLORS.good; btnAct = "start" end
    btn(btnAct.."_"..i, x+2, by, tileW-4, 3, btnText, btnColor)
    btn("del_"..i, x+tileW-3, y, 3, 1, "X", COLORS.bad, COLORS.text)
else
    center(x, y+10, tileW, "НЕТ СВЯЗИ", COLORS.bad, COLORS.tileBg); center(x, y+11,
tileW, "ПРОВЕРЬ КАБЕЛЬ", COLORS.label, COLORS.tileBg)
    btn("del_"..i, x+2, y+tileH-2, tileW-4, 1, "УДАЛИТЬ", COLORS.bad)
end
else if i == #reactors + 1 then btn("add", x+4, y+12, tileW-8, 3, "ДОБАВИТЬ",
COLORS.good) end end

```

```

end

local areaY = math.floor(H * 0.66) + 2; local col1 = 3; local col2 = 40; local col3 =
80; local line1 = areaY + 2; local line2 = areaY + 4; local bgF = 0x181818
gpu.setBackground(bgF); gpu.fill(col1, line1, 30, 1, " "); gpu.set(col1, line1,
"АКТИВНО: " .. activeCount .. " / " .. CFG.maxReactors)
drawVal(col2+18, line1, 15, fmt(totalGen).. " RF/t", COLORS.good, bgF); drawVal(col2+18,
line2, 15, fmt(totalCost).. " RF/t", COLORS.warn, bgF); drawVal(col3, line2, 15,
fmt(totalProfit).. " RF/t", COLORS.energy, bgF)
end

local function drawWizard()
gpu.setBackground(COLORS.bg); term.clear(); local cx = math.floor(W/2); local cy =
math.floor(H/2)
center(1, cy-7, W, "=== БЫСТРАЯ НАСТРОЙКА ===", COLORS.energy, COLORS.bg)
if wizard.step == 1 then
center(1, cy-4, W, "Поиск свободного ядра...", COLORS.text, COLORS.bg)
if wizard.coreAddr then
center(1, cy-3, W, "Ядро найдено: " .. wizard.coreAddr:sub(1,8).. "...",
COLORS.good, COLORS.bg)
center(1, cy-1, W, "ПОИСК ГЕЙТОВ...", COLORS.energy, COLORS.bg)
center(1, cy+1, W, "1. Подойдите к гейту ЩИТА (ВХОД)", COLORS.label, COLORS.bg)
center(1, cy+2, W, "2. Установите на нем поток: 100 RF/t", COLORS.warn, COLORS.bg)
center(1, cy+3, W, "Второй гейт определится АВТОМАТИЧЕСКИ.", COLORS.good,
COLORS.bg)
center(1, cy+5, W, "Жду сигнала 100...", COLORS.text, COLORS.bg)
else center(1, cy-2, W, "СВОБОДНОЕ ЯДРО НЕ НАЙДЕНО!", COLORS.bad, COLORS.bg) end
end
btn("cancel_wiz", cx-8, cy+7, 16, 1, "ОТМЕНА", COLORS.bad)
end

local function tickWizard()
if not wizard.active then return end
if not wizard.coreAddr then
for addr, type in component.list("draconic_reactor") do
local used = false; for _, r in ipairs(reactors) do if r.proxy.address == addr then
used = true; break end end
if not used then wizard.coreAddr = addr; drawWizard(); break end
end
end
end

```

```

if not wizard.coreAddr then return end
if #wizard.candidates == 0 then
  for addr, type in component.list("flux_gate") do
    local used = false; for _, r in ipairs(reactors) do if r.gateIn.address == addr or
r.gateOut.address == addr then used = true; break end end
    if not used then table.insert(wizard.candidates, component.proxy(addr)) end
  end
end
for _, gate in ipairs(wizard.candidates) do
  local flow = tonumber(gate.getSignalLowFlow()) or 0
  if flow == 100 then
    wizard.inAddr = gate.address
    for _, other in ipairs(wizard.candidates) do if other.address ~= wizard.inAddr then
wizard.outAddr = other.address; break end end
    if wizard.outAddr then
      computer.beep(1000, 0.2)
      table.insert(reactors, { proxy = component.proxy(wizard.coreAddr), gateIn =
component.proxy(wizard.inAddr), gateOut = component.proxy(wizard.outAddr), auto = false,
isStarting = false, isOnline = true, temp=0, gen=0, shieldCost=0, fuelPct=0, fieldPct=0,
satPct=0, targetTemp=CFG.defaultTargetTemp })
      saveConfig(); wizard.active = false; needFullRedraw = true; tickReactors(); break
    end
  end
end
end
end

local function handleTouch(x, y)
  for id, b in pairs(buttons) do
    if x >= b.x and x < b.x + b.w and y >= b.y and y < b.y + b.h then
      if id == "exit" then running = false
      elseif id == "add" then wizard.active = true; wizard.step = 1; wizard.candidates =
{}; wizard.coreAddr = nil; drawWizard()
      elseif id == "cancel_wiz" then wizard.active = false; needFullRedraw = true
      else
        local act, idx = id:match("(%a+)_(%d+)"); idx = tonumber(idx); local r =
reactors[idx]
        if r then
          if act == "del" then table.remove(reactors, idx); saveConfig(); needFullRedraw =
true

```

```

        elseif act == "start" then r.isStarting = true; r.auto = false;
computer.beep(1000, 0.1); updateLiveDisplay()
        elseif act == "stop" then safeCall(r.proxy.stopReactor); r.auto = false;
r.isStarting = false; r.modeStr = "СТОП"; saveConfig(); updateLiveDisplay()
        elseif act == "tup" then r.targetTemp = (r.targetTemp or 8000) + 50; if
r.targetTemp > 20000 then r.targetTemp = 20000 end; saveConfig();
computer.beep(1200,0.05); updateLiveDisplay()
        elseif act == "tdn" then r.targetTemp = (r.targetTemp or 8000) - 50; if
r.targetTemp < 2000 then r.targetTemp = 2000 end; saveConfig(); computer.beep(1000,0.05);
updateLiveDisplay()
            end
        end
    end
    computer.beep(800, 0.05); break
end
end
end

gpu.setResolution(maxW, maxH); gpu.setBackground(COLORS.bg); term.clear()
if filesystem.exists(CFG.filename) then
    local f = io.open(CFG.filename, "r"); local data =
serialization.unserialize(f:read("*a")); f:close()
    if data then for _, d in ipairs(data) do table.insert(reactors, {
proxy=component.proxy(d.r), gateIn=component.proxy(d.i), gateOut=component.proxy(d.o),
auto=d.auto, isOnline=true, isStarting=false, targetTemp=(d.t or CFG.defaultTargetTemp) })
end end
end
needFullRedraw = true
local lastTick = 0

-- === ГЛАВНЫЙ ЦИКЛ ===
while running do
    if wizard.active then tickWizard(); local _, _, x, y = event.pull(0.2, "touch"); if x
then handleTouch(x, y) end
    else
        local e, _, x, y = event.pull(0.05, "touch"); if e then handleTouch(x, y) end
        if computer.uptime() - lastTick > CFG.updateInterval then tickReactors(); if
needFullRedraw then drawStaticInterface(); needFullRedraw = false end;
updateLiveDisplay(); lastTick = computer.uptime() end

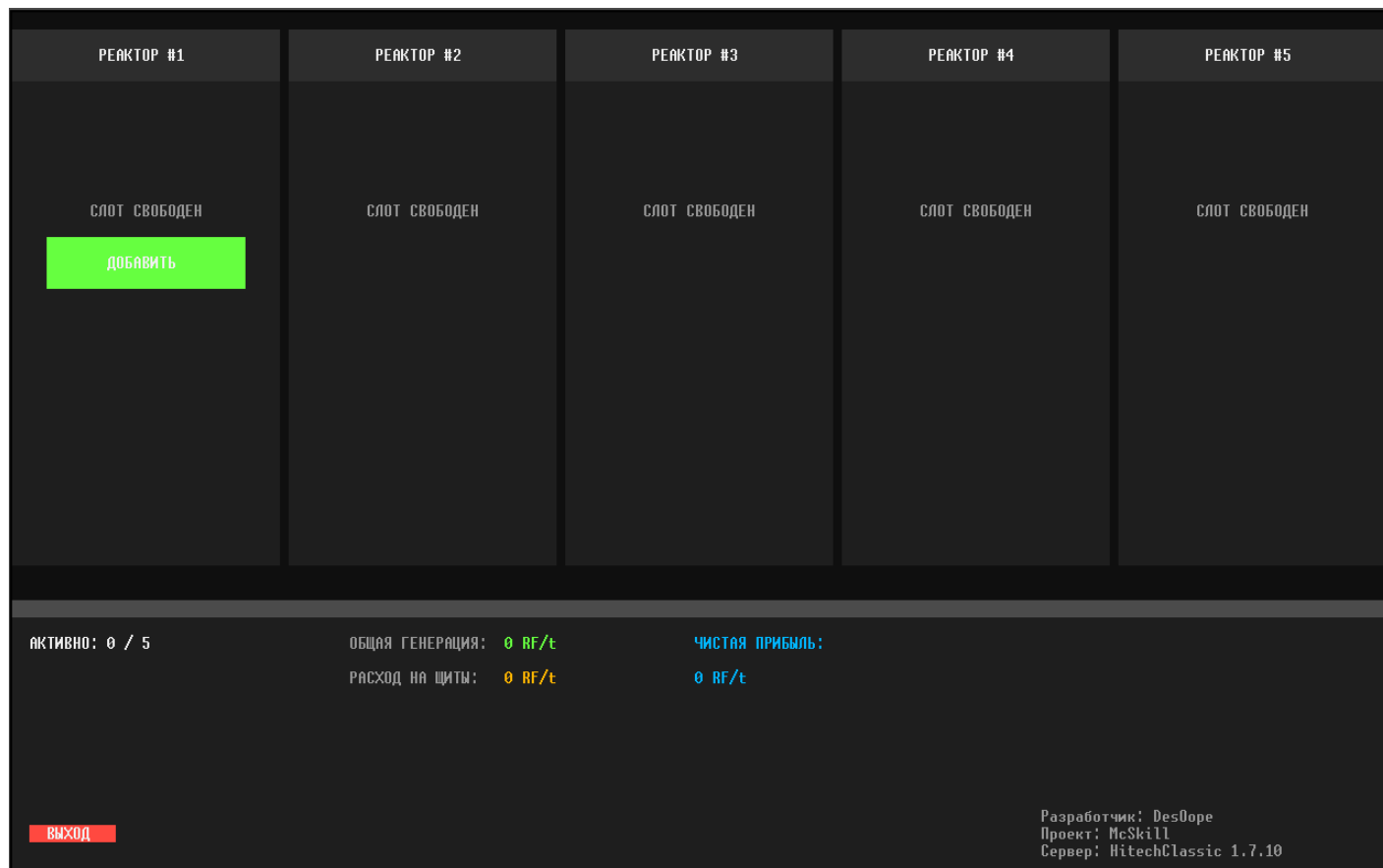
```

```
end  
end  
gpu.setResolution(maxW, maxH); gpu.setBackground(0x000000); term.clear()
```

Перезагрузите компьютер командой:

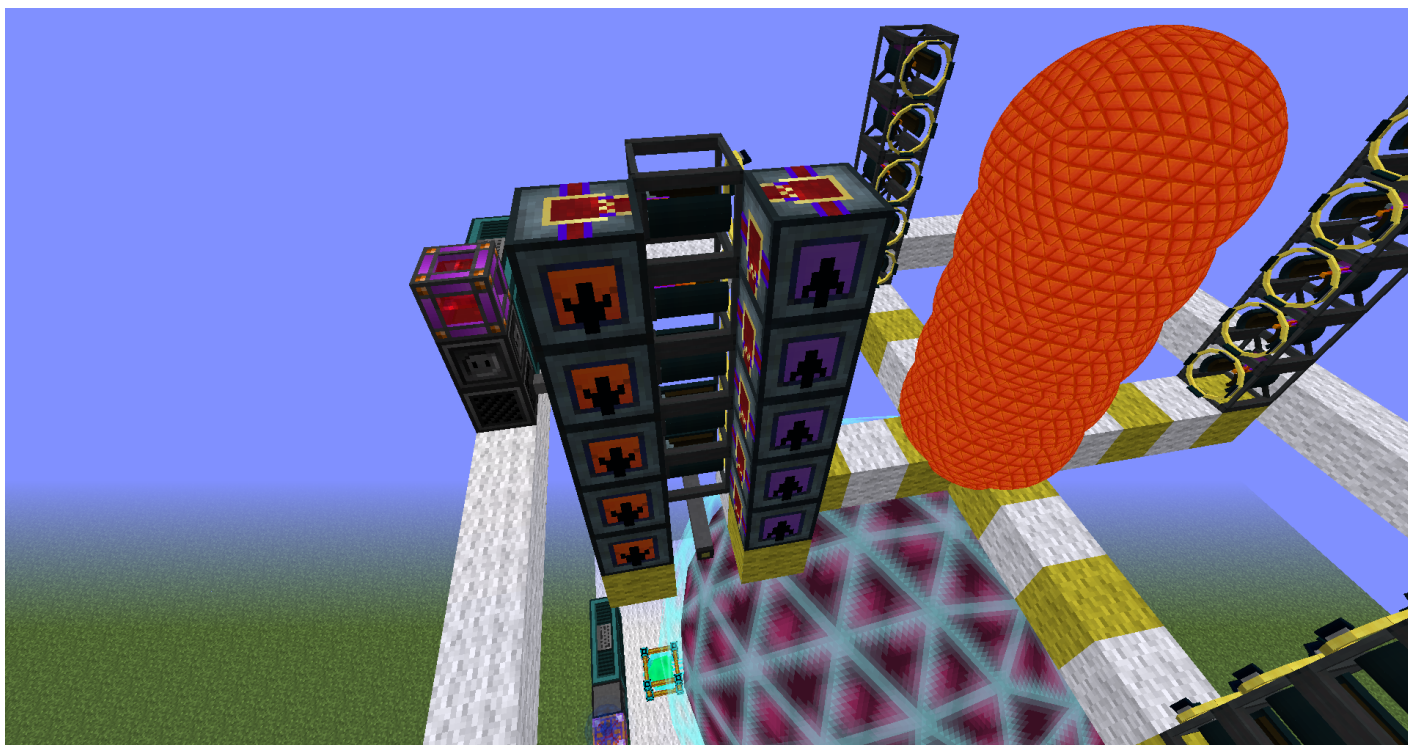
```
reboot
```

После перезагрузки мы видим такое окно программы



4. Подключение реакторов к сети

Гейты: Установите флаксовые гейты возле реакторов.



Трубы: Проложите криостабилизированные трубы.

Пилоны: Установите пилоны на трубы. Не забудьте активировать их стеклом снизу. Левый пилон переключите в режим приема энергии (**ПКМ** по синему ядру).

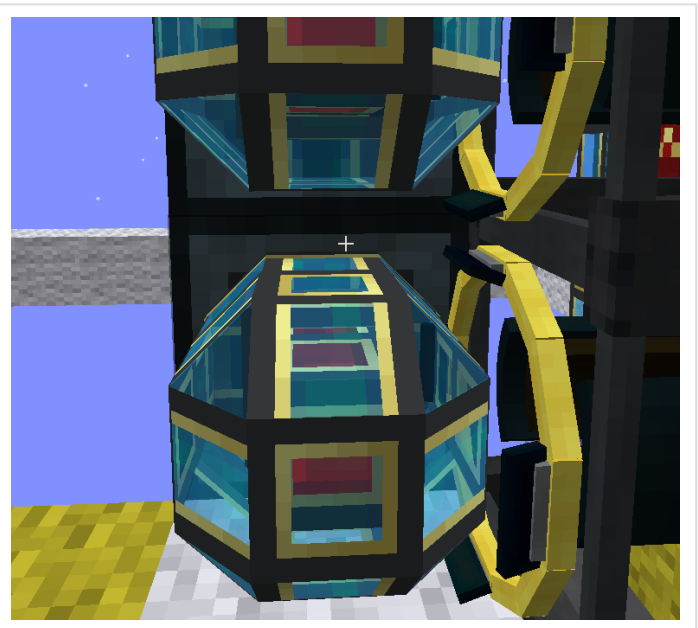
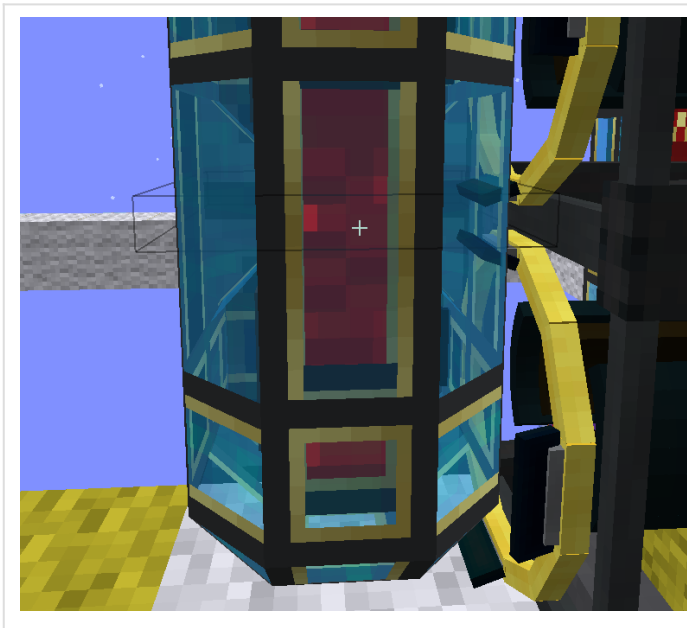


Трубы: Проложите криостабилизированные трубы с другой стороны.



Важно: С этой стороны трубы нужно разъединить гаечным ключом (например, Yeta из Ender IO), чтобы потоки не смешивались.

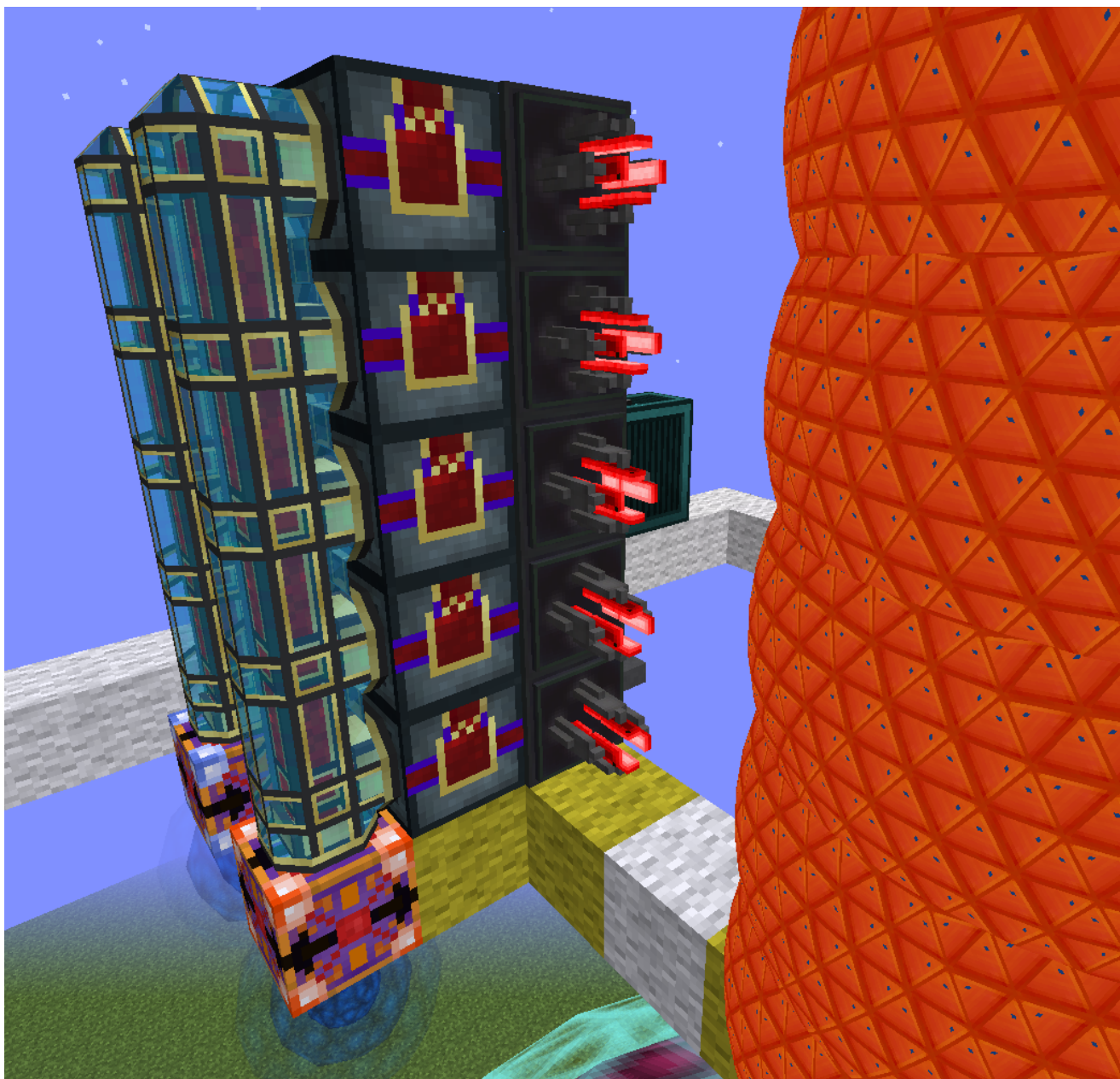
Наведите между труб и нажмите ПКМ держа в руках гаечный ключ.



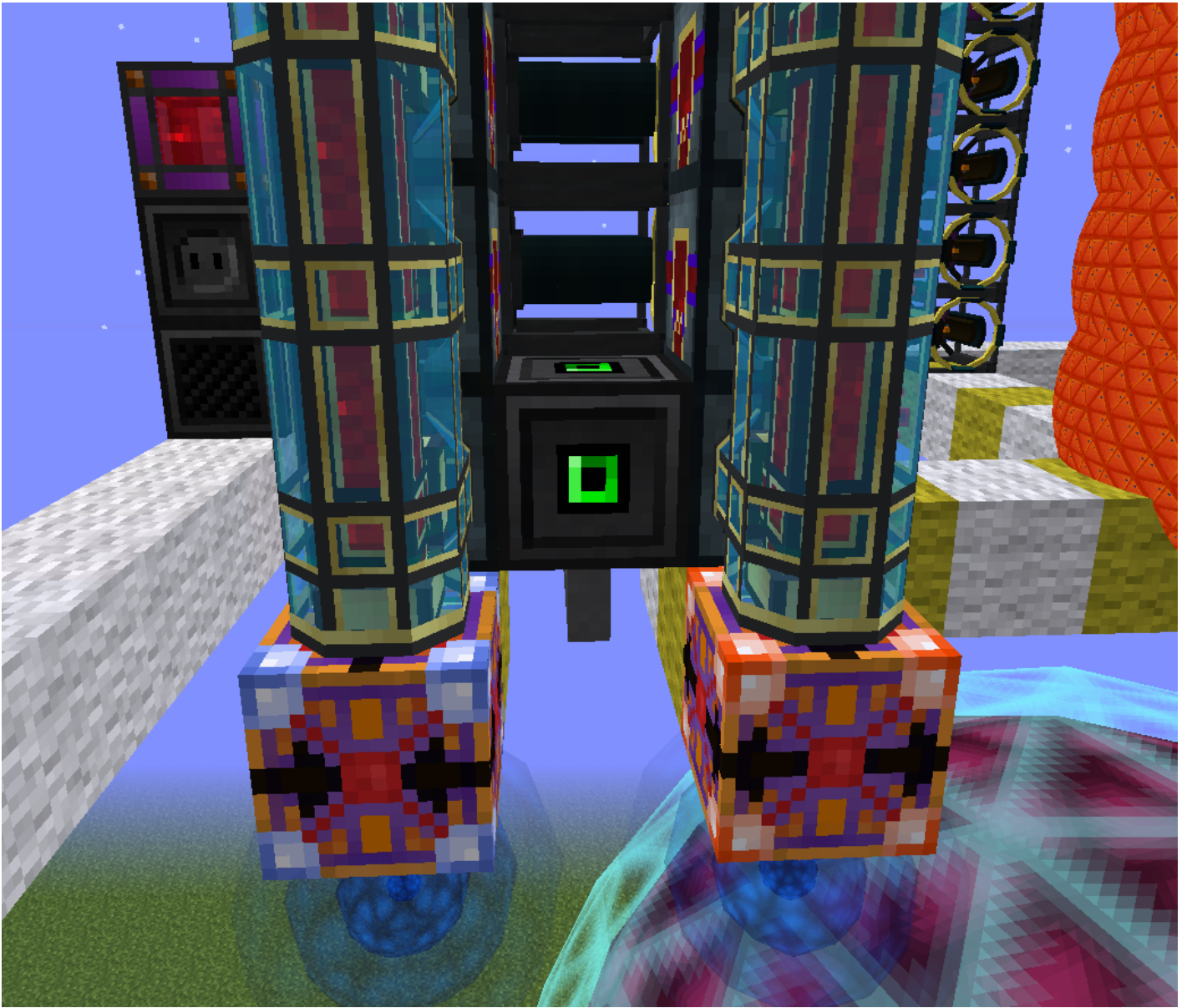
Повторите так со всеми трубами с этой стороны



Инжекторы: Поставьте инжекторы на трубы, направив их в сторону реакторов.



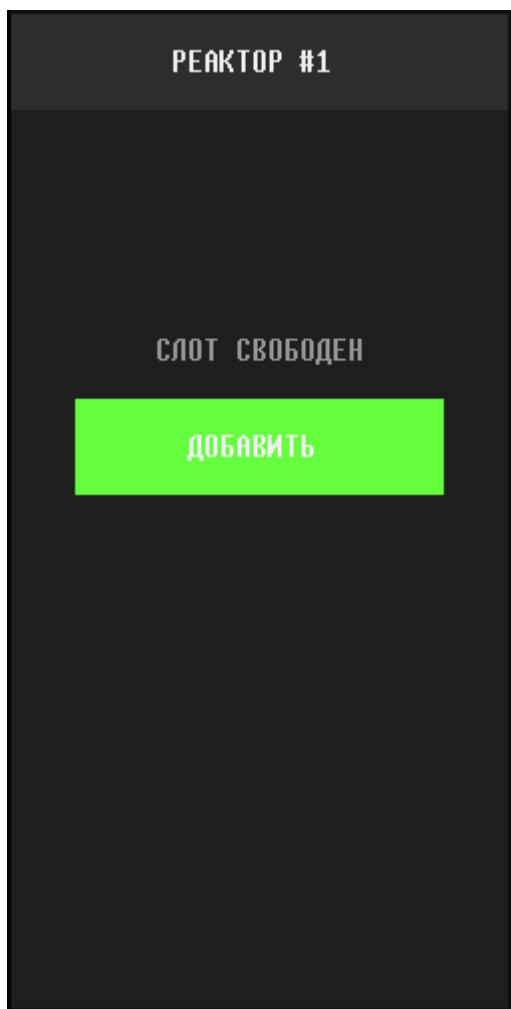
Связь: Установите по очереди (смотреть ниже) по одному **Адаптеру** между гейтами каждого реактора и соедините их кабелями с сервером.



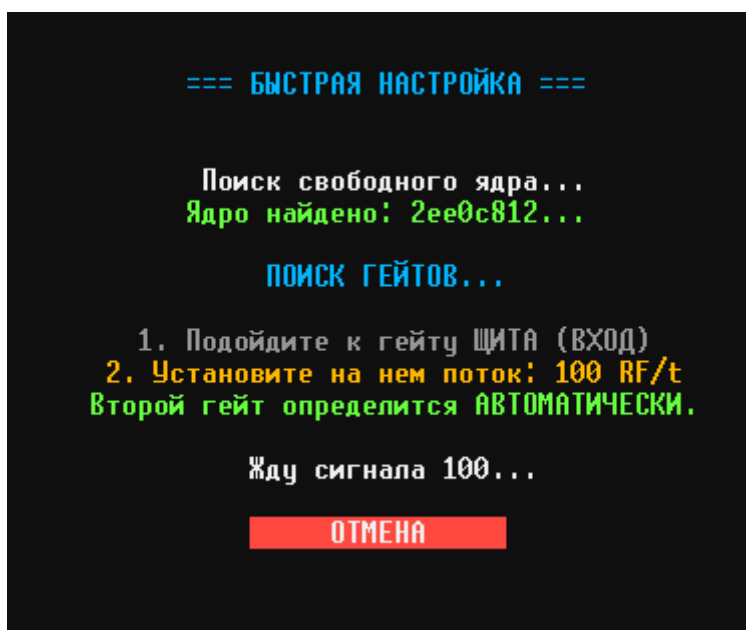
5. Запуск и эксплуатация

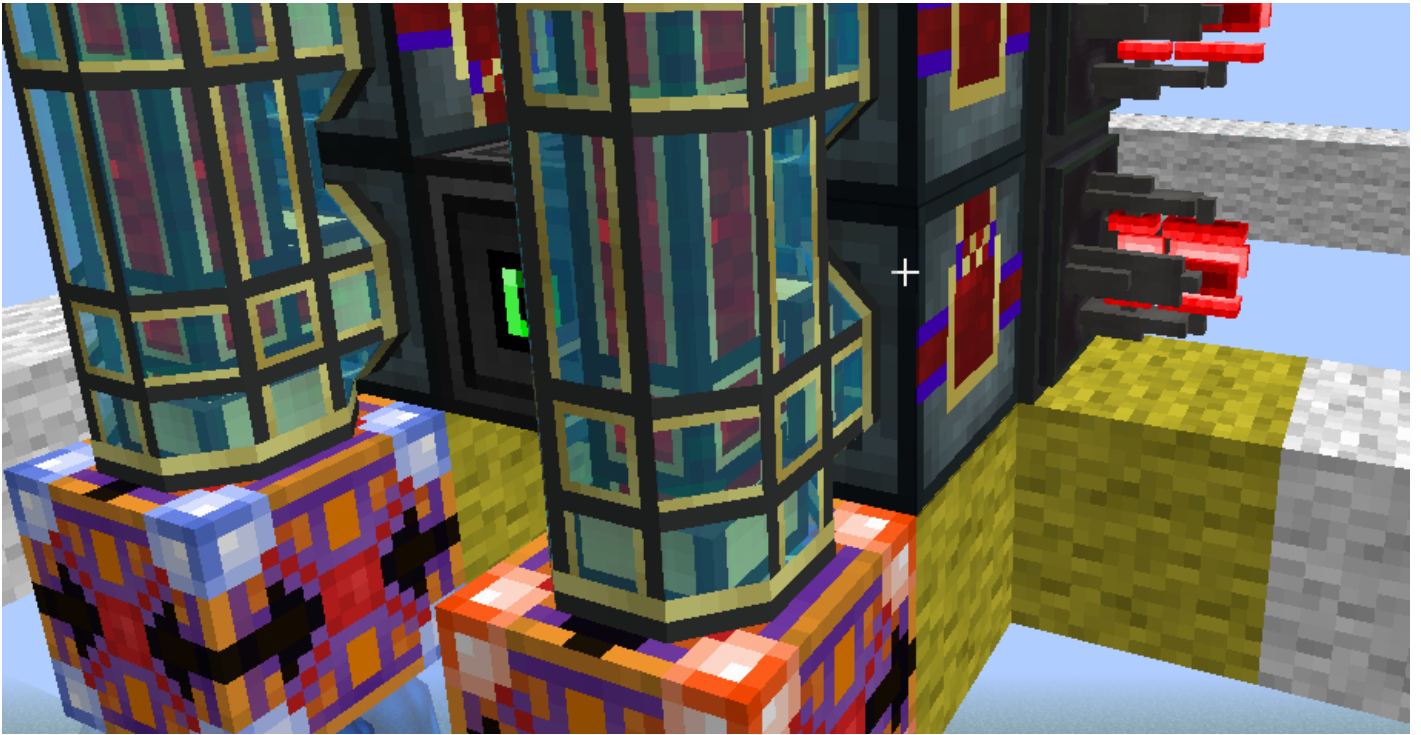
Вернитесь к монитору сервера и следуйте алгоритму:

- Нажмите кнопку "**Добавить**" в интерфейсе программы.



На экране появится запрос. Подойдите к адаптеру нужного реактора и в его правом гейте вручную выставьте поток **100 RF/t**.





Реактор автоматически отобразится в списке.

РЕАКТОР #1 ✕

OFFLINE
[СТОП]

- **20 С** +
(Цель: 8000)

ТОПЛИВО: **100%**

ЩИТ ПОЛЯ: **0%**

НАСЫЩЕНИЕ: **0%**

ВЫРАБОТКА: **0**

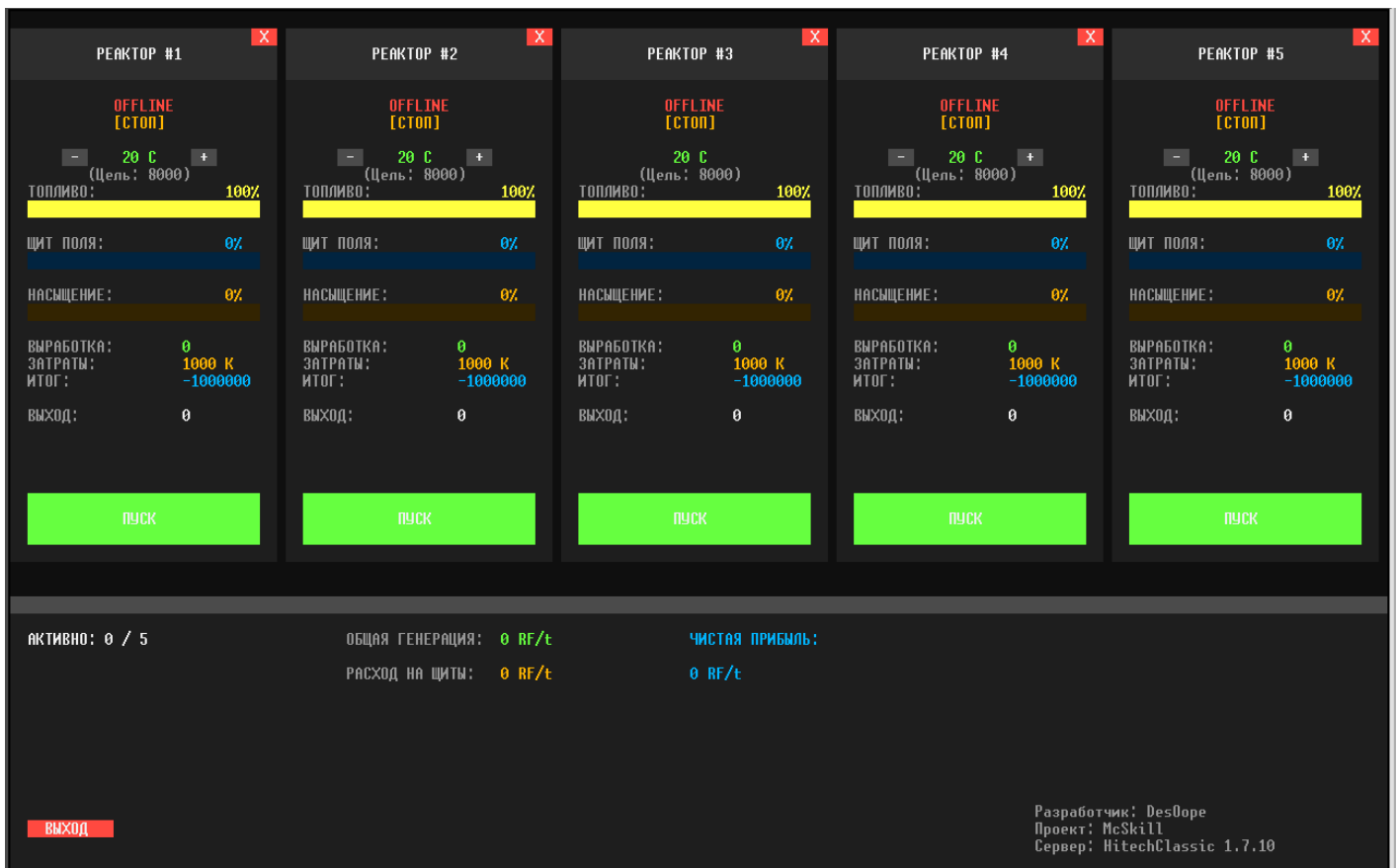
ЗАТРАТЫ: **1000 К**

ИТОГ: **-1000000**

ВЫХОД: **0**

ПУСК

Повторите эту процедуру с оставшимися реакторами



После того как мы добавили все наши реакторы - жмем пуск и наблюдаем

Что произойдет дальше?

Программа автоматически зарядит щиты, поднимет насыщение, нагреет реакторы, запустит их и они будут работать на стабильных 8000 градусах. В таком режиме система может работать стабильно около 13 реальных дней.

Совет для профи: В программе можно менять рабочую температуру. Если выставить 9000 градусов на 5-10 часов, а потом вернуть на 8000 - топливо выгорит быстрее, что позволит реактору выйти на высокие мощности в 1.5 - 2 млн RF/t. Но делайте это только на свой страх и риск.

РЕАКТОР #1	РЕАКТОР #2	РЕАКТОР #3	РЕАКТОР #4	РЕАКТОР #5
ONLINE [НАГРЕВ -> 8000]	ONLINE [НАГРЕВ -> 8000]	ONLINE [УДЕРЖАНИЕ 8000]	ONLINE [НАГРЕВ -> 8000]	ONLINE [НАГРЕВ -> 8000]
- 8000 С + (Цель: 8000)	- 8000 С + (Цель: 8000)	- 8000 С + (Цель: 8000)	- 8000 С + (Цель: 8000)	- 8000 С + (Цель: 8000)
ТОПЛИВО: 99%	ТОПЛИВО: 99%	ТОПЛИВО: 99%	ТОПЛИВО: 99%	ТОПЛИВО: 99%
ЩИТ ПОЛЯ: 25%	ЩИТ ПОЛЯ: 25%	ЩИТ ПОЛЯ: 25%	ЩИТ ПОЛЯ: 25%	ЩИТ ПОЛЯ: 25%
НАСЫЩЕНИЕ: 11%	НАСЫЩЕНИЕ: 11%	НАСЫЩЕНИЕ: 11%	НАСЫЩЕНИЕ: 11%	НАСЫЩЕНИЕ: 11%
ВЫРАБОТКА: 534 К	ВЫРАБОТКА: 534 К	ВЫРАБОТКА: 534 К	ВЫРАБОТКА: 534 К	ВЫРАБОТКА: 534 К
ЗАТРАТЫ: 162 К	ЗАТРАТЫ: 162 К	ЗАТРАТЫ: 162 К	ЗАТРАТЫ: 162 К	ЗАТРАТЫ: 162 К
ИТОГ: 372 К	ИТОГ: 372 К	ИТОГ: 372 К	ИТОГ: 372 К	ИТОГ: 372 К
ВЫХОД: 534 К	ВЫХОД: 534 К	ВЫХОД: 534 К	ВЫХОД: 534 К	ВЫХОД: 534 К
СТОП	СТОП	СТОП	СТОП	СТОП

АКТИВНО: 5 / 5	ОБЩАЯ ГЕНЕРАЦИЯ: 2.67 М РФ/т	ЧИСТАЯ ПРИБЫЛЬ: 1.86 М РФ/т
	РАСХОД НА ЩИТЫ: 812 К РФ/т	

ВЫХОД

 Разработчик: DesOope
 Проект: McSkill
 Сервер: HitechClassic 1.7.10

Revision #6

Created 2026-03-17 22:24:16 UTC by DesOope

Updated 2026-03-18 13:46:37 UTC by DesOope