Всем доброго времени суток. Переведу на русский язык свой проект, связанный с контроллером паяльника на жалах hakko t12. Оригинальная версия опубликована [тут](http://out.arduino.ru/?redirect=http%3A%2F%2Fwww.instructables.com%2Fid%2FThe-Hakko-T12%2F&baseU=http%3A%2F%2Farduino.ru%2Fforum%2Fproekty%2Fkontroller-payalnika-dl-zhal-hakko-t12). На мой взгляд, жала t12 - лучший выбор для построения качественного паяльника, хотя они стоят дороже, чем hakko 907.

Жала hakko t12 - замечательная вещь. Когда берешь в руку одно из них, возникает ощущение качественно сделанного предмета, в который вложили частичку души. Высокого качества встроеный сенсор позволяет точно измерить температуру жала, а стало быть ее поддерживать в узких пределах. Нагрев до рабочей температуры занимает всего 15-30 секунд (в моем случае, я пожертвовал врменем нагрева для большей стабильности поддержания температуры). Вы можете купить готовую паяльную станцию для этих жал, однако, построить собственную, на мой взгляд, гораздо веселее. Основная особенность представленного здесь контроллера - использование алгоритма [ПИД](https://ru.wikipedia.org/wiki/%D0%9F%D0%98%D0%94-%D1%80%D0%B5%D0%B3%D1%83%D0%BB%D1%8F%D1%82%D0%BE%D1%80), который обеспечивает поддержание требуемой температуры паяльника. Рассказ об алгоритме выходит за рамки данного проекта, все, кто хочет изучить его, рекомендую вооружитсья гуглом и терпением. Если быть кратким, должен сказать, что настройка алгоритма под конкретный контроллер является весьма кропотливым занятием и требует много времени. Зато позволяет очень точно удерждивать нужную температуру и позволяет быстро увеличивать необходимую мощность в случае, если вы прислоните паяльник к батарее отопления.

Основными особенностями контроллеря являются:

* Используетмый алгоритм ПИД позволяет точно удерживать температуру паяльника вблизи заданного значения с высокой точностью. Паяльник нагревается за считанные секунды. Контроллер позволяет увеличивать мощность в случае быстрого охлаждения паяльника.
* Контроллер имеет встроенный режим калибровки, который упрощает настройку контроллера паяльника на отображение корректных значений температуры и настройку коэффициэнта усиления сенсора.
* Контроллер поддерживает два режима работы: поддержание заданной температуры и поддержание фиксированной мощности.
* Контроллер не издает никаких звуков при работе. Для этого используется библиотека fastPWMdac которая заменяет стандартные функции управления скважностью (PWM) имеющейся в арсенале контроллера Arduino. Библиотека увеличивает частоту сигнала PWM до 31250 Гц вместо стандартных 490 Гц, которые выдает стандартная функция analogWrite().
* Контроллер использует "ускорение" стандартного Rotary Encoder. Если Вы вращаете рукоятку быстро, значения увеличиваются с шагом 5 единиц, что позволяет быстрее выбрать нужную температуру.
* Заданная температура сохраняется в памяти контроллера (EEPROM) всякий раз, когда вы ее изменяете. Чтобы увеличить "жизнь" контроллера Arduino, данные сохраняются блоками фиксированной длины (кратной объему памяти EEPROM) в разных частях памяти.
* Контроллер также сохраняет в памяти EEPROM параметры калибровк контролера и единицы измерения темпаратуры.
* Контроллер может автоматически выключить паяльник, если он не использовался в течение некоторого времени. Время бездействия также натраивается.

**Компоненты**

Для создания контроллера нам потребуется:

* Блок питания на  24 вольта, выдающий 4-6 ампер, можно импульсный
* Блок питания на 5 вольт для питания контроллера (выдающий около 500 мА) ОБЯЗАТЕЛЬНО изолированный, с трансформатором. Использовать конвертер 24->5 вольт не получится.
* Контроллер Arduino(например, arduino nano v3). Я использовал atmega328p-pu с осциллятором 16 МГц и двумя конденсаторами по 22 pF, он более компактен. Не забудьте записать загрузчик в контроллер.
* двухстрочный LCD-дисплей по 8 символов в строке, лучше с желтой подсветкой, он более контрасет и углы обзора у него лучше, чем у синего.
* Полевой транзистор irf9540. Можно использовать аналог, выдерживающий напряжене до 50 вольт и ток до 5 ампер. Обращаю внимание, что это p-channel транзистор, использовать n-channel не получится.
* npn биполярные транзисторы ksp10 (2n2222 - лучше) - 2 шт.
* pnp биполярный транзистор s9012h (2n2907 - лучше).
* стабилитрон 3.3 вольта.
* стабилитрон 18 или 15 вольт.
* высокочастотный диод типа FR304.
* Индуктивность порядка 200 милли Генри.
* Высокоточный операционный усислитель ad822 с полевыми транзисторами на входном каскаде. Можно использовать аналог. lm358n будет выдавать случайный значения вместо температуры.
* Мощный транзистор 0.22 Ома 5 ватт, для измерения потребляемого тока.
* резисторы 100 Ом - 2 штуки.
* резисторы 470 Ом-  2 штуки.
* резистор 220k.
* резисторы 1k - 3 штуки.
* резистор 5.1k.
* резистор 10k - 4 штуки.
* Подстроечный резистор 10k (много оборотов).
* Подстроечный резистор 500k (много оборотов).
* Керамический конденсатор 1 микрофарад.
* Керамические конденсаторы 100 нано фарад - 5 шт.
* Rotary Encoder с кнопкой
* "пищалка"
* разъем GX12-5

**Рукоятка**

Первым делом нужно выбрать рукоятку для будущего паяльника. Прочитал [хорошую статью](http://out.arduino.ru/?redirect=http%3A%2F%2Fwww.elenblog.ru%2F2-pajalnaja-stancija-dlja-hakko-t12-poluchenie-ishodnyh-dannyh%2F&baseU=http%3A%2F%2Farduino.ru%2Fforum%2Fproekty%2Fkontroller-payalnika-dl-zhal-hakko-t12), где сравнивались две рукоятки и остановил свой выбор на FX-9501, она показалось мне более качественной и я не доверяю защелкам. Забегая вперед должен сказать, что ручка в целом неплохая, провод достаточно мягкий и хорошо гнется. Есть лишь одна проблема: перекручивание провода при входе в рукоятку (от там припаивается к разъему) в результате которого со временем три тонких провода отламываются и пропадает контакт с паяльником. К счастью, ручка легко разбирается и провода можно припаять назад. Рекомендую во избежание перекручивания провода обмотать часть рукоятки с проводом изолентой, чтобы провод не перекручивался в рукоятке.

Все рукоятки имеют одну и ту же проблему - разъем. Найти ответную часть разъема, который подходил бы к купленной ручке, я не смог. Поэтому, считаю проще заменить разъем целиком на GX12-5 или подобный, который можно купить в Интернете. У моего паяльника в шнуре было три провода: красный - заземление, зеленый - +24 вольта для нагрева и измерения температуры, черный - минус. Заземления в квартире нет, поэтому я обошелся подключением лишь двух проводов.

**Подключение датчика температуры**

Вторая проблема заключается в том, что паяльник имеет всего два контакта, которые нужно использовать и для нагрева, и для снятия показаний температуры. В статье сказано, что внутри паяльника термопара соединяется с нагревательным элементом последовательно и вся конструкция двумя проводами подключается к контроллеру. Термопара при нагревании паяльника выдает напряжение порядка 5 мили вольт. Понятно, для того чтобы измерить столь небольшое напряжение, нам нужен операционный усилитель. Однако, для нагрева паяльника мы должны подать на него +24 вольта, что может не понравится нашему усилителю. К счастью для себя я нашел готовую схему, где эта проблема была решена при помощи стабилитрона. [Вот эта схема и статья к ней](http://out.arduino.ru/?redirect=http%3A%2F%2Fradio.aliot.com.ua%2F%3Fp%3D945&baseU=http%3A%2F%2Farduino.ru%2Fforum%2Fproekty%2Fkontroller-payalnika-dl-zhal-hakko-t12).

Контроллер использует обе цепи операционного усилителя: он измеряет температуру паяльника через первый усилитель (пины 1, 2, 3) и протекающий через паяльник ток при помощи второго усилителя (пины 5, 6, 7). Коэффициэнт усиления первого усилителя нужно настроить при помощи потенциометра на 500 кОм. Этот резистор нужна настроить следующим образом. При температуре паяльника, близкой к максимальной (450 градусов) на выходе операционного усилителя напряжение должно быть около 5 вольт («считываемое» значение на входе A0 должно быть 850-900). Чтобы облегчить подстройку резистора в контроллере применяется режим настройки, который доступен через меню изменения параметров контроллера. Процедура калибровки преследует две цели: облегчить настройку потенциометра и получить формулу пересчета значений сенсора из «внутренних» единиц в значение температуры, выраженное в градусах цельсия или фаренгейтах. Контроллер Arduino считывает значение напряжение на пине A0 в интервале 0-1023 и использует это значение в работе, однако нам неудобно пользоваться безразмерными величинами для измерения температуры и требуется перевести эти значения в привычные нам градусы цельсия.

Как было сказано выше, второй усилитель, собранный на пинах 5, 6 и 7, используется для измерения тока, протекающего через паяльник. Величина протекающего тока нам не важна, нужно лишь убедиться, что подача напряжения на разъем приводит к протеканию тока через паяльник. Если ток отсутствует, это может означать потерю связи с паяльником и контроллер прекратит подачу напряжения. При активной работе с паяльником, контроллер проверяет наличие тока через мощный резистор номиналом 0.22 Ома, измеряя падение напряжение на нем и отключает подачу напряжения сразу же, если ток меньше заданного значения. В режиме ожидания (когда паяльник выключен) контроллер кратковременно подает напряжение на паяльник и проверяет наличие тока через него. Если ток не обнаружен, выдается сообщение «no iron», означающее что пропал контакт с паяльником.

**Сделаем его тихим**

Чтобы подавать мощность на паяльник используется P-канальный мощный полевой транзистор. Контролер Arduino позволяет менять скважность сигнала, регулируя подаваемую на паяльник мощность. Первая версия контроллера паяльника была сделана с использованием стандартной библиотечной функции Arduino analogWrite(), с помощью которой менялась скважность выходного сигнала, а следовательно и мощность, подаваемая на паяльник. Однако, такое решение было достаточно назойливым, поскольку несущая частота сигнала PWM составляла 490 Гц. Во время работы паяльника контроллер издавал звуки различных частот и громкости, которые сильно досаждали. К счастью, на просторах Интернета мне попалась библиотека [Fast PWM DAC](http://out.arduino.ru/?redirect=http%3A%2F%2Fwww.avdweb.nl%2Farduino%2Fhardware-interfacing%2Ffast-pwm-dac.html&baseU=http%3A%2F%2Farduino.ru%2Fforum%2Fproekty%2Fkontroller-payalnika-dl-zhal-hakko-t12), которая позволяет генерировать PWM сигнал на пинах 9 и 10 с высокой частотой, около 35 кГц. Такую высокую частоту человеческое ухо не в состоянии уловить и паяльник кажется нам бесшумным при работе.

Тем не менее, чтобы заставить полевой транзистор переключаться с высокой скоростью, необходимо использовать специальную схему подключения, которая гарантирует быстрый заряд и разряд емкости, находящейся на затворе транзистора.  Можно найти детальное описание процессов, происходящих на полевом транзисторе при его работе. Использование двух транзисторов в схеме управления позволяет обеспечить быстрое переключение полевого транзистора между двумя состояниями «открыт» и «закрыт». В этих состояниях его сопротивление либо близко к нули, либо бесконечно велико, что приводит к отсутствию нагрева транзистора при его работе. Устанавливать радиатор на него не требуется. Чтобы закрыть транзистор, на затвор подается +24 вольта, точнее разница напряжений между затвором и истоком равна нулю. Чтобы открыть транзистор, на затвор необходимо подать «отрицательное» напряжение относительно истока. К сожалению, подключить затвор к «земле» не получится, поскольку в этом варианте разница напряжений между истоком и затвором превысит допустимые 20 вольт. Чтобы предотвратить это, в схеме используется второй стабилитрон на 18-15 вольт, который ограничивает отрицательное напряжение на затворе при открытии транзистора.

**Индуктивность**

В схеме присутствует индуктивность около 200 мили генри, которая снижает нагрузку на конденсаторы блока питания в выходном каскаде. Как сказал мой приятель, чтобы не подогревать конденсаторы. Диод fr304 снимает «обратное» напряжение с индуктивности и паяльника в момент выключения питания. Поскольку переключения происходят с высокой частотой диод должен быть высокочастотным.

**Собираем контроллер**

Поскольку я использовал отдельный блок питания, схема контроллера распалась на отдельные части. Вы тоже можете поступить подобным образом. Скажем, разместить контроллер Ардуино, операционный усилитель и подстрочные резисторы на небольшой плате, которую закрепить за экраном, а «силовую» часть с транзистором и обвязкой для него разместить на отдельной плате. Впрочем, компоновка сильно зависит от используемого корпуса и начинать нужно именно с него.

Для питания контроллера Ардуино и операционного усилителя необходимо подать стабилизированное напряжение +5 вольт. Изначально я планировал использовать преобразователь DC-DC чтобы получить из 24 вольт 5. Однако, практика показала, что в этом случае не получается измерить показания на термопаре, потому что появляется паразитное смещение напряжение между «землей» и входным каскадом операционного усилителя. Одним словом, питание 5 вольт должно быть изолировано от +24 вольт. Автор оригинальной схемы использовал тороидальный трансформатор, выдающий оба напряжения. Я решил пойти по тому же пути и построил отдельный трансформаторный блок питания для контроллера. Идея использовать тороидальный трансформатор мне понравилась, однако импульсный блок питания легче и компактнее. Я собрал отдельный блок питания на трансформаторе от старинного блока питания и установил за ним стабилизатор напряжения 7805 на +5 вольт. Вы можете купить готовый источник питания, но обязательно изолированный.

**Управление контроллером**

Как было сказано выше, контроллер имеет несколько режимов работы:

* режим ожидания, когда паяльник выключен.
* основной режим работы (поддержание заданной температуры).
* режим поддержания заданной мощности.
* режим настройки параметров.
* режим калибровки контроллера.

При включении контроллера он переходит в режим ожидания. В этом режиме паяльник выключен. На верхней строке отображается заданная температура и сообщение о статусе паяльника – «OFF» - выключен. На нижней строке отображается текущая температура паяльника или служебные сообщения. Если паяльник не подключен к контроллеру, на нижней строке появится надпись «No iron». Когда паяльник достигает комнатной температуры после использования на нижней строке появляется надпись «cold», что означает, что паяльник может быть безопасно положен в ящик стола или жало паяльника можно заменить, взявшись за него рукой.

Заданную температуру можно изменить вращая ручку Rotary Encoder. При плавном вращении значения изменяются на 1 градус, при быстром – на 5. Это позволяет быстро выставить заданную температуру. Температура паяльника отображается корректно только после калибровки контроллера.

Чтобы включить паяльник, слегка нажмите на рукоятку Rotary Encoder. Контроллер перейдет в основной режим работы, в котором он будет стараться поддерживать температура паяльника около заданной, регулируя подаваемую мощность. В этом режиме на верхней строке экрана отображается заданная температура и сообщение о текущем режиме работы паяльника:

* «ON» - паяльник включен и нагревается
* «rdy» - паяльник готов к работе
* «wrk» - паяльник активно используется

На нижней строке отображается текущее значение температуры паяльника и подаваемая на него мощность в % от максимального значения. Мощность подается на паяльник плавно, в зависимости от разности заданной и текущей температур, в соответствии с алгоритмом ПИД. Это обеспечивает более точное поддержание заданной температуры. Вращая рукоятку Rotary Encoder можно изменить требуемую температуру паяльника. Контроллер снова отобразит в верхней строке сообщение «ON» до тех пор, пока новая заданная температура не будет достигнута. В режиме поддержания температуры контроллер постоянно проверяет ток, протекающий через паяльник. Если ток не протекает, контроллер выдает сообщение «Failed» и отключает питание паяльника. В этом случае необходимо слегка нажать на ручку Rotary Encoder чтобы перейти в режим ожидания.

Режим поддержания заданной мощности можно активировать долгим нажатием на рукоятку Rotary Encoder пока контроллер находится в основном режиме. В режиме поддержания мощности вы можете настраивать подаваемую на паяльник мощность, вращая рукоятку. На экране отображается подаваемая мощность во внутренних единицах 0-255 и текущая температура паяльника в градусах. Легкое нажатие на рукоятку Rotary Encoder включает или выключает подачу мощности на паяльник. Чтобы вернуться в основной режим, долго нажмите на рукоятку Rotary Encoder.

**Режим настройки параметров**

Чтобы переключиться в режим настройки параметров, нажмите и удерживайте ручку Rotary Encoder примерно 1.5 -2 секунды. Меню настройки имеет пять пунктов:

* таймаут автоматического выключения паяльника при бездействии [3-30] минут или выключен.
* выбор единиц измерения температуры (градусы Цельсия или Фаренгейты).
* калибровка контроллера.
* сохранение параметров.
* выход из меню без сохранения параметров.

Чтобы выбрать нужный пункт меню, вращайте ручку. Чтобы изменить параметр в выбранном пункте меню, нажмите на ручку Rotary Encoder. По окончании установки параметров выберите пункт меню «save» или долго удерживайте нажатой кнопку Rotary Encoder. Измененные значения параметров сохраняются в EEPROM контроллера и загружаются при следующем запуске.

**Режим калибровки контроллера**

*Требуется внешний термометр чтобы провести калибровку контроллера.*

Контроллер требует калибровки, которая включает настройку потенциометра, управляющего операционным усилителем. Как было сказано выше, при настройке потенциометра нужно обеспечить как можно более полный диапазон работы аналогового входа A0. Когда температура паяльника достигает 450 градусов Цельсия, значения на входе A0 должны составлять 850-900 единиц.

В режиме калибровке вы можете настроить потенциометр и откалибровать процедуру преобразования внутренних единиц температуры в градусы цельсия. В программе считается, что рабочий диапазон паяльника составляет 180-450 градусов цельсия. В случае необходимости можно изменить эти умолчания в программе контроллера.

Меню калибровки имеет следующие пункты:

* верхняя температура.
* нижняя температура.
* загрузить параметры по умолчанию.
* вернуться в режим ожидания без сохранения изменений.

Перед началом калибровки имеет смысл записать параметры по умолчанию в EEPROM контроллера, выбрав соответствующий пункт меню.

Калибровка контроллера производится следующим образом. Сначала выберите пункт «upper», чтобы установить верхнюю температуру паяльника. Паяльник начнет нагреваться. На экране контроллера будет отображаться температура во внутренних единицах, с входного пина A0 (0-1023). Вращая ручку Rotary Encoder вы можете изменять подаваемую на паяльник мощность. Отстройте температуру паяльника, равную 450 градусов по цельсию (используйте внешний термометр). Отстройте подаваемую мощность таким образом, чтобы температура оставалась постоянной, как можно ближе к 450 градусам.

Теперь вращайте потенциометр таким образом, чтобы значения температуры на входном пине A0 составили 850-900 единиц. Убедитесь, что контроллер имеет запас значений свыше указанных значений. Для этого поверните потенциометр чуть еще и проверьте, что значения считанные с входного пина составляют 930-950 единиц. Верните потенциометр в положение, где значение близко к 900 единицам. Эта проверка гарантирует, что контроллер сможет зафиксировать температуры, выше максимальной и начать охлаждать паяльник, если нужно. Во время настройки потенциометра следует удерживать температуру вблизи 450 градусов. Поэтому, чем точнее вы определите подаваемую на паяльник мощность вначале процедуры, тем точнее вы откалибруете контроллер.

Нажмите на ручку Rotary Encoder чтобы вернуться в меню калибровки. Теперь вы успешно откалибровали потенциометр и контроллер сохранил значение датчика температуры (безразмерные единицы) для верхней температуры 450 градусов по Цельсию.

Можно немного подождать пока паяльник остынет и затем выбрать пункт «Lower» для калибровки нижнего значения температуры. Выровняйте изменяя подаваемую мощность на паяльник его температуру возле значения 180 градусов по Цельсию. Здесь также нужно пользоваться внешним термометром. Чем стабильнее вы зафиксируете температуру и чем ближе она окажется к 180 градусам, тем точнее вы выполните калибровку паяльника. Когда вы подберете мощность, достаточную для поддержания температуры паяльника равной 180 градусам, нажмите на ручку Rotary Encoder. Контроллер запомнит считанное значение сенсора для 180 градусов.

Теперь в нашем распоряжении оказались сохранены оба значения внутреннего сенсора в безразмерный единицах: для 450 и для 180 градусов по Цельсию. Контроллер использует функцию map() чтобы используя эти граничные значения переводить температуру из внутренних единиц в градусы Цельсия.

Вы можете повторно настроить верхнюю и нижнюю температуры при необходимости, выбрав соответствующие пункты меню. Всякий раз контроллер будет запоминать в ОБЫЧНОЙ памяти эти значения. Если оба значения определены, вы можете сохранить калибровку контроллера, записав эти значения в EEPROM нажав и удерживая кнопку Rotary Encoder 1.5-2 секунды.

**Автоматическое отключение паяльника**

Контроллер поддерживает атематическое отключение питание паяльника в случае, если он не используется в течение некоторого времени. Таймаут отключения может быт задан через меню настроек. Также в этом меню, автоматическое отключение может быть выключено.

Поскольку паяльник не имеет датчика положения или датчика удара или что-то подобное, определить период неиспользования паяльника можно только путем анализа подаваемой на него мощности. Алгоритм ПИД, который используется для поддержания температуры паяльника имеет три параметра. Эти параметры зависят от характеристик паяльника и блока питания. Для поддержания температуры паяльника алгоритм может использовать параметры в широком диапазоне, однако, для уменьшения математической дисперсии температуры и мощности в режиме «простоя» требуется довольно точная отладка этих параметров. Для своего блока питания мне удалось подобрать параметры таким образом, чтобы привести обе дисперсии к минимуму. Этим пользуется алгоритм автоматического отключения. В режиме «простоя» мощность постепенно снижается до минимального значения и дисперсии температуры и мощности тоже падают. Когда паяльник используется (что-то нагревается с его помощью) подаваемая мощность растет и обе дисперсии также возрастают в первый момент, поскольку алгоритм ПИД пытается увеличить мощность для поддержания заданной температуры. Возможно, этот способ не очень надежный критерий определения использования паяльника, тем не менее, контроллер успешно используется в течении полугода и показал стабильные результаты.

Я бы не был столь оптимистичен... Не знаю, как оригинальные, но про китайские Т12 отзывы ещё те -  
*"Минусы: Калибровать температуру желательно после установки каждого жала. Разброс параметров немаленький, от 10 до 50 градусов.  
...  
Калибровка температуры не учитывает нелинейность датчика. Откалибровав датчик при одной температуре - получаем отклонения при более высокой (более низкой) температуре порядка 15-20 градусов."*

Думается, на нелинейность можно забить, она не десятки градусов составляет, да и калибровка по двум точкам лучше, чем по одной.

ну и такие измерения -  
[](http://arduino.ru/sites/default/files/u10235/bilderman_2017.03.06_22-55-14.png)