Однажды я решил порадовать себя и купил часы citizen. Кроме всего прочего, в этих часах заявлена возможность синхронизировать часы по радиоканалу. В момент заказа я не сильно представлял, что это такое, думал, что оно как-то само всё сделает как надо. Но оказалось, что синхронизация по радио в России практически не работает. Я решил исправить ситуацию. Первое, что я нашёл была статья посвященная усилению сигнала. Предлагалось намотать большую катушку с конденсатором, положить на окно, выходящее на запад и ждать. [1]. Также в этой статье упоминался протокол dcf77, по которому часы синхронизируют время, и я понял, что он не очень сложный. Хотя на этот момент в электронике я был полный ноль, но мне представлялось, что с цифровой частью я справлюсь, а с остальным я как-то разберусь. И я решил стартовать проект по разработке устройства синхронизации часов по протоколу dcf77.

Здесь я опишу то, как я работал над этим проектом, с какими проблемами столкнулся, будучи новичком в электронике, интересные решения, которые мне попались при работе над проектом, и попробую оправдать их. Этот текст, может быть интересен программистам, которые решили заняться электроникой. Поэтому со стороны программирования многие вещи будут опускаться.

# Концепция устройства

Первое с чего я начал это сбор информации о протоколе, возможных готовых устройствах, и продумал требования к устройству. Начал я продумывания основной концепции.

1. Синхронизация должна выполняться в пределах комнаты
2. Сам синхронизатор может иметь погрешность 2с
3. Синхронизатор должен быть и часами.
4. Должен обладать системой самодиагностики.
5. Должен питаться от батареи. (Я не уверен, что смогу сделать устройство, которое может работать в сети безопасно для окружающих ☺ )
6. От батареи устройство должно работать примерно месяц.
7. Все должно быть красиво. Корпус. Экран. Кнопки. И т.д.

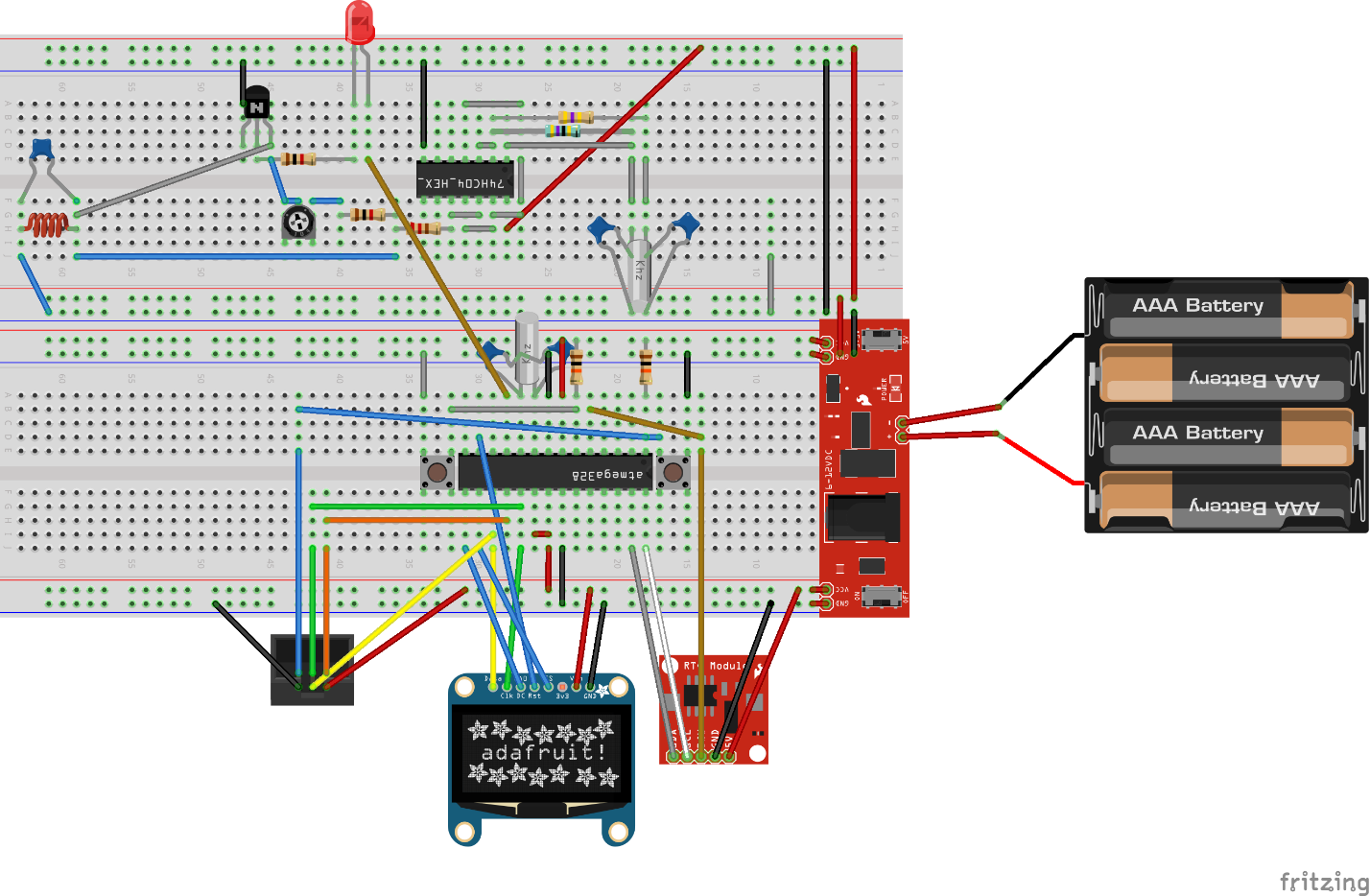
# Первые шаги

Далее я стал собирать материал о протоколе и подобных устройствах. Где-то на немецких ресурсах были какие-то готовые устройства [2]. Нашел описание протокола на википедии. [3]

С этого все и началось.

Что такое dcf77? Это протокол, в котором дата и время передается нулями и единицами амплитудно-модулированным сигналом. В начале каждой секунды амплитуда сокращается до четверти на 0.1 или 0.2 с. Если длинна сокращенного сигнала 0.1 с то 0 если 0.2 с то 1. Что достаточно хорошо видно если моргать светодиодом. Подробнее здесь [3], [15].

Первое что я сделал реализовал протокол dcf77 на большом компе на c++. Это я сделал быстро. Понятно, что я пробовал уже делать мигалки на ардуино, поэтому сделать мигалку протоколом dcf77 на светодиоде не составило труда. Фоток или схем самых первых версий у меня, к сожалению, не сохранилось. Но схема была примерно такая, как на рисунке. Только без генератора и на Arduino Uno.



По пути к такой схеме столкнулся с некоторыми трудностями.

Первое с чем я столкнулся это то, что в Arduino ide в ino файле нельзя реализовывать классы. Пришлось всё переписывать на C. Не догадался сразу что можно было просто подключить его из отдельного файла. Эта проблема разрешилась быстро.

Из-за того, что я начал с того, что визуализировал сигнал светодиодом, то резистор последовательно светодиоду поставил в районе 200 ом (на рисунке уже не так). Для того, чтобы он светил ярче. Это все привело к тому, что транзистор постоянно был приоткрыт и схема нормально не работала. Я методом тыка догадался поставить два диода последовательно. Транзистор вроде стал закрываться. Но это как-то смотрелось диковато, даже для меня, и я понял, что нужно изучать теорию. Азы знать очень важно. По крайней мере, как транзистор работает в ключевом режиме. И здесь я столкнулся со следующей проблемой: мало деталей и мало инструментов. Например, чтобы сделать антенну, нужен провод и феррит. Из Китая их ждать долго. К счастью hex invertor был в Arduino started. Зато кварца 77500 я в России вообще не нашел, и даже в Китае, на ali, он был всего в одно месте, партией 50 штук. К этому времени я уже понял, что даже если я соберу генератор я не смогу даже понять, работает он или нет. И я понял, что нужны инструменты: нужен частотомер и/или осциллограф. Это дало мне передышку в работе над устройством. И я начал изучать электронику. Прослушал ряд YouTube роликов по электронике, например, это [4]. Статьи по теме [5]. Статьи по антеннам. Очень волновало, что антенны для этой частоты должны быть около 700 метров длинной. Но все же решил попробовать сделать магнитную антенну в приемниках [2].

Для антенны нужен контур из магнитной катушки и конденсатора в резонансе. Резонанс как-то надо искать. Это привело к проекту transistor tester. Хотел его между делом тоже реализовать. Тк там был измеритель индуктивности, а несколько Arduino mini (плз хейтерам Ардуино сразу не реагировать. Я потом немного исправился ☺ ) у меня было. Но понял, что его я сделать пока сам не готов, заказал DYI, посмотрел на его схему и нашел, его источник [6].

Так же узнал о некоторых полезных программах для телефона, это EveryCircuit и ElectroDroid. ElectroDroid я в этот момент использовал для расчета резонанса антенны. EveryCircuit позволяет моделировать небольшие схемки на телефоне. Решил документировать и схему. Где-то в интернете увидел красивую картинку на макетной плате. Понял, что такую делает Fritzing, кроме того, он может и принципиальную схему, и плату проектировать. Начал пользоваться fritzing.

Тут уже подошло время как стали приходить первые посылки из Китая. Кое-что пришлось уже срочно дозаказывать в «чипе и дипе». Здесь же появилась первая идея как синхронизировать время. Синхронизация было решено сделать на wifi модуле esp8266.

Случайно наткнулся на подобный проект [7]. Решил взять из него основные идеи. Радио часть полностью позаимствовал из него. Без переделки использовать ее нельзя было, но

1. Он рассчитан на 12 вольт.

2. Основан на микроконтроллере pic.

# Первые успехи

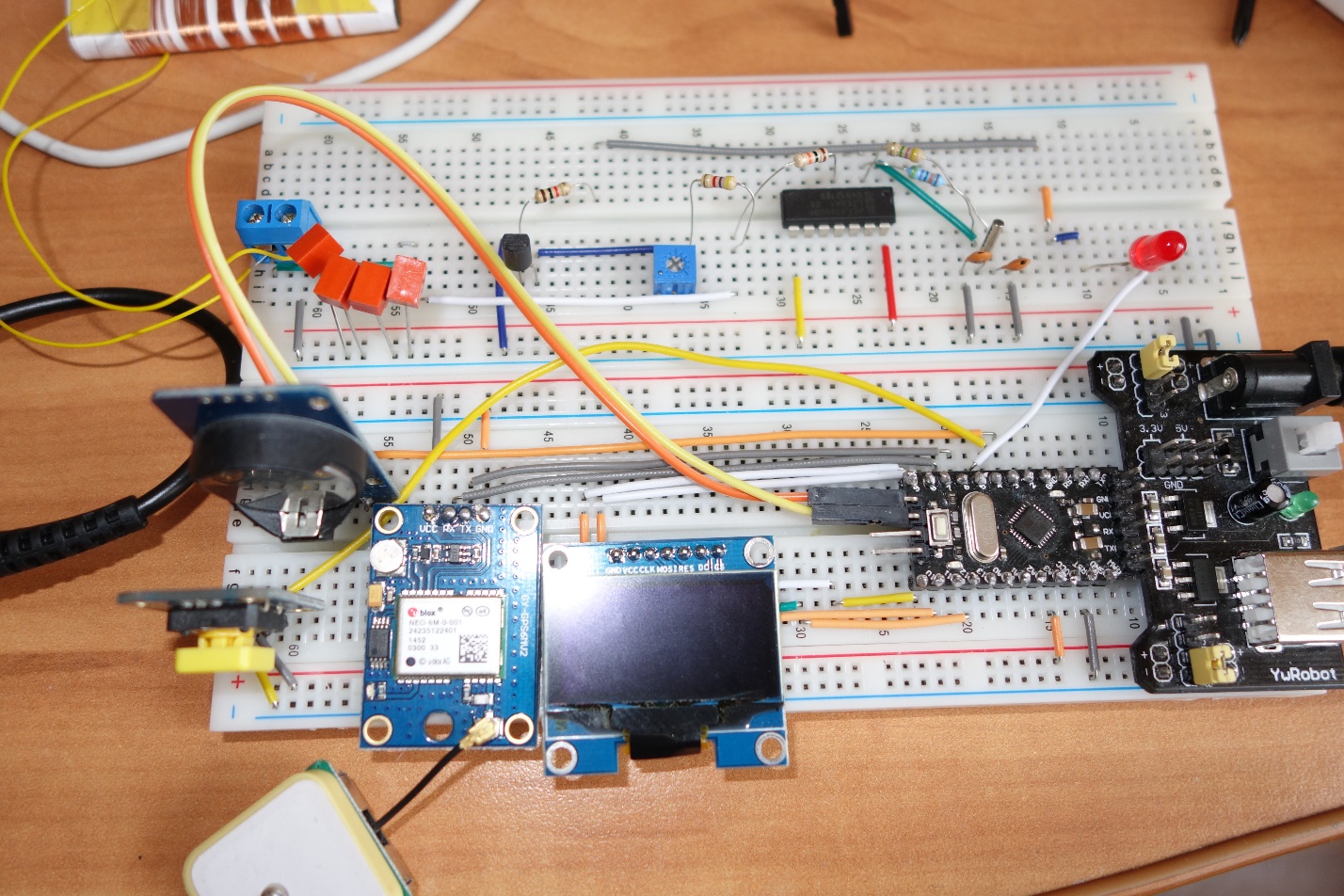
По мере того как прибыли детали я стал улучшать устройство. После того как пришел кварц и транзистор тестер появилась возможность наконец-то собрать первую версию, способную передавать время. Намотал катушку на ферритовом стержне – это антенна. Посмотрел ее индуктивность. Подобрал конденсаторы по 100 pf. Всё вроде собрал. Должно работать. А не работает. И, понятное дело, диагностировать проблему не получается. Осциллограф еще не приехал. Даже мультимер самый простой. По нему мало чего можно понять. Хорошо, что у моего тестя были приборы и осциллограф и частотомер. Поехал к нему со своим девайсом. Посмотрели частотомером вроде частота есть. Посмотрели осциллографом тоже вроде ничего. Синус не чистый но все же похож ☺. Поехал домой разбираться дальше. Проблема в том, что как разбираться непонятно. По часам можно понять только то увидели они сигнал или нет. Понять же в чем ошибка невозможно. Но все же, понятно, что проблема не в радио части, а в модуляторе. Пробовал варьировать параметры. Не уходя далеко от описания с Википедии. В конце концов решил удлинить импульсы на в частотном модуляторе. И в этот момент часы наконец то зацепили сигнал. И, после нескольких попыток, ДНЕМ удалось выполнить первую синхронизацию. Тогда решено было заняться синхронизацией времени на самом устройстве.

В процессе поиска проблем также обнаружил [8], что digitalwrite работает неэффективно. Переделал все так, чтобы его не использовать. Потом я встречал это и в других источниках. Но в первый раз увидел здесь. Это было первое отступление от Arduino.

Мне очень нравятся oled экраны, поэтому было решено было использовать экран ssd1306. Но библиотека от adafruit ужасна. Она очень медленная. Использует много оперативной памяти. Нашел альтернативу u8glib. Очень мне понравилась. [11]

Первое решение, которое я пробовал было основано на модуле esp8266 и подразумевало синхронизацию времени через Интернет. Но мне не понравилось, что достаточно сложно выполнять соединение через этот модуль. Да и в тот момент он был не очень стабильным. Поэтому пришло в голову попробовать синхронизацию времени через GPS. Модуль для ardino такой был. Прошивка для синхронизации через GPS была проще, но были проблемы, с которыми я столкнулся в будущем.

Наконец удалось собрать полнофункциональное устройство на Макетной плате.

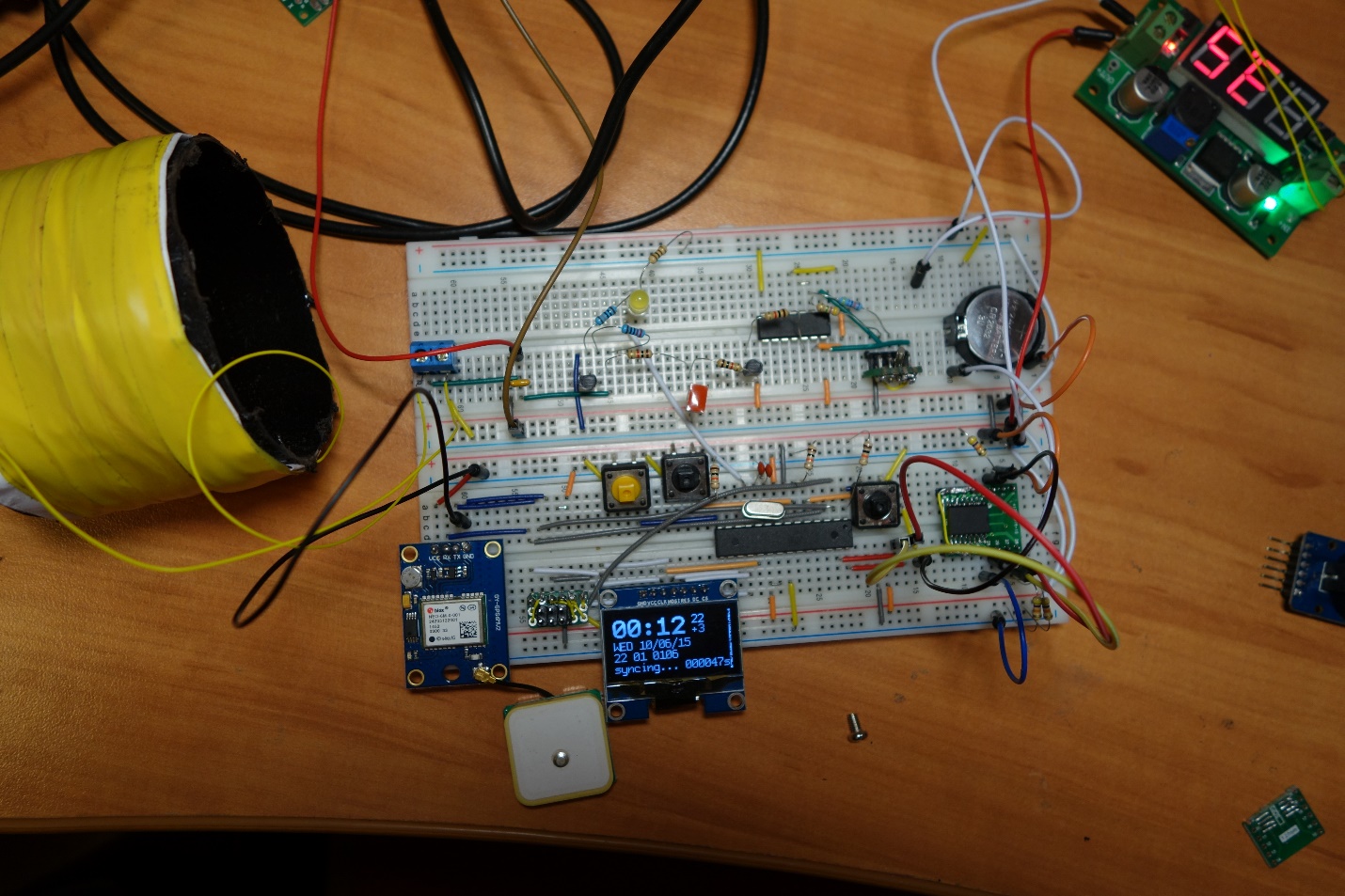


# Отказ от Arduinio

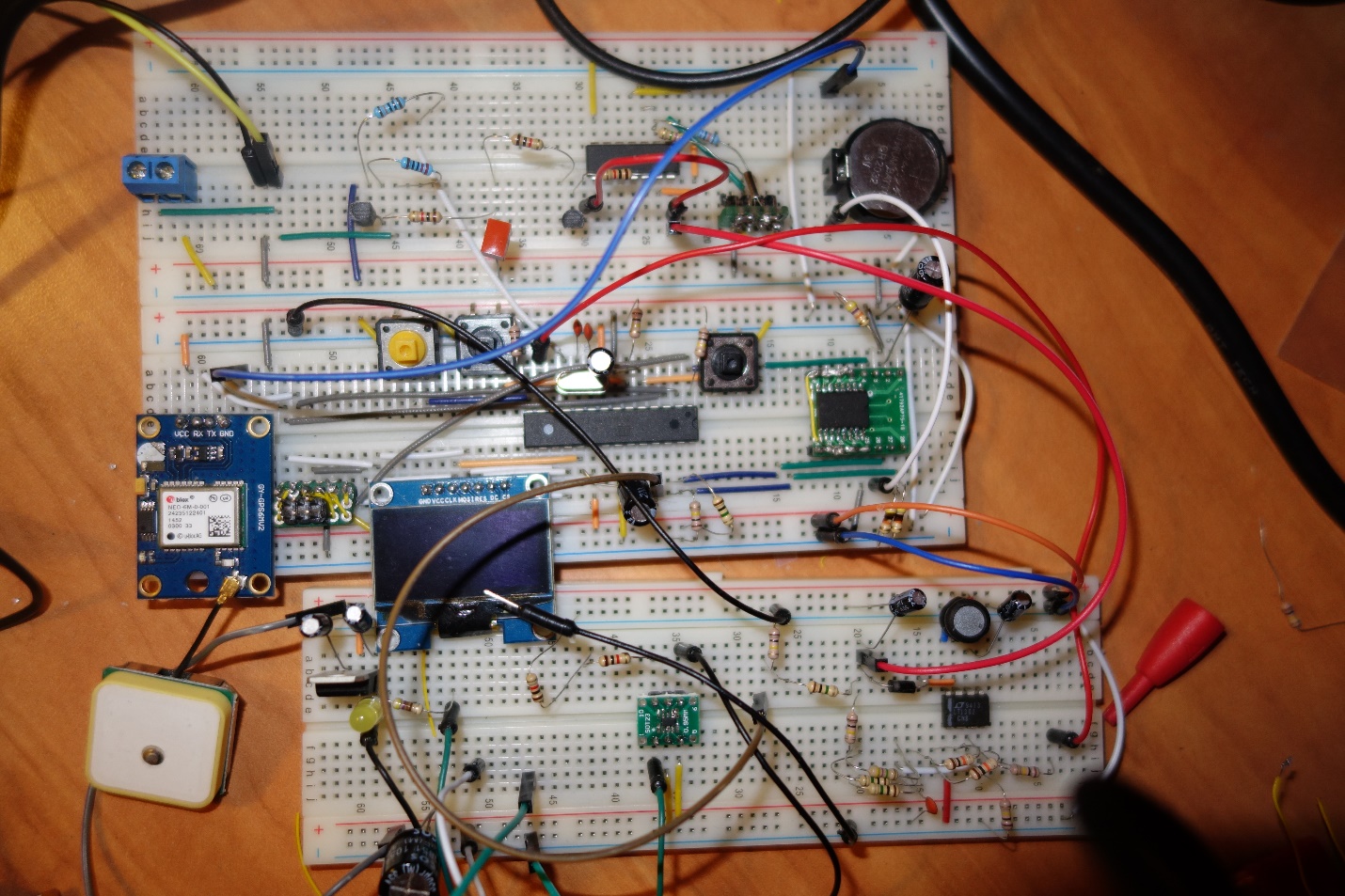
Я нашел несколько ссылок на интересные dyi в корпусах. Для меня было важно, чтобы устройство имело опрятный вид. Я увидел несколько видео [9], где прошивают atmega328 в Arduino. Прошивка к этому времени была уже непростой. Переносить ее на что-то другое не хотелось. Arduino IDE мне при таких раскладах показалось хорошим инструментом. Прошивал Arduino используя usbasp, который тоже заказывал в Китае.

Я не хотел связываться с самостоятельным травлением плат. Поэтому решил, что устройство буду распаивать на специальной макетной плате под пайку. Поэтому предпочитал микросхемы в dip корпусе. В тех же случаях, когда хороших альтернатив в dip корпусе нет, использовал переходники.

# Улучшение Передачи

Успех в передаче позволил наконец начать проводить эксперименты с передатчиком и приемником. Я ставил передатчик и носил часы по всей комнате в поисках лучшего сигнала. Выяснил, что примем лучше всего, когда Антенна направлена своей осью на часы. И часы находятся на расстоянии не больше метра – полутора. Кроме того, наконец то приехал осциллограф. Он тоже позволил получить понимание о происходящих процессах. Проводил эксперименты с катушкой, которую присоединял к щупам. Можно было видеть, что амплитуда сигнала меняется, подтверждая наблюдения с часами. Максимальная амплитуда на катушке если она находится рядом и вдоль оси антенны. Полтора метра – мало. Были решено увеличить мощность передатчика. Вопрос как? В схеме, которую я взял за основу были установлены резисторы 1 кОм. Решил уменьшить это сопротивление. Кроме того, измерил сопротивление антенны. Было что-то около 300 ом. Забегая вперёд скажу, что здесь я не подумал о реактивном сопротивлении. Но обо всем по порядку. Уменьшил сопротивление передающей антенны за счет использования провода сечением 0.3. 

Чтобы уже работало наверняка решил поставить установить step-up модуль для передающей части. И иметь возможность отключать его для экономии энергии. В такой конфигурации всё стало заметно лучше. В пределах 2-3 метров на оси передатчика было много успешных синхронизаций. Вне оси успешные синхронизации были на расстоянии метра двух.

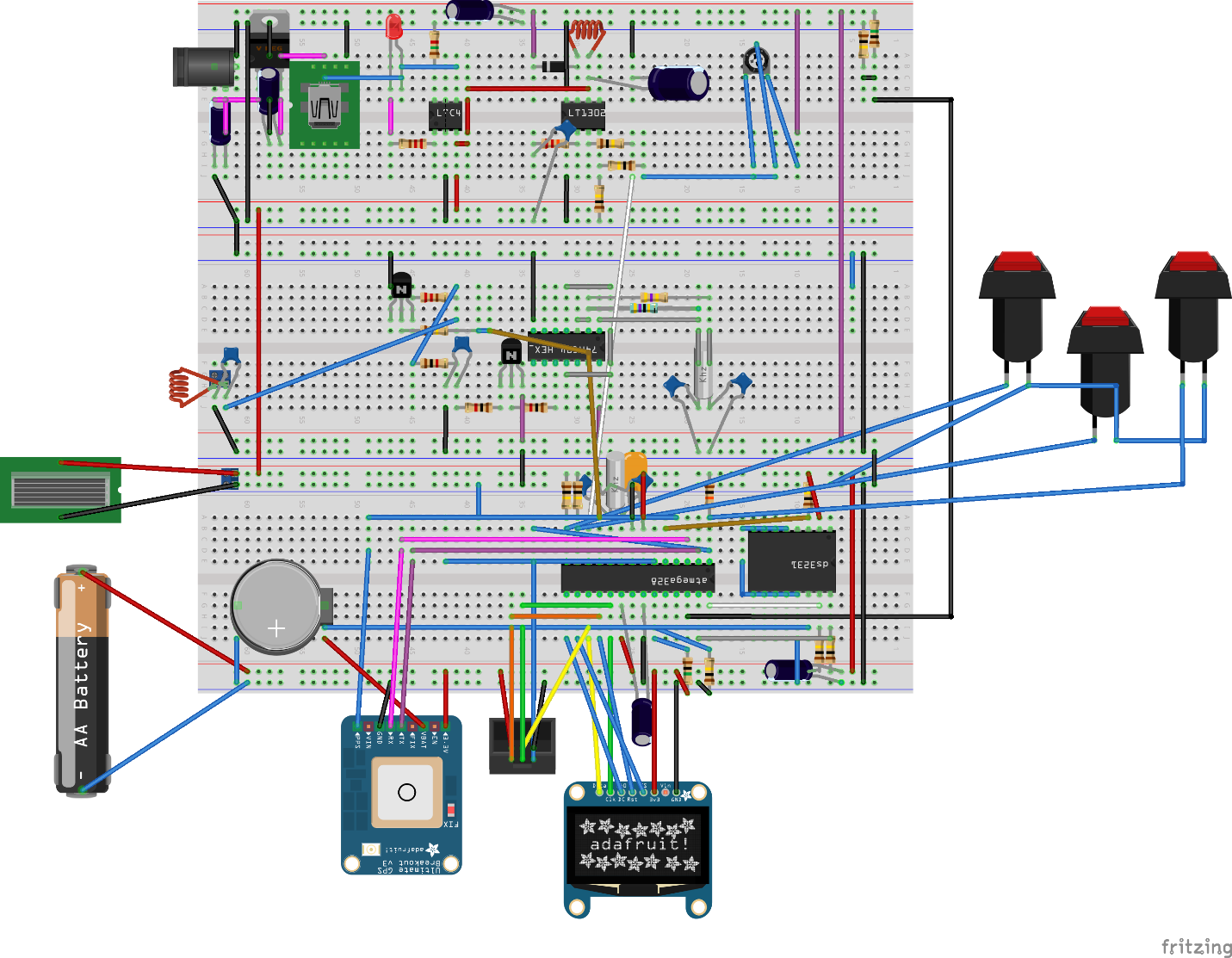


Для повышения напряжения использовал lt1302. Кроме того, нашел интересную зарядку [10] на ltc4054-4.2, которая хорошо вписывалась в концепцию устройства.

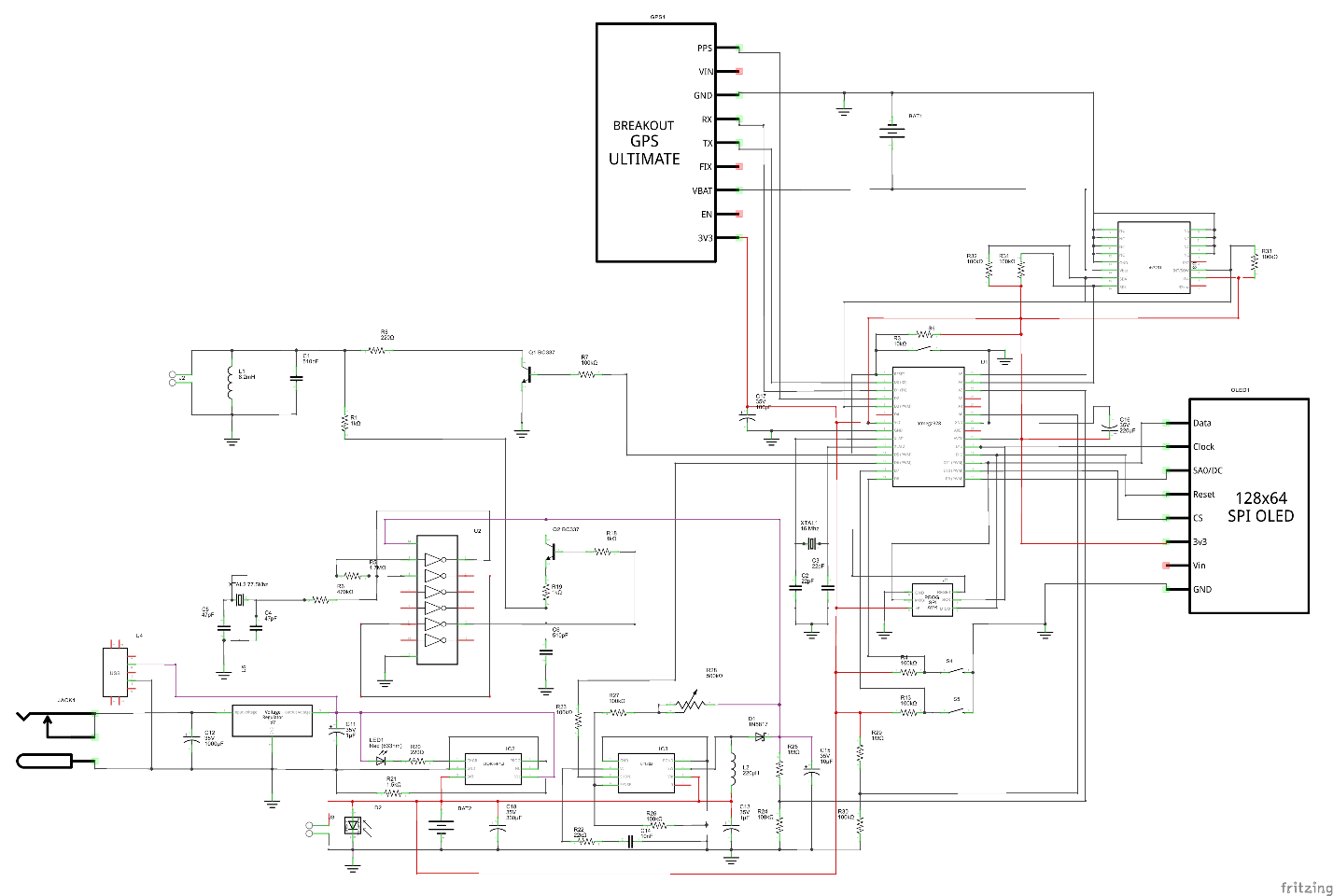
На этом этапе я понял зачем нужны даташиты. Они очень помогают при использовании микросхем, особенно когда дело касается специализированных устройств. Схемы подключения lt1302 и ltc4054-4.2 я уже брал из даташита. И завел привычку помещать в git кроме схем, исходников еще и даташиты всех использованных устройств.

# Первая попытка собрать устройство

Итак, я подобрался к пайке проекта. На этот момент я имел схему устройства. И проект на fritzing, который повторял макетную плату.

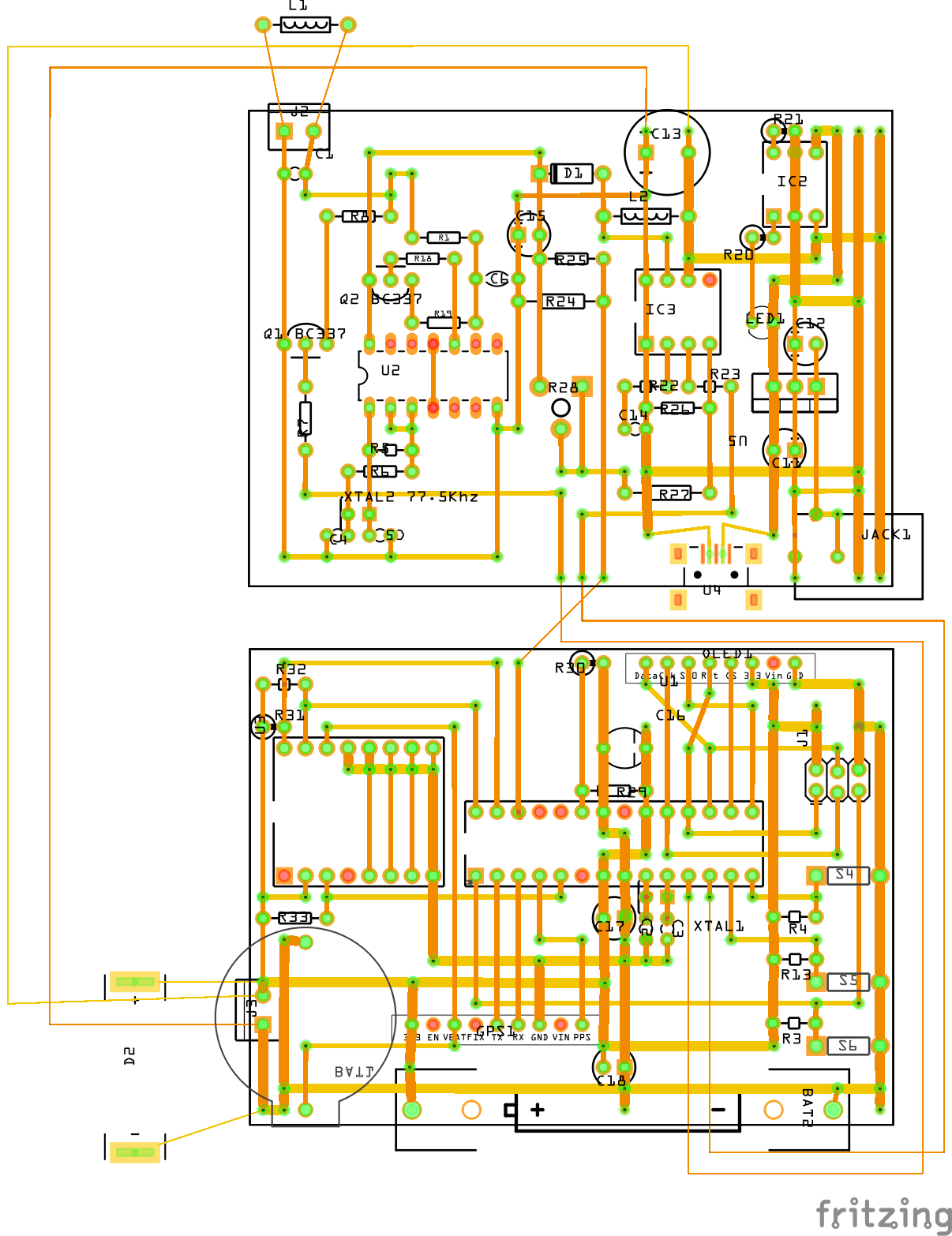


Отдельный геморрой – создавать свои компоненты для fritzing. Как оказалось, это неприятная часть вообще в любой САПР.



Еще одна задача, которая длиться достаточно долго – разведение платы. Это может длиться несколько дней. Но в принципе вполне посильно. Основной момент: размещаем элементы, потом их соединяем. Если что-то неудобно соединяется пробуем немного переставить компоненты. Заметил у Fritzing недостаток. Если что-то переставляешь на макетной плате или на схеме, то это влияет на макетную плату. И могу возникать лишние соединения. Поэтому при небольших изменениях в одном месте их нужно делать в другом. В общем попробуйте, но утомляет сильно. Хотя для простых схем fritzing неплох.

Вот что получилось.



Использовал макетную плату 15x9. Распилил ее пополам. Кстати еще один нужный прибор для радио поделок – дремель.

Паял пару дней. Вроде заработало. Но было много навесного монтажа. Выглядело некрасиво.

Пишут, что платы нужно отмывать после пайки. Решил, что попытаюсь отмыть. Использовал для этого flux-off. Но как-то не заладилось. Отмывал в субботу на работе, когда почти никого не было. Но от него стала кружиться голова, хотя использовал немного и осторожно. Короче отмыл и выкинул его. При этом отмылось все плохо. Пока решил забить.

Вот что получилось.



Здесь уже другой модуль GPS. Я хотел, чтобы устройство работало от батареи долго. GPS модуль потребляет достаточно много, я решил найти GPS устройство с линией enable, чтобы его можно было отключать. Enable не заработал у меня. Зато GPS на mtk3339 оказалось, что потребляет в 2 раза меньше чем neo-6m. 30ma против 60ma. И GPS модуль mtk3339 можно погружать в сон.

# Переосмысление проекта

Собрал устройство, рассчитывая, что после пайки заработает все гораздо лучше. Но так не случилось. Всплыли старые проблемы – неустойчивая синхронизация. Была также проблема при включении устройства без батареи. Оно не работало. Удалось выяснить, что проблема возникала из-за того, что step-up при запуске заряжал большой конденсатор, и на старте устройство потребляло большой ток, больше чем ток, на который был рассчитан блок питания, который уходил в защиту. С батареей все было нормально, поэтому пока решил не обращать внимание на проблему. Просто подобрал конденсатор, чтобы не было помех и отключение случалось меньше. Резистор ставить не хотелось, чтобы не увеличивать и без того большое потребление устройства. Устройство работало от силы – день. Что не удовлетворяло требованиям. Проблему с неустойчивой синхронизации удалось сильно уменьшить после того, как я нашел ошибку в описании протокола на русской Wikipedia [3] и в [15]. В немецкой версии все было нормально. Ошибку на Wikipedia я исправил. Оказалось, что амплитуда должна опускаться не до 75% а до 25%. Я это заметил, проводя эксперименты, но потом нашёл и подтверждения, например, на немецком wiki. Так, с синхронизацией все решилось, но радиус все равно был небольшим. На дисплее с краю была полоска с мусором, с которую я не мог побороть.

Мне не понравился переключатель, который выключает питание. Он оказался очень громоздким. Кроме того, в transistor tester я подсмотрел идею, что можно включать питания нажатие на кнопку, которая может использовать и для других целей. Transistor tester при этом в режиме простоя не потребляет энергии практически.

Форма сигнала оставляла желать лучшего. Анализатор спектра показывал много лишних гармоник.

Я стал думать, как улучшить устройство. Нагуглил еще один вариант синхронизатора [12]. Интересно в нем то, что за логическим элементом есть фильтр на дифференциальном усилителе lvm824, а за ним усилитель на транзисторе. Разобрал устройство. Благо оно состояло из двух частей. Использовал логическую часть для управления, а передающую часть стал собирать на макетной плате и экспериментировать.

Как есть я не хотел брать схему. Потому, что падение напряжение на биполярном транзисторе 0.6 вольт, а напряжение питания и так примерно 4v. Стал думать, что делать.

Рассматривал варианты со звуковыми усилителями. Но, во-первых, они рассчитаны на частоту до 40khz, а во-вторых у них падение напряжение 2 вольта. Не нашел ничего подходящего.

Я уже встречал характеристику rail-to-rail которая означает, что усилитель работает во всем диапазоне питания, поэтому стал искать подобные. Я понял, что нужно разобраться как делаются и рассчитываются фильтры. Нашёл ресурсы, где можно посчитать фильтры[13].

Транзистор Q1 в схеме [14] использует как усилитель. Чтобы не терять 0.6 вольта, я по началу решил сделать то же самое на полевом транзисторе, но читая дальше об усилителях, я обнаружил интересную схему усилителя – мостовую, который лучше усиливает сигнал в противофазе. Попробовал такую схему на lvm822 с транзистором на выходе. Преимущество такой схемы в том, что при том же питании амплитуда сигнала удваивается. А это значит, что в идеале мощность учетверяется. Также я заметил, что есть микросхема ad8532, которая по выходной мощности достаточна для моих целей и таким образом не нужно делать ставить выходной транзистор. Такое подключение сильно уменьшила помехи и позволило отказаться от step up.

На второе улучшение передатчика я тоже наткнулся случайно. Я видел, что реактивное сопротивление моего колебательного контура 1.5 kOm. Это ограничивает мощность передатчика. Когда я экспериментировал с уровнем передачи на катушку я подумал, что можно сделать повышающий трансформатор после усилителя. Тогда с выхода усилителя на вход антенны будет подаваться большее напряжение. Это удалось сделать, намотав несколько витков на передающую Антенну. На самой антенне примерно 200 витков, а на другой обмотке примерно 10 витков. Точно количество подбирал осциллографом и тестером. К этому времени у меня уже был тестер с частотомером.

Такие изменения дали большую прибавку к мощности передачи. Теоретически увеличение мощности по сравнению со старой схемой будет примерно 400 раз. Кроме того, сигнал стал почти чистой синусоидой. Но пришлось добавить на антенну блокирующие конденсаторы, из-за постоянной составляющей, которая пролезала в сигнал.

Радиус действия увеличился сильно. Можно было даже синхронизировать часы в другой комнате, не обращая внимания на расположение антенны.

Но на работе устройство не работало. Думаю, когда много включенных компьютеров – много помех.

Еще одна проблема -- передатчик иногда не запускался из-за того, что не запускался генератор. Это удалось решить за счет того, что корпус кварца я заземлил и за счет более стабильного питания.

# Финальный вариант

Кроме улучшений передающей части хотелось сделать ключ выключения как в транзистор тестере, но там он сделан на биполярном транзисторе. Я чтобы уменьшить потери решил его сделать на полевом с низким сопротивлением. Здесь я его смоделировал [16]. Также опробовал на макетной плате. Все было отлично: падение напряжения было примерно 0.05в, что меня устраивало.

Хотелось следить за энерго-эффективностью. Поэтому решил встроить амперметр ina139. На макетной плате опробовал. Но пришлось искать сопротивление 0.05Om.

Передатчик сделал отключаемым все через тот же полевик AO3401 (Rds=60mOm). В сумме падение напряжение 0.1в на передатчике.

Стал переделывать схему в fritzing. И тут fritzing стал показывать с худшей стороны. Схема стала разваливаться. Уже проведенные соединение оставались, но контакты считались отсоединенными. Исправить проблему не удалось. Чем больше исправляешь, тем больше схема разваливалась. На работе к этому времени у нас организовался клуб единомышленников. Я стал интересоваться у знакомых, кто чем пользуется для проектирования схем. Мне посоветовали kiCad и ToporLite. Я переделал схему на kiCad. Сделал новые компоненты, которых не было. Это все заняло очень много времени. Воспользовался автотрассировкой от topor. В принципе понравилось. Немногим позже я столкнулся с deeptrace. Но переходить на него уже не решился. Хотя подозреваю, что deeptrace всё-таки лучше.

Сразу решил, что уже не буду затачиваться на dip. Все буду делать на smd, а, чтобы не заморачиваться с травлением, плату закажу на али или в Зеленограде.

Корпус у меня уже был давно подготовлен. Поэтому плату разводил под корпус.

Разводил плату несколько дней. И решил все-таки заказывать в Зеленограде. Сделали быстро. Получилось красиво. Стал распаивать и обнаружил несколько ошибок. Перепутал rx и tx gps. Перепутал + и 0 датчика тока. Ошибся с размерами на пару миллиметров с каждой стороны. Отверстия под крепление дисплея сделал меньше чем надо. Кое-где на компонентах не было маски. Неправильно развел линейный стабилизатор для питания от 9В – ну учел его охлаждение. Нужно было сделать большой полигон под него. И неправильно развел сам разъём 9в. Если разводку разъёма можно было исправить, то от перегрева было избавиться трудно. Поэтому 9в разъем я убрал.

И еще был ряд мелких проблем. Самые неприятные первых две. С ними очень долго промучился.

Еще одна проблема разводки в том, что я невнимательно читал даташит. Не обратил внимание на то как разводить модуль GPS. Там оказалось необходимым сделать отверстие. Все подобные исправление сделал уже после дремелем. Некоторые исправления пришлось делать поверхностным монтажом. Хуже всего было с датчиком тока.

Вообще паять было сложно. Сложнее всего было распаять GPS модуль. Даже без ошибок очень просто замыкаются соседние дорожки.

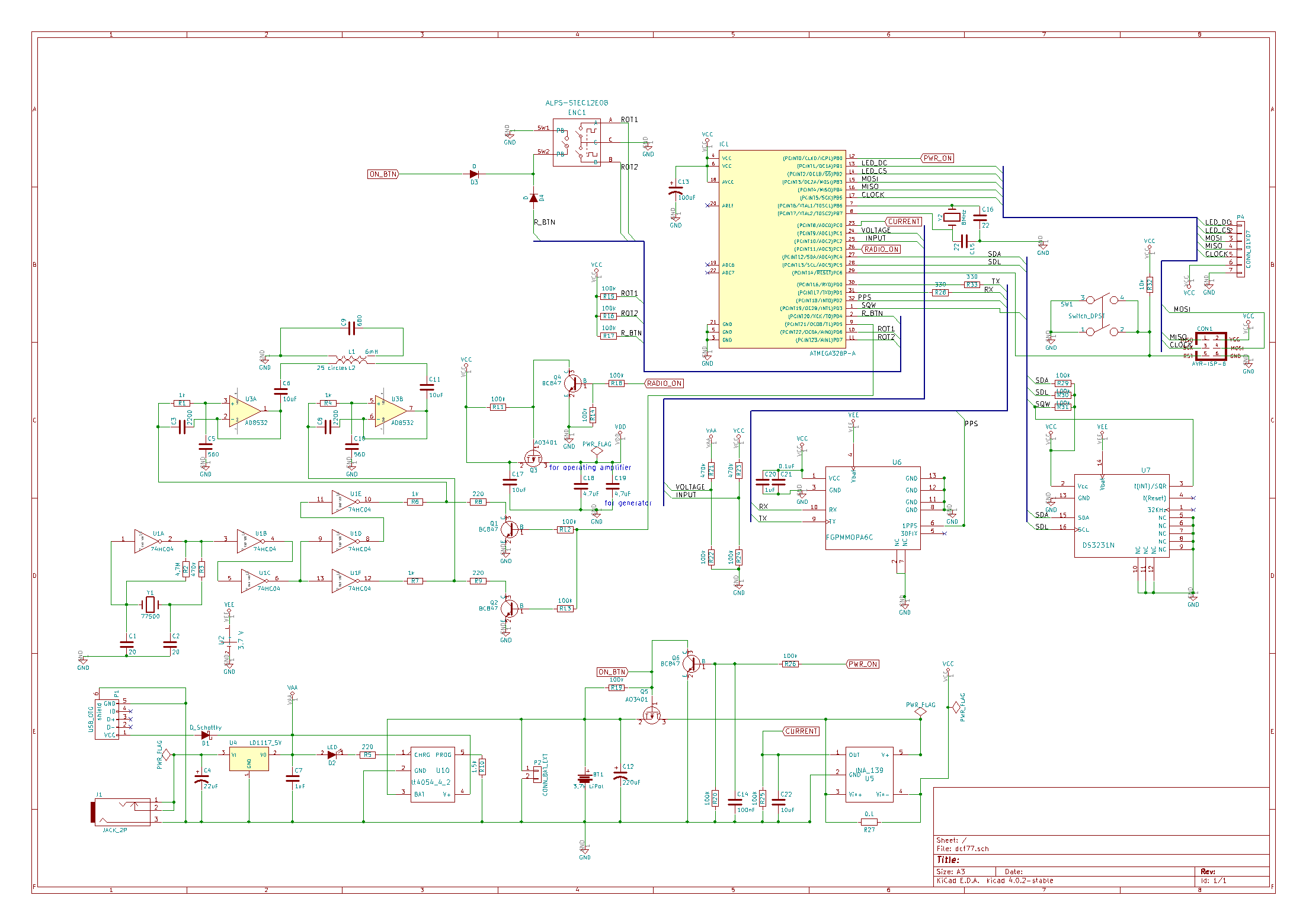
Промахнулся с шлейфом экрана. Не очень аккуратный разъём получился. И громоздкий.

Но устройство собрал. Опять попробовал отмыть флюс. На этот раз отмывал изопропиловым спиртом. Так отмывалось гораздо лучше и без последствий.

Вот что получилось







# Последние штрихи. Доработка программного обеспечения

Последним штрихом было уменьшения потребления за счет управления питанием, контролем за потреблением и использованием солнечной батареи для подзарядки устройства.

Дорабатывал шрифты, чтобы прошивка влезла в память. Исправил проблему с полоской на экране. Оказалось, что китайцы продали мне экран на sh1106 вместо ssd1306 как было написано в заказе. Ну да бог с ними.

Для контроля за нагрузкой используется round robin database для значения тока.

Еще один тонкий момент – определение, когда gps синхронизировал время, чтобы его отключить. Для этого пришлось дорабатывать библиотеку gpstiny.

Сейчас со всеми доработками устройство работает примерно месяц без зарядки.

# Ретроспектива

Начал я проект в январе 2015 года. Первые результаты я получил в апреле. Спаял на макетной плате в июле. После этого был перерыв до марта 2016 года. Железную версию на заказанной плате собрал в июне 2016. В 31 июля 2016 был последний комит по улучшению софта. Таким образом я потратил на это устройство полтора года. Можно ли было к проекту подойти по-другому? Наверное, да. Но знаний очень не хватало на начало проекта. Теперь понятно, что для такой работы нужны инструменты и какие-то детали. И ясно какие. Нужны знания. Они появились. Я, конечно, до сих пор путаю эмиттер и коллектор на схеме, и не смогу по памяти рассказать, как транзистор работает в каждом из режимов подключения. Но знаю где это быстро посмотреть. Было потрачено много денег на проект. Дешевле, наверное, было бы купить часы с синхронизацией по GPS ☺. Можно было бы сократить затраты? Да можно, но не принципиально. Можно ли продавать устройство? Да можно. Но трудозатраты на его производство слишком большие. Выгоднее просто работать там, где я работаю сейчас. Я видел в продаже устройства попроще за 150 евро, а мне его нужно собирать дня 2-3. И, думаю, что продавать такое тяжело. Так что, коммерческого успеха у устройства не вижу. Более того, сейчас стали появляться часы с синхронизацией по gps. Они скоро станут дешевле. Получил ли я удовольствие от процесса? Да получил. Было очень интересно изучать новое. Мне как программисту это было полезно и для увеличения кругозора. Выложу ли я его исходники? Может быть. Если это нужно кому-то.

# Возможные улучшения

Попытки синхронизации делаются каждый день, но все же часы синхронизируются через раз. Почему? Пока ответа нет. Если запускаешь синхронизацию руками, то она происходит всегда. При этом ночью передатчик включается, судя по показаниям амперметра.

Новую ревизую платы я развел, но не заказывал.

Нужно улучшить систему питания контроллера. Нужно отключать питание gps полностью, так как в режиме ожидания gps всё равно ест 1 mа.

Нужно сделать часы более точными. Есть идея как это сделать, но только на stm32. Там можно использовать функцию подстройки хода по сигналу с GPS. Также stm32 позволит отказаться от лишней микросхемы. Хотя ds3231 более точная, но для stm32 можно сделать подстройку.

Попробовать использовать e-ink экран для экономии энергии. Сейчас в основном экран съедает всю батарею. Он потребляет где-то 4 ma из 6 ma в режиме ожидания, когда показывается только текущее время. Передатчик потребляет 60 ma когда включен. Это 20 минут в день. GPS потребляет 30 ma, но он синхронизирует время за минуту.

# Ссылки

[1] <http://journal.gladkiy.me/radiosinhronizatsiya-chasov-s-vy-shkoj-dcf77-v-germanii/> (на данный момент ссылка не доступна)

[2] <http://www.elv.de/dcf-empfangsmodul-dcf-2.html>

[3] <https://ru.wikipedia.org/wiki/DCF77>

[4] <https://www.youtube.com/user/Zefar91/videos>

[5] <http://ikit.edu.sfu-kras.ru/CP_Electronics/pages/mm/index.html>

<http://digteh.ru/Sxemoteh/ShVklTrz/OK/>

[6] <http://www.mikrocontroller.net/articles/AVR_Transistortester>

[7] <https://www.sundgren.se/1-recreation/2-electronics/dcf77_simulator.htm>

[8] <http://robotosha.ru/arduino/digitalwrite-optimizing-arduino.html>

[9] <https://www.youtube.com/watch?v=sNIMCdVOHOM&index=1&list=PLnOALHW_SnwoVHhkoyYadEQ9IuQ6nEj6M>

[10] <http://radiokot.ru/circuit/power/charger/33/>

[11] <https://www.youtube.com/watch?v=lkWZuAnHa2Y>

[12] <http://pda.teron.ru/index.php?s=74c6aae329db3030d1b6cb47ad47051b&showtopic=909610&st=20>

[13] <http://sim.okawa-denshi.jp/en/OPseikiLowkeisan.htm>

[14] <http://pda.teron.ru/index.php?app=core&module=attach&section=attach&attach_rel_module=post&attach_id=516440>

[15] <http://www.kaligraf.narod.ru/DCF77_protokol.html>

[16] <http://everycircuit.com/circuit/5072084672184320/p-mosfet-based-electronic-switch>