

СОДЕРЖАНИЕ
(с гиперссылками)

РАБОТА С КОНФИГУРАТОРОМ

1. НАЗНАЧЕНИЕ ОРГАНОВ УПРАВЛЕНИЯ ПРОГРАММОЙ

- [1а. Кнопки работы с файлами и режимов работы](#)
- [1б. Кнопки опций прошивки](#)
- [1в. Кнопки вывода на график](#)
- [1г. Переключатель выбора корректируемой таблицы](#)
- [1д. Кнопка интерполяции и перестройки](#)
- [1е. Поле ввода таблиц](#)

РАЗРЕШЕНИЕ (ПРОГРАММНАЯ ТОЧНОСТЬ) ПРОГРАММЫ

- [1ж. Параметры прошивки](#)
- [1з. U ДАД, V](#)
- [1и. Время искры](#)

[Линейная интерполяция уоз \(от гиперболы оборотов\)](#)

[Напряжения с индуктивного датчика](#)

[Типовые графики](#)

[Разработчикам](#)

ОПИСАНИЕ ФУОЗ
Принцип работы ПО

- [1. Предназначение ФУОЗ](#)
- [2. Подключение к штатной системе](#)
- [3. Организация обработки прерываний](#)
- [4. Режим работы оборудования \(инициализация\)](#)
- [5. Общее описание работы](#)
 - [5.1. Включение питания](#)
 - [5.2. Запуск двигателя](#)
 - [5.3. Операции вне искры \(расчёты\)](#)
 - [5.4. Работа мотора после пуска](#)
 - [5.5. UART](#)

Методы проверки работоспособности ФУОЗ, а также порядок настройки кривых, оборудование для этого и методы подгонки характеристик совместно с сервисной программой наладки будут приведены отдельным пакетом со своим Мануалом.

РАБОТА С КОНФИГУРАТОРОМ

При первом запуске программы желательно в свойствах файла разблокировать его, как из Интернета.

Также при первом запуске программы или после копирования на другой носитель, комп будет постоянно спрашивать включения макросов (жёлтая строка вверх программы). Включите, иначе программа работать не будет.

1. НАЗНАЧЕНИЕ ОРГАНОВ УПРАВЛЕНИЯ ПРОГРАММОЙ

1а. КНОПКИ РАБОТЫ С ФАЙЛАМИ И РЕЖИМОВ РАБОТЫ (справа внизу)

"ЗАГРУЗИТЬ ФАЙЛ" - загружает из директории с этой программой файл типа ***.inc**. Не загружайте файлы конфигурации МК, имеющие то же расширение.

"СОХРАНИТЬ" - сохраняет полученный файл данных УОЗ и ДАД в директорию с программой под именем **Data-1.inc**. Перед компиляцией переименовать его в **Data.inc**.

"КОМПИЛЯЦИЯ" - компилирует файлы программы, данных и конфигурации МК в папку с Конфигуратором, под именем **IGN.hex**. Для компиляции программы в папке с Конфигуратором необходимо наличие следующих файлов:

- **avrasm2.exe** - сам компилятор;
- **m8def.inc** - файл конфигурации AT Mega 8;
- **IGN.asm** - файл программы;
- **Data.inc** - файл данных УОЗ и ДАД.

Во время работы компилятора в папке с Конфигуратором создаётся файл **Compiler.bat**, после окончания компиляции он автоматически удалится, минуя Корзину.

"UART" - запуск программы (от **Flash#13**) обмена данных между МК и компьютером для просмотра УОЗ в работе и устранения неполадок. Запускается на подключённом к моту и компьютеру ФУОЗ с прошитой программой. Для работы этой функции, в папке с Конфигуратором необходимо наличие файла **FUOZ_UART.exe**, а также шнурок USB-TTL и установлены драйвера.

1б. КНОПКИ ОПЦИЙ ПРОШИВКИ (справа сверху)

Это переключатели, их состояние индицируется под ними (в сером поле, фиолетовым цветом).

"ОПТИКА" - переключатель "Индуктивный датчик / Оптика (или Холл)".

"Хар ДАД" – переключатель «Линейная / Квадратичная» характеристика ДАД.

Линейная характеристика определяет линейную зависимость изменения угла от нагрузки моту, т.е. с линейным изменением нагрузки (положением ручки газа) угол будет меняться линейно. Характеризуется увеличением резвости моту на динамических режимах, но на может добавлять нежелательные динамические рывки на трансмиссии.

Квадратичная же характеристика уменьшает изменения угла при незначительных изменениях нагрузки, характеризуется плавностью изменения динамических характеристик. Влияет также и на изменение угла при статических нагрузках (перевозим мешок с песком + пассажир).

В любом варианте, настройку кривой ДАД начинают с линейной характеристики (квадратичная притупляет ощущения), а затем, выбирают ту или иную характеристику. Кнопки опций прошивки не влияют на углы УОЗ.

"UART" – включает/отключает в прошивке возможность обмена по UART.

1в. КНОПКИ ВЫВОДА НА ГРАФИК (справа по центру, синего цвета)

Кнопки фиксируемые, независимые друг от друга.

"Таб1", "Таб2", "Таб3" - кнопки вывода на график соответствующей таблицы. Таблицы можно вывести как одну, так и все вместе, или все отключить - в любой комбинации.

На содержание данных состояние кнопок никак не влияет, служит только для вывода на экран.

1г. ПЕРЕКЛЮЧАТЕЛЬ ВЫБОРА КОРРЕКТИРУЕМОЙ ТАБЛИЦЫ (слева внизу коричневые кнопки).

Состояние выбранной таблицы индицируется зелёным флажком слева от соответствующей кнопки.

"Таб1", "Таб2", "Таб3" - кнопки, чтобы не загромождать поле ввода данными из других таблиц. При нажатии содержание предыдущей таблицы выгружается в буферную память, а на её место загружаются данные выбранной таблицы.

1д. КНОПКА ИНТЕРПОЛЯЦИИ И ПЕРЕСТРОЙКИ ГРАФИКОВ (слева сверху)

Кнопка не фиксируемая.

"ПЕРЕСТРОИТЬ" – перестраивает график по новой таблице с применением интерполяции и формирует буфер выходных данных.

Суть работы алгоритма: По нажатию кнопки все таблицы (могут быть неполными) сортируются по оборотам и проверяются на корректный ввод данных, сбрасываются в буфер на 32*3 точек. Затем этот буфер копируется в буфер на 256 точек * 3 таблицы на соответствующие точкам места. Затем анализируются пропущенные точки и обороты и углы границ пропусков. На основании этих данных включается алгоритм интерполяции, который рассчитывает и пишет недостающие ячейки.

Сама математика алгоритма приведена ниже. После этого заполненный буфер копируется в выходной буфер, откуда данные отправляются на экран и, когда надо (сохранение), на формирование выходного файла.

Термины предупредительных сообщений:

- **Обороты < 460.** Программой предусмотрено, что обороты должны быть неотрицательные и не менее 460.
- **Дублирование скоростей.** В графе оборотов имеются дубли скоростей, удалите лишние дубли, особенно если они имеют разные углы.
- **Склеивание скоростей.** На вид скорости разные, но при расчёте кодируются одним числом. Касается особенно высоких скоростей – увеличьте шаг между указанными скоростями. Либо увеличьте текущую скорость, либо уменьшите предыдущую. Пример склейки скоростей: **6000** и **6010 об/мин**. На графиках склейка хорошо заметна по горизонтальной ширине ступеньки.

1е. ПОЛЕ ВВОДА ТАБЛИЦ (3 столбца слева)

Здесь, в принципе, всё интуитивно понятно, замечу только, что:

1. Ячейки по скорости можно вводить не по порядку, например, сначала края таблицы, потом середину. Программа отсортирует автоматом при перестроении или формировании выходного файла.
2. Заполнять таблицу полностью не обязательно, можно ввести хоть 3 точки. Все недостающие точки при перестройке графика или сохранении будут определены.
ВНИМАНИЕ. При сохранении, в выходном файле в виде комментариев сохранится массив именно Ваших **32 БАЗОВЫХ точек построения** (может быть и 2 точки), поскольку программа из массива данных не знает, какие из 256 точек являются Вашими **БАЗОВЫМИ**. Не удаляйте этот массив комментариев, компилятором он не используется, используется только Конфигуратором.
3. Если Вам хочется вклеить промежуточную точку (а всё занято), обратите внимание на точки фиксированного УОЗ – это горизонтальная прямая в правой части графика. Удалите все точки между началом и концом прямой, вставьте нужные Вам точки на их место. Нажмите **"ПЕРЕСТРОИТЬ"** и продолжайте корректировку. При некорректных данных вылезет предупредительное сообщение.
4. Для улучшения восприятия, углы ДАД, равные **0**, при загрузке в ячейках не отображаются.
5. Углы ДАД отсчитываются от основной кривой и только вниз. Т.е. имея угол, например, 20 ° и поправку ДАД = 5 °, получим при максимальной нагрузке результат 20 ° - 5 ° = 15 °. Если нет ДАДа, оставляем пустое поле, либо забиваем туда 0 °, либо просто стираем содержимое ячеек.

РАЗРЕШЕНИЕ (ПРОГРАММНАЯ ТОЧНОСТЬ) ПРОГРАММЫ

Программа формирует задержки, кратные 2 мкс (2 байта в задержке). Т.е., например, для 9000 об/мин минимальная задержка составляет $(9000 \text{ об/мин} = 150 \text{ об/сек} = 6,67 \text{ мсек/об} = 18,5 \text{ мкс/}^\circ$. $2 \text{ мкс/} 18,5 = 0,11^\circ$. На меньших оборотах точность пропорционально выше.

Тем не менее, для упрощения составления таблиц, углы округлены до **0,1°**, а обороты - **10 об/мин**.

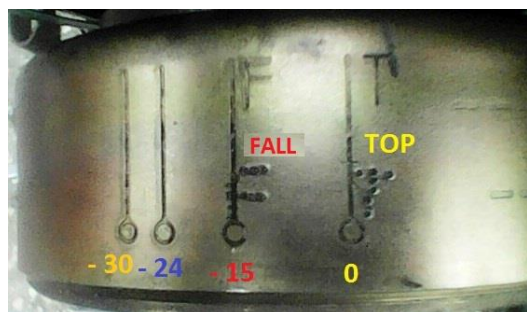
Разрешение оборотов связано со счётчиком периодов МК и имеет 1 байт, посему на высоких оборотах оно значительно снижается. Поскольку ступенька имеет 2 катета (обороты - углы), то на высоких оборотах высота ступеньки будет независима от разрешения угла (при его больших приращениях).

Поскольку УОЗ выходит на максимум на оборотах, не превышающих 4000 и тем более, подходит плавно, без изгибов, то с этим, можно как-то считаться. Другой вопрос, ДАД ведь, в большинстве случаев не выходит на больших оборотах на горизонтальную кривую. Тем не менее, сильно он там не изменяется, поэтому этим можно пренебречь. ДАД тоже кодируется 2 байтами.

1ж. ПАРАМЕТРЫ ПРОШИВКИ

"НАЧАЛЬНЫЙ УГОЛ, °" - угол от начала метки (наплыва на роторе) до ВМТ. Пишется в файл данных ТОЛЬКО для работы Конфигуратора.

ПРИМЕЧАНИЕ: Для его определения я измерил диаметр ротора, умножил на π и разделил на 360 °. Затем измерил расстояние от метки до ВМТ, разделил на разрешение и получил 30°.



Ignition	Flywheel magneto
Point gap	0.3 – 0.4 mm (0.012–0.016 in)
Ignition timing "F" mark	<u>15° BTDC, Static</u>
Full advance	<u>30° BTDC at 3,100 rpm</u>
Starting system	Kick starter and starter motor
Valve clearance (cold)	IN: 0.05 mm (0.002 in) EX: 0.05 mm (0.002 in)
Engine weight	18 kg (39.7 lb)
Idle speed	<u>1,500 ± 100 rpm</u>
Carburetor type	Piston valve, 14 mm (0.55 in) venturi bore

А теперь - цифры. У меня мопед Дельта-72сс (у 110 сс размеры те же).

- Диаметр ротора = **110 мм**,
- Длина окружности = **345,6 мм**,
- $1^\circ =$ **0,960 мм**.

ВМТ на роторе маркируется меткой **T** ("Top"), а само начало метки - самая левая метка. Длина составила **28,8 мм**. Получаем: $28,8/0,96 = 30^\circ$.

Метка **F** ("Fall"- задний) - это конец метки (**15°**).

Измеренные данные полностью совпадают с данными Honda C70 - прародитель китайских движков.

15° - холостой ход (**1500 ± 100 об/мин**);

30° - обороты выхода на фиксированный УОЗ (**3100 об/мин**).

Про метку 24° ничего не знаю, это около **2500 об/мин**.

13. U ДАД, V - напряжение с ДАД. Требуется какого-либо преобразования, чтобы выловить минимальное напряжение за оборот. В исходном состоянии напряжение составляет около 4,5 V (атмосферное давление). При возникновении разряжения напряжение падает где-то до 1V, в зависимости от его величины.

"ХОЛОСТЫЕ" - напряжение с ДАД на холостых оборотах (разряжение максимальное, сигнал минимальный).

"АТМОСФЕРНОЕ" - напряжение с ДАД при заглушенном моторе, в программе от него берётся 95%, воздухан вносит свою лепту. Эти данные вносятся в Конфигуратор и прошиваются.

Наличие сглаживания ОБЯЗАТЕЛЬНО, в крайнем случае, поставьте в разрыв шланга бензофильтр (совсем плохо), либо простейший RC-фильтр (лучше, но чувствительность будет в 2-3 раза хуже, соответственно, ухудшится и помехоустойчивость). Детектор минимума собирается на одном удвоенном ОУ типа **Rail-To-Rail**. Можно без переделки платы и на компараторе LM393 с RC-цепью на выходе. Не совсем удачно, компаратор шурует на частоте 1МГц, но у меня работает уже 2 месяца, аналоговый сигнал в норме.

Чтобы случайно не спалить чип в ДАДе, и для уменьшения помех, его желательно устанавливать либо рядом с ДАД, либо в его корпус. ДАД брать на 1 атм.

Я брал Газелевский, там размеры корпуса сохранили старые, но перешли на микро-чип. Поэтому плата занимает всего около 20% от внутренности. Впихнул туда детектор.

Можно ли ставить на 2Т? В общем принципе - не имеет смысла, т.к. там клапанов нет, резонансная система по впуску и выпуску. Но при использовании детектора, Вы измеряете именно вакуум, перепускные броски давления Вам по боку. А вакуум зависит именно от нагрузки. Лично не проверял, не на чем. Но, думаю, что будет работать нормально. Тем более, что детектор будет сглаживать все Ваши резонансные броски давления. Здесь уже RC-цепью без хотя бы диода не обойдёшься, нужен именно детектор. В общем, не важно, на какой схеме Вы его соберёте. Важно, чтобы он выхватывал из сигнала минимальное напряжение и держал его на время нескольких оборотов.

ДАД измеряет разряжение, связанное с вязкостью потока газов и его скорости. Что его потребляет и на каких оборотах – абсолютно без разницы. Только на высоких оборотах сигнал из-за той же вязкости принимает синусоидальный характер и немного усредняется.

Где врезать ДАД - до или после карбюратора? Характеристики вакуума до и после карбюратора имеют взаимно-обратные характеристики. Так, на холостых, после карба будет максимальное разряжение, до - минимальное и наоборот. К тому же, связь с мотором по вакууму до карбюратора будет гораздо ниже, соответственно, и чувствительность тоже ниже. Поэтому врезать нужно ДАД **между карбом и мотором**, программа составлена именно под этот режим.

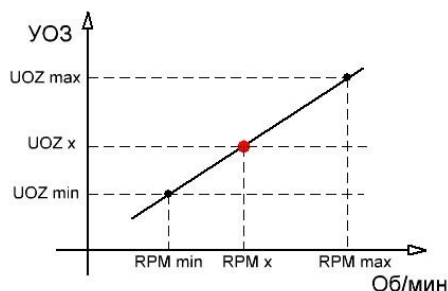
На графике присутствуют 3 маркера: кружки **красного**, **жёлтого** и **зелёного** цветов - нужны для облегчения построения кривой. Для этого их можно выделить и переместить в характерные точки, по которым потом строить кривую.

Кривая УОЗ должна быть плавной, выходить на максимум без всяких изломов. Для информативности на график нанесена дуга синего цвета - ей следовать не надо, но надо видеть, как должна выходить кривая УОЗ на постоянный угол. Если мешают (маркеры и дуга), можете удалить. Описания таблиц (слева внизу) можно заменить на свои..

Заводские характеристики допускают отклонения ± 200 об/мин (для заданного угла) или $\pm 2^\circ$ (для заданной скорости) – ошибиться трудно...

1и. Время искры – сюда заносится время искры в **мкс**, по умолчанию – **400 мкс**.

Линейная интерполяция УОЗ (от гиперболы оборотов)



$$\frac{\Delta UOZ}{\Delta RPM} = \frac{UOZ x - UOZ min}{RPM x - RPM min}$$

$$UOZ x = \frac{\Delta UOZ}{\Delta RPM} * (RPM x - RPM min) + UOZ min$$

←

$$\begin{aligned} \Delta UOZ &= UOZ max - UOZ min \\ \Delta RPM &= RPM max - RPM min \\ \Delta N &= N max - N min \end{aligned}$$

N – счётчик длительности периода

$$RPM max = \frac{117187}{N min} ; RPM min = \frac{117187}{N max} ; RPM x = \frac{117187}{N x}$$

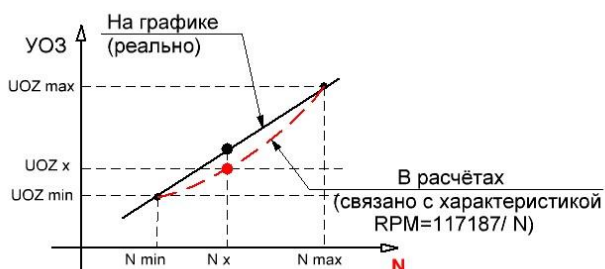
117187 в формулах сокращаются, не учитываем. Подставляем, преобразовываем, получаем:

$$UOZ x = \frac{\Delta UOZ}{\Delta N} * N min * \left(\frac{N max}{N x} - 1 \right) + UOZ min$$

Тьфу, ты.. По невнимательности не дописал. Итак, с нелинейностью разобрались. Теперь надо развернуть ось оборотов в обратную сторону (счётчик периода меньше – скорость больше). Поэтому окончательно формула принимает следующий вид:

$$UOZ x = UOZ max - \frac{\Delta UOZ}{\Delta N} * N min * \left(\frac{N max}{N x} - 1 \right) ;$$

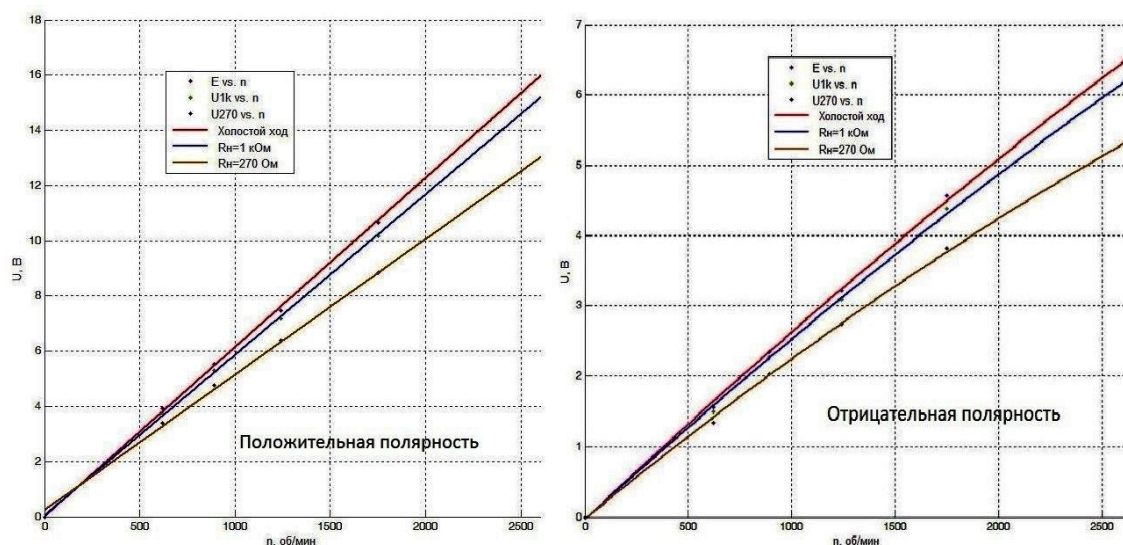
Где: $\Delta UOZ = UOZ max - UOZ min$;
 $\Delta N = N max - N min$;



В данном случае, как и на графиках УОЗ, нарисована линейная шкала, в реальности, шкала оборотов нелинейна, так как датчиком оборотов МК является счётчик длительности периода. На

оси_оборотов можно написать что угодно, в реальности, МК измеряет длины периодов, но не обороты. В области малых оборотов графика это незаметно.

НАПРЯЖЕНИЯ С ИНДУКТИВНОГО ДАТЧИКА

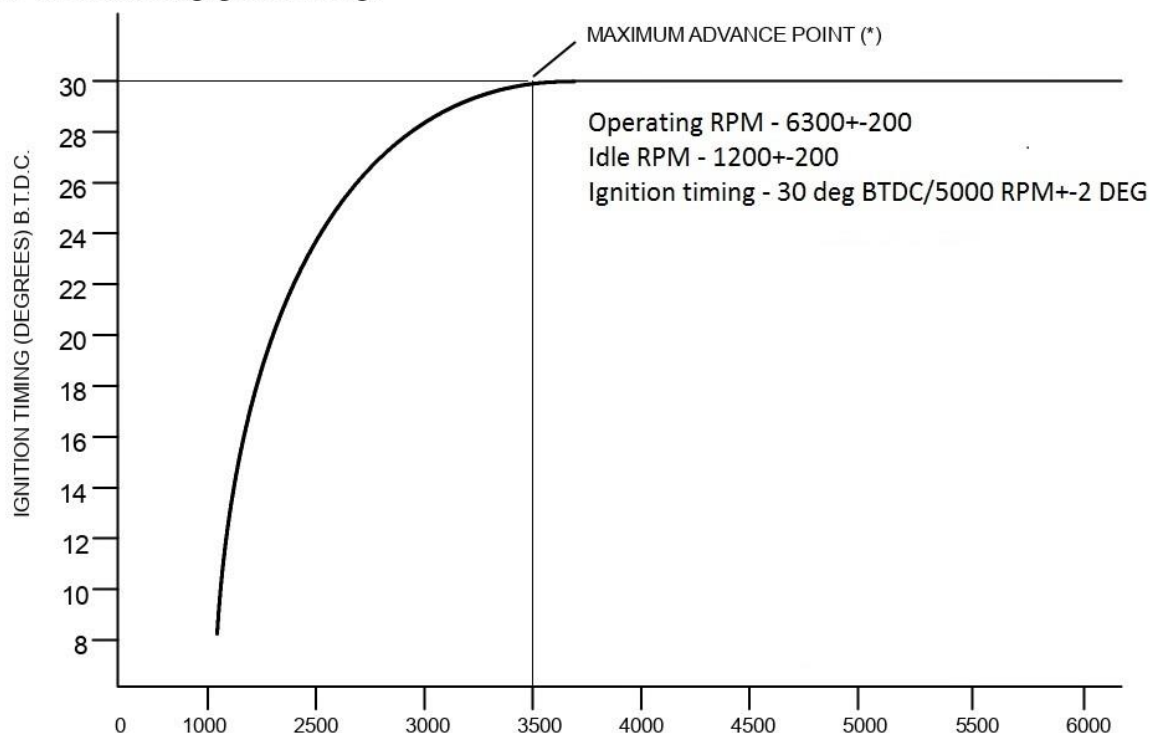


ТИПОВЫЕ ГРАФИКИ

TYPICAL IGNITION TIMING CURVE

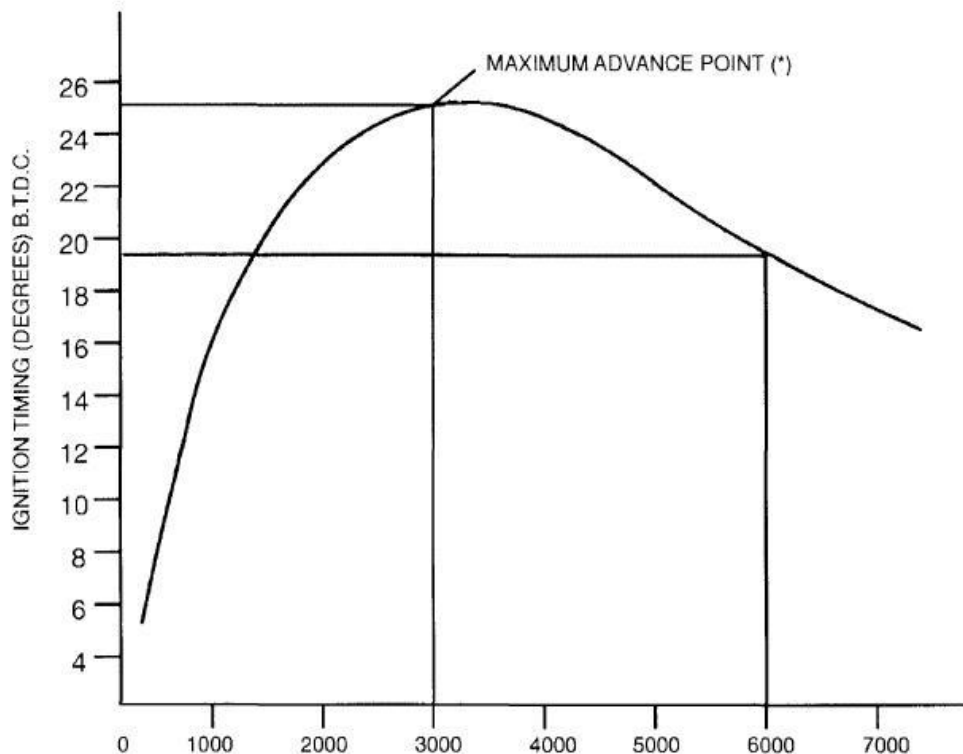
Sportsman-400 4T 425cc

* Actual advance point may vary by several hundred RPM either above or below 3500. Use the point of maximum advance when checking ignition timing.



* Actual advance point may vary by several hundred RPM either above or below 3000. Use the point of maximum advance when checking ignition timing.

Typical 2-Stroke



Разработчикам

Это краткое пояснение для тех, кто хочет изменить что-то в программе, без последствий. ☺ К примеру, изменить путь или название для сохраняемого файла или что-то другое...

При изменении структуры программы (вставка или удаление строк/столбцов, их объединение или разбивка) программа автоматически пересчитывает новые адреса ячеек. Но... Используемые в программе макросы (а их, вроде, 14) используют «жёсткие» адреса ячеек... Посему, если Вы не разобрались в программе, **не производите указанных выше действий, а также не задействуйте якобы пустые ячейки внутри рабочего поля программы – это приведёт к её полной неработоспособности.**

Первым делом зайдите в макросы и посмотрите, как они работают и какие ячейки используют, там в комментариях хорошо расписано... Начинайте с макроса при открытии книги. Фоновых макросов в программе нет. При неправильном назначении адресов ячеек, формулы в рабочих ячейках затираются со скоростью пули ☺ и без предупреждения. Номера всех столбцов (в цифрах, для макросов) смещены на 2! Не помню, где-то применил offset...

Макросы использованы по нескольким причинам:

1. Ячейка не может быть одновременно приёмником и источником данных (если это приёмник – там должна быть формула). При ручном вводе данных (источник) эти формулы затираются. Поэтому приёмом данных занимаются макросы...
2. Средства Excel не имеют циклической обработки данных и рекурсивного обращения ...
3. Средства визуализации - различные кнопки и переключатели..
4. Открытие файлов, сохранение, создание и запуск посторонних программ...

Формулы программы можно увидеть, если раздвинуть столбцы справа от программы (смотрите по буквам столбцов), макросы – через контекстное меню на кнопке...

На всякий случай всегда сохраняйте копию исходника... На свой страх и риск..

С другой стороны, все программные коды открыты и могут быть отредактированы под себя..

ОПИСАНИЕ ФУОЗ

Принцип работы ПО

1. ПРЕДНАЗНАЧЕНИЕ ФУОЗ

ФУОЗ предназначено для замены штатных коммутаторов в 4Т и 2Т ДВС. Оно построено на микроконтроллере (МК) AT Mega 8 и при формировании УОЗ учитывает степень нагрузки двигателя по показаниям датчика абсолютного давления (ДАД).

Для настройки, правки углов УОЗ и ДАД, к ФУОЗ прилагается программа Конфигуратор.

Часть технических решений заимствована с параллельной ветки форума пользователя под ником **Flash.#13**. К ним относятся: организация обработки прерываний (частично), организация обмена данными с ПК по UART + программа обмена, инициализация таймера T1.

2. ПОДКЛЮЧЕНИЕ К ШТАТНОЙ СИСТЕМЕ

В своей работе ФУОЗ может использовать либо индуктивный датчик, либо оптический датчик (или датчик Холла). Активным уровнем входов считается уровень **лог «0»**.

В случае индуктивного датчика импульсы на входе разной полярности - первым должен идти «+» импульс, следующий - «-».

ФУОЗ использует следующие 14 выводов МК:

- **в.4 МК (PD2, INT0)** – вход внешнего прерывания по Метке 1 (Датчику 1) – раннему сигналу. В китайских мопедах Дельта, например, этот сигнал соответствует, **30°** до ВМТ.
- **в.5 МК (PD3, INT1)** - вход внешнего прерывания по Метке 2 (Датчику 2) – позднему сигналу. В тех же мопедах этот сигнал соответствует, **15°** до ВМТ.
- **в.12 МК (PD6), в.23 МК (PC0)** – выходы для выдачи искры сигналом с уровнем **лог «1»**. Длительность искры задаётся Пользователем (по умолчанию - **400 мкс**). Оба порта работают практически одновременно, это сделано для совместимости с разными топологиями плат. Неиспользуемый порт можно приспособить для подключения тахометра, резервного AC-CDI, отключения Flyback (в системах DC-CDI), диагностики, либо оставить, как резервный. Запараллеливать порты недопустимо.
- **в.25 МК (ADC2)** – вход для подключения ДАД (пока только ДАД, в следующих разработках будет комбинированным). Соединение выходит на разъём программирования **ICSP**. В программаторах этот штырёк обычно никуда не подключён. Но перед программированием убедиться лишний раз, что он не подключён к массе или питанию. Такая конфигурация сделана для того, чтобы задействовать и защитить от внешних воздействий разъём, не используемый в штатных условиях.
- **в.27 МК (PC4), в.28 МК (PC5)** – входы для переключения таблиц УОЗ и ДАД, активный уровень – лог «0».
 - PC4, PC5 = "1" – Таблица 1;
 - PC5 = "0" – Таблица 2;
 - PC4 = "0" – Таблица 3.

Переключение таблиц реализуется либо тумблером на 3 положения, либо «поворотником». После отладки его можно снять.

В будущем, PC4, PC5 = "0" планируется отдать под сервисный режим.

- **в.16 МК (PB2)** – вывод под LED диагностики (при включении питания однократно моргает, на холостых оборотах и заглушенном моторе – выключен, на рабочих – горит).
- **в.1 МК (RESET), в.17 МК (MOSI), в.18 МК (MISO), в.19 МК (SCK)** – выводы интерфейса последовательного программирования **ICSP**;
- **в.2 МК (PD0, RxD), в.3 МК (PD1, TxD)** – выводы интерфейса под **UART** (в программе режим работы с UART может быть отключён).

3. ОРГАНИЗАЦИЯ ОБРАБОТКИ ПРЕРЫВАНИЙ

В программе существует 7 видов прерываний (сторожевой пёс выключен):

- **INT0** – внешнее прерывание по Датчику 1 (Д1) – для определения скорости двигателя, перезапуска Т1 и старта отсчёта задержки.
Суть работы: по прохождению Д1 считываются показания Т1 (для определения оборотов), Т1 перезапускается, стартует отсчёт задержки и защита от помехи (таймер Т2, на пол оборота), устанавливаются флаги «Д1 обработан» и «МК занят». Если Т2 работает, значит, это помеха и прерывание не обрабатывается;
- **INT1** – внешнее прерывание по Датчику 2 (Д2) – для принудительной подачи искры во время первого оборота при запуске (обороты < 460 об/мин) - во это время ещё не определена скорость двигателя. Служит для исключения отдачи при запуске двигателя.
Суть работы: в обычном режиме обработка прерывания Д2 запрещена. Во время пуска (обороты < 460) переполняется Т1, разрешающее обработку прерывания по Д2. После чего переполнение сбрасывается. При прохождении Д2 без задержки подаётся искра.
- **blok_d1** – прерывание по совпадению таймера Т2. По этому прерыванию Т2 останавливается и сбрасывается.
Суть работы: каждый оборот, по прохождению Д1 обрабатывается его прерывание, а затем запускается таймер Т2, при этом в него записывается число его импульсов за 1/2 предыдущего оборота, и он начинает отсчёт.
 Обороты могут плавать, но это отражается незначительно. Во время своей работы Т2 блокирует обработку прерываний с Д1. По окончании счёта (через 1/2 оборота), Т2 останавливается и сбрасывается, разрешая обработку прерываний по Д1.
 Таким образом, мы имеем **динамическую защиту от помех** на первые пол оборота мотора, когда имеют место помехи от искры. В оригинальном варианте оно называется **ослеплением** и по времени фиксировано.
 Данная защита является дополнительной и в полной мере от помех не защищает, особенно когда вокруг неблагоприятная электромагнитная обстановка.
- **timer1comp** – прерывание по сравнению таймера Т1 - для отсчёта задержки на искрообразование.
Суть работы: в Т1 записывается 2-байтное число, равное времени задержки искрообразования. При совпадении счётчика Т1 с этим числом – идёт обработка этого прерывания с последующим сигналом на искрообразование (**искра с задержкой**).
 Для правильной обработки прерывания введена его блокировка. После записи задержки в Т1, разрешается обработка этого прерывания, а после отработки – опять запрещается.
 В оригинальной версии задержка тоже 2-байтная, но младший байт незначащий (256 значений углов по 0,7°), в этом отличие.
- **timer1** – переполнение Т1, для определения, что двигатель остановлен (<460 об/мин).
Суть работы: при переполнении Т1 управление программой передаётся на ожидание Д2, по приходу которого искра подаётся без задержки. Переполнение при этом сбрасывается.
- **uart_rx** – квитирование ПК приёма информации, при этом ПК передаёт код квитирования **212**. При удачном приёме – разрешает передачу следующей информации;
- **adc_comp** – окончание замера АЦП, по прерыванию загружается регистр R4 и устанавливается флаг окончания замера.

4. РЕЖИМ РАБОТЫ ОБОРУДОВАНИЯ (ИНИЦИАЛИЗАЦИЯ)

- **Порты:** входы: PC4, PC5, PD0, PD2, PD3
 выходы: PB2, PC0, PD1, PD6
- **Регистры:** свободные регистры R14..R15, R22..R23 = 4 шт (2 ст.)
- **Таймеры:** Т1 – 16 бит, 500 кГц, пуск, прерывания по: совпадению, переполнению;
 Т2 – 8 бит, 4 кГц, старт-стоп, прерывания по совпадению (сброс).
- **АЦП:** вкл., опора – AVcc, выравнивание – слева, канал – ADC2, 125 кГц, прерыв. от компар.
- **UART:** Tx, Rx – разрешены, флаг опустошения регистра, асинхронный режим (/8),

прерывание по завершению приёма, чётность, стоп – 2, разрядов – 8, спад Tx (фронт Rx), 9600 Бод.

5. ОБЩЕЕ ОПИСАНИЕ РАБОТЫ

5.1. Включение питания

При подаче питания на схему происходит инициализация:

- Назначаются вектора прерываний;
- Инициализируется оборудование (п.4);
- Считываются из памяти параметры - минимальное значение ДАД (ДАД мин), максимальное значение ДАД (ДАД макс), тип датчика и характеристика ДАД;
- Рассчитывается Диапазон ДАД по датчику (Дд): $Дд = ДАД\ макс - ДАД\ мин$;
- Рассчитывается константа ДАД (Кд): $Кд = 4080 / Дд$;
- В зависимости от положения тумблера выбирается одна из 3 таблиц. При отсутствии тумблера выбирается таблица №1.

Инициализация закончена, МК переходит в свой основной рабочий цикл.

Проверяется, сработал ли Д1? Поскольку мотор даже не заведён, то Д1 не сработал. Уходим на проверку занятости МК. По старту МК занят, поэтому уходим на формирование искры.

Так как искра выключена, возвращаемся в главный цикл. Цикл крутится так до тех пор, пока не начнём заводиться. Таймер Т1 при этом переполняется.

5.2. Запуск двигателя

Где останавливается поршень при глушении? В китайских мопедах **Дельта-72 cc** (ход поршня - **41,5 мм**, длина шатуна – **121,3 мм**), **15°** соответствует недоходу поршня до ВМТ – **0,8мм**.

Соответственно, **30° - 3,2мм**.

Пересчитать под свои углы можно по программе **Недоход поршня - угол.xlsx**, на 1 стр. ветки.

Так вот, при глушении моего мопеда компрессией откидывает поршень далеко назад, за первую метку, тем более длина метки составляет **1/24** оборота. Шансов остановиться мотору в промежутке между метками практически нет. **4%**, и то, если нет компрессии. Практика это доказала. Итак, заводка мопеда начинается с прохода Д1.

По срабатыванию Д1 проверяется флаг переполнения Т1. При заводке он переполнен, поэтому программа изменяет следующие флаги:

- разрешения работы Д2 – устанавливает;
- обработки Д1 – сбрасывает;
- переполнения Т1 – сбрасывает.

Далее программа возвращается в начало главного цикла, ожидая Д2.

По срабатыванию Д2 включается искра на портах D6 и C0, сбрасывается флаг разрешения работы Д2. Программа уходит в начало главного цикла.

Поскольку флаг свободы МК сброшен (МК будет занят искрой), программа уходит на формирование длительности искры. Включается в работу счётчик главных циклов программы. Одно кольцо главного цикла, в данном случае, занимает **11** тактов МК, т.е. **2,75 мкс**. Для формирования длительности искры, например, **400 мкс** счётчик должен насчитать $400 / 2,75 = 145$ главных циклов. Несколько тактов МК используются в цикле одноразово (при входе и выходе) – ими можно пренебречь.

По окончании счёта (равенство значений счётчика длительности искры), порты D6 и C0 сбрасываются, счётчик циклов обнуляется. Флаги приходят в следующее состояние:

- МК свободен – устанавливается;
- Выбор таблиц и вакуума – сбрасывается.

Формировка искры при пуске закончена, МК освобожден.

5.3. Операции вне искры (расчёты)

В процессе дальнейшей работы программа крутится в главном цикле до тех пор, пока не встретит опять Д1. Происходит это так.

Поскольку Д1 не обработан и МК освободился, проверяется состояние флага выбора таблиц и вакуума. При первом проходе после искры этот флаг сброшен. Поэтому вызывается подпрограмма выбора таблиц тумблером. В зависимости от положения тумблера, в регистры записывается стартовый адрес выбранной таблицы. Возвращаемся из подпрограммы.

Идём в подпрограмму формирования ДАД. Стартуем замер АЦП по каналу ADC2 (повторно запуска не будет, работает постоянно). Если замер сделан, то обновляем регистр работы с ДАДом и сбрасываем флаг замера АЦП. При каждом выполнении замера его флаг устанавливается и значение записывается в R4.

Проверяем, не вышло ли значение за пределы **ДАД мин ...ДАД макс**. Если вышло, то ограничиваем. **ДАД макс** в Конфигураторе рассчитывается как **95%** от атмосферного. Там же значения напряжений с ДАД переводятся в число в диапазоне **0...255**.

Затем определяется отклонение сигнала ДАД от минимума в абсолютных единицах – **Д абс**:

$$\text{Д абс} = \text{ДАД} - \text{ДАД мин};$$

Далее нам надо определить какой % диапазона по датчику **Дд** занимает уровень сигнала ДАД.

$$\text{Д отн} = \frac{\text{Д абс} * 255}{\text{Дд}} = \frac{\text{Д абс} * 255 * 16}{16 * \text{Дд}} = \frac{\text{Д абс}}{16} * \frac{4080}{\text{Дд}} = \frac{\text{Д абс}}{16} * \text{Кд}$$

В окончательном виде получаем:

$$\text{Д отн} = \frac{\text{Д абс}}{16} * \text{Кд}$$

Где: **Кд** – константа ДАД, рассчитанная при инициализации, дополнительное умножение на 16 служит для увеличения точности константы.

Далее накладываем на ДАД характеристику. Если характеристика – **линейная**, то оставляем **Д отн** как есть. Если же характеристика – **квадратичная**, то возводим его в квадрат и, для сохранения разрядности – делим на **256** (отсекаем младший байт).

$$\text{Д отн} = \frac{\text{Д отн} * \text{Д отн}}{256}$$

Обращаю особое внимание, **Д отн** – это **оцифрованная характеристика Вашего датчика ДАД, выражающая степень нагрузки Вашего двигателя**.

Чипы на измерение вакуума в ДАД, термокомпенсированы и имеют линейную характеристику, обычно с точностью **0,5 %**.

Д отн лежит в пределах **0...255** и соответствует нагрузке **0...100%** с заданной характеристикой.

Затем, согласно выбранной таблицы (1...3) и оборотам Вашего двигателя (для первого оборота **ДАД=0**), из Flash-памяти извлекается 2-байтная задержка угла по ДАД – табличный диапазон ДАД (**Д табл**). Термин мой. Он характеризует **100%** загрузку Вашего двигателя при текущих оборотах.

Перемножаем её на относительный ДАД и получаем поправку (П):

$$\text{П} = \frac{\text{Д табл} * \text{Д отн}}{256}$$

Поправка вычисляется, исходя из прошлого оборота. Поскольку под нагрузкой обороты плавают незначительно, да и сигнал ДАД должен быть достаточно инерционным, этими факторами пренебрегаем.

Для исключения загрузки МК во время формирования задержки и искры, повторяю, расчёт ведётся в свободное время.

Запоминаем поправку в регистрах и устанавливаем флаг выбора таблиц и вакуума. Возвращаемся из подпрограммы.

Далее, если текущая скорость мотора выше **1700**, загорается диагностический **LED**, в противном случае – потухает. При отладке – крайне полезная вещь. Кроме всего прочего, при подаче питания, если всё исправно, он кратковременно 1 раз моргнёт. При зависаниях МК (бывает с UART на проводах) – мотор заглох, он постоянно горит. При имитации датчика, с проверкой вне мопеда – тоже помогает.

После этого вызывается UART (если выбрано в опциях). Выбор таблиц, расчёт поправки по ДАД и изменение состояния **LED** происходит сразу после искры и только ОДИН раз за ОБОРОТ мотора.

Затем программа уходит в начало главного цикла и всё повторяется.

5.4. Работа мотора после пуска

В главном цикле, после срабатывания Д1 и отсутствия переполнения Т1, в соответствии с:

- показаниями старшего байта Т1 (измеритель периода оборота);
- выбранной таблицей УОЗ (1...3);
- рассчитанной поправкой по ДАД за прошлый оборот

из памяти выбирается 2-байтная задержка на УОЗ (рассчитанная в Конфигураторе), прибавляется поправка ДАД (УОЗ, соответственно, уменьшается).

Затем полученная задержка сравнивается с **18 мкс (0,97° на 9000 об/мин)** – время на отработку цикла, прерывания и записи искры с запасом, минимально необходимо 13,5 мкс.

На низких оборотах (задержка выше) она записывается в регистр сравнения Т1. Флаги приходят в следующее состояние:

- задержка выбрана;
- Д1 не обработан.

Программа уходит в главный цикл. При совпадении Т1 с задержкой (если задержка выбрана – флаг), включается искра, флаг выбора задержки сбрасывается и программа возвращается в главный цикл. Время искры формируется, как описано выше.

Если же требуемая задержка < **18 мкс** (опоздали !!), то экстренно подаётся искра на оба порта, минуя счётчик. Программа опять уходит в главный цикл на формирование времени искры.

5.5. UART

Для диагностики неполадок и проверки углов в работе **Flash.#13** разработал программу **FUOZ_UART.exe**. Поскольку исходников на программу нет, и не предвидится, а своих мозгов мало, то попробуем прикрутить её к нашей программе.

Часть функций (длина метки, Д2, режим работы) не используется, вместо них использованы либо константы, либо приближённые расчёты, на график не влияет. Лишние коды – это подгонка параметров.

Как это работает? В главном цикле, за искрой (когда МК свободен) вызывается подпрограмма обмена по UART. В подпрограмме проверяется – разрешена ли передача? Если нет – то выход.

Если же передача разрешена, то продолжаем. Если буфер передачи пуст – то в ОЗУ копируется 11 байт передаваемых данных. Далее, если порт занят – возвращаемся в главный цикл. В противном случае, читаем из ОЗУ 1 байт и передаём. Выходим в главный цикл.

При ответе ПК о приёме байта (квитирует выдачей байта **212**), обработка прерывания по UART разрешает передачу следующего байта.