Speeduino Thoughts – Part Two

This project is taking me into uncharted territory. Making the decision to undertake this project was a big decision. Deciding what would be the best aftermarket ECU for my requirement and budget. Committing to it by purchasing the Speeduino unit consisting of the Speeduino interface board, VR conditioner, and extra items has firmed up the project as well.

These initial decisions are all good. The next step in the process has been to learn about the Speeduino project, and try to marry the Speeduino to the OEM CFI system.

It was mentioned early on that maybe a piggyback option would be acceptable, but I have ruled this out mainly because the operation of the 1985 GL1200 Limited Edition FI motorcycle is dependent on the ECU for most of the motorcycle operation including the travel computer and LCD dash. I won't be without a ride as I have a second motorcycle, so this will allow me to work on this project without worrying about the need to rush.

I am very familiar with the design of the CFI system on my 1985 GL1200 Limited Edition FI motorcycle. My familiarity was at the component level and the purpose of each within the CFI system. Reviewing the abundant documentation that you can find on line has had me delve deeper into the workings of the CFI system on my motorcycle.

I realize that it is no longer acceptable to just understand the CFI system from a component standpoint, but I need to understand how each component relates to the ECU. I need to know what the input/outputs are, and how each is sent and received. It has been necessary to look at the OEM CFI system design and match it against the aftermarket ECU to determine what can be used.

There is virtually no information on how the OEM ECU is programmed, and how it relates to the CFI system. This makes it a challenge to marry up the various components to the aftermarket ECU.

Any ECU you choose will need to be configured to your requirement. This entails knowing the specifications for the components you want to use be they as installed by the OEM, or new aftermarket components.

There is a new language to learn, that of the ECU world. The ECU world has a set of acronyms that need to be learned and understood. I think it is necessary to list these acronyms and what each stands for. My work life was in our Navy and government worlds is filled with acronyms. The issue with acronyms is that after a while you accept the acronym as being the standard, whereas when asked what it stands for you generally come up blank and need to find the expanded version. This was made clear to me the other day when discussing acronyms. Two acronyms that were mentioned were scuba and radar. I know what scuba stands for, but radar, maybe in the past but since it is so commonplace, never paid much attention to it. I have since looked it up to refresh my memory – **RA**dio **D**etection **A**nd **R**anging. You can find the full name for an acronym on line, since there are so many from so many differing occupations and agencies, make sure you have the correct one.

You may find that the modern day ECU requires additional components that are not fitted on your vehicle of choice. An O2 sensor, narrow/wideband, is recommended, but is not installed on my motorcycle. The O2 sensor is used primarily at startup and to get a good idle. The Honda engineers compensated for this by using two cam sensors.

Now to back up a bit and discuss the OEM CFI system. It is quite complex and for the time, state of art regarding CFI systems. I would submit that this CFI system could still be considered an excellent system considering that it is the forerunner of Honda's PGM-FI system, and changes to the base system has been minimal.

I have found that the technical requirement(s) for this new ECU is quite detailed. It is necessary to know the specifications of the components so that these parameters can be entered into the tuning software, TunerStudio. It is not good enough to just pick out a sensor/component and install it. From learning this, I would think that this is part of the frustration people have when embarking on a project like this.

The initial configuration constraints dialogue box, Engine Constants, needs to be user defined as well. This dialogue box sets the stage for everything that follows.

Required Fuel, calculated value needs to be entered. This section also has a parameter input box for the injector pulse width in milliseconds. In the case of my motorcycle CFI system installation, I will have to try to discern the injector pulse width.

Control Algorithm – Speed Density or Alpha-N. Speed Density uses

Alpha-N uses throttle position (Alpha) and RPM (N) to calculate the amount of fuel to inject instead of the MAP and RPM in Speed Density mode. It has been mentioned that there is an "ITB" mode, just not mentioned as such. Will be looking into this aspect. Will set initial mode to Speed Density. It has also been mentioned that the Speeduino project does not have an ITB mode, but has something similar called

Fuel Squirts per Engine Cycle – This is the number of fuel injector squirts over a 720 degree engine rotation period. Will set initial setting at "2".

Injector Staging – configures the timing strategy for the injectors. Will use "alternating".

Engine Stroke – self explanatory – 4 stroke.

Number of cylinders – self explanatory – 4 cylinders.

Injector Port Type – apparently not use at this time but will be set to "port".

Number of Injectors – self explanatory – 4 injectors.

Engine Type – Should be "Even Fire" for my application.

Injector Layout – Specifies how the injectors are wired. Should be "Paired" for my application.

Board Layout – Should be "Speeduino v0.4".

MAP Sample Method – Should be "Cycle Average" for my application.

There are two MAP sensors (PB) connected in the system, one for the right cylinder bank and the other for the left. It has been mentioned that both of these may probably be used, one for the MAP requirement and the other for the barometric pressure requirement. The MAP sensors I have installed are not Honda OEM components, but from a later model Suzuki motorcycle. These MAP sensors have similar, extremely close specifications. As to using the two, will be looking into this as it may be very beneficial.

I have swapped out the crank sensor with one designed for a carbureted Honda Goldwing. The resistance values are different than those specified for the installed crank (Ns) sensor. The crank (Ns) sensor resistance is supposed to be 297 to 363 ohms. The Sensors I have installed presently have a resistance reading of 324 ohms so all should be good. I have come to realize that installing replacement components in a CFI system such as that on my 1985 GL1200 Limited Edition FI motorcycle, I need to be a little less cavalier and ensure that I get components that will work and are as close as possible to the OEM component/part.

From my research, I have found that the coolant and air temperature sensors should not be swapped out with similar sensors that have the same thread profile. The new ECUs require the specifications of these sensors so that the software/firmware can be tuned to properly use these. Since there is no information available from Honda, or about the ECU on the Honda Goldwing, or the previous FI models, changing these sensors out with an aftermarket sensor could result in less than adequate engine performance.

The installed OEM IAT and CLT sensors will have to be tested for temperature resistance. Will do this during the fall maintenance period. Will be testing these sensors this fall while doing engine work. I will be trying to get three distinct resistance numbers for each to input into the tuning software.

Starting with the crank (Ns) sensor, the trigger wheel is of importance. Both the Megasquirt family (will only be mentioning the Megasquirt family on rare occasions) of ECUs and the Speeduino project can use the 8 tooth trigger wheel that is installed. In conjunction with this are the two cam sensors (Gr/Gl) used for injector timing. The issue here is that the Speeduino project can only use one cam sensor. Not a big issue, but one that must be considered. If you are not going to use sequential fuel injection and ignition, there is no need to use the cam sensors, you only need to use the a crank sensor and trigger wheel, but it must be a trigger wheel with a missing tooth. Since the crank trigger wheel on my motorcycle is not a missing tooth trigger wheel, I will use one cam sensor to provide the ECU with a location pulse.

The OEM sensors used are variable reluctance (VR) sensors. These work well, but the output signal is a sine wave. The ECU requires a digital signal, that being a square wave that is either high – DC volts applied, or low – no DC volts applied. To accommodate the OEM installed VR sensors, the output signal needs to be conditioned/converted to a digital, square wave signal. I suspect that this is done in the OEM ECU, but cannot confirm how or what is being used.

You can change out the VR sensors for Hall effect sensor(s) that output a digital, square wave signal. Issue here is space constraint, sourcing an acceptable alternative, and cost. There is a wealth of information regarding suitable alternatives, but the most cost effective solution is to use a VR conditioner board.

For my application I will be using the dual wheel trigger setup. This is using a primary multi-toothed wheel (no missing teeth) with a secondary single tooth wheel that gives off a single pulse. The primary multi-toothed trigger wheel must divide evenly into 360 when using crank sensor as the primary, or 720 if using the cam sensor as primary. The single tooth cam wheel (single pulse) is used to provide location information to the ECU, and needs to be used for the Dual Wheel mode.

In the Trigger Settings dialogue box, the Dual Wheel is found in the Trigger Pattern parameter. Each Trigger Pattern mode will provide for various options within the dialogue box related to the required Trigger Pattern Mode.

Using the dual wheel setup will assist in changing from wasted spark/batch injection to sequential engine operation if the want/need arises.

The next issue I have had to study, and come to grips with is the infamous tooth #1. This is defined as the first tooth on the primary trigger wheel after the pulse on the secondary, single tooth (single pulse). I have read the documentation in the Speeduino manual as well as the Megasquirt (both use the same methodology) and tried this procedure to figure out tooth #1 one for my motorcycle.

I used the GL cam sensor as the secondary single tooth wheel. I did find a #1 tooth in accordance with (IAW) the procedure and a trigger angle (in degrees) of 135. I had previously mentioned that the crank (Ns) trigger wheel is an 8 tooth wheel, and as such the degrees between each tooth is 45. I found that when tooth #1 passes the primary sensor, the trigger angle in degrees was 135.

This is a rather large number, and I expect it can be accommodated. From this exercise I understand why a 12 tooth trigger wheel is recommended as a minimum as it reduces the trigger angle number. I have been looking for a suitable multi-tooth alternative in the 24 tooth range. I feel this would be better suited for the new ECU application in that the Trigger angle would be significantly reduced, and the output pulses to the ECU would be more, providing more engine operating information resulting in a better engine tune.

The Trigger Settings setup page has a Skip Revolutions (cycles) constraint. This is the number of engine revolutions that will be skipped before the injectors and coils are fired on start.

The next section of the Trigger Settings constraints page requires configuration regarding Trigger Edge – leading (rising) or trailing (falling). This needs to be determined and is from the primary input, in my case the crank sensor.

The secondary trigger edge – cam input – is the same as for the primary trigger edge, leading/trailing.

Do not fully understand this concept, but will before I install the new ECU.

Understanding the timing and coil dwell is next on my list of items to fully understand.

Ignition timing is well documented in the Honda service manuals. There are two timing marks, one with a “T” designation, and the second with an “F” designation. The “T” designation is for static timing such as when you do engine work that requires you to set initial timing prior to an engine start. The “F” designation is used to time the engine when it is operating. This is all done with a timing light, and if you are fortunate to have a timing light that has an engine degrees advance feature, you can determine what the timing advance is with the engine at operating temperature and specified idle.

You can also use this type of timing light to determine the timing advance when engine is first started. I did just this and found the timing advance at initial engine start to be approximately 25 degrees BTDC.

The second issue and not quite as simple to determine is coil dwell. This is the time the coil is allowed to charge before discharging through the spark plug. I had to research this to get an understanding of the concept as it has been many years since I have had to consider this word, yet alone try to determine what the dwell for a project such as this. Fortunately, my father was a mechanic and had a lot of this test equipment (Snap On tools was his friend). I am in the process of having my brother locate my father's tach/dwell meter and send it to me.

The dwell settings required for the tuning software are cranking dwell and running dwell, both in milliseconds (ms). Cranking is defined as when the engine is RPM is above “0”, but below the Cranking value as defined in the Cranking Settings. Running Dwell will be used when the engine is operating normally.

These figures are important because the tuning software applies a voltage correction such that when the nominal voltage is high dwell time is reduced, conversely when voltage is low, dwell time is increased. Dwell time for voltages between 12 VDC and 14.8 VDC (I'd prefer to see 14.5 VDC) the dwell time is kept at 100%. As voltage increases or decreases, dwell time is adjusted as mentioned.

The third Dwell Settings parameter is the Spark Duration setting. This is the time in milliseconds that the coil takes to fully discharge – opposite of dwell time that is required to fully charge the coil. This is a parameter I have to determine.

There is Overdwell Protection, an ON/OFF parameter. If this parameter is “ON”, you need to set the maximum dwell time, and the recommended maximum dwell time based on this parameter is recommended to be at least 3ms higher than the highest dwell time, cranking or running.

The Overdwell protection system operates independently of standard ignition schedules and monitors the time each ignition output has been active. If the active time exceeds the max dwell time as entered in the tuning software, the output signal to the coil(s) will be ended to prevent damage to the coil(s).

In conjunction with the timing and dwell settings is the Spark Settings requirement. This dialogue box contains the options as to how the ignition outputs will function, and which of the four ignition output channels are used and how.

The Spark Output mode is a drop down box and from this yous elect your specific requirement. The mode that I will be using at the onset is Wasted Spark using two injector channels.

I had mentioned above that the new ECU only requires a crank angle input depending on the requirement such as for wasted spark mode; however, for my application I require both crank and cam angle references because there are no missing teeth on the crank trigger wheel.

The Cranking Advance Angle in degrees is the timing advance that will be set to when cranking. This overrides all other timing advance modifiers during cranking.

A critical parameter in the Spark Settings dialogue box is the Spark Output Triggers. The choice is either Going High or Going Low. The majority of ignition setups use Going Low where the coil charges when the Spark output trigger is high and will fire through the spark plug when the signal goes low.

Now on to the injectors. There are high impedance injectors (resistance greater than 8 ohms) and low impedance injectors (resistance less than 3 ohms) but can be the same flow rate, but the injectors installed in my motorcycle engine are low impedance. The difference between the two when used is that the high impedance injectors can be installed and used immediately without any other system compensation. Low impedance injectors require a resistor in series with each injector so that the current flow is reduced to prevent the injector(s) from receiving an over current and "burning" up.

The installed OEM injectors, being low impedance, already have resistors in series. The injectors are connected so that these fire in pairs, batch injection. The firing sequence is 1-3-2-4. Injectors 1-3 and 2-4 are connected together and inject at the same time. The resistor bank has a 3 ohm resistors per set of injectors. This should be adequate for the new ECU installation.

You have to enter the injector characteristics in TunerStudio. The parameter – Injector Open Time, the time the injector takes to open completely once triggered, plus the time necessary to close. This is specific to each injector type and version.

Injector latency is what you are looking for and is the amount of time it takes for an injector to fully open once it receives its signal to fire. It is usually measured in milliseconds or microseconds and is an important parameter when tuning. This is also called "dead" and "lag" time. This parameter is generally provided by the injector manufacturer, but since Honda is not forthcoming regarding this information, nor have I found it during my searches on line, a systematic trial and error approach is probably required. There are several Goldwing forum threads that a person can peruse to get started in the right direction. It is a parameter that can be measured providing you have the proper test gear. I will be trying to determine this parameter with an oscilloscope.

The tuning software requires an injector flow rate for tuning. This may be in lbs/hr or cc/min. I have had the injectors on my motorcycle engine cleaned and flow tested. At 40 PSI these injectors flow 65 cc/min. Using this parameter, the injectors flow 65cc/min. In lbs/hr this is approximately 6.5 lbs/hr using a conversion factor of 1 lb/hr = 10.23 cc/min. Using these values the combined fuel requirement in lbs/hr would be 25.4 lbs/hr. Have to reconcile this with an 85% duty cycle as well. I have read several Goldwing forum threads and most mention that the OEM injectors are quite over size, but according to my values, I do not know how this can be – stranger things have happened.

The Speeduino has the ability to control a secondary fuel stage for engines that have 2 injectors per cylinder. This is not applicable to my requirement and as such the Staging Enabled parameter in this dialogue box will be set to OFF.

The Honda fuel pump requirement is for a fuel pump to deliver a minimum of 630 cc/min (21.3 oz/min) at a pressure of 2.0 to 2.4 Kg/cm2 (28-34 PSI). This pressure may have to be increased with a new ECU and programming, but I will use these values initially.

The throttle position sensor (TPS) is a 3 wire TPS and is in essence a rheostat. The TPS uses a 5 VDC input signal. The Speeduino project can apparently use just about any 3 wire TPS as long as it uses a 5 VDC input signal. I have an aftermarket TPS installed, but this should not present any problems. The TPS would be calibrated as per the OEM service manual. The tuning software, TunerStudio, needs the tune calibrated so that the engine at idle reads "0" and with the throttle wide open "100".

The TPS is used in conjunction with Acceleration Enrichment. For the Acceleration Enrichment to operate properly. A TPS must be installed. Acceleration Enrichment adds extra fuel during transition periods following an increase in throttle. Acceleration Enrichment is similar to the function of the accelerator pump on a carbureted engine.

Acceleration Enrichment is based on the rate of change of the throttle position known as TPSdot where "dot" means delta over time. This is measured in % throttle opening per second.

My reading indicates that values of 100% per second, even 1000% per second are normal. 100% per second means the throttle is opened from 0% to 100% opening in 1 second. 1000% per second means the throttle is opened from 0% to 1000% in 0.1 second.

I understand the premise for TPSdot and how it interacts with the % increase in fuel. I expect there will be a similar dialogue box that will be used for when the throttle is already open to a specific percentage such as highway driving.

For Acceleration Enrichment to operate properly, not only is a TPS required, but the TPS signal should not be "noisy" - electrical signal chatter. This chatter could result in Acceleration Enrichment being triggered when not required. The best solution to this issue is to replace the TPS with a new TPS that has been bench tested for a "clean" signal output. The second best solution is to adjust the TPSdot Threshold parameter in the Acceleration Enrichment dialogue box by increasing this value in 5% increments, pausing between each to observe the results, repeat as necessary.

The Accel Time parameter in milliseconds is use to determine the time necessary for Acceleration Enrichment to be in force. This timing ensures a correct air fuel ratio (AFR) on acceleration. If this time is too short, Acceleration Enrichment will initially be adequate for the requirement, but the engine will briefly go lean. When this happens there is a need to have the Acceleration Enrichment time increased to ensure the engine does not go into a lean condition. Recommended Accel Time increase increments are in the range of 10 to 20 milliseconds.

The Rev Limits, is a spark based rev limit with hard and soft cuts. The Soft Rev Limit in RPM locks the timing at an absolute value to slow further acceleration. Should the RPM continue to increase to the Hard Rev Limit, also in RPM, ignition events will cease until the RPM drops below the Hard Rev Limit threshold.

The Rev Limiter dialogue box settings are user inputs. There are four user inputs, Soft Rev Limit in RPM, Soft Limit Absolute Timing in degrees, Soft Limit Max Time in seconds, and Hard Rev Limit in RPM.

Soft Rev Limit, in RPM, is the limit at which the soft cut ignition timing will be applied. It will remain in effect until the RPM falls below this threshold.

Soft Limit Absolute Timing in degrees is the timing value that will be applied and maintained at during the period when the Soft Rev Limit is applied.

Soft Limit Max Time is the time that the Soft Rev Limit will be applied, after which the Hard Rev Limit will be applied.

Hard Rev Limit, in RPM, is the threshold at which all ignition events will cease.

Inlet Air Temperature (IAT) Density, represents the change in oxygen density of the air being drawn into the air chamber as the temperature increases. I mention the air chamber because this is where the IAT sensor is located on my motorcycle. The base tune for this parameter is apparently suitable for most applications and as such, should be acceptable without change.

Staged Injection and the Flex Fuel requirements will not be used for my application.

This is my first cut at putting together an understanding of the OEM installed CFI system and the Speeduino project. I envision a lot more deliberation and comparison between the two before I am finished.