



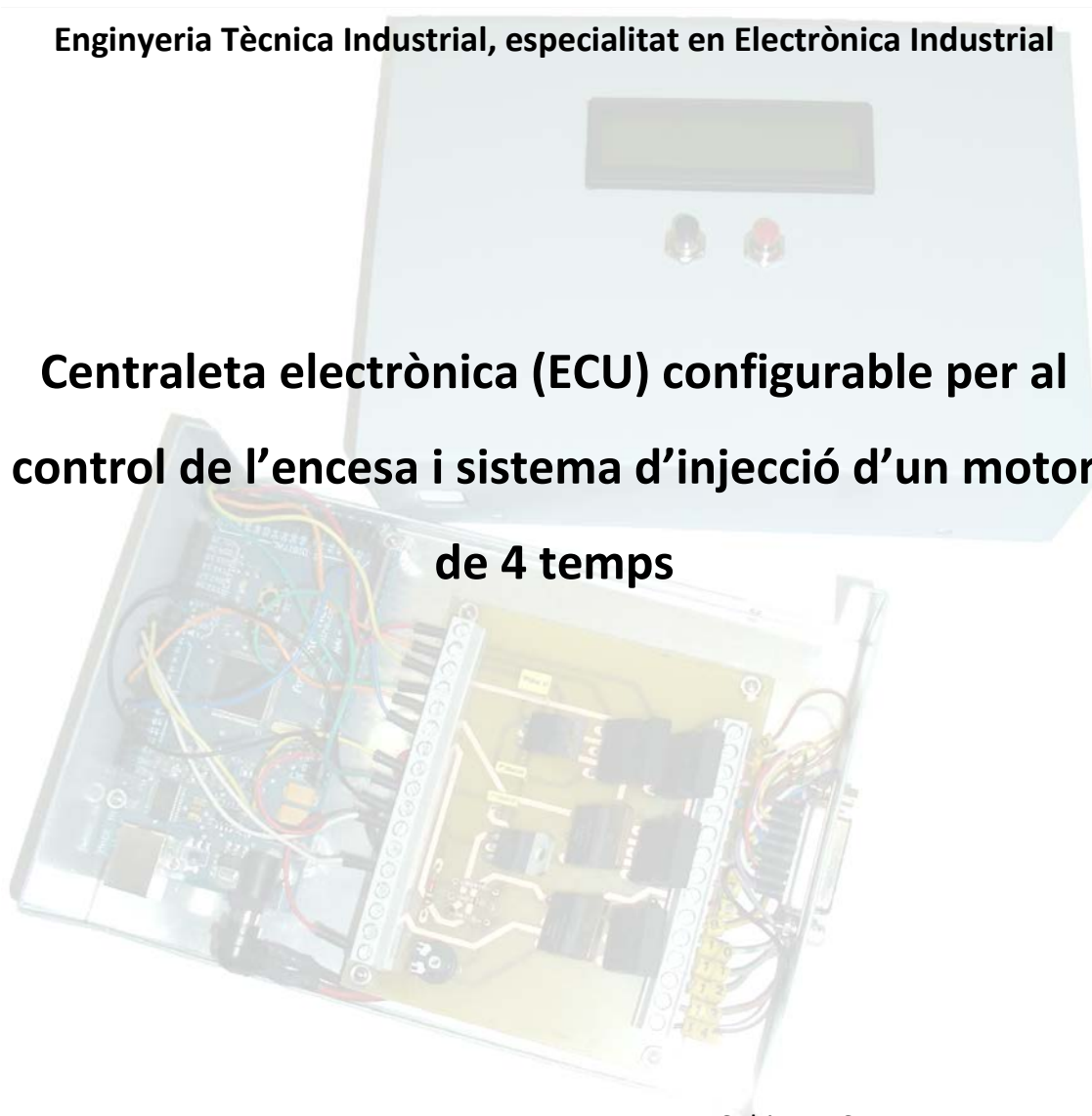
Escola Politècnica Superior
d'Enginyeria de Manresa

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Projecte Final de Carrera

Enginyeria Tècnica Industrial, especialitat en Electrònica Industrial

Centrala electrònica (ECU) configurable per al control de l'encesa i sistema d'injecció d'un motor de 4 temps



Marc Subirana Costa

Nº Registre: 2040

Director PFC: Antoni Escobet Canal

Manresa, Gener del 2011

Primer de tot volem agrair a totes aquelles persones que ens han ajudat a la realitzaci3 d'aquest projecte final de carrera: En primer lloc, a l'Antoni Escobet Canal del departament de Disseny i Programaci3 de Sistemes Electrònics (DiPSE), que ens ha guiat i ajudat en l'elaboraci3 i supervisi3 del projecte. Al Fermi Parcerisa pel seu assessorament tècnic i la seva experiència en el món de la competici3. Al Miquel d'Auto reparaci3 Puig per la seva confiança i l'informaci3 aportada.

Agrair als nostres amics la seva comprensi3 pel poc temps que els hem pogut dedicar aquests 3 mesos.

Finalment, a les nostres famílies, que ens han donat suport i ànims durant tot el projecte, en especial, l'Anna i la Natàlia, per recolzar-nos durant tota la carrera.

Sumari

Sumari	5
1. Glossari	8
2. Introducció	9
2.1 Objectius	9
2.2 Motivació	9
2.3 Antecedents	10
2.4 Estudi previ.....	11
3. Fonaments Teòrics	13
3.1 Història	13
3.2 Principi de funcionament dels motors de gasolina o cicle Otto.	14
3.2.1 Cicle de 4 temps	15
3.3. Estudi de les parts a controlar	16
3.3.1 Encesa convencional	16
3.3.2 Encesa electrònica (transistoritzada)	18
3.3.3 Sistemes d'injecció.....	19
3.3.4 Gestió del ralenti	21
3.4 RECURSOS UTILITZATS	23
4. Gestió de l'aplicació	27
4.1 Diagrama de blocs.....	27
4.2 Organigrama	28
4.3. Gestió del motor APQ (Volkswagen)	32
4.3.1 Determinació del règim de gir del motor (RPM)	32

4.3.1.1	Descripció del sistema de sincronisme	32
4.3.1.2	Lectura del règim de gir	333
4.3.1.3	Descripció del codi per la lectura de les RPM.....	33
4.3.1.4	Càlcul del règim de gir del motor	37
4.3.2	Control de l'encesa	39
4.3.2.1	Descripció del sistema d'encesa.....	39
4.3.2.2	Descripció del codi per l'encesa	41
4.3.2.3	Circuit de potència de l'encesa	43
4.3.3	Control de la Injecció	43
4.3.3.1	Descripció del sistema d' injecció	43
4.3.3.2	Descripció del codi per la injecció	47
4.3.3.3	Circuit de potència de l'injecció	50
4.3.4	Control del Ralentí	51
4.3.4.1	Descripció del sistema de ralentí	51
4.3.4.2	Descripció del codi pel ralentí	51
4.3.4.3	Circuit de potència del ralentí	52
4.4	Sistema de transferència	53
4.4.1	Control de flux	53
4.4.2	Format trama	53
4.4.3	Funcionament parada i espera.....	54
4.4.4	Enviament "EEPROM TRANSFER v1.0" → microcontrolador.....	56
4.4.5	Recepció i Reconeixement microcontrolador	56
4.4.6	Registre de la EEPROM (veure annex)	59
5.	Disseny del circuit imprès	60
5.1	Etapa d'alimentació del microcontrolador.....	60
5.2	Etapa d'alimentació pel control del ralentí.....	60
5.3	Adaptació senyal del sensorHall.....	60

5.4 Etapa de potència dels injectors, encesa i control del ralenti.....	60
5.5 Esquema elèctric	62
5.6 Circuit imprès	63
6. Pressuposts.....	65
7. Propostes de millora	67
8. Conclusions.....	68
9. Bibliografia.....	69

1. Glossari

RPM	Revolucions per minut
ECU	Unitat Control Motor (Engine Control Unit)
SensorHall	Sensor de Hall
PMS	Punt Mort Superior
PMI	Punt Mort Inferior
TPS	Sensor posici3 papallona (Throttle Position Sensor)
PWM	Modulaci3 per Ample de Pols (Pulse-width modulation)
TIMER	Temporitzador
E/S	Entrades / Sortides
μ C	Microcontrolador
PC	Ordinador

2. Introducció

2.1 Objectius

L'objectiu principal del projecte és el disseny i fabricació d'un sistema electrònic per controlar l'injecció i l'encesa d'un motor de 4 temps de gasolina. El sistema ha de ser totalment configurable i s'ha de poder fer a través d'un entorn entenedor per a qualsevol usuari.

Per garantir una vida útil i fiabilitat del motor a llarg termini, els fabricants limiten la potència dels motors entre un 25% i un 30%. Amb la unitat de control electrònica dissenyada, la finalitat és poder obtenir una part d'aquesta potència modificant els temps d'injecció.

2.2 Motivació

La motivació d'aquest projecte, sorgeix de les mancances detectades en alguns equips que participen en competicions automobilístiques de nivell provincial o comarcal a l'hora de configurar la part de control del motor. En aquestes competicions, on bàsicament s'utilitzen cotxes de sèrie modificats i homologats per al seu ús, molts dels equips opten per utilitzar un dispositiu mecànic (carburador) per la barreja d'aire-combustible substituint-lo pel sistema de control electrònic que porta inicialment el vehicle de sèrie. Això és degut a les limitacions de molts d'aquests dispositius de fabricació antiga i a la falta d'informació per poder fer una reprogramació en els dispositius més moderns. La falta de recursos econòmics també influeix en la utilització de dispositius mecànics.

Aquest, és un projecte que des del primer moment presenta la possibilitat d'unir una afició amb l'aprenentatge en un camp molt important en la nostra societat: l'automoció. A més, al tractar-se d'un projecte de desenvolupament real, suposa una gran experiència per la formació.

2.3 Antecedents

S'ha trobat diversos fabricants al mercat internacional de productes semblants al nostre. Alguns d'aquests són:

Haltech: empresa dedicada al sistema de gestió de motors que, entre d'altres, fan sistemes de control per la conversió personalitzada de carburació a injecció. Una de les ECU que utilitzen per fer una conversió d'aquest tipus és:

Platino Sport 1000:



Capaç de controlar la injecció seqüencial en motors de 4 cilindres o, semi seqüencial de fins a 8 cilindres. Aplicacions típiques:

- Control d'injecció en motors de combustible modificats.
- Conversió de carburació a injecció de combustible
- Disseny i finalitats de desenvolupament
- Ús educatiu de a les universitats i escoles tècniques

El preu aproximat del PS1000 és de €1,452.94EUR.

Greddy: empresa fundada i dirigida per corredors i aficionats amb l'objectiu principal de desenvolupar productes d'alt rendiment que augmentin la potència. Exemple de ECU adaptable:

e-Administración:



Sistema de gestió del motor programable que permet sincronitzar correctament amb el control del motor. Aquest sistema utilitza la majoria de cablejat de fàbrica de les ECU's japoneses i els sensors existents. El preu aproximat del kit és de 795 a 1295€.

2.4 Estudi previ

A continuaci3 es pot observar la planificaci3 de les tasques que s'han dut a terme per a la realitzaci3 del projecte i una breu descripci3 de cada una d'elles.

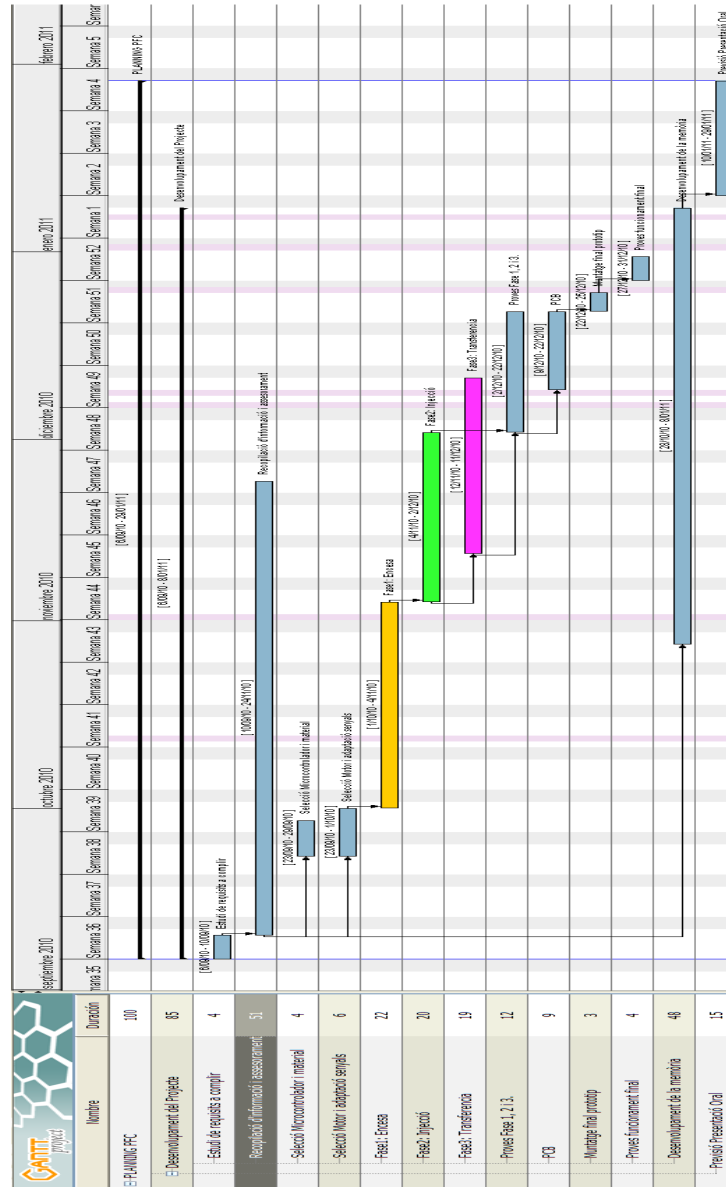


Figura 1: Diagrama de Gantt del calendari previst.

Desenvolupament del projecte: Es va fer un calendari del projecte per poder organitzar totes les tasques a realitzar i poder finalitzar-lo en el temps establert.

Estudi de requisits a complir: Es van identificar les principals parts a controlar, injecci3, encesa i transferència.

Recopilaci3 d'informaci3: És la part fonamental que permet veure la viabilitat del projecte d'una manera molt evolutiva al pas del dia a dia.

Selecci3 microcontrolador i material: És l'etapa on es selecciona el microcontrolador i elements varis per al control a realitzar.

Selecci3 motor: Es busca i es selecciona el motor gasolina de 4 temps a controlar i s'adapten les connexions necessàries.

Fase 1: És la part de l'encesa, on es fa l'adquisici3 de dades dels senyals originals, proves de simulaci3 i disseny de circuits de control, programaci3 del microcontrolador per el control d'encesa i proves pràctiques.

Fase 2: És la part de l' injecci3, on es fa l'adquisici3 de dades dels senyals originals, proves de simulaci3 i disseny de circuits de control, programaci3 del microcontrolador per el control d'injecci3 i proves pràctiques.

Fase 3: És la part de transferència, on es fa el disseny i programaci3 del software "EEPROM TRANSFER v1.0, programaci3 del microcontrolador per la transferència de noves dades i proves finals.

Proves Fase 1, 2 i 3: Es duen a terme les proves pràctiques del control d'encesa, injecci3 i transferència amb el motor.

PCB: Disseny, insolaci3 i muntatge de components a una placa de circuit imprès.

Muntatge final prototip.

Proves funcionament final.

Desenvolupament mem3ria.

Exposici3 oral.

3. Fonaments Teòrics

3.1 Història

Els orígens dels motors de combustió interna són molt remots. Els primers motors de combustió interna alternatius de gasolina que van assentar les bases dels que coneixem avui dia, van ser construïts quasi a la vegada per Karl Benz i Gottlieb Daimler. Els intents anteriors de motors de combustió interna no tenien la fase de compressió, aquests funcionaven amb una barreja d'aire i combustible aspirat o bufat durant la primera part del moviment del sistema. La distinció més significativa entre els motors de combustió interna moderns i els dissenys antics és la utilització de la fase de compressió

Considerats com a màquines completes i funcionals, i productores d'energia mecànica, hi ha alguns exemples de motors abans del segle XIX. A partir de la producció comercial de petroli a mitjans del segle XIX (1850) les millores i innovacions foren molt importants. A finals d'aquell segle hi havia una munió de varietats de motors usats en tota mena d'aplicacions.

En l'actualitat els motors de combustió interna, malgrat els problemes associats (crisis energètiques, dependència del petroli, contaminació de l'aire, augment dels nivells de CO₂,...), són encara imprescindibles i es fabriquen segons dissenys molt diferents i una gamma molt àmplia de potències que va des de pocs watts fins a milers de kW.

No va ser fins al 1980 que es va començar a utilitzar tècniques analògiques per mesurar i processar paràmetres d'aportació des del motor, llavors, s'utilitzava una taula emmagatzemada en un xip de memòria només de lectura digital per produir valors de sortida precomputats. Els sistemes posteriors computen aquestes sortides dinàmicament. El desavantatge de tals sistemes és que els valors precomputats són només òptims per a un motor idealitzat, nou. Com el motor es desgasta, el sistema no és capaç de compensar.

3.2 Principi de funcionament dels motors de gasolina o cicle Otto.

El motor de gasolina o motor Otto és un tipus de motor de combustió interna que utilitza l'explosió d'un combustible, provocada mitjançant una espurna, per expandir un gas i empènyer així un pistó que transmetrà l'energia. Aquest tipus de motors poden ser de 2 o 4 temps i el cicle termodinàmic utilitzat es coneix com a Cicle Otto.

Esquema d'un motor de 4 temps:

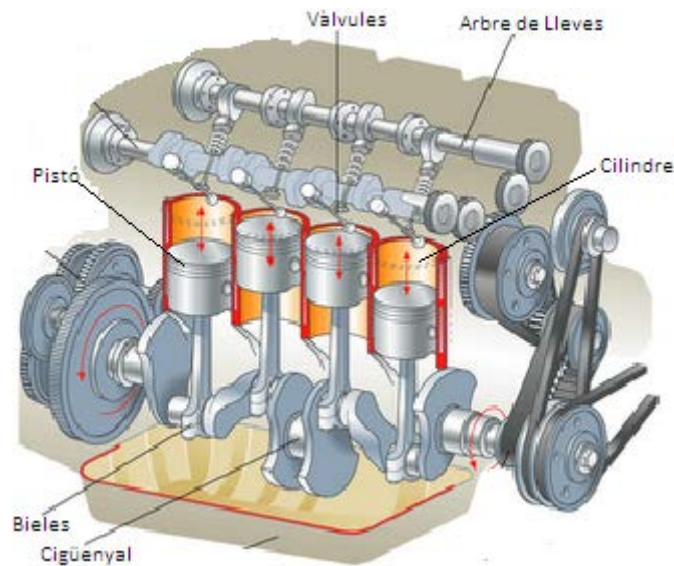


Figura 2: Parts fonamentals d'un motor de quatre temps

El motor de combustió interna de quatre temps és d'ignició mitjançant l'espurna produïda per una bugia. Les parts més importants d'aquest tipus de motors (figura 2) són les vàlvules, pistons, cigonyal, cilindres, bieles i bugies. S'ha de destacar que el motor està compost per una gran massa d'elements però que aquests són els més bàsics.

Els cilindres són els recipients pels quals es desplaça el pistó de forma lineal i la part superior del cilindre és la culata. El volum determinat entre la culata i el pistó representa la cambra de combustió, que és on es crema la barreja del combustible amb l'aire.

Les vàlvules són d'admissió i escapament. La vàlvula d'admissió permet l'entrada de la barreja de combustible i aire necessari per la combustió. La vàlvula d'escapament permet la sortida dels gasos. Les vàlvules estan connectades a l'eix del motor

mitjançant l'arbre de lleves i uns engranatges que permeten un sincronisme d'obertura i tancament adequat.

La combustió comença amb l'espurna de la bugia. L'expansió dels gasos empeny el pistó cap avall amb un moviment rectilini, on el cigonyal, mitjançant la biela, transfereix en forma de moviment rotatori a l'eix del motor.

3.2.1 Cicle de 4 temps

En aquest apartat veurem els quatre temps que realitza un motor de combustió interna. Durant un cicle podem observar diferents etapes com són l'admissió, compressió, explosió i escapament:

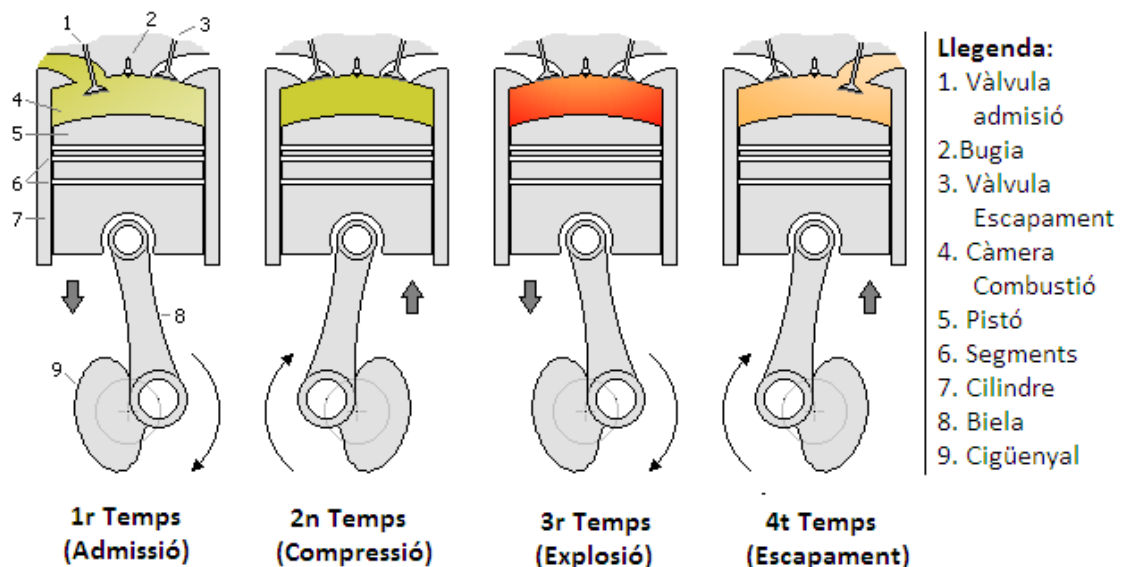


Figura 3: Cicles del motor de combustió interna

Primer temps o admissió: en aquesta fase el descens del pistó aspira la mescla d'aire i combustible. La vàlvula d'admissió es manté oberta, mentre que la de sortida està tancada. El primer temps el cigonyal dóna 180° i l'arbre de lleves dóna 90° , la vàlvula d'admissió es troba oberta i la seva cursa és descendent.

Segon temps o compressió: Al arribar al final de cursa inferior, la vàlvula d'admissió s'obre, comprimint-se el gas contingut a la cambra per l'ascens del pistó. Al 2n temps el cigonyal dóna 180° i l'arbre de lleves dóna 180° , a més les dues vàlvules es troben obertes i la seva cursa es descendent.

Tercer temps o explosi3: En arribar al punt mort superior (PMS) el gas ha assolit la pressi3 m3nima, en aquest moment salta l'espurna a la bugia provocant la inflamaci3 de la mescla. Una vegada iniciada la combusti3, aquesta progressa r3pidament incrementant la temperatura a l'interior del cilindre i expandint els gasos que empenyen el pist3. Aquesta 3s l'3nica fase en la qual s'obté treball f3sic. En aquest temps, el cigonyal d3na 180° mentre que l'arbre de lleves d3na 270°, les dues v3lvules es troben tancades i la seva cursa es descendent

Quart temps o escapament: En aquesta fase, el pist3 empeny en el seu moviment ascendent els gasos de la combusti3 a trav3s de la v3lvula d'escapament. En arribar al PMS, es tanca la v3lvula d'escapament i s'obre la d'admissi3 reiniciant el cicle. En aquest temps, el cigonyal, d3na 720° i l'arbre de lleves d3na 180° i la seva cursa 3s descendent.

3.3. Estudi de les parts a controlar

3.3.1 Encesa convencional

El sistema d'encesa 3s la part necess3ria capaç de produir l'ignici3 de la barreja de combustible i aire dins del cilindre en els motors de gasolina, coneguts tamb3 com motors d'encesa per espurna. Als motors de gasolina resulta necessari produir l'espurna entre dos electrodes separats en l'interior del cilindre en el moment just i amb la pot3ncia necess3ria per iniciar la combusti3. El sistema d'encesa b3sic est3 compost pels seg3ents components:

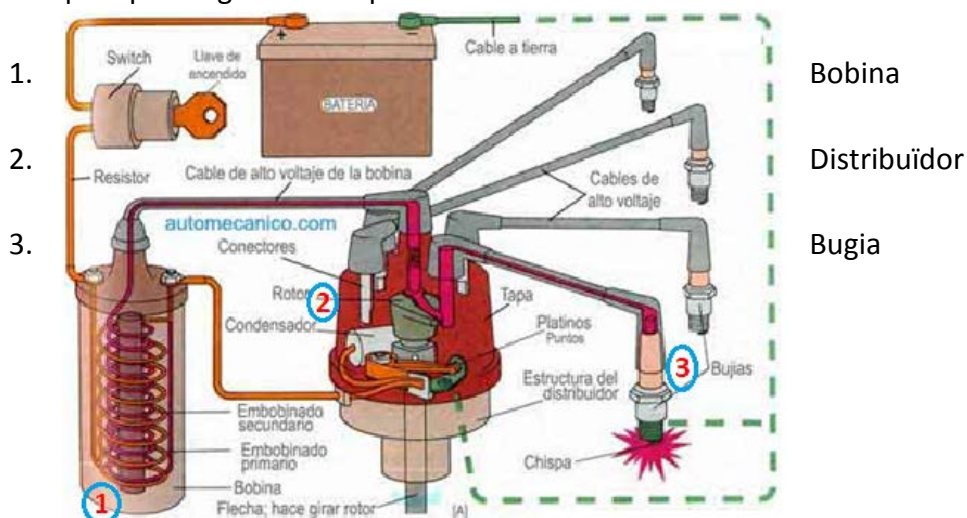


Figura 4: Sistema d'encesa controlat mec3nicament per platins.

El principi en el qual es basa el funcionament de **la bobina**, es pot considerar un dels grans descobriments de la ciència, ja que fins a l'actualitat es manté intacte. La bobina està composta per dos circuits: circuit primari i circuit secundari.

Quan s'activa la clau del contacte, la corrent (+) és connectada a la bobina, però perquè aquesta funcioni necessita el senyal (-), que en l'esquema anterior, els "platins" són els encarregats de donar el senyal (-) a través d'una commutació realitzada per un èmbol. Quan això succeeix, circula una corrent elèctrica pel circuit primari de la bobina, en el moment d'obrir-se el contacte, aquesta corrent s'interromp produint un canvi molt ràpid del valor del camp magnètic generat en el nucli del transformador, i per tant, la generació d'un alt voltatge per poc temps en el secundari. Com que la relació entre el número de voltes del primari i del secundari és molt alta, el voltatge generat al secundari és extremadament alt (15000V i 18000V) capaç de fer saltar l'espurna a la bugia.

El distribuïdor és un element fonamental del sistema de l'encesa del motor en els motors de gasolina. Aquest és l'encarregat d'enviar la corrent d'alt voltatge, procedent de la bobina, mitjançant un rotor giratori fins a les bugies de cada cilindre.

Per motius de fiabilitat en el funcionament i per eliminar el ròssec mecànic i problemes de derivacions i electroerosions entre el rotor i contacte de sortida d'alta, en l'actualitat, els vehicles de gasolina implementen un sistema d'encesa directa (DIS), en el qual les bobines produeixen l'espurna simultàniament en els dos cilindres que es troben en el Punt Mort Superior, aprofitant l'espurna només en un d'ells.

La bugia és l'element que produeix la ignició de la mescla de combustible i aire als cilindres. Aquesta té un electrode central aïllat que està connectat per un cable complementari aïllat a la bobina d'encesa. La bugia té dues funcions primàries:

Inflamar la mescla aire combustible.

Dissipar la calor generada a la cambra de combustió fins al sistema de refrigeració del motor.

Les bugies converteixen l'energia elèctrica generada per la bobina de l'encesa en un arc elèctric.

3.3.2 Encesa electrònica (transistoritzada)

Amb la introducció de l'electrònica als sistemes d'encesa convencionals, es millora la fiabilitat i rendiment del sistema evitant també el desgast dels components mecànics. En l'encesa convencional, el número d'espurnes subministrades està limitat a 18000 per minut, i en l'encesa electrònica a unes 21000.

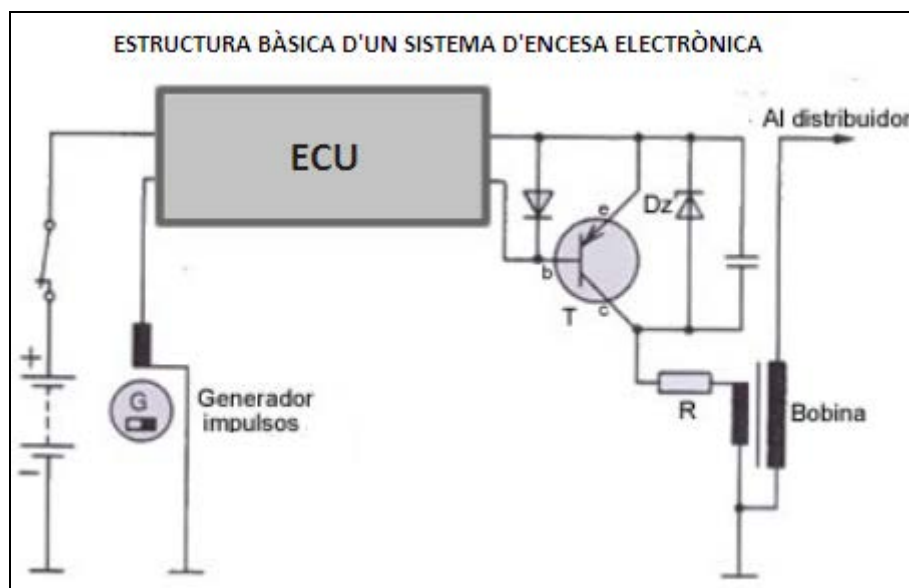


Figura 5: Esquema encesa transistoritzada.

A la figura anterior, es pot veure l'estructura bàsica d'un sistema d'encesa electrònica, on la corrent que travessa el primari de la bobina és controlada per un transistor (T). El transistor està controlat pels polsos que genera la ECU fent-lo funcionar com un interruptor, aconseguint així la commutació del circuit primari de la bobina.

El generador d'impulsos (G) és el senyal analògic llegit per la ECU que s'utilitza per realitzar el sincronisme de l'encesa. En l'instant en que un dels cilindres tingui que rebre l'espurna d'alta tensió, el generador G crea un impuls de tensió que és enviat al circuit electrònic, el qual aplica a la base del transistor, tallant la corrent del primari de la bobina i generant així al secundari de la bobina l'alta tensió que fa saltar l'espurna a la bugia. Passat aquest instant, la base del transistor es posa altre cop a massa repetint el cicle.

El **generador d'impulsos d'inducci3** és un dels més utilitzats en els sistemes d'encesa electrònics. Instal·lat en el mateix distribuïdor, genera un senyal elèctric que s'envia a la ECU per poder generar el tall de corrent a la bobina en el moment precís. El generador està constituït per una roda dentada d'acer magnètica que produeix, durant la seva rotaci3, una variaci3 del flux magnètic de l'iman permanent que indueix d'aquesta forma una tensi3 a la bobina que fa arribar a la ECU. La roda té tantes aspes com cilindres té el motor.

Exemple Generador d'impulsos:

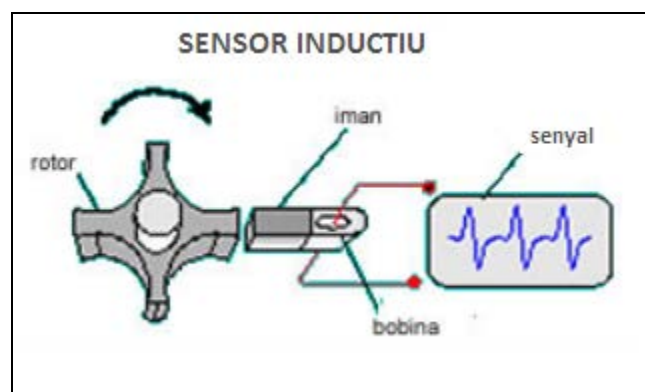


Figura 6: Sensor inductiu

3.3.3 Sistemes d'injecci3

El funcionament del motor de gasolina es basa en la combusti3 d'una barreja homogènia de combustible el més polvoritzada possible i aire. El rendiment del motor i les emissions contaminants depenen bàsicament de la composici3 de la barreja que s'introdueix al cilindre. Els sistemes d'injecci3 de gasolina pretenen aconseguir una dosificaci3 de combustible el més ajustada possible a les condicions de funcionament del motor. Tots els sistemes actuals que efectuen la injecci3 del combustible al col·lector d'admissi3, ho fan mitjançant uns injectors que en obertura, presenten sempre la mateixa secci3 de pas i, gràcies a la forma del forat de sortida, polvoritzen el combustible creant una bona barreja amb l'aire.

Podem aplicar tres classificacions pels sistemes actuals de gasolina:

Pel número d'injectors:

Monopunt: quan existeix un sol injector en una posició similar a la que tindria un carburador.

Multipunt: es disposa d'un injector per cada cilindre

Pel sincronisme de la injecció:

Simultània: es basa en injectar el combustible simultàniament a tots el injectors a una certa freqüència que coincideix amb l'obertura de la vàlvula d'un determinat cilindre, de manera que el combustible injectat s'acumula al col·lector d'admissió, mentre la vàlvula d'admissió està tancada. Al obrir-se la vàlvula, es produeix l'entrada de la barreja acumulada.

Seqüencial: l' injecció es produeix només en el moment d'obertura de la vàlvula d'admissió. L'injector regula la quantitat de combustible en funció del temps en que es troba obert, i la freqüència d'obertura depèn directament del règim de gir del motor.

Pel sistema de control i accionament dels injectors:

Mecànica: Control i accionament mecànic dels injectors (sistema en desús).

Mecànica – electrònica: Control electrònic i accionament mecànic dels injectors.

Electrònica: Control i accionament mecànic dels injectors.

Existeixen vàries formes d'injectar el combustible controlat **electrònicament**. A la següent figura es veuen les diferents configuracions en funció de les classificacions anteriors:

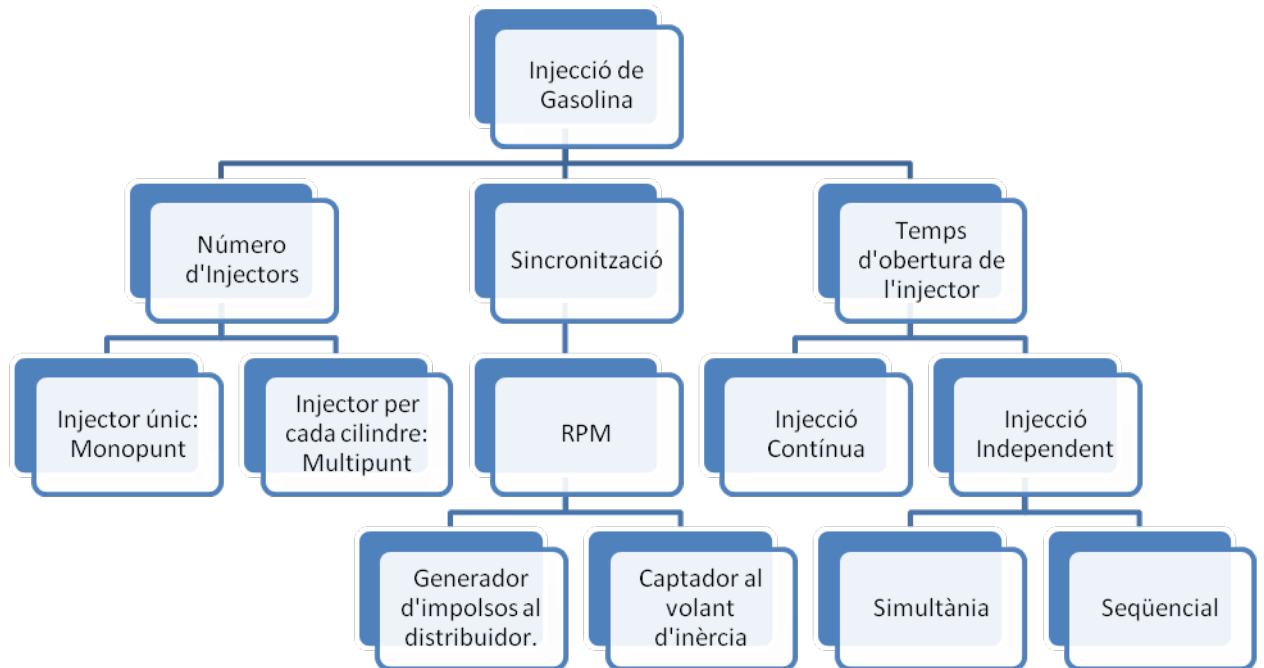


Figura 7: Diagrama sistemes d' injecció

3.3.4 Gestió del ralenti

Sense un sistema de control de ralenti no seria possible controlar les RPM's mínimes d'un motor. Aquest control és una part important, ja que qualsevol canvi en l'aplicació de la càrrega del motor, causa un caiguda de la velocitat del motor que pot provocar que aquest es pari. Per tan, els objectius principals del control de velocitat del ralenti són eliminar o minimitzar les caigudes de velocitat quan es deixa d'aplicar càrrega al motor. També s'utilitza per estabilitzar la velocitat del ralenti del motor durant arrancades en fred i després de condicions d'operació després d'un període d'escalfament.

En general, en la majoria de models de quasi tots els autom3bils es poden trobar quatre tipus de v3lvules per controlar el ralenti:

V3lvula de Papallona

Aquest sistema utilitza un petit motor de corrent cont3nua per controlar els graus d'obertura de la papallona.

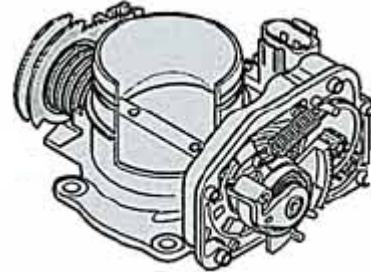


Figura 8: TPS

Solenoides Rotatius amb Control de Treball

Aquest sistema utilitza un solenoide rotatiu per generar l'estabilitzaci3 de la velocitat del ralenti. El control del desviament de l'aire, s'aconsegueix per mitj3 d'una v3lvula m3bil de gir rotatiu que bloqueja un "bypass".

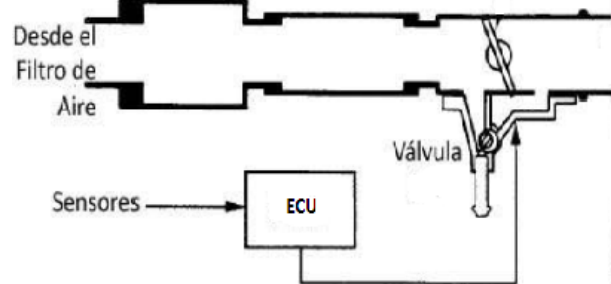


Figura 9: Solenoide rotatiu amb control de treball

V3lvula de control d'aire amb control de Treball

Aquest sistema s'encarrega de regular l'aire desviat al utilitzar una v3lvula de control d'aire (ACV) controlada per la ECU mitjançant senyals digitals de cicle el3ctric de treball.

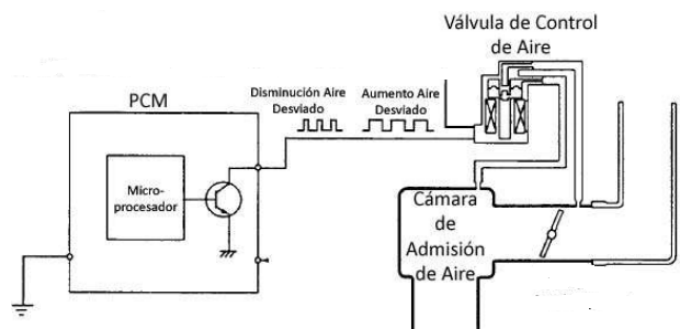


Figura 10: V3lvula de control d'aire amb control de treball

Vàlvula interruptora del buit ON/OFF

Aquest sistema utilitza una vàlvula switch del buit (VSV) normalment tancada per controlar una purga fixa d'aire en l'admissió. Aquest tipus de vàlvula és controlada per senyals de la ECU.

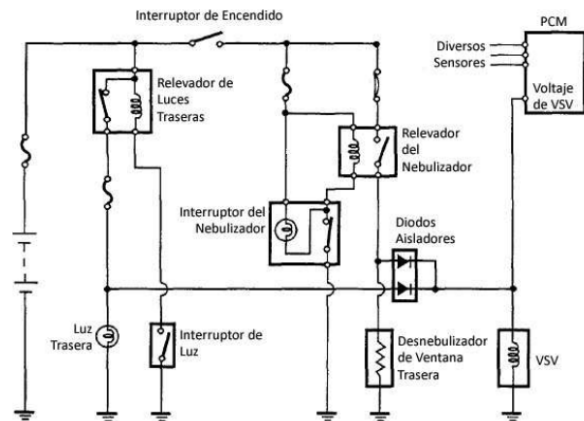


Figura 11: Vàlvula interruptora del buit

3.4 RECURSOS UTILITZATS

En el mercat actual es disposa d'una gran varietat de fabricants en microcontroladors com és el cas de Microchip, Atmel, Intel, Motorola entre altres, els quals ofereixen una àmplia gamma de productes per satisfer pràcticament qualsevol possible necessitat.

En el sector automobilístic és un dels més importants per el qual s'han dissenyat els microcontroladors, i la importància de fer una bona elecció, es basa en tota una sèrie de factors a tenir en compte, tal hi com es detalla a continuació.

El primer factor, és el què permet fer una clara classificació podent dividir-los pel nombre de bits, tenint així de 4, 8, 16 o 32 bits. Principalment els què ofereixen més prestacions són els de 16 i 32 bits, però alhora tenen certs inconvenients com ara el preu degut a la complexitat de fabricar-los. Per altre banda els microcontroladors de 4 bits resisteixen a desaparèixer, on es pot veure de forma clara en l'actualitat que la majoria de sistemes controlats per microcontroladors és de 8 bits. És per aquest motiu, que s'ha buscat un microcontrolador de 8 bits, degut a què les limitacions que ofereix per al prototip envers als de 16 i 32 bits són mínimes, també són fàcilment solucionables amb la programació d'aquest.

El segon factor, es basa en el nombre d'E/S a utilitzar, i havent fet una primera aproximació per al prototip d'entre 15 i 20 entrades digitals, 1 o 2 entrades analògiques i alguna sortida amb senyal PWM ha permès limitar els possibles candidats per la nostra unitat de control.

El tercer i quart factor, és el de la velocitat de rellotge i la capacitat de memòria que pot oferir el microcontrolador, per aquest motiu s'ha estudiat les diferents senyals a controlar i després d'observar que caldria fer treballar el microcontrolador amb temps de control de l'ordre de pocs mil·lisegons, sembla molt factible utilitzar-ne un que pugui oferir una velocitat de 16MHz. En referència a la capacitat, cal tenir en compte un cinquè factor, el nombre de timers, aquest va lligat amb el primer factor, el nombre de bits, degut que l'interès principal del prototip és controlar l'injecció i l'encesa, i la utilització d'un nombre de 4 a 6 temporitzadors amb resolució de fins a 16 bits repercuteix en l'ocupació d'espai del registre EEPROM, fent una previsió més que suficient d'entre 4kb i 8kb de capacitat d'aquesta memòria.

Després de fer una primera recerca d'informació pel què fa als diferents possibles candidats de Microchip com el PIC16F877 o el PIC18F4550, limitats per la velocitat de rellotge o el nombre limitat de timers per un costat, i per l'altre la poca experiència amb assemblador s'ha escollit un microcontrolador de la marca Atmel que permet ser programat en llenguatge C, amb llicència lliure i una gran plataforma al seu darrere, anomenada Arduino.

Pel què fa a la família Arduino, s'ha escollit l'Arduino MEGA1280 basada en xip Atmel d'entre l'amplia gamma com Duemilanove, Diecimilia, Nano, Mini, etc... per ser la més completa en nombre d'entrades i sortides digitals, entrades analògiques, registre de memòria eeprom més gran que la resta i gran compatibilitat en la resta d'accessoris que la plataforma ofereix.

Les especificacions de l'Arduino MEGA1280, han estat clau per assolir totes les necessitats. Tenint les següents característiques:

Voltatge entrada (limit): **6-20V**

Pins E/S digitals: **54** (14 proporciones sortida PWM)

Pins d'entrada analògics: **16**

Intensitat per pin: **40mA**

Intensitat al pin 3.3V: **50mA**

Memòria Flash **128kb** dels 4kb els fa servir el gestor d' arrancada (bootloader)

Timers: 6 (Timer0-2 de 8bits, Timer 1-3-4-5 fins a 16bits)

SRAM: **8kb**

EEPROM: **4kb**

Velocitat de rellotge: **16Mhz**

Un altre factor important ha estat el software, el qual utilitza un programari lliure i que permet fer la programació a un alt nivell com és en C, mentre que Microchip i altres fan servir assemblador.

I per últim, i no menys important, ha estat el del preu, ja que per 49€ s'ha adquirit una placa amb moltes possibilitats tenint al darrere una plataforma electrònica oberta per a la creació de prototips en software i hardware flexibles, fàcils de fer servir i amb molta informació i desenvolupament constant.

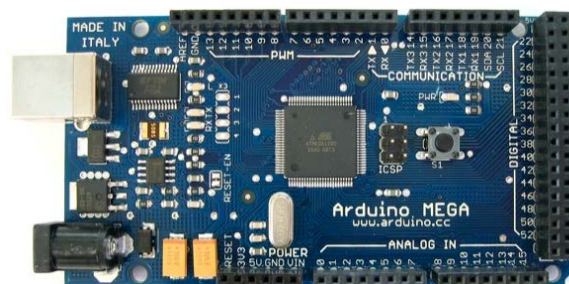


Figura 12: Placa Arduino Mega1280

L'entorn de programació del microcontrolador s'ha utilitzat el què la pròpia plataforma Arduino ofereix l' anomenat "Arduino 0021" compatible en sistemes operatius basats en Windows, Linux i Mac OS X. Es tracta d'un entorn senzill, intuïtiu i gratuït amb una gran comunitat al seu darrere, i amb gran quantitat d'exemples i biblioteques, tot i que és del tot possible programar amb altres programes com AVR C o C++.



Figura 13: Logotip programari Arduino

L'entorn triat per a la programació d' interfície gràfica amb l'usuari, amb el nom de "EEPROM TRANSFER v1.0", ha estat elaborat amb Labwindows CVI 7.1 que ofereix la marca National Instruments. És un programari basat en codi C capaç de generar codi de forma senzilla i eficaç d'una forma més gràfica i didàctica, i del qual ja se'n tenia experiència al llarg de la carrera, envers altres possibles com haurien pogut ser Visual Studio, Dev C++, etc.



Figura 14: Logotip programari Labwindows

L'entorn utilitzat per a realitzar el disseny del circuit ha estat l'EAGLE 5.10.0. El projecte s'ha desenvolupat amb aquest programa perquè és una eina molt eficaç, de senzilla utilització i intuïtiu alhora d'aprendre'n el seu funcionament.



Figura 15: Logotip programari Eagle

4. Gestió de l'aplicació

La gestió de l'aplicació es compon per un sistema funcional que permet la seva utilització a través de dos polsadors, un de selecció i l'altre de canvi de mode, podent visualitzar per una pantalla LCD el mode en què es troba, tal hi com es pot apreciar als punts a continuació.

4.1 Diagrama de blocs

A continuació es mostra el diagrama de blocs que permet veure de forma senzilla l'estructura principal del programa i en el següent punt de forma més ampliada.

Tal hi com es pot observar, hi ha 3 modes de funcionament bàsics:

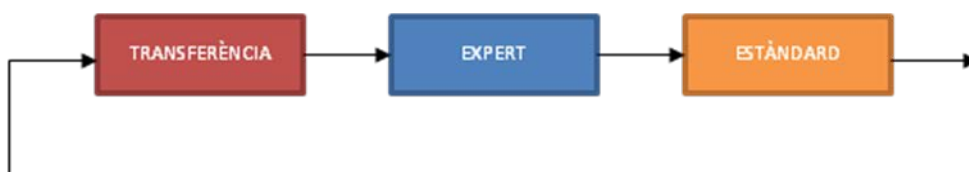


Figura 16: Diagrama blocs

Transferència

Aquest mode permet establir la comunicació amb el software de transferència per fer l'intercanvi de paràmetres de la EEPROM del microcontrolador.

Expert

Aquest mode permet fer funcionar el motor amb valors d'injecció modificats.

Estàndard

Aquest mode permet fer funcionar el motor amb valors d'injecció estàndards i que en cap cas es poden modificar.

4.2 Organigrama

Organigrama del programa i una breu explicació a continuació.

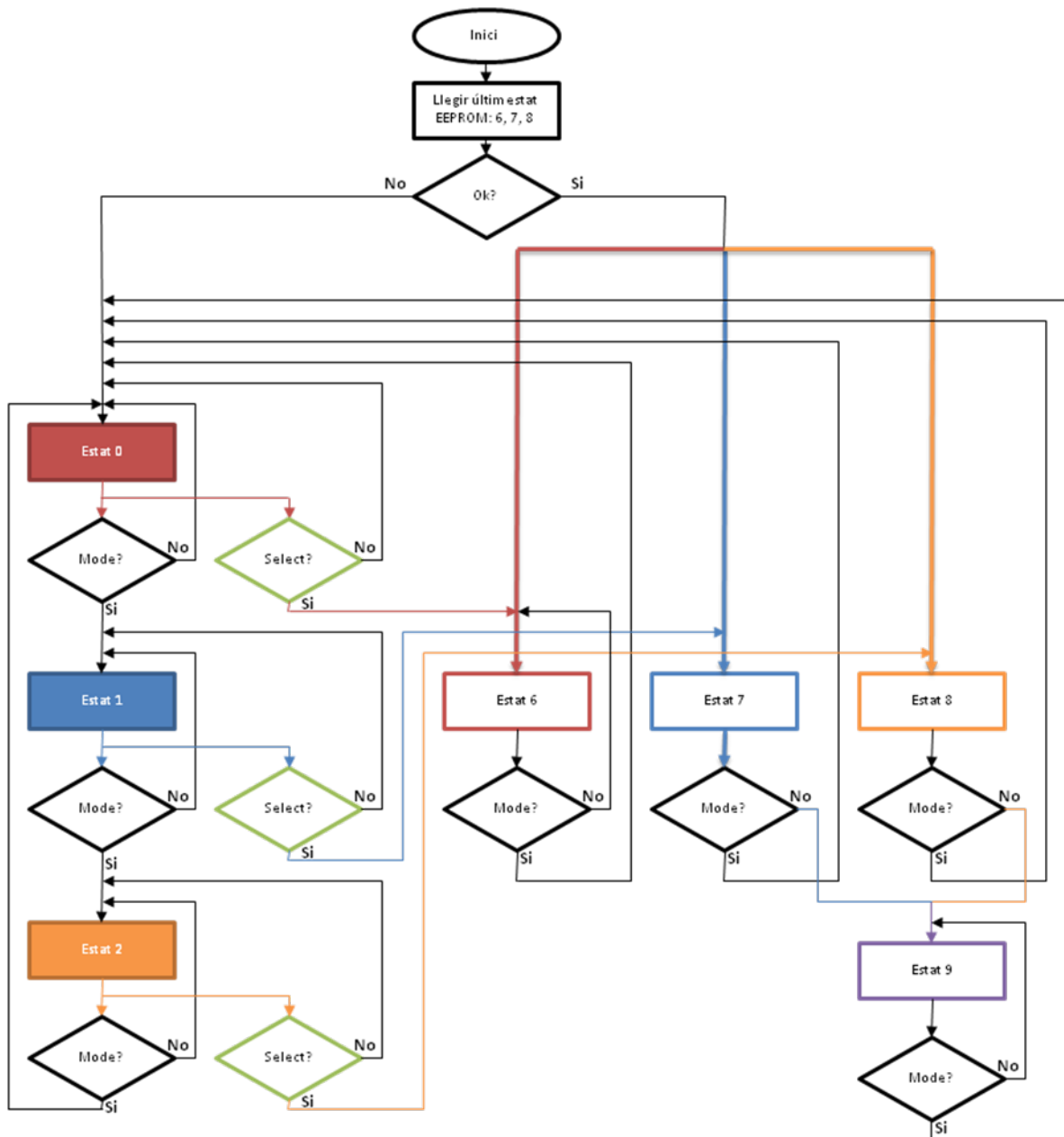


Figura 17: Organigrama

Definició dels Estats:

Estat 0 = Menú on es pot seleccionar estat de transferència.

Estat 1 = Menú on es pot seleccionar valors EXPERTS d'injecció.

Estat 2 = Menú on es pot seleccionar valors STANDARD d'injecció.

Estat 3, 4 i 5 = Reservats possibles futures ampliacions.

Estat 6 = Estat on es permet transferència amb EEPROM TRANSFER v1.0

Estat 7 = Estat on es carreguen valors EXPERT de la EEPROM.

Estat 8 = Estat on es carreguen valors STANDARD de la EEPROM.

Estat 9 = Estat de funcionament de control motor.

Com es pot observar primer de tot s'inicialitza l'aplicació mirant l'últim estat en què es trobava el programa, aquest estat emmagatzemat a l'espai 4000 del registre de la memòria EEPROM del microcontrolador només pot ser el 6, 7 o el 8 (Registre memòria EEPROM detallat a l'annex). D'aquesta manera si es troba en l'estat 6, romandrà en aquest fins que s'hagi fet la transferència de nous valors d'injecció o s'hagi polsat el polsador "Mode" que permetrà canviar d'estat dirigint-se al 0, 1 i 2 a mesura que es va polsant el mateix polsador "Mode" de forma cíclica 0-1-2:



Figura 18: LCD Menú Selecció Mode transferència



Figura 19: LCD Menú Selecci3 Mode Expert



Figura 20: LCD Menú Selecci3 Mode Standard

Només s'accedeix dins dels estats 6, 7 i 8 prement el polsador "Select", amb el següent ordre:

Estat 0 → "Select" → Estat 6

Estat 1 → "Select" → Estat 7

Estat 2 → "Select" → Estat 8

L'estat 6 tal hi com s'ha comentat és l'estat de transferència, que a través del software creat, l'EEPROM TRANSFER v1.0, es permet l'adquisici3 de nous valors per l'injecci3. El software de transferència s'explica i es complementa en els següents punts.



Figura 21: LCD Menú transferència



Figura 22: LCD transferència completa

L'estat 7 permet fer la recàrrega dels valors que es poden modificar, és a dir els anomenats com a "EXPERT MODE" i un cop feta la recàrrega dels valors del registre 0 fins al 1557 de la EEPROM, automàticament es dirigeix a l'estat 9 de funcionament de control del motor.

Estat 7 → Estat 9



Figura 23: LCD càrrega valors Expert

L'estat 8 fa el mateix que el 7, però recarrega els valors anomenats com a "STANDARD MODE", aquests valors no són modificables i són ocupats al registre des del 2000 fins al 3557 de la EEPROM.

Estat 8 → Estat 9



Figura 24: LCD càrrega valors Standard

4.3. Gestió del motor APQ (Volkswagen)

4.3.1 Determinació del règim de gir del motor (RPM)

4.3.1.1 Descripció del sistema de sincronisme

Per al control d'un motor de cicle Otto de quatre temps s'ha de poder determinar el moment exacte de l'encesa i l'injecció. Per determinar aquest moment és imprescindible saber el règim de gir del motor.

Per tant, la primera lectura que s'ha fet del motor ha estat les RPM. En aquest tipus de motor s'utilitza el senyal d'un mateix sensor per realitzar el càlcul de les revolucions, la sincronització de l'encesa i la sincronització de l'injecció. Pel nostre motor en qüestió es tracta d'un sensor de Hall situat al distribuïdor:

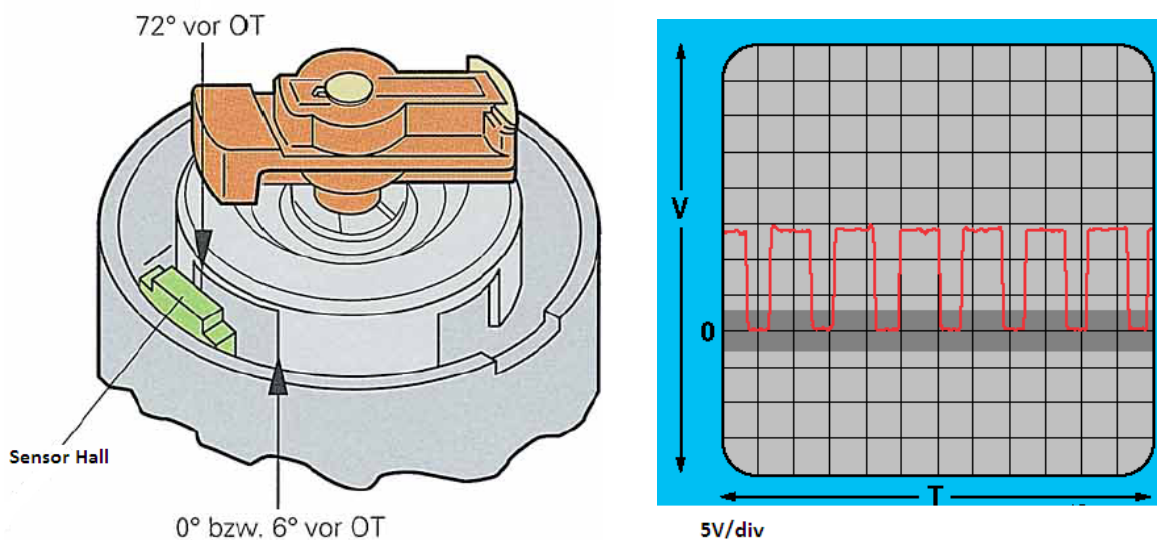


Figura 25: Distribuïdor motor APQ amb sensorHall.

La posició del sensorHall indica si un pistó del motor es mou cap al PMS o si es troba en temps de compressió o d'escapament. Al diagrama de la figura 25 es pot veure el senyal generat pel sensorHall, en que cada un dels polsos generats correspon a la posició esmentada anteriorment d'un pistó.

Per poder realitzar el càlcul de les revolucions per minut del motor s'ha tingut en compte que:

1 Revolució = 1 Cicle Motor (1 – 3 – 4 – 2)

1 Cicle Motor = 4 Pulsos (sensorHall)

4.3.1.2 Lectura del règim de gir

La tensió utilitzada en els automòbils és de 12 a 14,5V, la mateixa amb la que treballen la majoria de sensors. Podem veure en la figura 25 que l'amplitud del senyal generat pel sensor és de 14V. Tenint en compte que les E/S del microcontrolador operen a 5V i que cada pin pot rebre una intensitat màxima de 40mA, s'ha utilitzat el circuit de la figura 26 per poder realitzar la lectura del senyal. Aquesta lectura es fa a través d'una entrada digital.

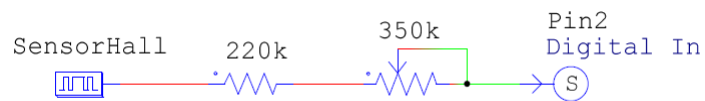


Figura 26: Circuit Condicionador de senyal SensorHall

4.3.1.3 Descripció del codi per la lectura de les RPM

La funció d'interrupcions externes combinada amb la funció d'interrupció per desbordament d'un dels temporitzadors del microcontrolador és la forma més precisa per poder fer la lectura de les RPM. D'aquesta manera s'assegura que per cada pols del sensor s'executin les lleis de control necessàries.

```
Int hallsensor = 2;
```

```
pinMode(hallsensor, INPUT);
```

//Es defineix la variable a la que s'associarà la lectura del sensor i el pin d'entrada digital del microcontrolador (2). Amb el "PinMode" es defineix la variable "hallsensor" com una entrada associada al pin que s'indica en la definició de la variable.

```
attachInterrupt(0, RPM, RISING);
```

//“attachInterrupt” especifica la funció que s’ha de cridar quan es produeix una interrupció externa. Es defineixen tres paràmetres de la interrupció:

- 0 → Número d’interrupció utilitzada.
- RPM → Nom de la funció d’interrupció.
- RISING → Mode en què la interrupció ha de ser disparada. En aquest cas és per flanc de pujada.

//Temporitzador 1 per Revoluciones:

```
TIMSK1=0x01;
TCCR1A=0x00;
TCNT1=0xE796;
TCCR1B=0x04;
```

Per la configuració dels temporitzadors del microcontrolador s’ha utilitzat un llenguatge diferent al llenguatge “C” utilitzat en la resta de programa. La configuració dels temporitzadors s’ha portat a terme amb el llenguatge ensamblador per la flexibilitat que ens dóna a l’hora de configurar-lo per cada situació. En el nostre cas definim quatre registres del temporitzador per poder configurar-lo:

```
TIMSK1=0x01;
```

TIMSK1 – Timer/Counter 1 Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
(0x6F)	–	–	ICIE1	–	OCIE1C	OCIE1B	OCIE1A	TOIE1	TIMSK1
Read/Write	R	R	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Bit 5 – ICIE1: Timer/Counter, Input Capture Interrupt Enable
- Bit 3 – OCIE1C: Timer/Counter, Output Compare C Match Interrupt Enable
- Bit 2 – OCIE1B: Timer/Counter, Output Compare B Match Interrupt Enable
- Bit 1 – OCIE1A: Timer/Counter, Output Compare A Match Interrupt Enable
- Bit 0 – TOIE1: Timer/Counter, Overflow Interrupt Enable

Posant els quatre bits baixos a 1 aconseguim habilitar el mode d'interrupció per desbordament (Bit 0). La resta de bits no habilitaran cap altre mode, ja que van amb combinació amb l'activació d'un bit d'un altre registre.

TCCR1A=0x00;

TCCR1A – Timer/Counter 1 Control Register A

Bit	7	6	5	4	3	2	1	0	
(0x80)	COM1A1 COM1A0 COM1B1 COM1B0 COM1C1 COM1C0 WGM11 WGM10								TCCR1A
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Table 17-3. Compare Output Mode, non-PWM

COMnA1 COMnB1 COMnC1	COMnA0 COMnB0 COMnC0	Description
0	0	Normal port operation, OCnA/OCnB/OCnC disconnected.
0	1	Toggle OCnA/OCnB/OCnC on compare match.
1	0	Clear OCnA/OCnB/OCnC on compare match (set output to low level).
1	1	Set OCnA/OCnB/OCnC on compare match (set output to high level).

En aquest registre hi ha varis modes de funcionament que en el nostre cas no s'han utilitzat. Els bits 6 i 7 són els que ens interessa posar a "0" per poder deixar constància del mode normal d'operació. Aquest mode és el que necessitem per poder treballar amb el temporitzador com a comptador sense haver de comparar amb el valor d'altres registres.

TCNT1=0xE796;

TCNT1H and TCNT1L – Timer/Counter 1

Bit	7	6	5	4	3	2	1	0	
(0x85)	TCNT1[15:8]								TCNT1H
(0x84)	TCNT1[7:0]								TCNT1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Aquests registres són els que porten el comptatge de polsos de rellotge del microcontrolador i el seu valor representa el temps que ha passat. El valor que introduïm als registres és el punt en el qual començarà el comptatge fins a arribar al valor màxim de "0xFFFF", aquí s'activarà la bandera de desbordament i el temporitzador tornarà a començar el comptatge des del valor carregat al registre.

TCCR1B=0x04;

TCCR1B – Timer/Counter 1 Control Register B

Bit (0x81)	7	6	5	4	3	2	1	0	
	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

CSn2	CSn1	CSn0	Description
0	0	0	No clock source. (Timer/Counter stopped)
0	0	1	clk _{IO} /1 (No prescaling)
0	1	0	clk _{IO} /8 (From prescaler)
0	1	1	clk _{IO} /64 (From prescaler)
1	0	0	clk _{IO} /256 (From prescaler)
1	0	1	clk _{IO} /1024 (From prescaler)
1	1	0	External clock source on Tn pin. Clock on falling edge
1	1	1	External clock source on Tn pin. Clock on rising edge

Amb els tres bits baixos d'aquest registre configurem el preescalat del temporitzador en funció del temps que volem temporitzar. Aquest preescalat pels temporitzadors de 16 bits que s'utilitzen funciona de la següent manera:

El microcontrolador utilitzat té una velocitat de rellotge de 16MHz, per tant, sense preescalar el temporitzador tindrem:

$$\text{Constant de temps mínim} = \frac{1}{16000000} = 62,5ns$$

$$2^{16} = 65536(\text{constants } _ \text{Temps})$$

$$\text{Constant de temps màxima} = 65536 * 62,5ns = 4,096ms$$

Amb els preescalats, l'únic que es fa és reduir la velocitat de rellotge i obtenir constants de temps més elevades. Per exemple, si es fa el preescalat de 1024, s'obindrà un rang de temporitzador amb constants de temps més altes que es podran seleccionar pel nostre temporitzador:

$$\text{Preescalat} = \frac{16000000}{1024} = 15625$$

$$\text{Constant Temps m\u00ednim} = \frac{1}{15625} = 64\mu s$$

$$2^{16} = 65536(\text{constan ts_Temps})$$

$$\text{Constant de temps m\u00e0xima} = 65536 * 64\mu s = 4,194304s$$

```
Void RPM(){
```

```
Pulses++;
```

//Es declara la funció d'interrupció externa en la que s'hi ubicar\u00e0 tot el codi de les lleis de control que s'executaran quan es cridi la interrupció. Per fer la lectura desitjada, cada vegada que entrem a la funció d'interrupció externa s'incrementa el valor de la variable "pulses".

//Declaraci\u00f3 de la funció d'interrupció per desbordament del Timer1:

```
ISR(TIMER1_OVF_vect){
```

```
TCNT1=0xE796;
```

```
rpm=(pulses * 150);
```

```
pulses = 0
```

//Cada vegada que el Timer 1 es desborda, el programa entra en aquesta rutina d'interrupció i executa les lleis de control relacionades amb la lectura del r\u00e8gim de gir del motor.

4.3.1.4 C\u00e0lcul del r\u00e8gim de gir del motor

La lectura del r\u00e8gim de gir del motor funciona de la seg\u00fcent forma:

Per cada senyal del sensorHall el programa entrar\u00e0 a la funció d'interrupció (RPM) i incrementar\u00e0 la variable "pulses". Aquesta variable ser\u00e0 incrementada durant el temps de comtatge del Timer1, el qual al desbordar-se, posar\u00e0 la variable "pulses" a "0" i tornar\u00e0 a comen\u00e7ar el comtatge deixant incrementar el valor de la variable "pulses" altre vegada. D'aquesta manera i fent la seg\u00fcent conversi\u00f3 s'obtindr\u00e0 el valor del r\u00e8gim de gir del motor:

$$\frac{\text{Polsos}}{0,1\text{Segons}} = \frac{60s}{1\text{min}} = \frac{1\text{revolució}}{4\text{polsos}} = 150\text{RPM}$$

$$\text{Pulses} * 150 = \text{RPM}$$

S'ha fet diferents proves amb la finalitat de trobar un temps fiable per fer la captura dels polsos del sensorHall. Utilitzant un rang de temps de 0,1 segons ha estat la forma més precisa per obtenir les revolucions. Per tan, el valor que s'ha de carregar al registre TCNT1 ha de ser el corresponent perquè el comptatge sigui de 0,1s:

$$\text{Pre-escalem} \rightarrow \frac{16000000}{256} = 62500\text{Hz}$$

$$\text{Const}_{\text{mín}} = \frac{1}{62500} = 16\mu\text{s}$$

$$\text{Const}_{\text{màx}} = 16\mu\text{s} * 65536 = 1,048576\text{s}$$

Es resol el valor que s'ha de carregar al registre amb el següent càlcul:

$$65536 \text{-----} 1,048576\text{s}$$

$$X \text{-----} 0,1\text{s}$$

$$X = 6250$$

Per tan, el valor a carregar:

$$\boxed{65536 - 6250 = 59286d \Rightarrow E796h}$$

4.3.2 Control de l'encesa

4.3.2.1 Descripci3 del sistema d'encesa

El senyal de sincronisme pel correcte funcionament de l'encesa 3s el sensorHall.

Per determinar el moment en que el microcontrolador ha de donar el senyal per generar l'espurna a la bugia, s'ha testejat el senyal del sensorHall juntament amb el de l'encesa. (figura 27).

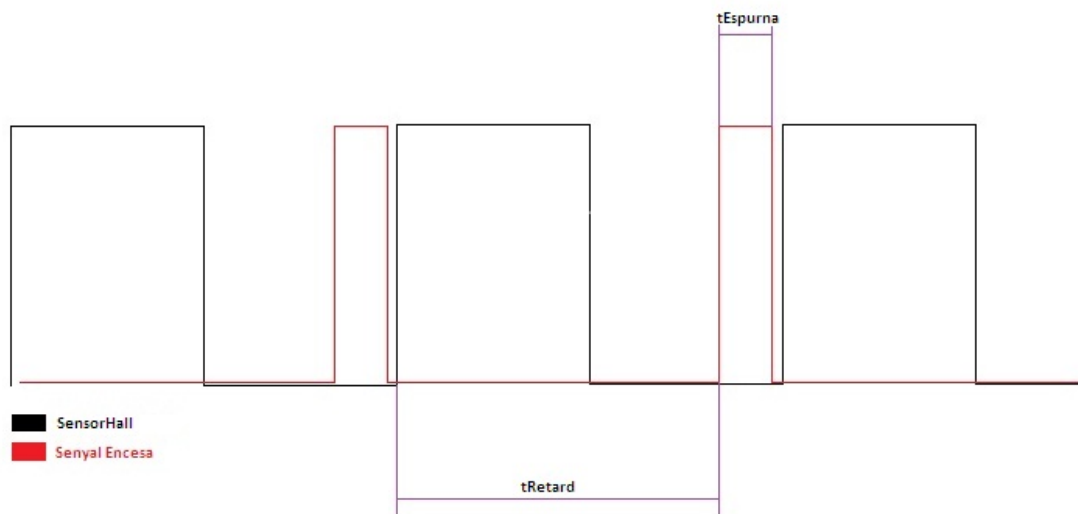


Figura 27: Senyal sensorHall i senyal d'encesa

S'han pres mostres de la posici3 del senyal de l'encesa en funci3 del senyal de sincronisme (sensorHall) a diferents revolucions per minut, obtenint la següent taula de temps:

RPM	TEMPS D'ENCESA (ms) (t_{Retard})	POLS ENCESA (ms) ($t_{Espurna}$)
5000	0,2	5
4500	0,9	5
4000	1,75	5
3500	3	5

3000	4,25	5
2500	5,5	5
2000	8	5
1500	13	5
1000	24	5
820	27	5

Amb els valors obtinguts s'ha creat una gràfica per poder obtenir una expressió el més aproximada possible a la funció d'encesa original:

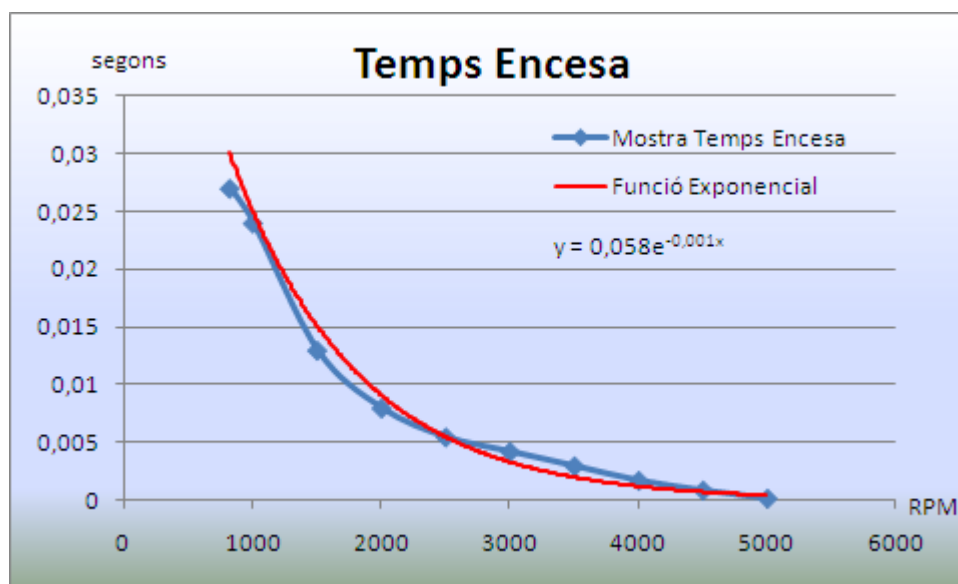


Figura 28: Gràfica funció Encesa

Per tant, la llei de control que s'aplicarà al nostre microcontrolador per determinar el moment de donar el senyal per activar l'encesa és la següent:

$$y = 0,058 \cdot e^{-0,001x}$$

4.3.2.2 Descripci3 del codi per l'encesa

Per portar a terme el funcionament de l'encesa s'ha necessitat 2 temporitzadors, ja que tal i com es pot veure a la figura 27 tenim dos temps que s'han d'executar, el "tRetard" i el "tEspurna". El temps "tRetard" serà el resultat d'aplicar l'expressi3 que s'ha determinat de la gràfica i el temps "tEspurna", sempre serà fix.

//S'utilitzen tres variables i dos temporitzadors definits al inici de programa per portar a terme el càlcul del "tRetard":

```
float IgnitionTime, PrelgnitionTime, IgnitionLoad;  
  
//Temporitzador 3 per encesa:  
  
    TCCR3A=0x00;  
  
    TCNT3=0x0000;  
  
    TCCR3B=0x02;  
  
//Temporitzador 4 per contar 5ms generaci3 espurna:  
  
    TCCR4A=0x00;  
  
    TCNT4=0xD8F0;  
  
    TCCR4B=0x02;
```

//Els temporitzadors s'han preescalat a 8, obtenint un valor mínim de 0,0005ms i un màxim de 32.768ms, rang suficient per poder aplicar el temps d'encesa, ja que com es pot comprovar a la taula de valors, el temps màxim és de 27ms i el més petit és de 0,2ms.

// Dintre del "programa principal de control", la funci3 de les variables és la següent:

```
IgnitionTime = 1000*(0.058*exp(-0.001*rpm));  
  
PrelgnitionTime = (IgnitionTime*65536)/32.768;  
  
IgnitionLoad = 65536-PrelgnitionTime;
```

//S'utilitza la variable "IgnitionTime" per aplicar la llei de control determinada de la funci3 de la gràfica i deixar-la en mil·lisegons. Amb la variable "PrelgnitionTime" i

“IgnitionLoad” s’acaba determinant el valor que s’ha de carregar al registre TCNTn del temporitzador utilitzat. (Veure exemple 4.3.1.4)

//S'utilitza la funció d'interrupció externa (veure funció i codi al 4.3.1.3) per actualitzar cada vegada el “tRetard”. A continuació es pot veure com el registre TCNT3 actualitza el valor d'“IgnitionLoad” a cada interrupció del sensorHall:

```
Void RPM(){  
    Pulses++;  
    TCNT3 = IgnitionLoad;
```

//Cada vegada que el registre TCNT3 obté un nou valor, el temporitzador comença el comptatge fins a desbordar-se, activant en aquest cas la rutina de servei d'interrupció del Timer3. Tot seguit, es desactiva la sortida digital 8 i es fa la càrrega del registre TCNT4.

```
ISR(TIMER3_OVF_vect){  
    digitalWrite(8, LOW);  
    TCNT4 = 0xD8F0;
```

//Al registre del comptatge del Timer4 s’hi carrega un valor fix, aquest correspon al temps “tEspurna” que sempre és de 5ms. Quan el Timer4 ha fet el comptatge dels 5ms, a través de la seva Rutina de servei d'Interrupció torna a deixar la sortida 8 en estat alt.

```
ISR(TIMER4_OVF_vect){  
    digitalWrite(8, HIGH);
```


4.3.2.3 Circuit de potència de l'encesa

El mòdul de potència està constituït físicament per una placa electrònica que rep a través d'un connector del cotxe els senyals necessaris per al seu control. Internament, la part de l'encesa és la següent:

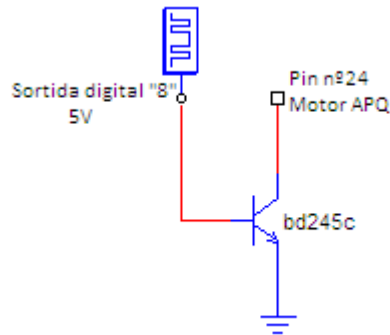


Figura 29: Esquema del funcionament de l'encesa

A l'hora de seleccionar el transistor s'ha tingut en compte el corrent que circula per la bobina que a 5000rpm és de 3A, i la seva tensió de 12V a 14,5V.

El senyal d'obertura i tancament en lògica TTL procedent del microcontrolador s'utilitza per excitar la porta d'un transistor de potència (BD245C), el qual veu excitada la seva porta i per tant permet el pas de la corrent entre col·lector i emissor.

4.3.3 Control de la Injecci3

4.3.3.1 Descripci3 del sistema d' injecci3

En els sistemes actuals s'utilitzen injectors governats electr3nicament. En aquest cas, s3n dispositius de "tot o res" ja que només tenen dos estats, oberts o tancats. Quan estan oberts permeten el pas del combustible, i quan estan tancats el bloquegen.

La ECU és l'encarregada de donar els impulsos elèctrics que governaran l'obertura dels injectors. El temps de duraci3 dels impulsos determina el temps en què l' injector estarà obert permetent el pas del combustible. La quantitat de combustible injectat serà proporcional al temps de duraci3 de l'impuls elèctric.

S'ha testejat el senyal dels injectors juntament amb el de la posici3 de la papallona (TPS) i les RPM per poder determinar el temps d'injecci3 en cada moment. S'ha extret la següent taula de valors:

TPS(v)	RPM	Temps Injecci3 (ms)	Temps injecci3 (ms) TPS = 1,8V
4,07	300	4,6	26
3,87	870	4,6	26
3,8	1320	4,1	26
3,75	1690	3,6	26
3,7	2040	3,3	26
3,65	2250	3,6	26
3,6	2490	3,8	26
3,55	2760	3,9	26
3,5	2940	4	26
3,45	3240	4,1	26
3,4	3450	4,2	26
3,35	3600	4,3	26
3,3	3800	4,4	26
3,25	4080	4,6	26
3,2	4200	4,7	26
3,15	4320	4,7	26
3,1	4500	4,8	26
3,05	4860	4,9	26
3	5200	5	26

Amb la mostra de valors anteriors s'ha dissenyat una taula de temps d'injecció. La taula ha estat creada a partir dels dos valors de temps d'injecció per cada valor d'RPM (veure taula anterior). D'aquesta forma, i en funció de la posició del TPS s'ha omplert la taula de forma lineal.

A la figura 30 i 31 es pot veure la cartografia o mapa d'injecció que utilitzarà el nostre microcontrolador.

Detall del mapa d'injecció dissenyat amb els temps d'injecció en mil·lisegons:

RPM	POSICIÓ PAPALLONA (TPS)					POSICIÓ PAPALLONA (TPS)				
	4	3,85	3,8	3,75	3,7	3,65	3,6	3,55	3,5	3,45
870	0,0036	0,00465	0,005	0,00535	0,0057	0,00605	0,0064	0,00675	0,0071	0,00745
1000	0,0026	0,003695	0,00406	0,004425	0,00479	0,005155	0,00552	0,005885	0,00625	0,006615
1320	0,0026	0,003695	0,00406	0,004425	0,00479	0,005155	0,00552	0,005885	0,00625	0,006615
1690	0,0017	0,00284	0,00322	0,0036	0,00398	0,00436	0,00474	0,00512	0,0055	0,00588
2040	0,0014	0,002555	0,00294	0,003325	0,00371	0,004095	0,00448	0,004865	0,00525	0,005635
2250	0,0007	0,001885	0,00228	0,002675	0,00307	0,003465	0,00386	0,004255	0,00465	0,005045
2490	0,0004	0,0016	0,002	0,0024	0,0028	0,0032	0,0036	0,004	0,0044	0,0048
2760	0,0001	0,001315	0,00172	0,002125	0,00253	0,002935	0,00334	0,003745	0,00415	0,004555
2940	0	0,00093	0,00134	0,00175	0,00216	0,00257	0,00298	0,00339	0,0038	0,00421
3240	0	0,000645	0,00106	0,001475	0,00189	0,002305	0,00272	0,003135	0,00355	0,003965
3450	0	0,00036	0,00078	0,0012	0,00162	0,00204	0,00246	0,00288	0,0033	0,00372
3600	0	7,5E-05	0,0005	0,000925	0,00135	0,001775	0,0022	0,002625	0,00305	0,003475
3800	0	0	0	0,000275	0,00071	0,001145	0,00158	0,002015	0,00245	0,002885
4080	0	0	0	1E-04	0,00054	0,00098	0,00142	0,00186	0,0023	0,00274
4200	0	0	0	0	0,00017	0,000615	0,00106	0,001505	0,00195	0,002395
4320	0	0	0	0	0	8,5E-05	0,00054	0,000995	0,00145	0,001905
4500	0	0	0	0	0	0	0	0,000385	0,00085	0,001315
4860	0	0	0	0	0	0	0	0,00013	0,0006	0,00107
5220	0	0	0	0	0	0	0	0	0,00035	0,000825

Figura 30: taula temps d'injecció en mil·lisegons

Detall del mapa d'injecció dissenyat amb els valors calculats per carregar al registre TCNT5 del Timer5:

RPM	POSICIÓ PAPALLONA (TPS)					POSICIÓ PAPALLONA (TPS)				
	4	3,85	3,8	3,75	3,7	3,65	3,6	3,55	3,5	3,45
870	58336	56236	55536	54836	54136	53436	52736	52036	51336	50636
1000	60336	58146	57416	56686	55956	55226	54496	53766	53036	52306
1320	60336	58146	57416	56686	55956	55226	54496	53766	53036	52306
1690	62136	59856	59096	58336	57576	56816	56056	55296	54536	53776
2040	62736	60426	59656	58886	58116	57346	56576	55806	55036	54266
2250	64136	61766	60976	60186	59396	58606	57816	57026	56236	55446
2490	64736	62336	61536	60736	59936	59136	58336	57536	56736	55936
2760	65336	62906	62096	61286	60476	59666	58856	58046	57236	56426
2940	65536	63676	62856	62036	61216	60396	59576	58756	57936	57116
3240	65536	64246	63416	62586	61756	60926	60096	59266	58436	57606
3450	65536	64816	63976	63136	62296	61456	60616	59776	58936	58096
3600	65536	65386	64536	63686	62836	61986	61136	60286	59436	58586
3800	65536	65536	65536	64986	64116	63246	62376	61506	60636	59766
4080	65536	65536	65536	65336	64456	63576	62696	61816	60936	60056
4200	65536	65536	65536	65536	65196	64306	63416	62526	61636	60746
4320	65536	65536	65536	65536	65536	65366	64456	63546	62636	61726
4500	65536	65536	65536	65536	65536	65536	65536	64766	63836	62906
4860	65536	65536	65536	65536	65536	65536	65536	65276	64336	63396
5220	65536	65536	65536	65536	65536	65536	65536	65536	64836	63886

Figura 31: Taula de temps d'injecció amb valors calculats per carregar al registre TCNT5

El senyal de sincronisme pel correcte funcionament de la injecció és el *sensorHall*.

Per determinar el moment en que el microcontrolador ha de donar el senyal d'injectar la gasolina, s'ha testejat el senyal del sensorHall juntament amb el dels 4 injectors:

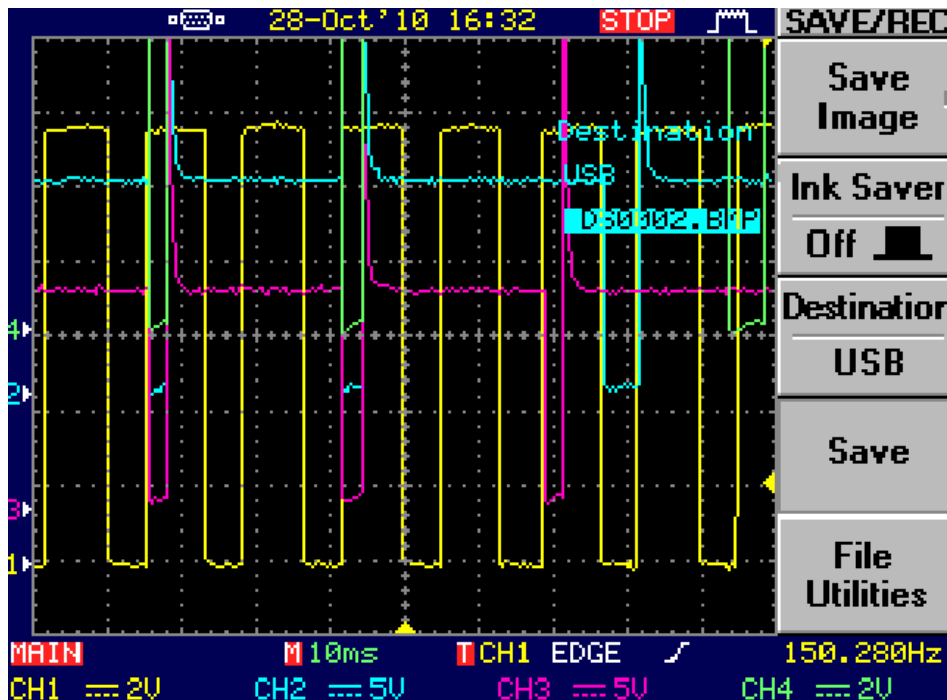


Figura 32: Oscil·lograma del sensorHall i injectors 1, 2 i 3.

El sistema d'injecci3 del motor APQ 3s multipunt. A l'oscil·lograma es pot comprovar que t3 dues maneres d'actuar, de forma simult3nia i de forma seqüencial:

La forma simult3nia 3s la que es pot veure en la primera part de l'oscil·lograma, en la que actuen els 4 injectors a la vegada cada 2 polsos del sensorHall. El sistema d'injecci3 treballa de forma simult3nia cada vegada que s'engega el motor i sempre que aquest no pogu3s sincronitzar-se per treballar de manera seqüencial.

La forma seqüencial consisteix en fer actuar un injector per cada pols del sensorHall (Figura 32). El sistema ha de funcionar de manera sincronitzada, per tant, ha de saber a quin injector li toca actuar. A la figura 33 es pot veure en qu3 es basa el sistema per saber quin injector ha d'actuar:

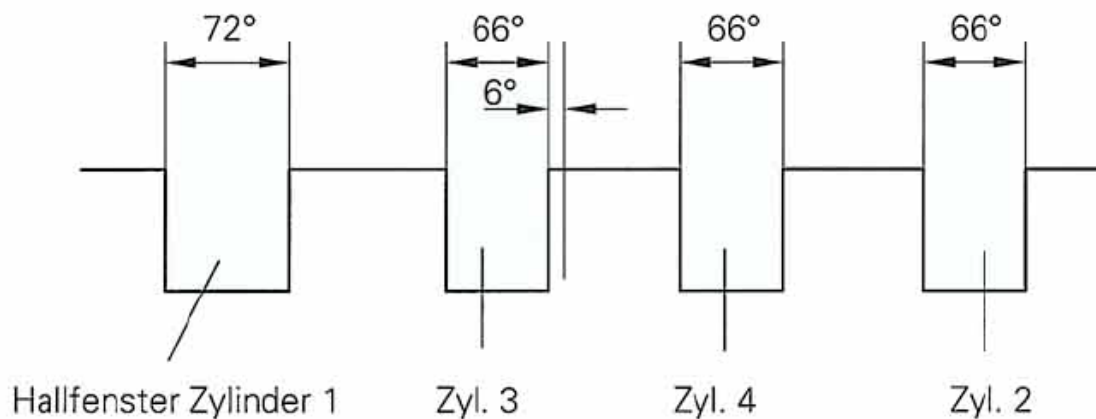


Figura 33: Diagrama de funcionament sensorHall

Per determinar l'injector que ha d'actuar s'utilitza com a refer3ncia la detecció de l'ample de pols del senyal de sensorHall per al cilindre 1. Aquest 3s lleugerament m3s ample que els altres. Amb la detecció del pols m3s ample ja es pot aplicar la seqü3ncia de funcionament pels injectors, que sempre ser3 la mateixa.

4.3.3.2 Descripci3 del codi per la injecci3

//Pel funcionament de forma simult3nia del sistema d'injecci3 s'ha utilitzat les següents variables:

```
int PulseCounter = 0, AnalogPosition = 4, ValueAnalogPosition = 0;  
unsigned int InjectorTime = 0;  
float TPS = 0;
```

//La variable "PulseCounter" s'utilitza per determinar el pols en què es donarà el senyal als injectors, aquests seran activats cada dos polsos del *sensorHall* per al mode de funcionament simultani (figura 32).

//La variable "AnalogPosition" llegeix un senyal d'entrada analògic que genera el *TPS* del motor APQ i que té un rang de 0 a 5V, per tant, el microcontrolador el pot llegir directament.

```
//Temporitzador 5 per ample de pols d'injecció
```

```
TIMSK5 = 0x01;
```

```
TCCR5A=0x00;
```

```
TCNT5=0x0000;
```

```
TCCR5B=0x02;
```

//El temporitzador s'ha preescalat a 8, obtenint un valor mínim de 0,0005ms i un màxim de 32.768ms, rang suficient per poder aplicar el temps d'injecció, ja que com es pot comprovar a la figura 30, el temps màxim d'obertura d'injecció és de 26ms i el més petit és de 0s.

//S'han creat vàries matrius per guardar els valors de temps d'injecció:

```
int rpm9[41] = {58336, 56236, 55536, 54836, 54136, 53436, 52736, 52036, 51336, 50636,  
int rpm10[41] = {60336, 58146, 57416, 56686, 55956, 55226, 54496, 53766, 53036, 52306,  
int rpm11[41] = {60336, 58146, 57416, 56686, 55956, 55226, 54496, 53766, 53036, 52306,  
int rpm12[41] = {62136, 59856, 59096, 58336, 57576, 56816, 56056, 55296, 54536, 53776,  
int rpm20[41] = {62736, 60426, 59656, 58886, 58116, 57346, 56576, 55806, 55036, 54266,  
int rpm22[41] = {64136, 61766, 60976, 60186, 59396, 58606, 57816, 57026, 56236, 55446,  
int rpm24[41] = {64736, 62336, 61536, 60736, 59936, 59136, 58336, 57536, 56736, 55936,  
int rpm27[41] = {65336, 62906, 62096, 61286, 60476, 59666, 58856, 58046, 57236, 56426,
```

//Una matriu o "array" és una col·lecció de variables que s'hi pot accedir mitjançant un número d'índex. En el nostre cas, es seleccionarà un valor d'una de les matrius en funció de les *RPM* i el *TPS*.

//Igual que en l'encesa i la lectura de les *RPM*, s'utilitza la funció d'interrupció externa activada pel *sensorHall* per aplicar les lleis de control. En aquest cas, s'incrementa la variable "PulseCounter" a cada interrupció, utilitzant una condició que determina l'activació de les sortides (22, 24, 26, 28) dels injectors:

```
Void RPM(){  
    Pulses++;  
    TCNT3 = IgnitionLoad;  
    PulseCounter++;  
    If(PulseCounter ==1){  
        digitalWrite(22, HIGH);  
        digitalWrite(24, HIGH);  
        digitalWrite(26, HIGH);  
        digitalWrite(28, HIGH);  
    TCNT5 = InjectionTime;  
    TIMSK5 = 0x01;  
    else{ pulseCounter = 0; } }
```

//També es fa la càrrega del registre TCNT5 amb el valor del temps d'injecció.

// Després de transcórrer el temps d'injecció i activar la bandera de desbordament del Timer 5, el programa entra a la rutina d'interrupció del Timer5 i desactiva els senyals de control dels injectors.

```
ISR(TIMER5_OVF_vect){  
    digitalWrite(22, LOW);  
    digitalWrite(24, LOW);  
    digitalWrite(26, LOW);  
    digitalWrite(28, LOW);
```

Internament, el microcontrolador porta un convertidor Analògic/Digital de 10 bits. La conversió del senyal analògic "AnalogPosition" a digital dins del programa es fa de la següent manera:

```
ValueAnalogPosition = analogRead(AnalogPosition);
```

TPS = ((ValueAnalogPosition * 4.9)/1023);

//Es guarda el valor llegit de la variable "AnalogPosition" en una nova variable (ValueAnalogPoition) i es calcula el TPS aplicant una regla de tres:

$$\begin{aligned} &ValueAnalogPosition = 4,9V(m\grave{a}xim) \\ &2^{10} = 1024 \\ &4,9V \text{ ----- } 1023bits \\ &TPS \text{ ----- } ValueAnalogPosition \\ &TPS = \frac{(ValueAnalogPosition * 4,9)}{1023} \end{aligned}$$

//La variable "InjectionTime" és on s'emmagatzema el valor del temps d'injecci3 que s'ha de d'executar a cada moment, per aix3, el TCNT5 carrega el valor d'aquesta variable al seu registre a cada activaci3 d'interrupci3 externa del sensorHall.

//Exemple de condici3 dintre del "programa principal de control" per triar el valor del temps d'injecci3 de la matriu en funci3 de les RPM i despr3s del TPS:

```
if ( 1320 < rpm && rpm < 1690) { if (TPS >=4){InjectionTime = rpml1[0];} else if(TPS >=3.85)
    else if(TPS >=3.35){InjectionTime = rpml1[11];} else if(TPS >=3.3){InjectionTime = rpml1[12];}
    else if(TPS >=2.7){InjectionTime = rpml1[21];} else if(TPS >=2.6){InjectionTime = rpml1[22];}
    else if(TPS >=1.7){InjectionTime = rpml1[31];} else if(TPS >=1.6){InjectionTime = rpml1[32];}
```

4.3.3.3 Circuit de pot3ncia de l'injecci3

El m3dul de pot3ncia est3 constituït f3sicament per una placa electrònica que rep a trav3s d'un connector del cotxe els senyals necessaris per al seu control. Internament, la part de l'injecci3 és la següent:

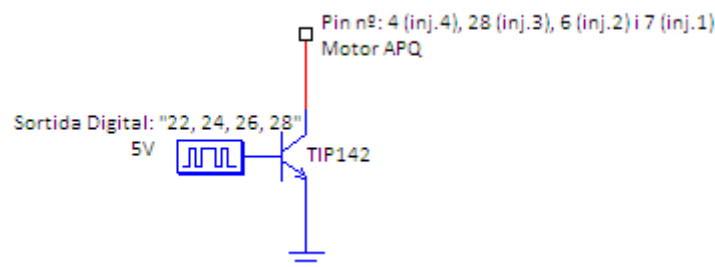


Figura 34: Esquema del funcionament de la Injecci3

El motiu principal a l'hora d'escollir el transistor ha estat el temps d'espera. És un factor molt important pel funcionament dels injectors ja que el transistor es fa treballar sempre com a interruptor i aquesta commutació ha de ser molt ràpida. També s'ha tingut en compte el corrent que circula per la bobina dels injectors que a 5000rpm és de 240mA, i la seva tensió de 12V a 14,5V.

4.3.4 Control del Ralentí

4.3.4.1 Descripció del sistema de ralentí

L'actuador del ralentí és el sistema que permet regular la posició del *TPS* mitjançant un senyal elèctric enviat pel microcontrolador. L'actuador està format per una papallona que permet, o no, el pas de l'aire de l'admissió controlat per un motor de corrent contínua que regula l'obertura entra 0 i 15°.

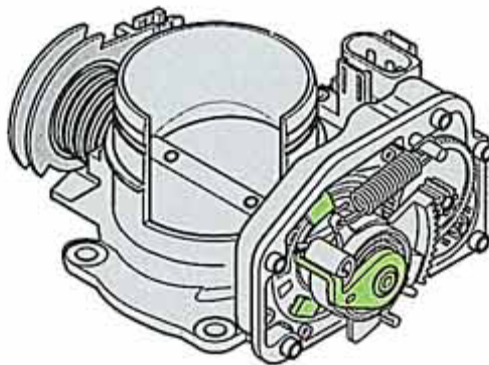


Figura 35: Unitat de control de l'accelerador amb actuador del ralentí integrat

Per governar el motor de corrent contínua que regula l'obertura del *TPS* s'utilitza un senyal de modulació per amplada d'impulsos (*MAP* o *PWM*).

4.3.4.2 Descripció del codi pel ralentí

Per generar un senyal *PWM* s'utilitza la funció "analogWrite". Aquesta funció, depenent del pin de sortida on es vulgui generar el senyal, anirà vinculat amb un dels temporitzadors. En aquest cas, el senyal es podrà generar als pins 9 i 10, ja que són els pins que utilitza l'únic temporitzador que queda disponible, el *Timer2*.

//La configuració del *Timer2* per la generació de senyal *PWM* ha estat per la freqüència en que es vol treballar:

//Pins 9 and 10: controlled by timer 2

Setting	Divisor	Frequency
0x01	1	31250
0x02	8	3906.25
0x03	32	976.5625
0x04	64	488.28125
0x05	128	244.140625
0x06	256	122.0703125
0x07	1024	30.517578125

//Temporitzador 2 per generar PWM pel control del ralenti

```
TCCR2B = 0x04;
```

```
analogWrite(10, 82);
```

//La funció "analogWrite" no té cap relació amb els pins d'entrada analògics. Dins de la funció es defineixen dos paràmetres:

"analogWrite(pin, valor)"

Pin: pin en el qual es vol generar el senyal *PWM*

Valor: cicle de treball desitjat comprès entre 0 (sempre apagat) i 255 (sempre encès).

4.3.4.3 Circuit de potència del ralenti

El circuit de potència utilitzat per controlar el motor de corrent contínua del controlador del ralenti, és exactament igual que l'utilitzat per fer el control de l'encesa. S'ha utilitzat el mateix transistor pel fet de ser un component que ja s'ha utilitzat en aquest projecte i, tot i estar sobredimensionat per aquest tipus de control, la seva resposta és correcta

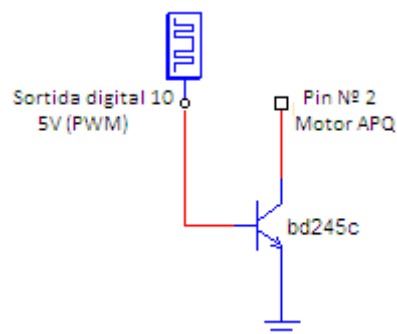


Figura 36: Esquema de control de l'actuador del ralenti

4.4 Sistema de transferència

4.4.1 Control de flux

El control de flux serveix per detectar situacions ambigües que es poden donar en una transmissió de dades. Com per exemple:

- Pèrdua de trames.
- Trames duplicades.
- Trames fora de seqüència.

En el prototip s'ha buscat solucions ràpides, fàcils i alhora robustes, és per això que s'ha decidit fer un control de "parada i espera" per la gran quantitat de trames a enviar cap a la memòria EEPROM del microcontrolador.

PARADA I ESPERA

El transmissor envia una trama i espera una indicació del receptor per enviar la següent. (Trama de reconeixement)

4.4.2 Format trama

El format triat per les trames de dades és el següent:

DC: Delimitador de començament d'un bit i format d'un caràcter **A=0x41** codi ASCII.

IC: Informació de control de 3 bits i format amb valor enter de 3 xifres, "número trama".

Informació: Bloc d'informació de 5bits i format amb valor enter de 5 xifres, "valor injecció".

DF: Delimitador de final de trama format per un caràcter **F=0x47** codi ASCII.

DC	IC	Informació	DF
----	----	------------	----

4.4.3 Funcionament parada i espera:

El funcionament es basa en enviar tots els valors de temps inclosos dins la taula temps d'injecció que hi ha en el software "EEPROM TRANSFER v1.0" creat per aquesta labor. Com que la taula està formada per 41 columnes i 19 files, s'obté un total de 779 caselles amb els diferents temps d'injecció.

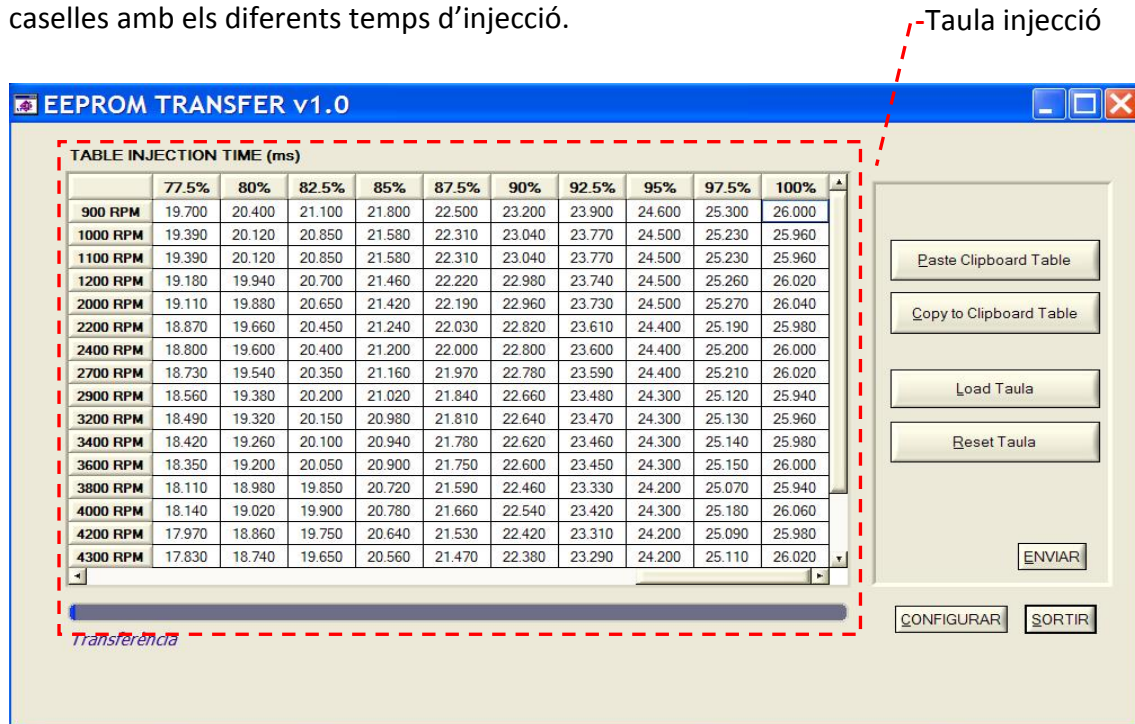


Figura 37: Captura software EEPROM TRANSFER v1.0

Tal hi com es pot observar a la taula, els temps estan representats en mil·lisegons, hi és sota programa on es creen les trames, amb format d'enter pel què fa al bloc d'informació per fer més fàcil i ràpida la bona comunicació entre ambdós programes.

El fet d'utilitzar valors enters és degut a què el temporitzador emprat per l'injecció és configurat amb una resolució de 16 bits, fet que pugui arribar a un valor de $2^{16} = 65536$.

Fent aquesta conversió amb el "EEPROM TRANSFER v1.0", on *val* és el valor de la casella, ja obtenim directament el valor que es vol donar al temporitzador d'injecció(*val1*), i per tant el valor a guardar a la EEPROM del microcontrolador.

```
val1 = val * (65535/32.768);  
val1 = 65535 - val1;
```

Per tant el nombre definit de trames a enviar és de 779, tenint en compte en tot moment l'informació de control IC que és l'indicador de la trama que s'envia, es pot veure en el diagrama següent com es té en compte la trama0 sent així la trama778 la última, i tenint controlades en tot moment cada trama amb un número IC entre 0 i 778.

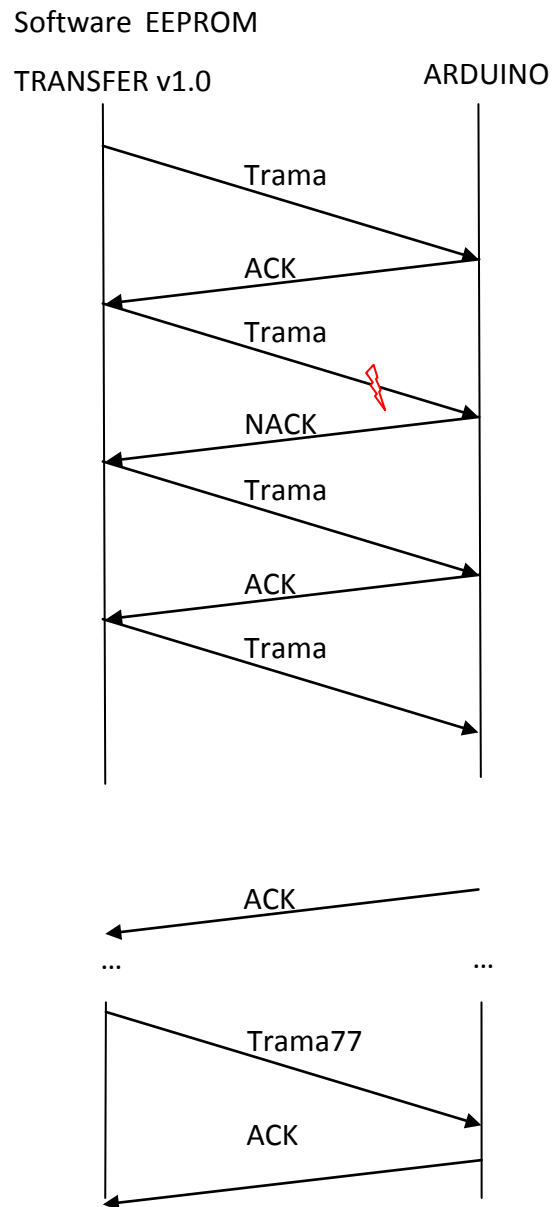


Figura 38: Diagrama comunicació PC i μC

4.4.4 Enviament "EEPROM TRANSFER v1.0" → microcontrolador

Les trames una a una són muntades i enviades de la següent forma:

```
Buffer[0]=0x41;           // Inici trama

Buffer[1]=cad[0];        // 3 Bits posició trama 000...778

Buffer[2]=cad[1];

Buffer[3]=cad[2];

Buffer[4]=cadena[0];     // 5 Bits valor informació 00000...65535

Buffer[5]= cadena[1];

Buffer[6]= cadena[2];

Buffer[7]= cadena[3];

Buffer[8]= cadena[4];

Buffer[9]= 0x47          // Final trama

ComWrt (comport,buffer,10); // S'envia al port sèrie tota la trama
```

Tal hi com s'observa la cadena de caràcters, és a dir la trama a enviar, és guardada al punter Buffer[], i un cop definit amb tota la informació s'envia pel port sèrie anteriorment configurat.

4.4.5 Recepció i Reconeixement microcontrolador

Dins el programa de transferència creat en el microcontrolador, rep les trames de la següent forma:

```
if (Serial.available() > 0) { // Quan arriben els caràcters sobre el port sèrie...

delay(4);                    // Espera una mica per assegurar tota la trama

if(Serial.read()==0x41) {   // Si rep 'A'

gg=0;                        // Increments a 0, per "posició" i "informació"
```

```
ff=0;

for(unsigned int w=0;w<3;w++){           // Llegir i guardar la "posició" de
                                          trama a buff[]

buff[ff]=Serial.read();

ff++;

}

IND = buff[2] - 48;                       // Conversió caràcter a enter el
                                          valor "posició"

IND = IND + (buff[1] - 48) * 10;

IND = IND + (buff[0] - 48) * 100;

for(unsigned int v=0;v<5;v++){           // Llegir i guardar la "informació" a
                                          buffer[];

buffer[gg]=Serial.read();

gg++;

}

VAL = buffer[4] - 48;                     // Conversió caràcter a enter el
                                          valor "informació"

VAL = VAL + (buffer[3] - 48) * 10;

VAL = VAL + (buffer[2] - 48) * 100;

VAL = VAL + (buffer[1] - 48) * 1000;

VAL = VAL + (buffer[0] - 48) * 10000;
```

Ara només faltaria el reconeixement “acknowledge” però abans cal guardar la nova informació a la EEPROM del microcontrolador, i és per això que cal partir la dada degut que el microcontrolador del prototip és de 8bits.

Aquest inconvenient fa que s'hagi d'ocupar el doble d'espai, passant a ser de 779 a 1558 valors a la EEPROM (veure taula posicions EEPROM a l'annex). La EEPROM té un total de 4kb és a dir 4096 bytes. On els valors modificables aniran entre el 0 fins al 1557.

```
WHB=highByte(VAL);      // Partir la dada en part “alta” i “baixa”.

WLB=lowByte(VAL);

IND=IND*2;              // “Posició” valor on guardar part “alta”

EEPROM.write(IND, WHB); // Guardar a la EEPROM en la “posició” anterior

IND++;                  // “Posició” valor on guardar part “baixa”

EEPROM.write(IND, WLB); // Guardar a la EEPROM en la “posició” següent

if (Serial.read()==0x47){ // Si rep ‘F’=0x47 “ack”

Serial.write(0x59);      // Enviar reconeixement “R”

LOADED++;               // Incrementar comptador

}

else {                  // Sinó rep ‘F’

Serial.write(0x55);      // Enviar negació de reconeixement “N”

LOADED--;               // Decrementar comptador

}

}
```



```
if (LOADED==779){           // Si no hi ha res al port sèrie i comptador és 779

lcd.setCursor(1, 0);

lcd.print("--TRANSFER-- ");

lcd.setCursor(0, 1);

lcd.print("  COMPLETED!!! "); // "Transferència completa per pantalla LCD"

LOADED=0;                   // Comptador dades correctes a 0

}

}

break;
```

La transferència és controlada amb un comptador de dades, el qual incrementa quan la dada reconeguda és correcte, enviant al software "EEPROM TRANSFER v1.0" una "R" i decrementa enviant una "N" quan la dada no és correcte, d'aquesta manera el control de flux de parada i espera permet l'enviament de la següent trama o el reenviament de la última.

4.4.6 Registre de la EEPROM (veure annex)

Valors modificables amb el software "EEPROM TRANSFER v1.0":

- REGISTRE EEPROM: 0 al 1557

Valors estàndards fixes:

- REGISTRE EEPROM: 2000 al 3557

Valor estat memoritzat:

- Registre EEPROM: 4000

5. Disseny del circuit imprès

5.1 Etapa d'alimentació del microcontrolador.

Per tal de que el circuit es pugui alimentar a partir de la bateria del vehicle, que dóna una tensió entre 12 i 14,5 volts, s'ha instal·lat un regulador de tensió comercial LM7809, aquest s'encarrega d'oferir una tensió de 9V estable al microcontrolador. El microcontrolador pot treballar amb una alimentació externa d'entre 6 i 20 volts. Si el voltatge subministrat és inferior a 7V, el pin intern de 5V pot proporcionar menys d'aquesta tensió i la placa pot tornar-se inestable, si s'utilitzen més de 12V els reguladors de tensió es poden sobreescalfar i malmetre la placa. El rang recomanable és de 7 a 12 volts.

5.2 Etapa d'alimentació pel control del ralenti

Pel funcionament original del sensor TPS, la ECU original subministra un voltatge d'alimentació de 5V. Per aquest motiu s'ha instal·lat un regulador de tensió comercial LM7805.

5.3 Adaptació senyal del sensorHall

Els *sensorHall* genera un senyal d'entre 12 i 14,5V d'amplitud. Aquest senyal ha de ser reduït a una amplitud de 5V que és el màxim que pot entrar al microcontrolador. Per fer aquesta reducció s'utilitza una resistència fixa de 220k Ω amb sèrie amb una de variable de 500k Ω , tenint així un mínim de seguretat alhora d'ajustar la sortida desitjada. En el nostre cas s'ha ajustat a 570k Ω , aconseguint una caiguda de tensió suficient per obtenir els 5V desitjats (figura 26).

5.4 Etapa de potència dels injectors, encesa i control del ralenti

Per fer el control dels injectors, de l'encesa i del ralenti s'ha utilitzat transistors de potència amb tecnologia TTL fent-los treballar com a commutadors en els tres casos.

Per l'etapa de potència dels injectors s'ha utilitzat el transistor "Tip142" (transistor Darlington), el Darlington és un dispositiu semiconductor que combina dos transistors

bipolars en cascada dins d'un únic dispositiu. Té un major desplaçament de fase en elevades freqüències que un únic transistor (figura 34).

Per l'etapa de potència de l'encesa i control del ralenti s'ha utilitzat transistors normals de potència "BD245C" (figura 29).

5.5 Esquema electrònic

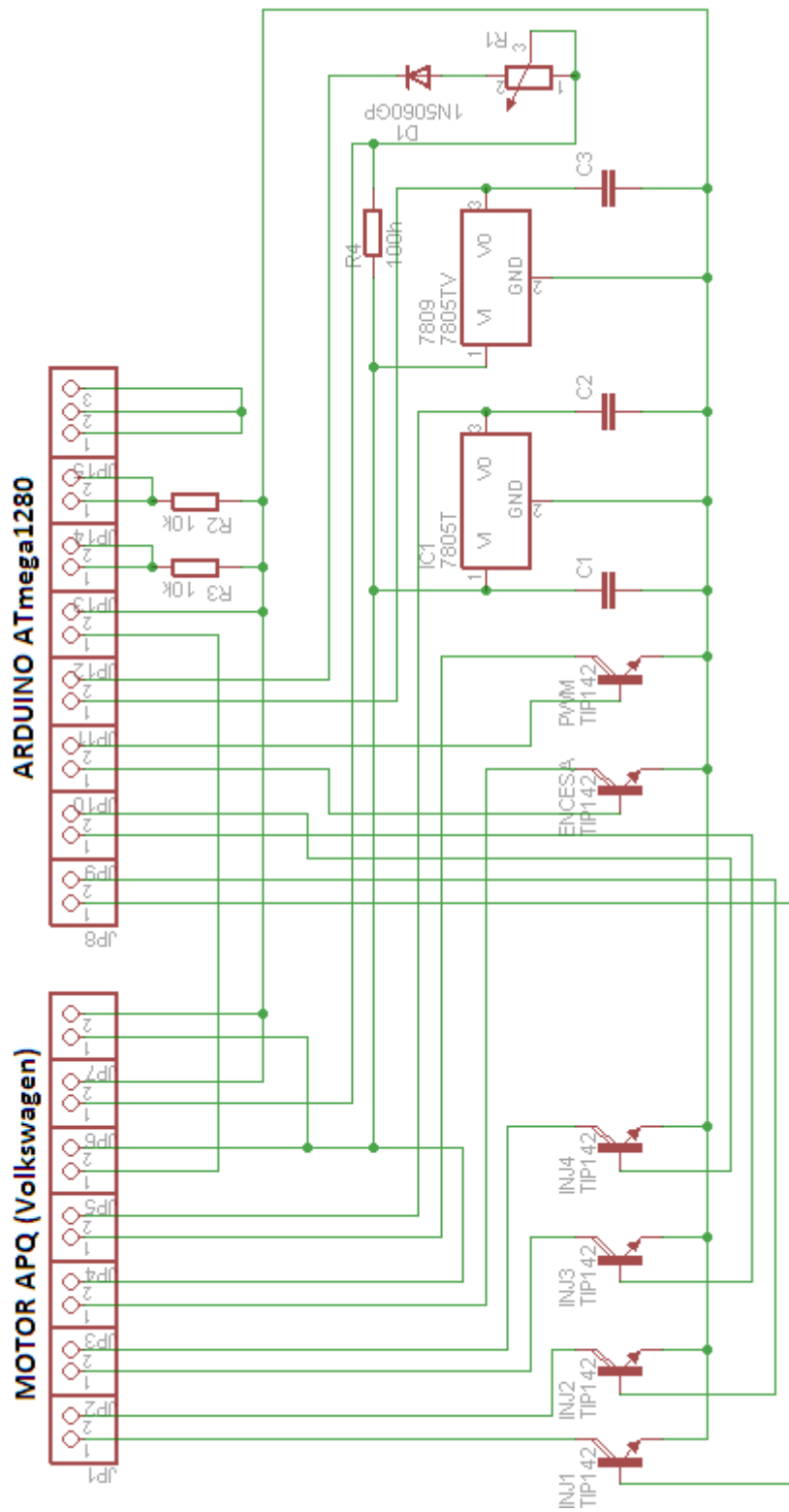


Figura 39: Esquema electrònic placa potència

5.6 Circuit imprès

Per tal de garantir el bon funcionament del circuit, s'ha decidit implementar aquest mitjançant una placa de circuit imprès de doble capa.

Per evitar o disminuir les reflexions, s'han eliminat els traçats de pista en forma de T o 90°, reduint-los mitjançant dos canvis de direcció de 45°.

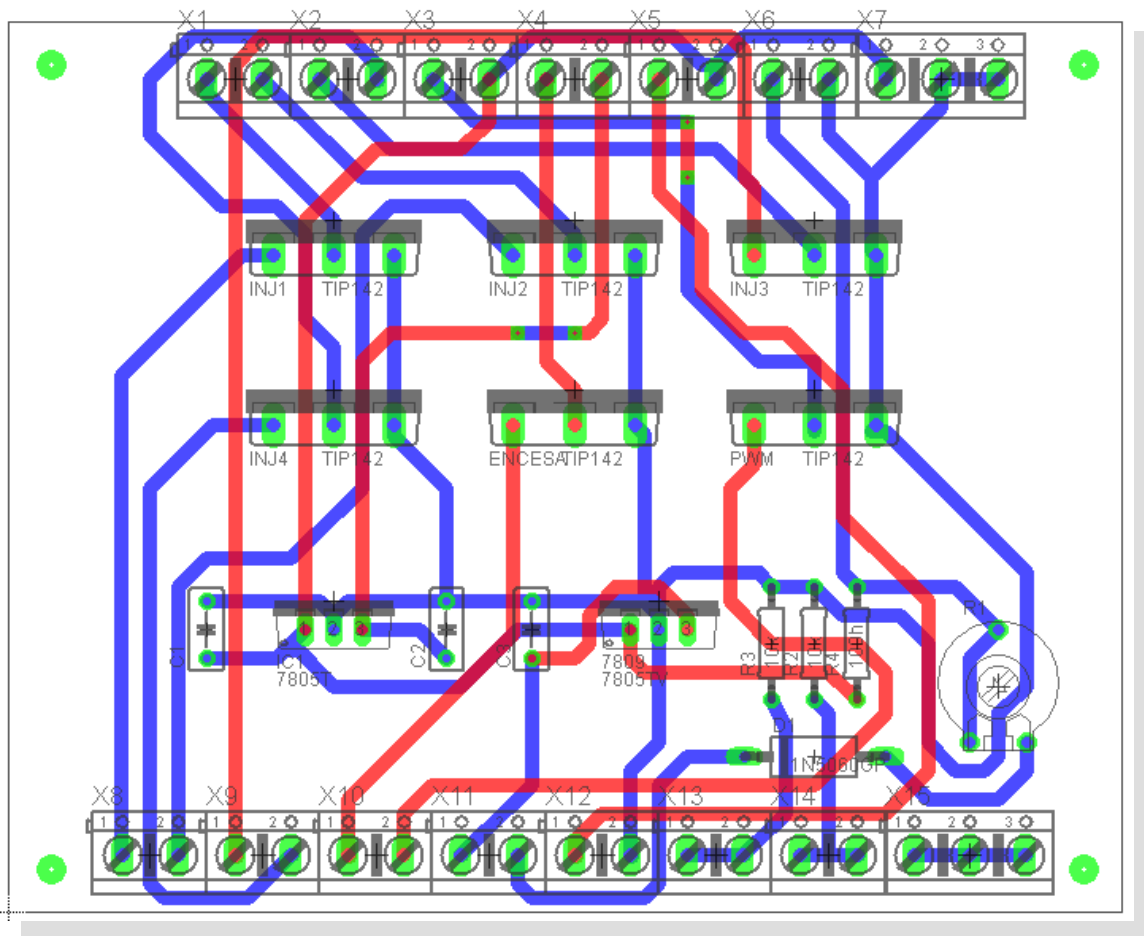


Figura 40: Captura disseny placa electrònica

Tal i com es pot veure la placa ha estat dissenyada en dues cares. El motiu pel qual s'ha dissenyat la placa així, és per simplificar espai i complexitat a l'hora de plasmar les pistes a la placa. En el cas d'haver-ho fet amb una sola capa, s'hagués necessitat més espai reduint així la robustesa i la utilitat del producte. La Insolació de la placa ha estat creada pel tècnic de laboratori de l'EPSEM. A la següent figura es pot veure els resultats de la placa abans de soldar components:

- Capa TOP de la placa

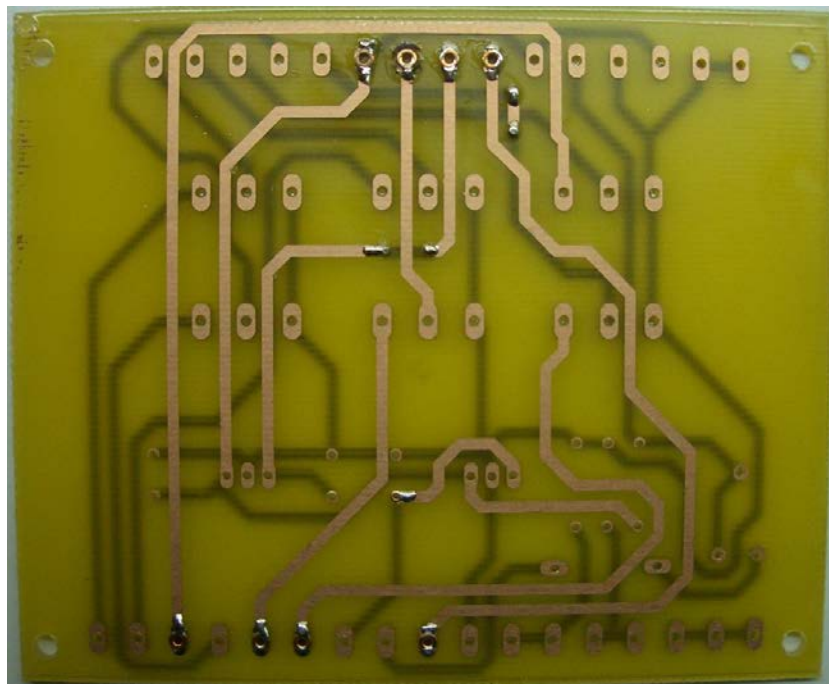


Figura 41: Placa electrònica vista superior

- CAPA BOTTOM de la placa

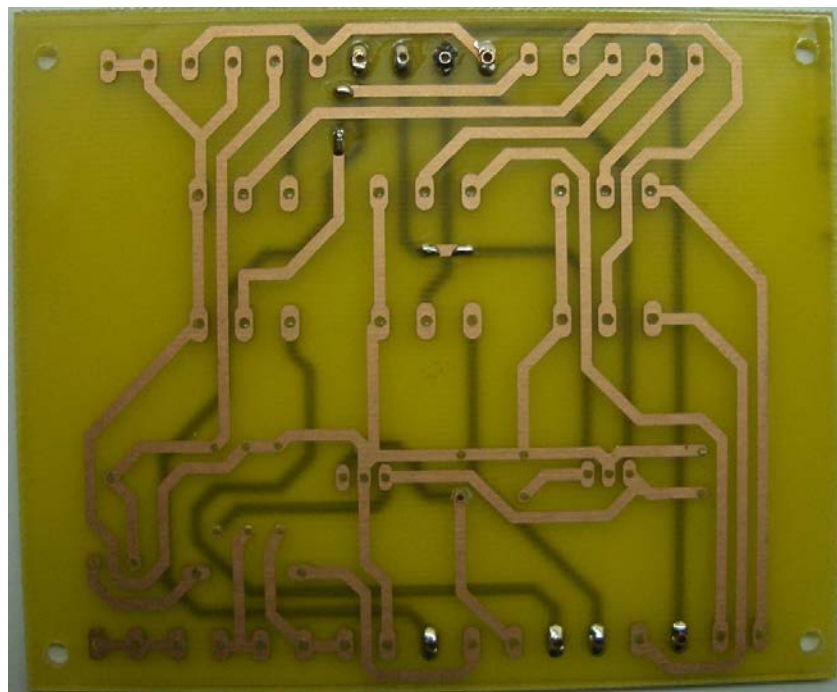


Figura 42: Placa electrònica vista inferior

6. Pressuposts

A continuació es pot observar el pressupost de material fet servir per al prototip tenint en compte el cotxe comprat de segona mà, i tot seguit el còmput d'hores d'enginyer.

Elements	Descripció	Quantitat	Preu Unitari	Preu
Integrats	Arduino MEGA1280	1	49,00	49,00
	Transistor TIP142	4	2,21	8,84
	Transistor BD245C	2	1,47	2,93
	Regulador LM7809	1	0,58	0,58
	Regulador LM7805	1	0,49	0,49
Resistències	Resistència 10k	2	0,06	0,12
	Resistència 100ohm	1	0,03	0,03
	Resistència 300k	1	0,03	0,03
	Pot. ajustable 572k	1	0,14	0,14
Condensadors	Condensador 0,1uF	2	0,15	0,30
	Condensador 680nF	1	0,47	0,47
PCB	PCB doble cara 100x160	1	6,98	6,98
Cablejat	Cable automòbil D1,5	4	3,62	14,49
	Cable automòbil D1	2	2,45	4,89
	Cable apantallat 16x0.25	1	1,84	1,84
	Cable arduino Jumper	1	5,00	5,00
LCD	Pantalla LCD 16x2	1	10,57	10,57
Polsadors	Polsador Negre	1	1,23	1,23
	Polsador Vermell	1	1,46	1,46
Terminals	Grapa ràpida automòbil	14	0,23	3,15
	Regleta connexió 2 bornes	16	0,41	6,58
Varis	Connector Sub-D mascle	1	0,70	0,70
	Connector Sub-D femella	1	0,70	0,70
	Tapa Sub-D cargols	1	1,92	1,92
	Kit cargols femella Sub-D	1	0,79	0,79
	Minibox 125x45x175	1	9,20	9,20

TOTAL MATERIAL	132,43 €
-----------------------	-----------------

Elements	Descripció	Quantitat	Preu Unitari	Preu
Enginyer	Marc Subirana Costa	115 hores	32 €/hora	3680 €
	Daniel Casals Sala	115 hores	32 €/hora	3680€

TOTAL HORES	7360 €
--------------------	---------------

*** Preus amb el 18% IVA inclòs.*

7. Propostes de millora

En relació a les propostes de millora, cal comentar que, tot i l'esforç dedicat al disseny i desenvolupament del projecte, és un punt que es podria ampliar força per molts i variats factors.

La utilització dels dos sensors principals, sensorHall i TPS, podrien no ser suficients si es busqués un estalvi en consum i caldria afegir sensors de pressió, temperatura i fins i tot de control d'emissions com ara la utilització de la sonda Lambda.

Pel què fa al software utilitzat per al programa de transferència de dades, el CVI Labwindows, seria un factor a modificar si el què es vol és introduir el producte al mercat, ja que en tractar-se d'un software sota llicència, incrementaria molt el cost final del producte.

També es podrien crear nous modes de funcionament, que permetessin canviar d'idioma, creació de diferents perfils, inicialització amb contrasenya, etc... que segur que donarien valor afegit per al possible client.

Caldria comentar que faria falta portar a la pràctica el prototip en un banc de proves, ja que és on es podria optimitzar més els temps d'injecció i encesa tenint en compte més factors, com seria el parell motor.

8. Conclusions

En primer lloc, cal esmentar que els objectius marcats del projecte han estat assolits amb èxit, tant per la part de control del motor com en la gestió i modificació de paràmetres.

En relació al control del motor, ha fet falta moltes proves pràctiques per aconseguir components que responguessin de forma ràpida i fiable, tot i la senzillesa dels circuits de control. A això, cal afegir-hi la poca informació dels fabricants de centraletes d'automòbil que encara ha complicat més el seu desenvolupament.

Per altre banda, la selecció del microcontrolador Arduino Mega 1280, ha facilitat molt les coses alhora de programar-lo ja que conté una plataforma força extensa i molta informació gratuïta, que han permès fer més senzilla una part de projecte que de bon principi semblava molt complicada.

Aquest projecte ha permès estudiar, desenvolupar i provar el control de l'electrònica en sistemes mecànics de combustió, arribant a controlar elements amb temps de l'ordre de mil·lisegons, com és el cas de l'injecció i encesa d'un automòbil. També cal comentar que alhora de dissenyar i programar un programa, on la velocitat de procés és elevada, es fa difícil fer-se a la idea de l'excel·lent temps de resposta del microcontrolador alhora d'escollir els diferents paràmetres un cop realitzat el càlcul de temps d'encesa o selecció del temps injecció.

L'apartat de comunicació per establir la transferència de paràmetres d'injecció, ha permès crear una part del projecte diferent i a la vegada molt útil per enllaçar les dues parts diferenciades del projecte, la configuració de paràmetres d'injecció i el control motor.

Pel què fa a resultats, cal comentar que en comparació amb les ECU PS1000 o Haltech, ja esmentades, són més complicades de configurar que el nostre prototip.

Un punt a citar, és el desenvolupament i aprenentatge que s'ha adquirit en el transcurs d'aquest projecte, gràcies a tots els conceptes i eines apreses al llarg d'una enginyeria d'aquestes característiques.

9. Bibliografia

Llibres

- Hermogenes Gil Mart3nez. La electr3nica en el autom3vil. Ediciones CEAC.
- Hermogenes Gil Mart3nez. Manual pr3ctico del autom3vil. Reparaci3n y mantenimiento. Editorial: Grupo Cultural.
- Apunts de l'assignatura *Inform3tica Industrial*. Impartida pel professor , Antoni ESCOBET. Curs 2009-2010 a L'EPSEM.

Adreces Internet

Microcontrolador

- <http://www.arduino.cc/>
- <http://www.greddy.com/products/>
- <http://www.haltech.com.au/>
- <http://es.wikipedia.org/wiki/Buj%C3%ADa>
- <http://automecanico.com/auto2002/sisencendido.html>
- <http://www.sabelotodo.org/automovil/sistencendido.html>
- <http://www.monografias.com/trabajos-pdf3/curso-sistemas-control-marcha-minima/curso-sistemas-control-marcha-minima.pdf>
- <http://www.mecanicavirtual.org/encendido-electronico-sin-contactos.htm>
- <http://www.todomecanica.com/sistemas-inyecciasolina/todas-las-paginas-3.html>
- <http://www.mecanicavirtual.org/sensores2-modelos.htm>
http://ca.wikipedia.org/wiki/Fitxer:Ciclo_de_cuatro_tiempos.png
- <http://tecnoblog-tecno.blogspot.com/>