

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Tilen Sotler

Analizator motorjev z notranjim izgorevanjem

**DIPLOMSKO DELO VISOKOŠOLSKEGA STROKOVNEGA
ŠTUDIJA**

Mentor: mag. Igor Škraba

Ljubljana, 2011



Št. naloge: 00171/2011

Datum: 03.10.2011

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **TILEN SOTLER**


Naslov: **ANALIZATOR MOTORJEV Z NOTRANJIM IZGOREVANJEM
ANALYZER FOR INTERNAL COMBUSTION ENGINES**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

V nalogi predstavite osnove delovanja motorjev z notranjim izgorevanjem in meritve, ki so potrebne za analizo njihovega delovanja. Za dano strojno opremo izdelajte programski modul za analizo in prikaz delovanja teh motorjev. Dobljene rezultate primerjajte z rezultati komercialnega programskega paketa.

Mentor:


pred. mag. Igor Škraba



Dekan:


prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani/-a Tilen Sotler,

z vpisno številko 63990136,

sem avtor/-ica diplomskega dela z naslovom:

Analizator motorjev z notranjim izgorevanjem

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal/-a samostojno pod mentorstvom (naziv, ime in priimek) pred. mag. Igor Škraba
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne _____ Podpis avtorja/-ice: _____

Zahvala

Zahvalil bi se rad dr. Juretu Knezu za uvod v zanimivi svet merilne tehnike, potrpežljivost in spodbudo pri raziskovanju le-tega.

Zahvalil bi se tudi mentorju mag. Igorju Škrabi in asistentu dr. Robertu Rozmanu za strokovno svetovanje, kritiko in popravke pri nastajanju tega diplomskega dela.

Hvala tudi Matjažu Pangeršiču za pregled besedila s strojniškega vidika in lektorici Anji Bakan za slovnični pregled. Brez njiju bi to delo vsebovalo mnogo več napak.

Ne nazadnje pa bi se zahvalil še svoji družini, ki me je ves čas podpirala in mi stala ob strani ne glede na moje odločitve v življenju.

Naj zaključim zahvalo s spodbudno mislijo za vse bralce tega besedila:

»The explorer is the person who is lost! --Tim Cahill«

Kazalo

SEZNAM KRATIC	2
POVZETEK	3
SUMMARY	4
1. MOTORJI Z NOTRANJIM IZGOREVANJEM	5
1.1 ZGODOVINA IN RAZDELITEV	5
1.2 OSNOVE DELOVANJA.....	6
1.2.1 Dvotaktni motor	6
1.2.2 Štiritačni motor	7
1.2.3 Motor s svečko.....	8
1.2.4 Motor z direktnim vbrizgom	8
2. OPIS PROBLEMA	9
3. OPIS MERILNE VERIGE ZA SPLOŠNE MERITVE	10
3.1 STROJNA OPREMA	11
3.2 PROGRAMSKA OPREMA	12
4. VHODNI PARAMETRI V ANALIZATOR	14
4.1 OSNOVNI PARAMETRI	14
4.1.1 Vrsta motorja	14
4.1.2 Kompresijsko razmerje	14
4.1.3 Politropni eksponent.....	14
4.1.4 Število valjev	15
4.1.5 Vrsta goriva	16
4.2 GEOMETRIJA IN NASTAVITVE VALJA	16
4.3 PREDLOGE ZA DOLOČEN MOTOR.....	19
4.4 SENZOR ZA KOT IN FREKVenco.....	19
4.4.1 Urejevalnik senzorjev za kot	21
4.4.2 Analogni senzor	22
4.4.3 Digitalni senzor	23
4.5 ZAJEM PODATKOV TLAKA Z EKSTERNO URO	24
4.5.1 CA-CPU.....	24
4.6 ZAJEM PODATKOV TLAKA Z INTERNO URO	25
4.6.1 Digitalni filtri.....	26
4.6.1.1 Filtri v frekvenčnem prostoru	26
4.6.1.2 Osnove NEO-filtrov	26
4.6.1.3 Osnove KEO-filtrov	28
4.6.2 Sprememba frekvence vzorčenja kot tehnika DSP.....	30
4.6.3 Decimacija	31
4.6.4 Interpolacija.....	32
4.6.5 Prevzorčenje	33
4.6.6 Polifazno prevzorčenje.....	33
4.6.7 Polifazno prevzorčenje in analizator.....	34
4.6.8 Zaključek.....	35
4.7 TLAK.....	35
4.7.1 Detekcija ZML	36
4.7.2 Poravnavanje tlaka na pravilno pozicijo kota.....	37

4.7.3 Poravnavanje tlaka na absolutno vrednost (ničenje)	37
4.7.3.1 Termodinamično ničenje tlaka.....	37
4.7.3.2 Ničenje iz vpisane vrednosti tlaka.....	37
4.7.3.3 Ničenje iz izmerjene vrednosti tlaka	38
4.8 MERITEV ZAČETKA IN KONCA VBRIZGA GORIVA TER DODATNI KANALI	38
4.10 DIAGRAMI NA ZASLONU ZA NASTAVITVE.....	40
5. REZULTATI	41
5.1 DIAGRAMI PODATKOV V ODVISNOSTI OD KOTA	42
5.1.1 Tlak	42
5.1.2 Povprečni tlak	43
5.1.3 Volumen valja.....	43
5.1.4 Odvod tlaka	43
5.2 SKALARNE VREDNOSTI.....	44
5.2.1 Povprečni efektivni tlak	44
5.2.2 Delo	45
5.2.3 Navor.....	45
5.2.4 Moč.....	46
5.2.5 Podatki o meritvi	46
5.3 TERMODINAMIKA (SKALARJI IN VEKTORJI)	47
5.3.1 Temperatura.....	47
5.3.2 Sproščanje toplotne energije (t. i. heat release).....	48
5.3.3 Klenkanje	50
6. PRIKAZ REZULTATOV	53
6.1 CA SCOPE	53
6.2 DELOVNI DIAGRAM.....	53
6.3 PRIKAZ V OBLIKI REKORDERJA.....	54
6.4 PRIKAZ V OBLIKI DIGITALNEGA MERILNIKA	55
7. ANALIZA	56
8. IZVOZ PODATKOV.....	57
8.1 IZVOZ PODATKOV V AVL CONCERTO	57
9. POVEZAVA S TESTNO MIZO	58
10. CA NOISE	59
11. PRIMERJAVA S PROGRAMOM CONCERTO (AVL).....	60
11.1 VOLUMEN	61
11.2 PV-DIAGRAM.....	61
11.3 POSAMEZNE VREDNOSTI.....	62
11.4 VEKTORJI.....	62
12. SKLEPNE UGOTOVITVE IN MOŽNOSTI ZA NADALJNI RAZVOJ.....	65
VIRI	67

SEZNAM KRATIC

FFT	Fast Fourier transform
NEO	Neskončni enotin odziv
KEO	Končni enotin odziv
STFT	Short-time Fourier transform
DAQ	Data acquisition
A/D pretvornik	Analogno/Digitalni pretvornik
GPS	Global Positioning System
CAN	Controller area network
IRIG	Inter-range instrumentation group
ZML	Zgornja mrtva lega
SML	Spodnja mrtva lega
XML	Extensible Markup Language
CA-CPU	Crankangle-central processing unit
DFT	Discrete Fourier transform
IDFT	Inverse discrete Fourier transform
DSP	Digital signal processing
CPE	Centralna procesna enota
MEP	Mean effective pressure
KF	Knocking factor
FS	Sampling frequency
pV diagram	Pressure volume diagram

POVZETEK

Namen dela je predstaviti programski modul za analiziranje motorjev z notranjim izgorevanjem. S takšnim programski paketom si lahko pomagajo strokovnjaki s področja avtomobilske, letalske in ladijske industrije za načrtovanje, preizkušanje in izdelavo motorjev.

V prvem delu so predstavljene zgodovina in osnove delovanja motorjev ter meritev in merilne tehnike na splošno. Ne dolgo nazaj je bila na področju merilne tehnike uporabljena namenska strojna oprema, ki je obdelovala podatke v čisto analognem smislu. Programska oprema je bila integrirana neposredno na čipe v takšnem inštrumentu in je služila bolj za prikaz podatkov kot za kakšne resne izračune. Slabost tega so veliki in težki inštrumenti, ki so narejeni samo za en namen. S povečanjem procesorske moči in pomnilniškega prostora na računalnikih pa se je začel razmah tehnike za digitalno procesiranje signalov. Strojna oprema le še zajame podatke, jih pretvori v digitalni zapis, od tam naprej pa se izvajajo digitalne tehnike za obdelavo podatkov. Inštrumenti so manjši in lažji, predvsem pa lahko izvajajo več različnih nalog.

Sledi predstavitev programskega modula za analizo motorjev. V tem delu si podrobneje ogledamo različne načine zajema fizikalnih podatkov in pretvorbo v računalniški zapis. Te podatke obdelamo z različnimi metodami s področja numerične matematike in digitalnega procesiranja podatkov v standardne parametre, ki jih potrebujejo inženirji za opis delovanja motorjev. Velikokrat so podatki lažje predstavljeni v obliki diagramov in tabel kot pa gole številke, zato je pomemben del tudi vizualizacija podatkov.

Zaključni del pa se nanaša na verifikacijo izhodov iz analizatorja. Izhode smo primerjali z obstoječim paketom podjetja AVL.

Ključne besede: programska oprema za izvajanje meritev, motorji z notranjim izgorevanjem, digitalno procesiranje signalov, filtracija podatkov

SUMMARY

The aim of this paper is to present the software module for internal combustion engines analysis. Such software package is very useful to car, airline or nautical industry experts when planning, testing and manufacturing engines.

The first part presents the history and the basics of engine operation, as well as measurements and measuring techniques in general. Not so long ago, an assigned hardware was used within measurement techniques in order to process data in a completely analogue matter. The software was integrated directly into the chip of an instrument, and was used mostly to present data rather than perform any serious calculations. The downsides of this are large and heavy instruments, mostly designed for one purpose only. But the increase of processor power and the amount of computers' memory space made way for the digital signal analysis technique. Hardware merely acquires the data, transforms it into digital recording, and digital data processing techniques are performed further on. The instruments are smaller and lighter and, most of all, able to perform several different tasks.

That chapter is followed by the presentation of software module for engine analysis where different methods of physics data acquisition and its transformation into computer recording are presented in a detailed manner. This data is processed with different methods stemming from numerical mathematics and digital data processing and transformed into standard parameters, needed by engineers when describing the engines' operations. Such data can often be understood more easily when presented in the form of diagrams and spreadsheets rather than bare numbers; therefore, the visualization of data is another important part of the process.

The final part of the paper deals with the verification of the data analyzer outputs. They were compared with the AVL company software package.

Keywords: DAQ software, internal combustion engines, digital signal processing, data filtration

1. MOTORJI Z NOTRANJIM IZGOREVANJEM

Okrog leta 1850 se je začelo komercialno vrtnanje in pridobivanje nafte. Posledično se je začel tudi razmah motorjev z notranjim izgorevanjem. Okrog leta 1880 je Otto izdelal prvi uporabnejši motor, katerega osnove delovanja se v tem času niso korenito spremenile. Večina sprememb se zadnje čase dogaja na področju optimizacije elektronske kontrolne enote (ECU). Za nas bosta najzanimivejša dvo- ter štiritačni motor, po drugi strani pa nas bo zanimala razlika med motorjem z vžigalno svečko (s pomočjo iskre) in motorjem s samovžigom.

1.1 Zgodovina in razdelitev

Namen motorjev z notranjim izgorevanjem je pridobivanje mehanske energije iz notranje kemične energije goriva. Mešanica goriva in zraka pred vžigom in izgoreli delci po vžigu sestavljajo delovno tekočino. Delo, ki direktno predstavlja moč, nastane kot posledica interakcije med delovno tekočino in mehanskimi deli motorja. Posvetili se bomo predvsem bencinskim (Ottov motor) in dizelskim motorjem. Ta tipa motorjev sta zaradi preprostosti in velikega razmerja med močjo in težo najbolj razširjena (zemeljski, zračni in morski prevozi).

Po letu 1860 so začeli prihajati prvi motorji z notranjim izgorevanjem v praktično uporabo. Kot mešanico za vžig so uporabljali premogov plin (pridobljen z destilacijo premoga) in zrak, vžgano pri atmosferskem tlaku s pomočjo tako imenovane preskakujoče iskre. Izkoristek takšnih motorjev je bil 5%. Pozneje sta Nicolaus A. Otto in Eugen Langen predstavila svoj motor, katerega izkoristek je bil okrog 11%. Za izboljšanje izkoristka tedanjih motorjev se je Ottu porodila ideja o štiritačnih motorjih (vsesavanje, komprimiranje, vžig/delo, izpuh) in prvi prototip je bil zagnan leta 1876. Ogromna razlika v velikosti in teži tega motorja v primerjavi s prvimi motorji je prispevala k vsesplošnemu uspehu – prodanih je bilo okrog 50.000 takšnih motorjev.

Nadaljnji razvoj je sledil hitro, ko je širša javnost spoznala, kaj je dosegel Otto. Do leta 1880 je nekaj inženirjev (Dugald Clerk – razvil dvotaktni motor leta 1878, patentiral 1881, James Robson, Karl Benz) razvilo dvotaktni motor (sočasno vsesavanje in komprimiranje, ekspanzija in izpuh s pridobivanjem mehanskega dela).

1882. leta je James Atkinson razvil Atkinsonov ciklični motor, ki je omogočil izvajanje sesanja, kompresije, vžiga ter izpuha v samo enem obratu posebne ročične gredi, ki je omogočila delovanje motorja z daljšim delovnim kot kompresijskim taktom. Posledica tega je bila ugotovitev, da je izkoristek direktna funkcija kompresije. Kompresijska razmerja so bila takrat omejena na razmerje ena proti štiri zaradi velikih problemov s klenkanjem.

Leta 1892 je dr. Rudolf Diesel razvil svoj Carnotov motor. Izgorevanje je povzročil samo z vbrizgom goriva v zrak, razgret zaradi kompresije (brez uporabe vžigalnih svečk), in s tem dvakrat povečal izkoristek v primerjavi z Ottovim motorjem.

Leta 1954 je Felix Wankel uspešno predstavil prvi rotirajoči motor. Rotor se obrača v ohišju ekscentrično, in sicer tako, da so njegovi trije tesnilni vogali vedno na steni ohišja. Med tremi stranicami rotorja in notranjo steno ohišja so trije delovni prostori, katerih prostornina se med

vrtenjem rotorja stalno spreminja. Večina Wanklovih motorjev ima uplinjač ali pa neposreden vbrizg goriva.

Na celoten razvoj motorjev je imelo velik vpliv tudi gorivo. Prvi motorji so uporabljali plin. Leta 1800 je postal na voljo bencin. Pred letom 1905 je bilo veliko težav s surovim bencinom zaradi učinka klenkanja. Po prvi svetovni vojni pa je sledil velik napredek. Odkrite so bile lastnosti tetraetilnega svinca. Ta namreč deluje proti klenkanju, zato so ga začeli dodajati bencinu [6].

Leta 1952 je prof. A. J. Haagen-Smit dokazal, da smog nastane pri reakciji izpušnih plinov na soncu. S tem je postalo jasno, da je avtomobil velik onesnaževalec, povzroča pa tudi veliko hrupa. Kot rezultat tega so se začeli razvijati standardi glede dovoljenega izpuha in zvoka. Poleg tega se je začelo gorivo (bencin in dizel) dražiti že leta 1970. Vse to je postavilo zahteve po motorjih s čim večjim izkoristkom, manjšim hrupom in manj izpuha. Za doseganje tega cilja pa so nujni raziskave in razvoj, pri čemer si tega dandanes brez takšnih ali drugačnih sistemov za analizo (v smislu strojne in programske opreme) ne predstavljamo več.

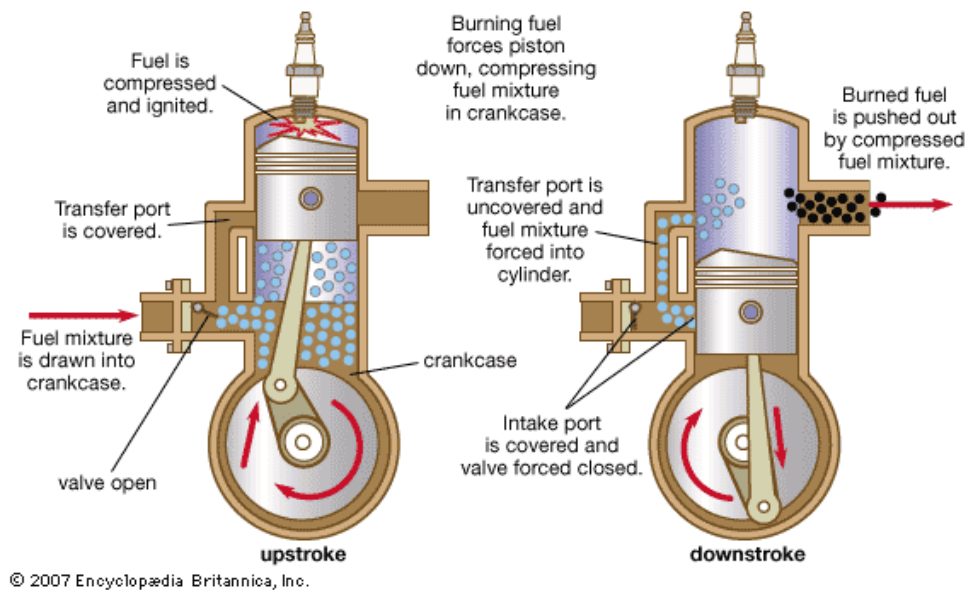
1.2 Osnove delovanja

Motorje z notranjim izgorevanjem lahko razdelimo glede na več meril. Za nas bosta najpomembnejši dve. Razdelitev glede na delovni takt (dvo- in štiritačni motorji) in razdelitev glede na vžig (vžig s svečko in samovžig). Pri vseh tipih je v osnovi delovanje takšno, da se bat v valju premika naprej in nazaj in pri tem poganja gred. Ko je volumen v valju največji, rečemo, da je bat v SML (spodnji mrtvi legi), ko je volumen najmanjši, je bat v ZML (zgornji mrtvi legi).

1.2.1 Dvotaktni motor

Dvotaktni motor potrebuje dva takta za en delovni cikel (slika 1):

- a) 1. takt: bat se premika iz SML v ZML, pri tem se delovne snovi vsesajo v blok motorja. Sledi stiskanje snovi in vžig.
- b) 2. takt: bat se pomika nazaj proti SML, delovni plini se dovedejo v valj. Sledi izpuh zgorelih plinov.



Slika 1. Prikaz delovanja dvotaktnega motorja

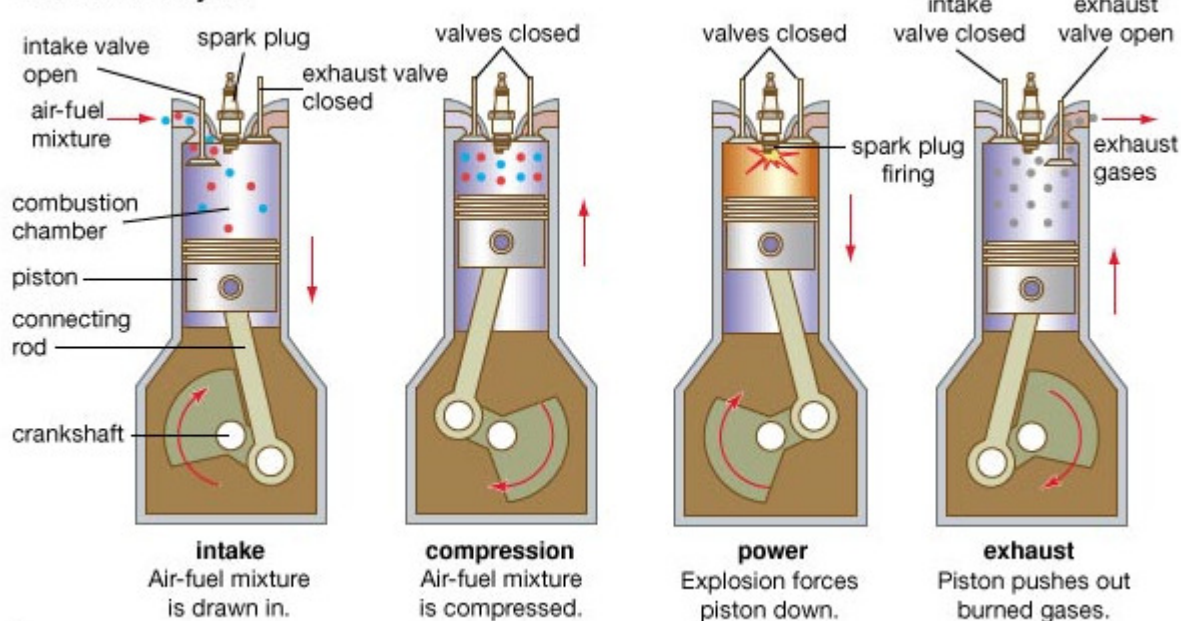
Dvotaktni motorji so preprostejši in imajo večjo moč pri enaki velikosti in teži kot štiritačni motorji. Imajo tudi veliko preprostejši sistem ventilov. Uporabljajo se predvsem za kosilnice, motorna kolesa, ladje (veliki valji, majhna hitrost) itn.

1.2.2 Štiritačni motor

Štiritačni motor potrebuje štiri takte za en delovni cikel (slika 2):

1. takt – sesanje: v valj se vsesa delovne pline.
2. takt – stiskanje (kompresija): vsi ventili se zaprejo in začne se stiskanje delovnih plinov.
3. takt – delovni: vžig delovne snovi. Plini, pod visoko temperaturo in tlakom, pritiskajo na bat in ga pomikajo navzdol.
4. takt – izpuh: odprejo se izpušni ventili in iz bata izpustijo ostanke zgorelih plinov.

Four-stroke cycle



Slika 2. Prikaz delovanja štiritaktnega motorja

Štiritaktni motorji manj onesnažujejo zrak kot dvotaktni, gorivo uporabljajo učinkoviteje, imajo tudi daljšo življenjsko dobo, boljši sistem hlajenja ter učinkovitejše mazanje. Uporabljajo se v avtomobilih, tovornjakih, letalih itn.

1.2.3 Motor s svečko

Poganjajo ga bencin in podobna goriva. Za vžig uporablja svečko, zrak in gorivo pa se zmešata že v uplinjaču.

1.2.4 Motor z direktnim vbrizgom

Takšen motor je lahko dizelski ali pa bencinski. Za razliko od motorja s svečko, se gorivo vbrizga direktno v valj, kjer se zmeša z zrakom. Bat stisne delovno snov, ta pa se posledično vžge. Prednosti teh motorjev sta večja učinkovitost pri uporabi goriva in večja moč. V večini primerov so nižje tudi emisije.

2. OPIS PROBLEMA

Zajem in analiziranje podatkov motorjev z notranjim izgorevanjem nam pomagata pri razumevanju zapletenega procesa izgorevanja goriva v valju motorja. Namen naloge je razviti programski modul za snemanje, vizualizacijo, analizo in izvoz parametrov, ki najbolj opisujejo to dogajanje. Izračunane vrednosti se uporabljajo za razvoj in preverjanje motorjev.

Kot vhod v modul je najbolj praktično uporabiti meritev tlaka iz valja, ker omogoča najpodrobnejši opis dogajanja (izgorevanja) v valju. Iz tlaka se da izračunati praktično vse pomembne vrednosti za razvoj oz. testiranje motorjev.

Za strojni del opreme bo uporabljena Dewetronova strojna oprema. Programski paket, znotraj katerega bo razvit modul, pa je Dewesoft.

Ker se motor vrti v določenem ciklu, je treba vhodne podatke tlaka povezati s tem ciklom. Na različnih pozicijah kota (valja) se izvedejo različne akcije (odpiranje ventilov, vbrizg goriva), čas pa za določene vrednosti ni pomemben.

Programski modul mora iz krivulje tlaka in pozicije kota prav tako izračunati določene vrednosti (povprečni tlak, odvod tlaka, klenkanje ...), ki bodo uporabljene za nadaljnji razvoj in testiranje motorjev.

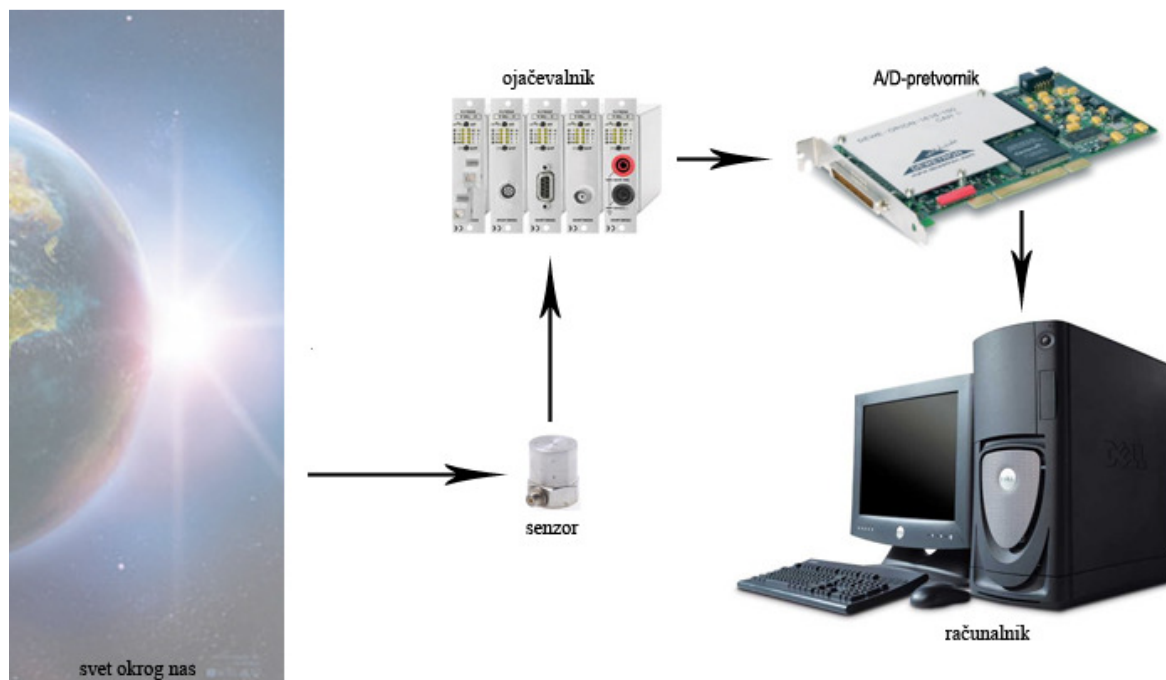
Vhodne in izhodne podatke je treba prikazati na zaslonu instrumenta, to pa zahteva izdelavo določenih diagramov. Uporabniki programa želijo videti podatke na določen način. Nekateri grafi pa so ključni za hitro razpoznavanje (že med meritvijo), kaj se dogaja z motorjem.

Modul mora tudi shraniti podatke za analizo. Uporabnik mora imeti možnost, da pozneje odpre podatkovno datoteko z določenim datumom in uro, se sprehodi po vseh podatkih cikla za ciklom ter analizira, kaj se je v določenem ciklu dogajalo. Mora pa imeti tudi določene vrednosti za celotno meritev.

Treba je izdelati še izvoz podatkov v različne druge programske pakete, ki omogočajo bolj specifično analizo parametrov. Najbolj razširjen je paket Concerto (AVL), ki za vhodne datoteke uporablja lasten binarni zapis. Programski modul mora izvoziti podatke v tem zapisu.

3. OPIS MERILNE VERIGE ZA SPLOŠNE MERITVE

Merilno verigo v današnjem času po navadi sestavljajo senzor, ojačevalnik (modul), A/D-pretvornik in računalnik (slika 3).



Slika 3. Merilna veriga

Senzorji so zelo različni in odvisno je, za kakšen namen ga potrebujemo (npr. senzor za temperaturo, vibracije, pomike, silo). Senzor po navadi meri neko fizikalno količino in jo pretvarja v drugo (npr. tok), ki jo bo ojačevalnik sposoben zaznati.

Ojačevalnik dobi vhod iz senzorja in ga ojača na pravo raven za A/D-pretvornik, običajno ± 5 V. Ojačevalnik ima lahko tudi nizko- oz. visokoprepustne filtre, ki skrbijo, da pride na A/D-signal samo v želenem frekvenčnem pasu oz. območju.

A/D-pretvornik skrbi za pretvorbo analognega signala v digitalni signal. Običajno se za pretvorbo uporabljajo sigma-delta pretvorniki. Nyquistov teorem pravi, da se lahko izognemo t. i. aliasom, če vzorčimo z vsaj dvakrat večjo frekvenco, kot je maksimalna frekvenca v signalu. To dosežemo tako, da imajo A/D-pretvorniki t. i. antialiasing filter in že pred vstopom signala v A/D-pretvornik odstranijo frekvence, višje od polovice vzorčevalne frekvence.

Na koncu verige pa je računalnik, ki skrbi za zajem podatkov, različne obdelave, shranjevanje in prikaz podatkov v človeku razumljivi obliki. Ker so današnji računalniki razmeroma zmogljivi, se je bolj razširilo tudi digitalno procesiranje signalov. Včasih so namenska vezja

opravila večino nalog (npr. filtriranje, detekcijo ovojnice signala, oktavni spekter...), danes pa se je večina tega preselila na računalnik.

3.1 Strojna oprema

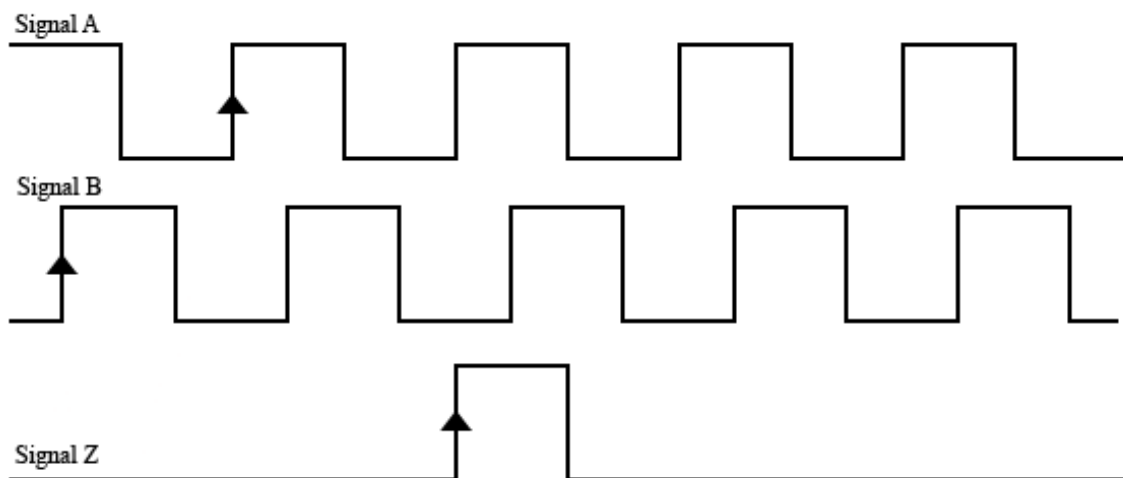
Za razvoj analizatorja je bila uporabljena Dewetronova strojna oprema.

Uporabljen senzor tlaka je produkt podjetja Kistler. Narejeno je tako, da je v avtomobilsko svečko (slika 4) vgrajen piezoelektrični senzor za meritve tlaka v valju.



Slika 4. Avtomobilska svečka z vgrajenim senzorjem za tlak [6]

Za meritev pozicije kota je bil uporabljen standardni enkoder. Enkoder ima tri digitalne izhode – A, B in Z (slika 5). A in B sta zamaknjena za 90 kotnih stopinj, s čimer dobimo smer vrtenja. Signal Z pa je pozitiven enkrat na obrat. Tako dobimo pozicijo vrtenja gredi od 0 do 360 kotnih stopinj. Če izračunamo odvod pozicije kota po času, dobimo kotno hitrost in s tem tudi frekvenco vrtenja.



Slika 5. Digitalni signali iz enkoderja

Uporabljen ojačevalnik za nabojne senzorje je DAQP-CHARGE-B, podjetja Dewetron. Ojačevalnik prenese naboj v kondenzator, ta pa generira napetost, proporcionalno vhodnemu naboju. Napetost gre direktno na A/D-pretvornik.

Vhod v ta modul je naboj. Lastnosti:

- izolirani vhodi
- izbira med 9 nizkoprepustnimi filtri

Uporabljen je bil A/D-pretvornik DEWE-ORION-1616 [4]. Poglavitne lastnosti:

- maksimalna vzorčevalna frekvenca je 100 KS/s (Kilo-Samples / Second)
- 16 kanalov (možno dodajanje do 32 kanalov)
- simultano vzorčenje
- 16-bitni kanali
- 2 digitalna vhodna kanala (možno dodajanje do 8 kanalov)
- štiri vhodna območja (+/-1,25 V do +/-10 V)

Ojačevalniki, A/D-pretvornik in računalnik pa so združeni v en inštrument, in sicer DEWE-5000 [5]. Gre za prenosni inštrument z zaslonom. Na straneh so že pripravljene vhodi za digitalni senzor, na sprednji strani pa so vstavljeni moduli (do 16) , ki se lahko po želji dodajo, odstranijo ali pa zamenjajo z drugimi.

3.2 Programska oprema

Dewesoft je eden vodilnih programskih paketov za zajem in od različice 7 naprej tudi za analizo podatkov [3]. Deluje na platformi Windows, tako da ga lahko namestimo na kateri koli osebni računalnik. Podpira ogromno različne strojne opreme:

- A/D-pretvorniki (National Instruments, Spectrum, Dewetron, Dewesoft, Data translation ...)
- GPS (NMEA, Javad, Dewetron ...)
- CAN-vodila (Dewetron, Softing, Vector ...)
- video (Basler, Marlin, vse kamere DirectX)
- točen čas (Gps, IRIG, datum ...)
- analogni izhod (National Instruments, Data translation)

Vgrajene ima standardne funkcije s področja digitalnega procesiranja signalov: FFT, ovojnico, NEO- in KEO-filtre, formule, klasifikacijo, točno frekvenco, STFT, oktavni spekter itn. Dosegljive so tudi aplikacije, kot so: meritve in izračuni iz električnih veličin, meritve zvoka, iskanje harmonikov, torzijske vibracije, modalna analiza, balansiranje itn. Analizator motorjev z notranjim izgorevanjem pa je bil dodan omenjenim aplikacijam.

Vsi podatki so na uporabniškem nivoju prikazani kot t.i. »kanali«. Spreminjamo jim lahko območje, skalirno vrednost, barvo, ime, enoto, itn. Kanal si lahko predstavljamo kot abstrakcijo za podatkovne skalarje, vektorje in matrike, katere preberemo iz A/D karte in jih tako predstavimo uporabnikom z različnimi diagrami.

Dewesoft dopušča tudi dodajanje zunanjih vključkov (t. i. »pluginov«), tako da lahko uporabniki sami napišejo svojo analizo signalov. Dewesoft v tem primeru zajame podatke in jih prek DCOM-tehnologije posreduje vključkom (»pluginom«). Ti jih obdelajo, vrnejo pa rezultate, ki jih zna Dewesoft prikazati na različnih grafih.

Vključke delimo na:

- DAQ
- matematiko
- vizualne kontrole

Zajem in shranjevanje podatkov je možno izvesti z interno ali pa eksterno uro. Omogoča tudi popolnoma sinhrono združevanje videa ter analogno in digitalno zajetih podatkov.

Notranja struktura programskega paketa Dewesoft je organizirana tako, da dokaj preprosto dopušča dodajanje matematik, aplikacij in vizualnih kontrol.

4. VHODNI PARAMETRI V ANALIZATOR

Program za analizo motorjev z notranjim izgorevanjem je za uporabnika videti kot t. i. črna škatla. Od uporabnika zahteva določene vhodne parametre, vrne pa mu izračunane vrednosti. Vhodni parametri so na primer vrednost tlaka za vsako kotno stopinjo vrtenja motorja, tip motorja, kompresijsko razmerje itn. Prav tako je treba nastaviti senzor za meritev kota zaradi pravilnega zajema podatkov za kot. Podatke lahko zajemamo z interno ali pa eksterno uro. Oba načina imata prednosti in slabosti.

4.1 Osnovni parametri

Osnovni parametri so potrebni za izračun prostornine valja in obnašanja samega bata. Prav tako parametri (vrsta motorja, kompresijsko razmerje, politropni eksponent število valjev in tip goriva) vplivajo na same izračune.

4.1.1 Vrsta motorja

Osnovni vrsti motorja sta štiritačni in dvotakti motor. Štiritačni potrebuje štiri takte in zato so vsi izračuni in diagrami narejeni od 0 do 720 kotnih stopinj. Standardna x-os za štiritačni motor se označi od -360 do 360 kotnih stopinj, ker je z 0 kotnimi stopinjami označena zgornja mrtva lega. Dvotakti motor pa konča en cikel motorja v dveh taktih, zato se os označi od -180 do 180 kotnih stopinj (0 kotnih stopinj je ZML). Ta polje je treba izbrati, ker so vsi izračuni in vizualizacije narejeni s takšno x-osjo, kot jo določa vrsta motorja. Vse vrednosti za kote v analizatorju so v kotnih stopinjah (ne v radianih), ker je to standard v že obstoječih programskih paketih, na tem področju.

4.1.2 Kompresijsko razmerje

Bat potuje v valju od spodnje mrtve lege proti zgornji mrtvi legi. Ko je bat v spodnji mrtvi legi, je volumen valja največji, ko pride v zgornjo mrtvo lego, pa najmanjši. Razmerju med največjim in najmanjšim volumnom v valju pravimo kompresijsko razmerje. Visoko kompresijsko razmerje je zaželeno, ker zaradi večje termične učinkovitosti dobimo več mehanskega dela. Prevelika razmerja pri bencinskih motorjih pa povzročajo klenkanje. Če se uporablja bencin z nizkim oktanskim številom, lahko pride celo do detonacij v valju. Dizelski motorji pa temeljijo ravno na principu večje kompresije. Gorivo se upira samovžigu in s tem pride do zakasnjene vžiga.

4.1.3 Politropni eksponent

Kompresijo in širjenje plinov lahko s fizikalnega vidika delimo takole:

- izotermni proces (konstantna temperatura),
- adiabatni proces (konstantni volumen),
- politropni proces.

Izotermni proces se mora izvesti počasi, da ostane temperatura konstantna. Adiabatski proces se mora zgoditi hitro, da ni pritoka oz. odtoka energije. Termodinamični procesi v valju so nekje med izotermnimi in adiabatskimi. Torej jih najboljše opisuje politropna enačba [6]. Slika 6 prikazuje spremembo tlaka glede na volumen in politropni eksponent.

$$pV^k = C \quad (4.1 a)$$

$$k = \frac{c_p}{c_v} \quad (4.1 b)$$

kjer so:

p = tlak [bar]

V = volumen [dm^3]

k = politropni eksponent

C = konstanta

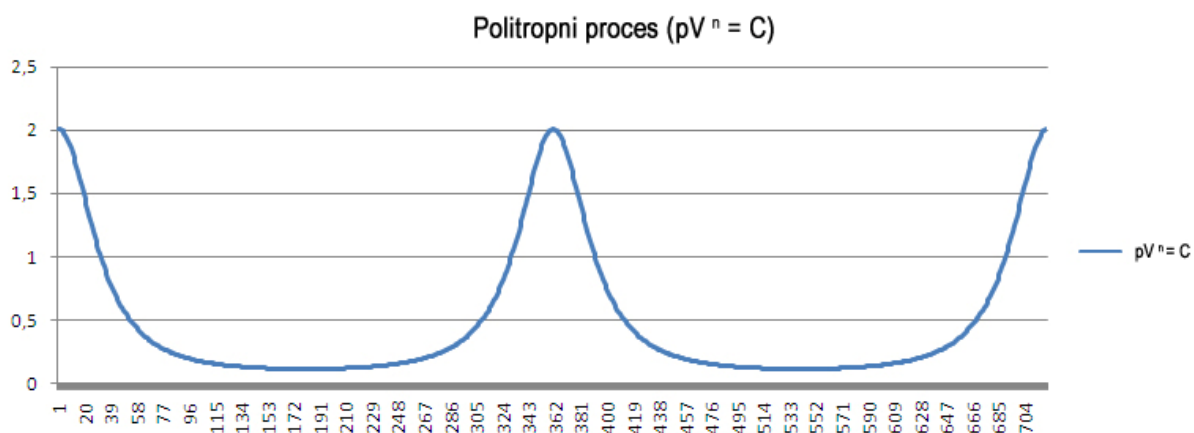
c_p = specifična toplotna kapaciteta pri konstantnem tlaku

$[\frac{\text{J}}{\text{kg} \cdot \text{K}}]$

c_v = specifična toplotna kapaciteta pri konstantnem

volumnu [$\frac{\text{J}}{\text{kg} \cdot \text{K}}]$

c_p in c_v nam povesta, koliko toplote je treba dodati plinu, da povečamo eno enoto mase plina za eno stopinjo Kelvina.



Slika 6. Diagram tlaka glede na politropno enačbo. Na osi y je tlak [bar], na osi x pa število vzorcev za en obrat glavne gredi motorja

Enačba (4.1 a) zelo dobro opisuje procese, ki se dogajajo v valju med stiskanjem in raztegovanjem delovnih tekočin. Uporabnik analizatorja mora poznati politropni eksponent za določen motor in ga podati v nastavitvah, ker se ta vrednost uporablja v nekaterih enačbah.

4.1.4 Število valjev

Število valjev v motorju je lahko zelo različno. Vsak valj ima kot vhod vrednosti iz senzorja tlaka. Možno je meriti tudi začetek in konec vbrizga goriva s posebnim senzorjem. Vsakemu

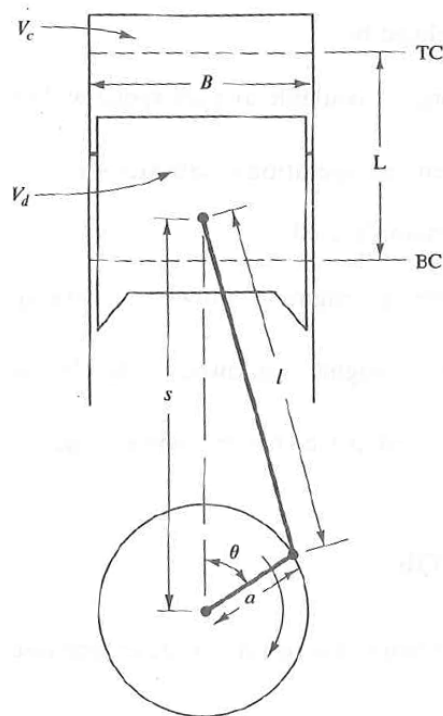
valju je možno dodati kanale. Vsi kanali v analizatorju so v kotni domeni. Vzame se vrednosti iz originalnih kanalov, zraven se preberejo vrednosti kota, podatki pa se poravnajo na en cikel motorja.

4.1.5 Vrsta goriva

Gorivo delimo na bencinsko in dizelsko. Pri bencinskem gorivu uporabljamo direktni vbrizg ali pa vžig s svečko. Pri dizelskem pa za vžig uporabljamo samo direktni vbrizg. Uporabnik mora izbrati vrsto goriva, ker se nekatere enačbe izračunajo drugače glede na vrsto goriva in posledično tudi vrsto motorja.

4.2 Geometrija in nastavitve valja

Bat potuje v valju od spodnje mrtve lege do zgornje mrtve lege. Izračuni in diagrami so vezani na premikanje bata v valju, zato je treba določiti osnovne parametre za volumen valja (slika 7).



Slika 7. Geometrijske lastnosti valja [6]

Volumen valja (V) [6]:

$$V = V_c + \frac{V_c}{2} (r_c - 1) * (R + 1 - \cos(\theta) - \sqrt{R^2 - \sin^2(\theta)}) \quad (4.2)$$

$$V_c = \frac{\pi * B^2 * 2R}{4 * r_c} \quad (4.3)$$

kjer so:

V_c = minimalni volumen (ko je bat v ZML) [dm³]
 R = razmerje med ročično gredjo [mm] in ojnico [mm] ($\frac{l}{a}$)
 B = premer bata [mm]
 θ = kot v radianih ($-\pi$ do π za dvotaktni in -2π do 2π za štiritaaktni motor)
 r_c = kompresijsko razmerje ($\frac{V_{max}}{V_{min}}$)

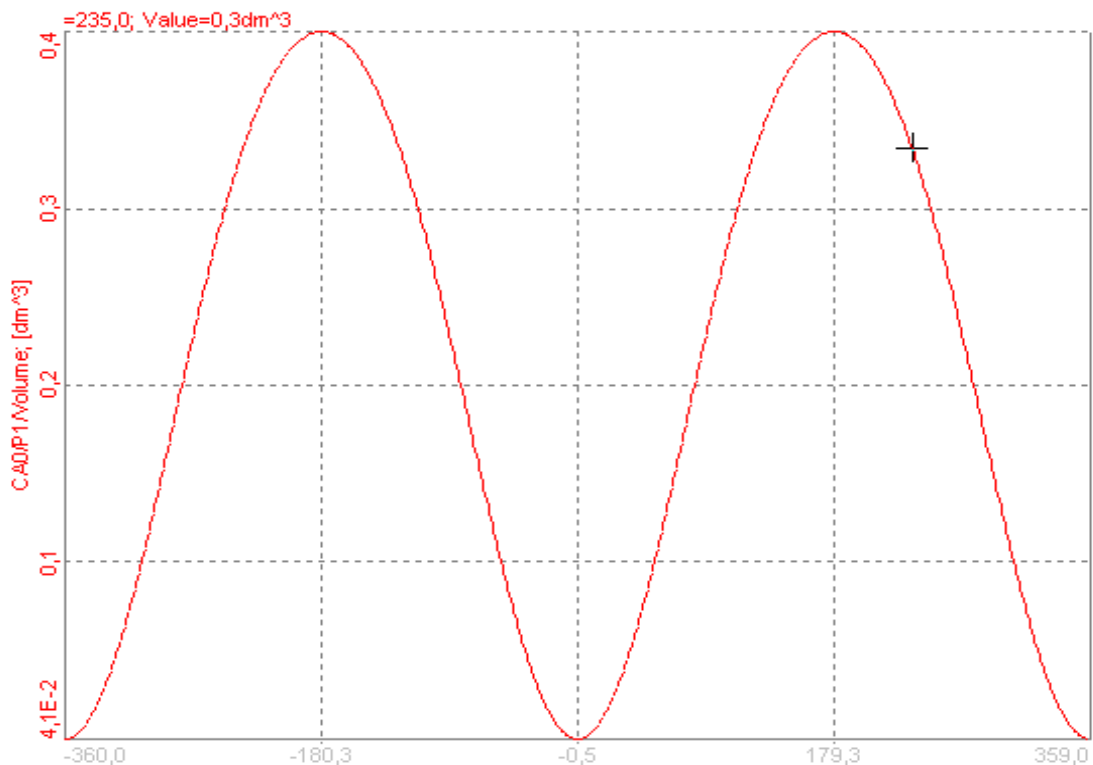
Glede na vpisane parametre se tudi izračunata minimalni in maksimalni volumen valja. Velja:

$$V_{max} = \frac{\pi * B^2 * L}{4} \quad (4.4)$$

kjer je:
 L = hod bata [mm]

$$V_{min} = \frac{V_{max}}{r_c} \quad (4.5)$$

Na sliki 8 lahko vidimo primer diagrama poteka volumna valja v odvisnosti od kota za en cikel štiritaaktnega motorja. Najmanjši volumen nastopi pri 0 kotnih stopinjah (ZML), največji pa pri -180 in 180 kotnih stopinjah (SML).

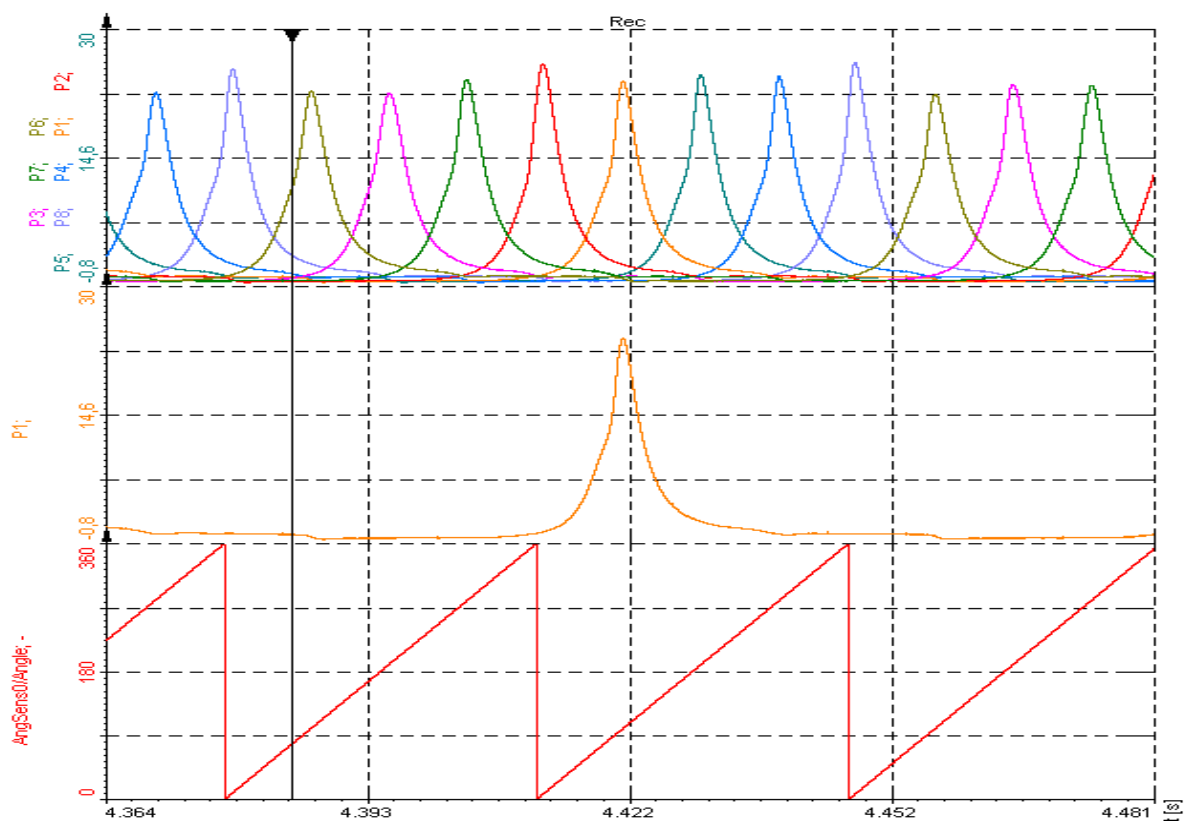


Slika 8. Volumen valja [dm³] na osi y in en obrat motorja na osi x [kotne stopinje]

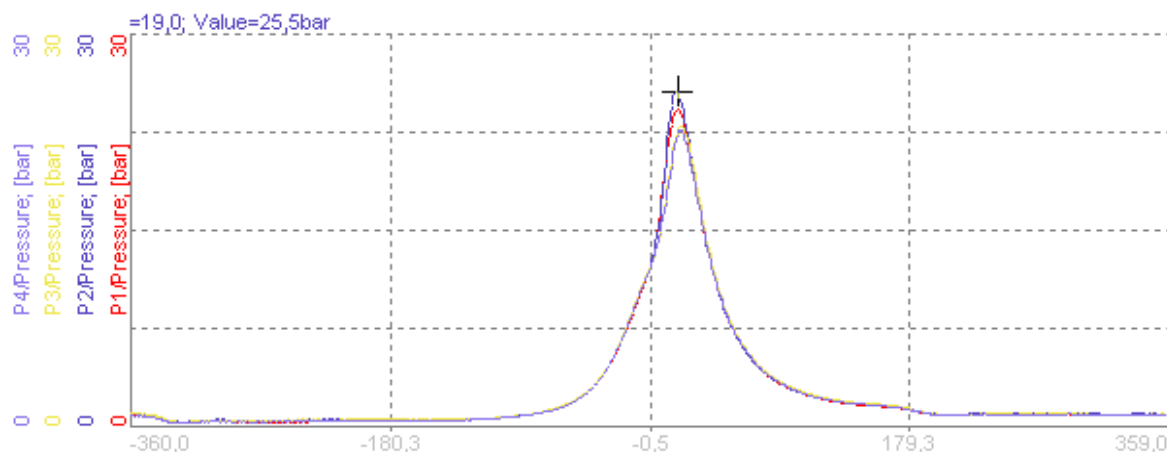
Vsak motor ima različno število valjev, zato je treba nastaviti tudi ta parameter. Za vsak valj je treba določiti, kateri analogni kanal z vrednostmi tlaka mu ustreza. Potem je treba nastaviti zamik kota za vsak valj. Bati se prožijo drug za drugim s fiksnim zamikom, določenim v kotnih stopinjah (slika 9). Če hočemo analogne podatke prestaviti v kotno domeno, moramo poznati ta zamik in ga vpisati v analizator. Treba je vpisati tudi zamik prvega valja, glede na ničlo senzorja za kot. Slika 10 nam prikazuje štiri tlake na štiriktaktnem motorju, poravnane v kotni domeni.

V vsakem valju se lahko merijo še drugi podatki (npr. začetek vžiga, temperatura ...). Če hočemo, da so ti podatki na x-osi poravnani s podatki za tlak, moramo v nastavitvah za trenutni valj izbrati tudi ta kanal.

Valj ima prav tako možnost meritve začetka in konca vbrizga goriva. Navadno imajo senzorji za vbrizg stopnico tam, kjer je prišlo do vbrizga goriva, zato moramo izbrati raven za začetek vbrizga in raven za konec vbrizga. Analizator podpira tudi možnost več vbrizgov.



Slika 9. Zamiki proženja valjev v časovni domeni. Na osi x je čas [s], na osi y pa je na spodnjem diagramu kot motorja [kotne stopinje], na zgornjih dveh pa je tlak [bar]



Slika 10. Tlaki v valjih, v kotni domeni, poravnani glede na ZML. Na y-osi je tlak [bar], na x-osi pa je prikazan en obrat motorja [kotne stopinje]

4.3 Predloge za določen motor

Programski paket Dewesoft shranjuje vse nastavitve za določeno meritev v nastavitvene datoteke. Te datoteke so zapisane v formatu XML. Prednost XML-a je v tem, da se da nastavitvene datoteke odpirati tudi v drugih urejevalnikih in popravljati določene nastavitve, zatem pa znova naložiti v programskem paketu Dewesoft.

Velikokrat želimo imeti eno nastavitveno datoteko z vsemi globalnimi nastavitvami, med posameznimi meritvami pa bi radi spreminjali samo geometrijo motorja, ker na motorju spreminjamo različne nastavitve. To smo dosegli tako, da smo dodali predloge za različne motorje. Uporabnik lahko te predloge doda, odvzame in spremeni. Trenutna geometrija se shrani v trenutno izbrano predlogo. Ko jo zamenjamo, se spremeni samo geometrija motorja, vse druge nastavitve pa ostanejo takšne, kot so bile naložene iz nastavitvene datoteke. Tudi te predloge so narejene v XML-u.

4.4 Senzor za kot in frekvenco

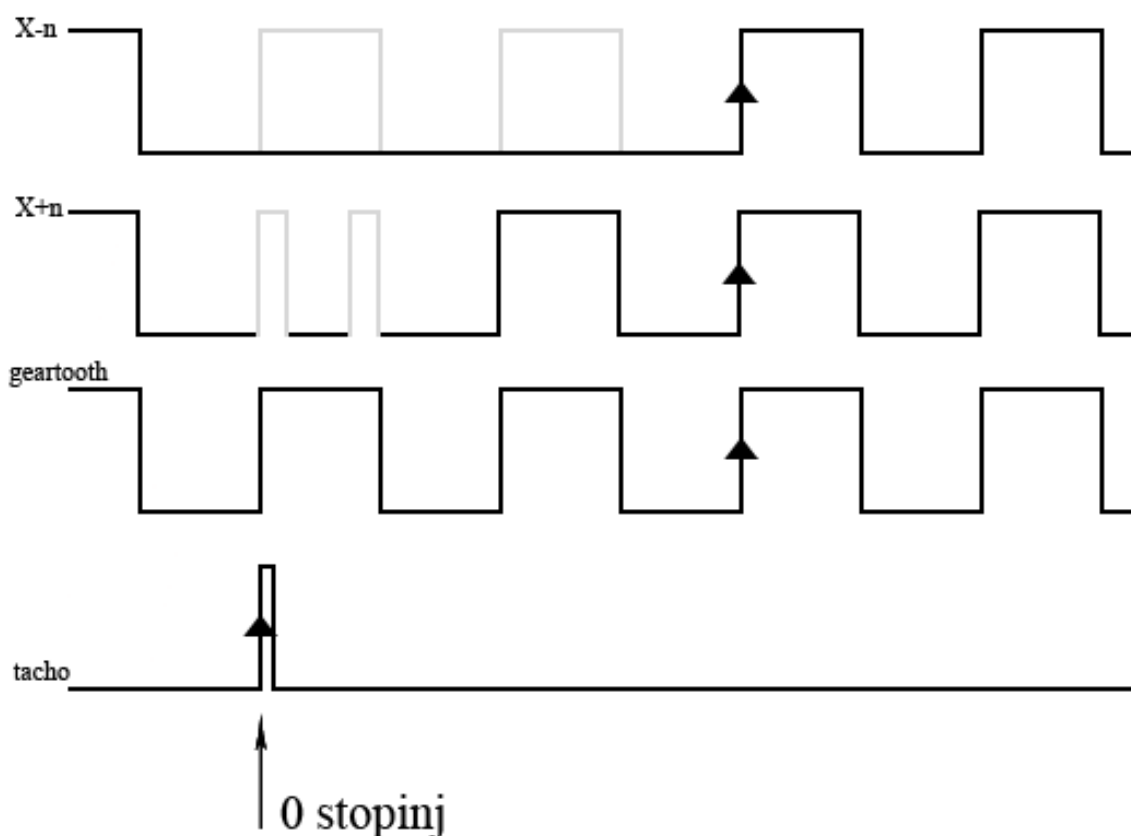
Natančnost večine izračunov je zelo odvisna od meritve točne pozicije tlaka v kotni domeni. Zahtevana natančnost za takšno meritev je lahko manjša od 0,1 kotne stopinje, kar pomeni, da potrebujemo res natančno meritev kota.

Najbolj tipični senzorji glede na izvedbo so:

- **Tacho senzor.** Izhod iz tachosenzorja je en impulz na obrat. Senzor ni preveč primeren za takšne aplikacije, ker moramo vmesne kote interpolirati, s tem pa ne dobimo deviacije hitrosti znotraj enega obrata.
- **Geartooth senzor.** Ima več impulzov na obrat, nima pa signala za reset. Zaradi tega ni najprimernejši senzor, ker ne vemo, kdaj se je izvedel en delovni takt.
- **CDM.** Ta senzor je enak kot geartooth, vendar ima še impulz za reset. Zato je zelo primeren za meritve kota na rotirajočih strojih.

- **Encoder.** Opisan je v poglavju 3.1. Ta senzor je najprimernejši za te aplikacije. Problem pa je, da ga je velikokrat težko vgraditi v motor. Zato se večinoma uporabljajo spodaj naštetih senzorji.
- **Gear tooth with missing teeth (X-n).** Velikokrat ga že proizvajalec vgradi v motor. Senzor ima nekaj zob z istim razmikom, potem pa jih nekaj vmes manjka. Prvi zob, ki manjka, lahko vzamemo kot ničlo. Takšen tip senzorja je velikokrat že vgrajen v motor. Predstavniki te skupine senzorjev je npr. senzor 60-2, ki je bolj podrobno opisan v nadaljevanju besedila.
- **Gear tooth with double teeth (X+n).** Tudi tukaj je večina zob z enakim razmikom, nekaj pa jih je bolj skupaj. Tako lahko te zobe spet vzamemo kot impulz za ničlo.
- **Tape sensor.** Ta senzor je pravzaprav samo trak z belimi in črni polji. Nalepimo ga okrog gredi motorja. Tam, kjer se konca stakneta, nalepimo večjo površino v črni barvi in tako dobimo impulz za ničlo. Za razpoznavo belih polj, ki služijo kot impulzi za vrednost kota pa se uporabi laserski ali pa optični senzor. Senzor ima TTL kompatibilen izhod oz. napetost v primeru optičnega senzorja.

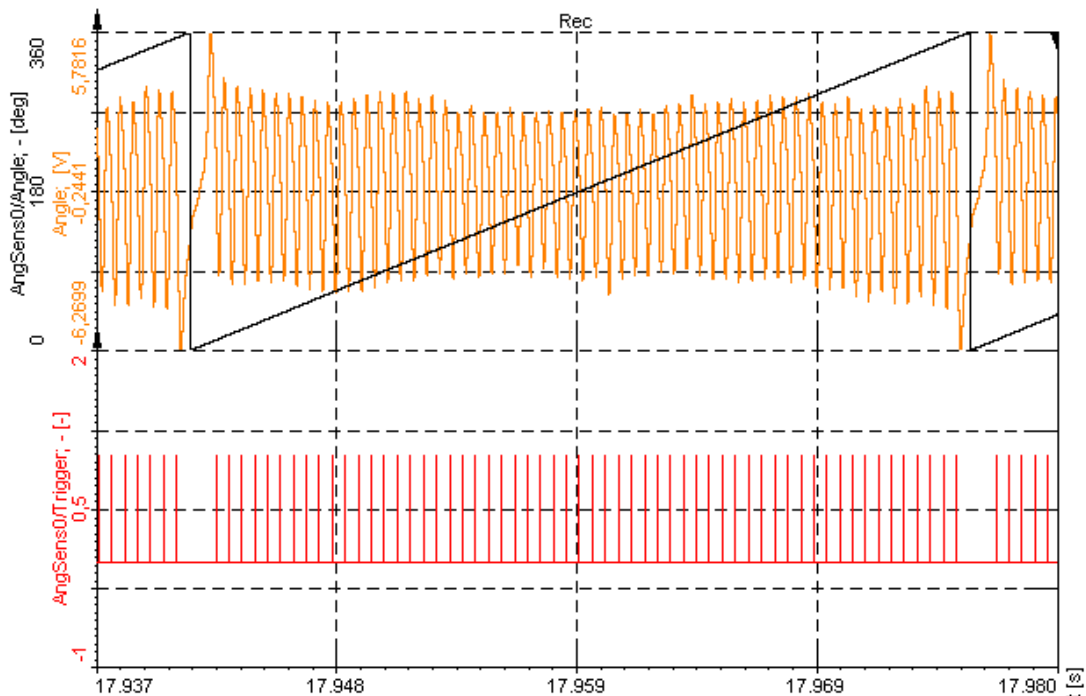
Slika 11 prikazuje digitalne signale nekaterih zgoraj naštetih senzorjev. S svetlo sivo so obarvani manjkajoči oz. podvojeni impulzi, oznaka 0 kotnih stopinj pa prikazuje začetek enega cikla.



Slika 11. Prikaz digitalnih signalov različnih senzorjev

Na sliki 12 lahko na spodnjem grafu vidimo točke proženja za senzor 60-2 (geartooth with missing teeth). Lepo se vidi, kje manjkata dva impulza. Na zgornjem grafu (slika 12) pa

vidimo analogni signal za ta tip senzorja (oranžna barva) in pa interpolirano vrednost za kot 0–360 kotnih stopinj (črna barva).

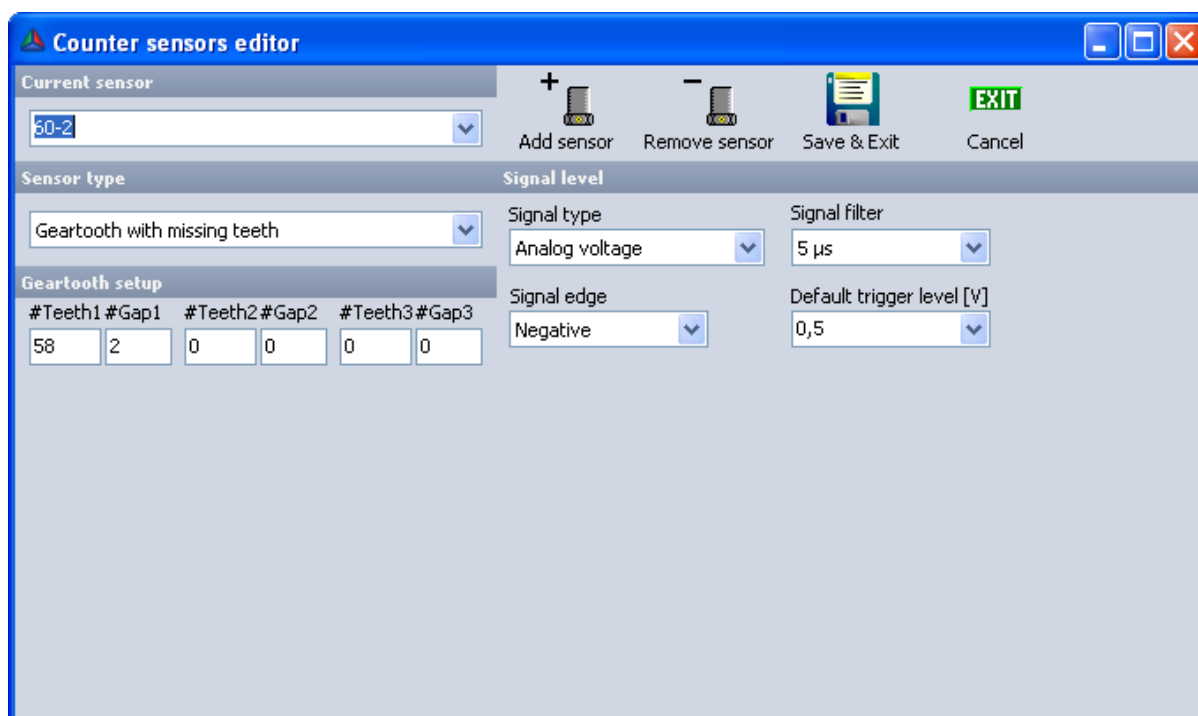


Slika 12. Izračun kota in proženje za senzor 60-2. Na x-osi je čas [s], na y-osi pa napetost [V]

V analizatorju je treba izbrati vrsto senzorja, na kateri kanal je priključen in resolucijo meritve. Vse našteje senzorje razen enkoderja lahko priključimo na digitalni ali pa analogni vhod na A/D-pretvornik. Ker osnovni podatki s senzorja za kot niso dovolj, da bi dosegli zahtevano resolucijo, je bilo treba izdelati dva posebna modula, ki zelo točno interpolirata kote med dejanskimi vrednostmi s senzorja. Pri digitalnem senzorju se lahko pozicije določijo veliko natančneje kot pri analognem.

4.4.1 Urejevalnik senzorjev za kot

Vsi senzorji za kot imajo vsaj nekaj osnovnih nastavitev. Programski paket Dewesoft omogoča preprosto dodajanje senzorjev in nastavljanje osnovnih parametrov z urejevalnikom senzorjev za kot (slika 13). Ti senzorji so dosegljivi z vseh aplikacij znotraj Dewesofta. Tipični parametri senzorja so: digitalni ali analogni senzor, filter, število impulzov na obrat itn. Zanimiva nastavev je filter. Na signal senzorja vplivajo motnje, ki se izražajo kot prehod na visoko raven signala in nazaj na nizko. Ti prehodi so veliko krajše kot pravi impulz. Zato imamo možnost nastaviti, koliko časa mora biti signal na visoki oz. nizki ravni, da se impulz šteje kot pravilen. Vse druge se označi kot motnje in se podatke zavrže.



Slika 13. Urejevalnik senzorjev za kot

4.4.2 Analogni senzor

Analogni senzori se uporabljajo samo pri internem vzorčenju. Pri analognem senzoru moramo kot osnovo nastaviti napetostni nivo proženja. Kadar signal vrednost preseže, to štejemo kot pravilen impulz. Ker je na senzorju velikokrat možno zaznati tudi šum, je treba nastaviti še dodatno raven proženja. Šele ko signal preseže drugo raven, je spet pripravljen na proženje. Prožimo lahko na pozitivno ali pa negativno fronto signala.

Vzemimo za primer senzor 60-2. Ta tip senzorja se največkrat uporablja v industriji, ker je že vgrajen v različne avtomobilske motorje. Ima 58 impulzov, potem pa dva manjkata. Na prvem manjkajočem impulzu je ničla. Vsaka visoka raven na tem signalu pomeni 6 kotnih stopinj, zadnji večji razmik pa prinese 12 kotnih stopinj. To pomeni, da je resolucija tega senzorja 6 kotnih stopinj, zadnja dva impulza pa imata slabšo resolucijo. Ker za analizator potrebujemo natančnejšo informacijo o kotu (vsaj 2 kotni stopinji), je treba vmesne vrednosti nekako dodati. Kot dobra možnost se je izkazala linearna interpolacija.

Vrednost interpoliranega kota določimo takole:

$$A_m = A_{n-1} + \frac{A_n - A_{n-1}}{n} * i \quad (4.6)$$

kjer so:

A_{n-1} = prejšnja prebrana vrednost za kot [kotne stopinje]

A_n = trenutna prebrana vrednost za kot [kotne stopinje]

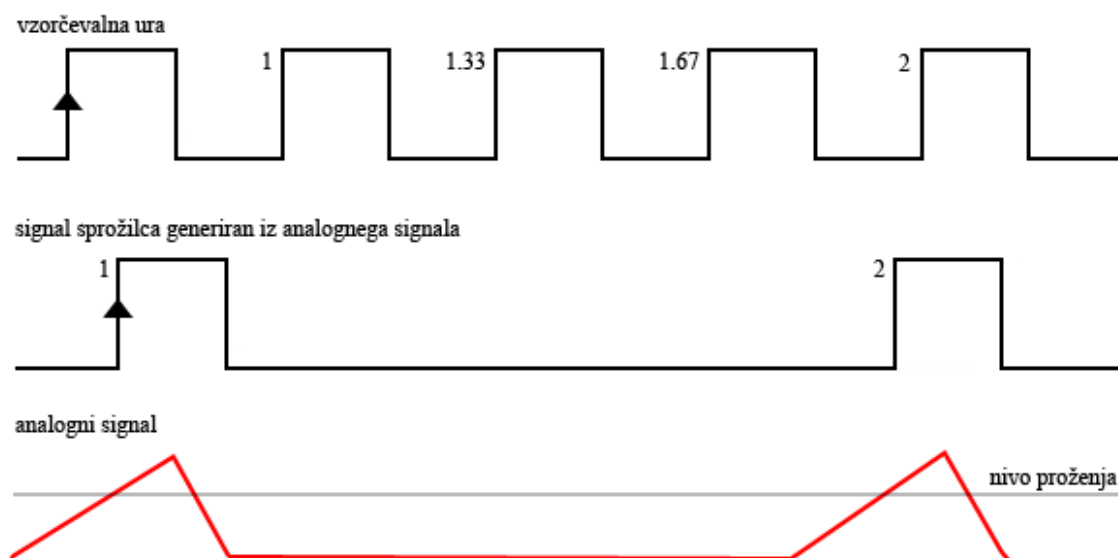
A_m = trenutna vrednost za kot [kotne stopinje]

n = število podatkov med dvema prebranimi kotoma

i = vrednost zajetega vzorca od prejšnjega do trenutnega kota (0 ... n)

Zelo pomembno pa je tudi, kako določimo trenutno vrednost kota. Če vrednost vzorca preberemo neposredno z A/D-pretvornika, je lahko nepravilna. Ta napaka je odvisna od frekvence vzorčenja f_s in je med 0 in $1/f_s$. To je posledica tega, da se vrednost vzorca lahko spremeni kadar koli (zvezno), mi pa ga v računalniku lahko preberemo samo v diskretnem času (diskretno vzorčenje podatkov). Nižja kot bo frekvenca vzorčenja, večja bo ta napaka v kotni domeni. Napake se pri analognih senzorjih ne da odpraviti, lahko pa se jo odpravi pri digitalnih.

Na sliki 14 vidimo primer analognega signala (rdeča barva). Ko analogni signal preseže vrednost prožilca, bi mi morali prebrati vrednost za kot (to prikazuje signal na sliki 14 v sredini). Ker pa naša vzorčevalna ura teče asinhrono s časovnimi podatki pa zaznamo pravilno vrednost šele nekaj časa kasneje (to prikazujejo številke na vzorčevalni uri – dejanske prebrane vrednosti). Kadar se motor vrti počasneje od vzorčevalne frekvence nam manjkajo podatki, katere moramo interpolirati z že omenjeno linearno interpolacijo. V spodnjem primeru sta to vrednosti 1.33 in 1.67, realno prebrana podatka pa sta 1 in 2 kotne stopinje.



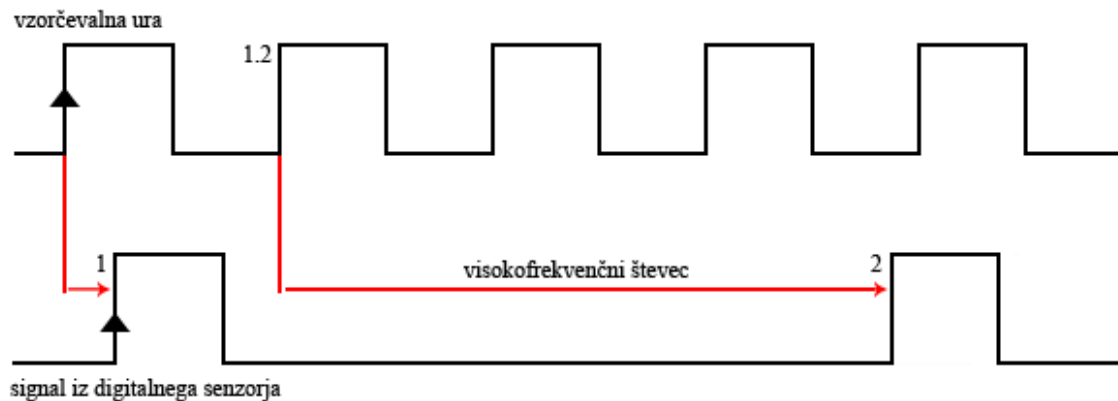
Slika 14. Interpolacija analognega senzorja

4.4.3 Digitalni senzor

Če uporabljamo eksterno vzorčenje, ne potrebujemo nobenega modula za pretvorbo podatka za kot, ker senzor vodi vzorčenje in so podatki zajeti že na pravih vrednostih kota. Če pobiramo podatke z interno uro imamo iste težave kot pri analognem senzorju.

Senzorji, ki jih priklopimo na digitalni vhod A/D-pretvornika, pa imajo to prednost, da lahko odpravimo napako vzorčenja, omenjeno v prejšnjem poglavju. To naredimo tako, da imamo še en dodaten visokofrekvenčni števec. V našem primeru števec šteje s frekvenco 80 MHz. Ta števec šteje od resničnega impulza na senzorju pa do začetka A/D-vzorca (slika 15). S tem pa lahko naslednji prebrani vrednosti za kot prištejemo napako A/D-pretvornika in tako dobimo zelo točno pozicijo za trenutno vrednost kota. Napaka je omejena z $1/80$ MHz in ni odvisna od vzorčevalne frekvence. To pomeni, da lahko z nizko vzorčevalno frekvenco dosežemo zelo

točno vrednost kota. Interpolacija drugih kotnih vrednost pa je tako kot za analogni senzor linearna.



Slika 15. Interpolacija digitalnega senzorja

4.5 Zajem podatkov tlaka z eksterno uro

Pri zajemu z eksterno uro senzor za kot vodi vzorčenje podatkov. Zaradi tega so vsi podatki s senzorjev za tlak, temperature itn. zajeti že v kotni domeni. Naloga analizatorja je samo, da vzame podatke od -180 do 180 za dvotaktni oz. od -360 do 360 kotnih stopinj za štiritaletni motor in jih pripravi za izračune in vizualizacijo. Imamo pa dve težavi:

- pojavi se t. i. aliasing. Frekvence, višje kot je frekvenca vzorčenja, se preslikajo v nižje frekvence in s tem povzročajo harmonike v frekvenčnem spektru signala. Tega bi se lahko znebili z antialiasing filtrom, ki ga nastavimo na polovico frekvence vzorčenja. Ker pa se hitrost motorja ves čas spreminja (tudi znotraj enega cikla), pri zunanji uri tega ne moremo določiti in tako ne moremo nastaviti filtra;
- resolucija je zelo odvisna od senzorja za kot. Če uporabimo prej omenjeni senzor 60-2, je resolucija meritve 6 kotnih stopinj, kar je za večino izračunov premalo, ker so rezultati pregrobi. Resolucijo meritve pa smo lahko povečali, tako da smo uporabili Dewetronovo napravo, imenovano CA-CPU.

4.5.1 CA-CPU

Naprava se priklopi med senzor za kot in digitalni vhod na A/D-pretvorniku. Iz zgodovine podatkov poizkuša napovedati, kdaj bo prišel naslednji impulz, in tako lahko vnaprej ekstrapolira vrednosti. Podpira vse našteje vrste senzorjev in frekvenco vhodnega signala do 1 MHz (št. impulzov senzorja * hitrost vrtenja motorja [Hz] < 1 MHz). Impulze iz vhodnega senzorja zna povečati do 100-krat. To pomeni: če imamo enkoder s 360 impulzi, ga lahko s faktorjem 10 povečamo na 3600 impulzov, kar pa ustreza resoluciji 0,1 kotne stopinje. Tudi prej omenjeni 60-2 senzor lahko povečamo za faktor 60 in spet dobimo resolucijo 0,1 kotne stopinje.

4.6 Zajem podatkov tlaka z interno uro

Pri zajemu podatkov z interno uro se podatki zajamejo neodvisno od obratov motorja. Z digitalnimi vhodi se zajame signal senzorja za kot. Ko imamo enkrat podatke kota, lahko druge podatke prestavimo v kotno domeno. Težava pa je v tem, da so podatki zajeti enakomerno s časovnim intervalom $1/(\text{frekvenca vzorčenja})$, ne pa enakomerno z vrednostjo kota. Zato moramo podatke nekako interpolirati na pravilne vrednosti (interval od -180 do 180 za dvotaktni oz. od -360 do 360 kotnih stopinj za štiritaletni motor), ker na teh podatkih temeljijo vse enačbe in tudi vizualizacija. Dobra lastnost zajemanja z interno uro je, da lahko na teh podatkih izvajamo vse standardne matematike (fft, filtri, itn.) in pa da lahko uporabljamo nizkoprepustni filter, da se znebimo t. i. alisov. Zelo pomembna lastnost je tudi, da lahko resolucijo nastavljamo neodvisno od števila impulzov senzorja, tako da podatke interpoliramo med dvema zajetima vrednostma za kot z želeno kotno resolucijo, saj je to veliko natančneje kot ekstrapolacija, ki jo pri eksterni uri izvaja CA-CPU. Slaba lastnost je, da je potrebna velika frekvenca vzorčenja in posledično tudi velika količina podatkov, ki predstavlja veliko obremenitev za računalnik. Recimo, da se motor vrti s 3000 obrati na minuto oz. s frekvenco 50 Hz. Radi bi merili z resolucijo 1 kotne stopinje, senzor za kot pa ima 360 impulzov.

Minimalno frekvenco vzorčenja (f_s) izračunamo po naslednji formuli: št. impulzov na obrat * frekvenca vrtenja * 2 (Nyquist). Torej:

$$f_s = 360 * 50 * 2 = 36000 \text{ Hz} \quad (4.7)$$

Lahko si predstavljamo, kaj se zgodi, če bi radi računali podatke z resolucijo 0,1 kotne stopinje.

Obstaja veliko interpolacijskih metod, naj naštejemo le:

- linearna interpolacija
- polinomska interpolacija
- metoda najbližjega soseda
- ...

Težava je v tem, da vsaka krivulja zahteva različne načine interpoliranja. In ne samo to, tudi na različnih delih bi bilo treba krivuljo različno interpolirati. Če vzamemo npr. krivuljo tlaka: bi se večina krivulje dala lepo interpolirati z linearno interpolacijo, v fazi stiskanja in delovnega takta pa polinom druge stopnje načeloma lepo opiše dogajanje. Če bi poizkusili tako interpolirano krivuljo odvajati, bi dobili nezveznosti. Linearna interpolacija v frekvenčnem spektru povzroči odziv na stopnico in s tem povzroči, da kakršni koli filtri, odvodi ... niso več gladka funkcija (pravimo, da je funkcija gladka, če v vsaki točki obstaja tangenta). Zato se v digitalni tehniki za interpolacijo uporabljajo filtri. S tem se ohrani harmonska vsebnost originalnega signala brez dodajanja aliasov nad Nyquistovo omejitvijo signala (torej nad $f_s/2$).

Za takšno interpolacijo lahko uporabimo dve vrsti digitalnih filtrov:

- NEO-filter
- KEO-filter

4.6.1 Digitalni filtri

Digitalni filter je diskretni sistem, ki izvaja matematične operacije na diskretnih podatkih, tako da spremeni spektralne komponente signala. Filter opisuje prenosna funkcija. Glede na namen uporabe jih delimo na:

- nizkoprepustne: prepuščajo frekvenčne komponente z nižjo frekvenco od mejne frekvence;
- visokoprepustne: prepuščajo frekvenčne komponente z višjo frekvenco od mejne frekvence;
- pasovno prepustne: prepuščajo neki frekvenčni pas;
- pasovno zaporne: prepuščajo vse drugo, razen določenega frekvenčnega pasu.

Glede na način realizacije pa jih delimo na:

- filtre v frekvenčnem prostoru,
- NEO-filtre,
- KEO-filtre.

4.6.1.1 Filtri v frekvenčnem prostoru

Signal s pomočjo diskretne Fouriereve transformacije (DFT) prestavimo v frekvenčno domeno. Tam ga lahko pomnožimo z želeno transferno karakteristiko in ga s pomočjo inverznega dft-ja preslikamo nazaj. Največja težava je v tem, da smo vzeli ven odsek originalnega signala, ga pomnožili z oknom (da se znebimo t. i. odtekanja oziroma leakagea). Ko takšen signal transformiramo nazaj v časovno domeno (iz frekvenčne), ni več zvezen na obeh koncih in je zato v našem primeru neuporaben.

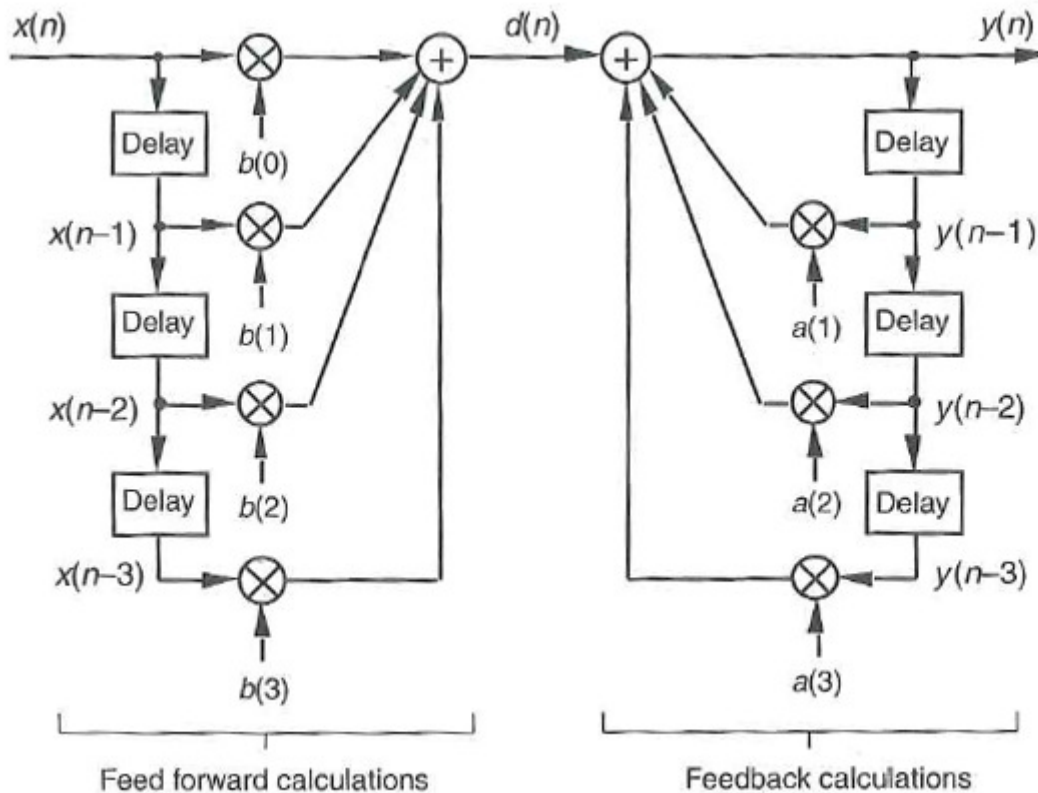
4.6.1.2 Osnove NEO-filtrov

NEO-filter pomeni filter z »neskončnim enotnim odzivom« oz. infinite impulse response (IIR). Vsak izhod iz takšnega filtra temelji na prejšnjih izhodih iz filtra, za razliko od KEO-filtrov, ki za vsako izhodno vrednost signala uporabljajo samo vhodni signal.

To ima dobre in slabe strani. Dobra je, da za zelo ostre filtre ne potrebujemo veliko koeficientov. Red filtra se označi s številkami od 1 ... n. Strmina prehoda iz prepustnega v zaporni pas filtra se izračuna iz: red filtra * 20 dB. To pomeni, da filter prvega reda reže 20 dB, filter 10. reda pa 200 dB na dekada. Današnji najboljše merilni sistemi lahko dosežejo 160 dB na dekada s tehnologijo dvojnega območja, kar pomeni, da je filter 10. reda zelo oster. Za implementacijo takšnega NEO-filtra potrebujemo 20 koeficientov, kar je zelo malo v primerjavi s KEO-filtri. Slaba lastnost pa je, da ima takšen filter nelinearno fazo. Ta fazni zamik lahko izničimo samo tako, da signal filtriramo naprej, potem pa nazaj (od zadnjega vzorca proti prvemu). To pa je izvedljivo samo v analizi, v meritvi tega ne moremo storiti. Težava je lahko tudi v izdelavi filtra, v določenih primerih lahko postane nestabilen.

Prenosna funkcija NEO-filtra v z-prostoru je videti takole [7]:

$$H(z) = \frac{\sum_{i=0}^P b_i z^{-i}}{1 + \sum_{j=1}^Q a_j z^{-j}} \quad (4.8)$$



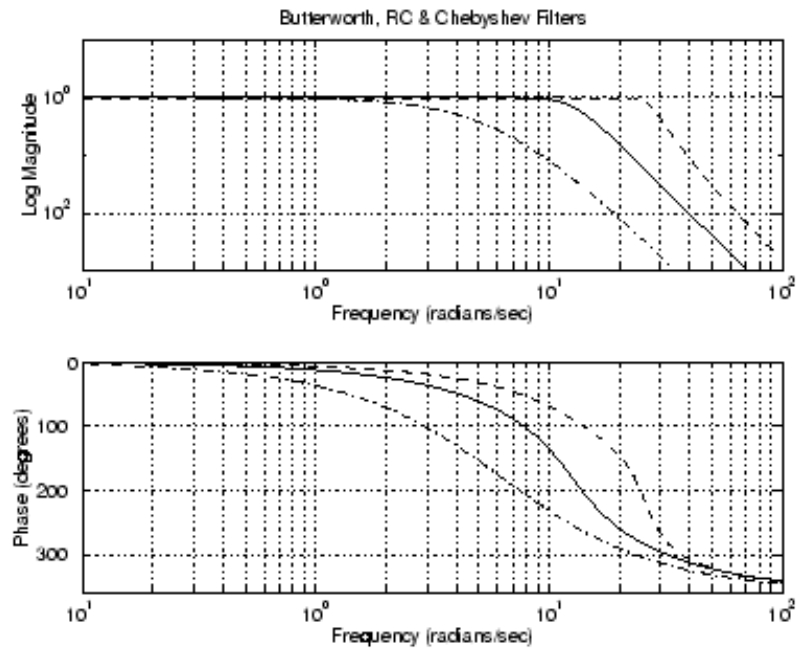
Slika 16. Računanje NEO-filtra [7]

Obstajajo različne vrste analognih filtrov na osnovi katerih projektiramo NEO-filtre:

- Butterworth
- Bessel
- Chebyshev
- Elliptic
- ...

Ti prototipi izhajajo iz analognih filtrov in so matematično opisani v Laplaceovi domeni. Tak filter se po navadi zapiše v z-domeni in se prek bilinearne transformacije koeficiente prestavi v časovno domeno, v kateri potem tudi filter apliciramo na podatke.

Razlika je predvsem v tem, kako ostro reže frekvence in koliko spremeni fazo (slika 17). Bessel najmanj kvari fazo, vendar pa poreže manj decibelov na dekada kot recimo Chebyshev. Chebyshev bolj reže, vendar pa močno pokvari fazo in povzroča valovanje v prepustnem in zapornem pasu. To pomeni, da tam signal zazveni. Butterworthov filter pa je ravno nekje vmes. Za interpolacijo podatkov ti filtri niso najprimernejši, ker pokvariyo fazo in ker so odvisni od prejšnjih vhodov. Mi pa bi radi interpolirali vrednosti kjer koli med obstoječimi vzorci. Za interpolacijo so KEO-filtri primernejši.



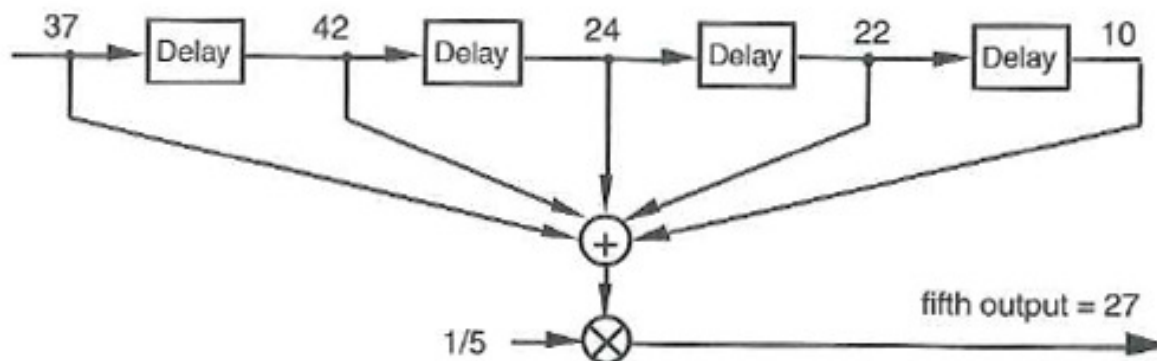
Slika 17. Primer obnašanja amplitude in faze nekaterih filtrov

4.6.1.3 Osnove KEO-filtrov

Filtri KEO (FIR) uporabljajo samo trenutne vzorce vhodnega signala za trenutni izhod filtra (niso rekurzivni tako kot NEO). Če damo v tak filter končni vhodni signal bo izhodni signal filtra končen. KEO-filtri so zelo podobni povprečenju podatkov, pravzaprav so se na začetku uporabljali prav za ta namen – da se na signalu poreže špice (tj. visoke frekvence), pozneje pa so ugotovili, da lahko spremenijo vrednosti in število koeficientov filtra in s tem še bolj vplivajo na vhodne podatke. Transferna krivulja je videti takole [7]:

$$H(z) = \sum_{i=0}^P b_i z^{-i} \quad (4.9)$$

KEO-filter se tako kot NEO-filter računa kot konvolucijo vhodnega signala in jedra filtra (slika 18). Matematična operacija konvolucija pomeni, da vhodni signal pomnožimo z drugim (filtrom), se na vhodnem signalu pomaknemo za en vzorec naprej in stvar ponovimo. Po definiciji konvolucije je treba koeficiente filtra obrniti okrog, ker pa sem v aplikaciji uporabljal simetrične filtre, lahko ta korak izpustimo. Simetrični filter pomeni, da ima levo in desno od sredinskega vzorca enake koeficiente.



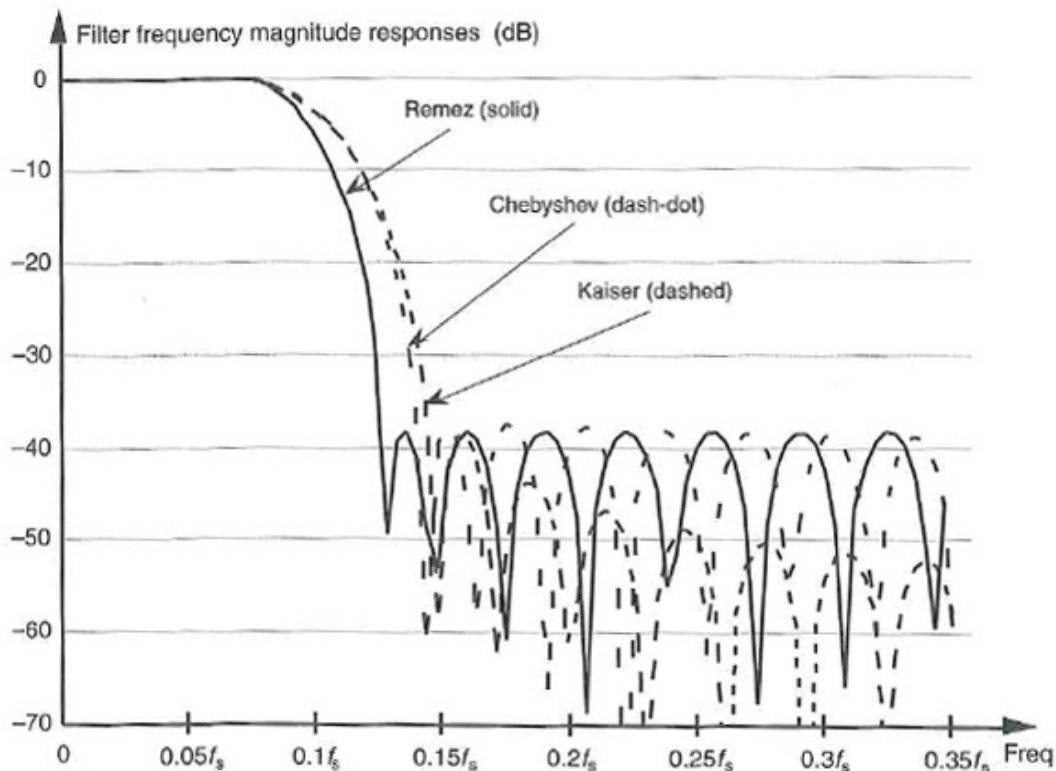
Slika 18. Diagram konvolucije KEO-filtra [7]. Številke zgoraj predstavljajo vhodne vrednosti v filter. Desno spodaj pa je izhodna vrednost filtra

Na lastnosti filtra vplivajo število koeficientov filtra (»taps«) in vrednosti teh koeficientov. Velja naslednja enačba:

$$y(n) = h(k) \times x(n) \underset{IDFT}{\overset{DFT}{H(m) * X(m)}} \quad (4.10)$$

pri čemer \times pomeni konvolucijo, DFT pa je diskretna Fourierjeva transformacija. Enačba pravi, da je konvolucija v časovni domeni enaka množenju v frekvenčni domeni. Ker pa je DFT reverzibilen, sledi, da je množenje v frekvenčni domeni enako konvoluciji v časovni domeni [10]. To nas pripelje do preprostega algoritma za načrtovanje KEO-filtra – metode okna (»window design method«).

V frekvenčnem prostoru določimo obliko idealnega filtra. Na tem filtru naredimo inverzno diskretno Fourierjevo transformacijo in dobimo filter v časovnem prostoru. Število linij DFT-ja nam pove, koliko koeficientov bo imel filter in s tem kako ostro bo rezal. Pri izdelavi filtra pride do tega, da ima filter valove v prepustnem in zapornem pasu. To pa zato, ker je vhodni signal v filter neskončen, mi pa nanj postavimo kvadratno okno (naš idealni filter). Pri prehodih tega okna iz ničle v enico nam filter zazveni in ustvari valove. To lahko rešimo oz. napako minimiziramo tako, da podobno kot pri Fourierjevi transformaciji (zaradi leakagea) uporabimo okno. Uporabimo lahko vsa standardna okna (Hanning, Hamming, Blackman, Kaiser itn.), odvisno od tega, kakšno karakteristiko filtra želimo (slika 19). Najprimernejša sta Chebyshev in Kaiser, ker imata nastavljivo vrednost za stop band v decibelih na dekada. Tako lahko naredimo filter, ki najbolj ustreza naši specifikaciji. Pri drugih oknih je težko predvideti, kako ostro bo filter rezal vhodni signal. To metodo sem uporabil tudi za izdelavo filtra za prevzorčenje signala. Naj samo omenimo, da obstaja boljša, t. i. Remezova metoda (»Remez exchange KEO-filter design method«). Ta je veliko kompleksnejša, rezultat pa je optimalen filter v smislu manjšega števila koeficientov in ostrejšega rezanja podatkov [7].



Slika 19. Primerjava uporabe različnih oken (Remez, Chebyshev, Kaiser) [7]

KEO-filter ima tudi lepo lastnost, da ima linearno fazo, ker so koeficienti filtra simetrični glede na sredinski koeficient. Kar pa pomeni, da lahko to fazo kompenziramo in dobimo filter brez faznega zamika.

Kompenzacija v številu vhodnih vzorcev za filter brez faznega zamika je enaka številu t , i. tapov filtra, deljeno z dva. To pomeni, da vzamemo pol vzorcev v preteklosti in pol v prihodnosti od vzorca, ki ga trenutno računamo.

4.6.2 Sprememba frekvence vzorčenja kot tehnika DSP

Podatke vzorčimo z določeno frekvenco. Velikokrat se pojavi želja, da to frekvenco spremenimo samo na vhodnih signalih. Če frekvenco vzorčenja spremenimo na A/D-pretvorniku, bo to vplivalo na vse podatke. Tukaj pa si spet lahko pomagamo z DSP-tehnikami, kot so decimacija, interpolacija in prevzorčenje. Frekvenco vzorčenja želimo spremeniti iz različnih razlogov:

- manjša obremenitev procesorja,
- manj podatkov za računanje, zato lahko podatke obdelujemo hitreje, s tem pa damo CPE več možnosti, za dodatne obdelave,
- cenejša strojna oprema,
- manjša poraba moči (daljša doba delovanja na baterije na npr. prenosniku),
- povečanje razmerja signal-šum.

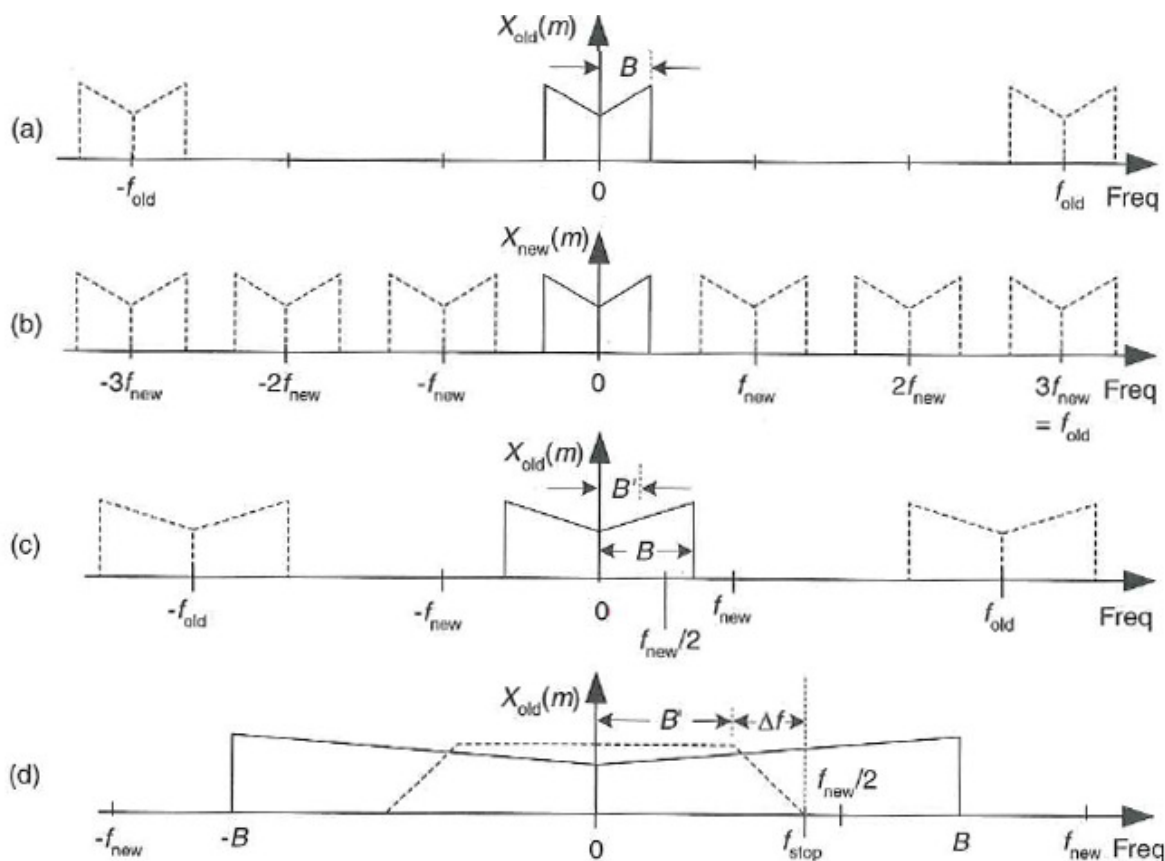
V našem primeru pa smo to tehniko potrebovali, da prestavimo podatke iz ene domene (časovna) v drugo (kotna).

4.6.3 Decimacija

Pri decimaciji zavržemo določeno število vzorcev:

$$f_{new} = \frac{f_{old}}{D} \quad (4.11)$$

To pomeni, da je nova frekvenca vzorčenja enaka stari frekvenci, zmanjšani za faktor D (ta pa mora biti celo število). Torej obdržimo vsak D-ti vzorec vhodnega signala, zavržemo pa vse druge. Ker je pasovna širina novega spektra manjša od vhodne pasovne širine signala, je treba signal prej filtrirati z nizkoprepustnim filtrom, da ne kršimo Nyquistovega pravila in dodamo aliasov v novi signal (slika 20). Za ta primer seveda uporabimo KEO-filtr, ker nima faznega zamika (če ga računamo po postopku, opisanem v prejšnjem poglavju). Za frekvenco rezanja pa določimo $\frac{f_{old}}{D}$. Ker smo število vzorcev zmanjšali za faktor D, moramo nizkoprepustnemu filtru ojačanje nastaviti na D, da dobimo pravilne amplitude na izhodu iz filtra.



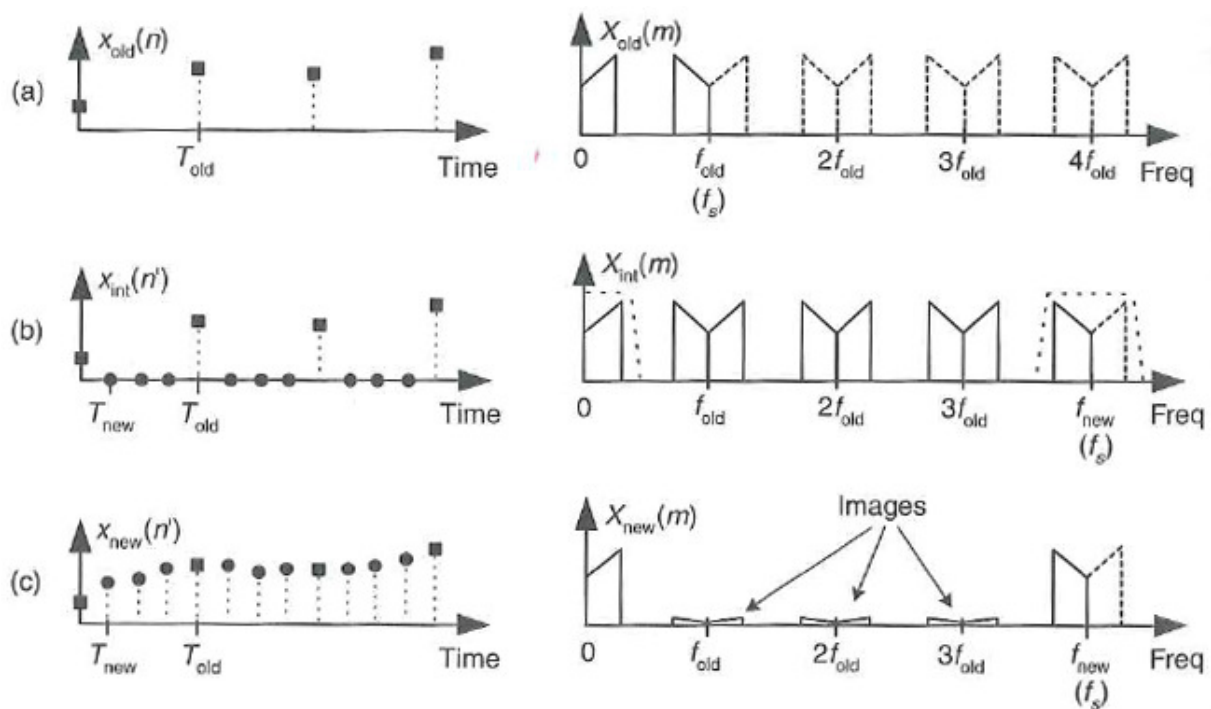
Slika 20. Frekvenčni spekter pri decimaciji vhodnega signala [7]. Vhodni signal decimiramo za faktor tri: (a) spekter originalnega signala; (b) spekter po decimaciji; (c) zahtevana pasovna širina $\gg B$; (d) frekvenca rezanja nizkoprepustnega filtra glede na pasovno širino $\gg B$

4.6.4 Interpolacija

Interpolacija podatkov pomeni, da med obstoječe vhodne podatke dodajamo nove.

$$f_{new} = f_{old} * M \quad (4.12)$$

Zgornja enačba nam pove, da bomo med obstoječe vzorce dodali M novih vzorcev. To naredimo s t. i. metodo zero-stuffing. Med obstoječe vhodne vzorce dodamo M vzorcev z vrednostjo nič, potem pa celoten signal filtriramo s KEO-filtrom (slika 21). Ker smo rekli, da je KEO neke vrste filter povprečnih vrednosti, bodo vzorci z vrednostjo 0, dobili povprečne vrednosti sosednjih vzorcev. Če bomo filter realizirali kot nizkoprepustni, bomo dobili signal s frekvenčno vsebino od 0 Hz do $f_{old} * M$. Od tega, kako dobro smo načrtovali filter (kako oster je), pa je odvisno, kako dober bo naš izhodni signal – kakšno frekvenčno vsebino nad $f_{old} * M$ bo prepuščal. Zato je zelo pomembno, kako dober KEO-filter bomo načrtovali. Tudi tukaj moramo tako kot pri filtru za decimacijo določiti frekvenčni dobiček oziroma t. i. ojačanje filtra v vrednost M .



Slika 21. Časovni in frekvenčni spekter pri interpolaciji vhodnega signala [7]. Interpolacija za faktor štiri: (a) originalni vzorčeno zaporedje in pripadajoči spekter; (b) dodane ničle v originalno zaporedje in pripadajoči spekter; (c) izhodno zaporedje interpolacijskega filtra in pripadajoči spekter

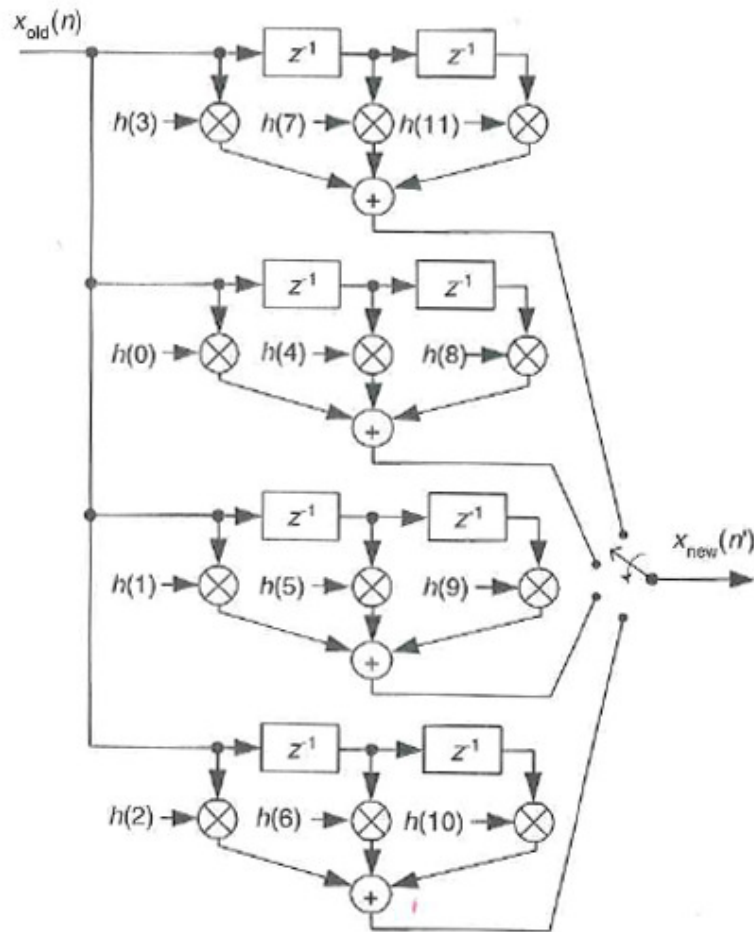
4.6.5 Prevzorčenje

Za analizator je bilo treba spremeniti frekvenco vzorčenja za neceloštevilski faktor. To lahko dosežemo s kombinacijo interpolacije in decimacije (torej M/D), in sicer interpoliramo signal za M , filtriramo z nizkoprepustnim filtrom s frekvenco rezanja M , potem pa filtriramo z nizkoprepustnim filtrom s frekvenco rezanja D in signal decimiramo. Takoj lahko opazimo, da bi lahko filter združili v enega s frekvenco rezanja M/D in s tem hitrostno optimirali algoritem. Če povzamemo: najprej povečamo frekvenco vhodnega signala za faktor M , potem filtriramo signal z nizkoprepustnim filtrom, ki ima frekvenco za rezanje nastavljeno na M/D . Nato pa signal decimiramo za faktor D . Seveda ne smemo spet pozabiti na ojačanje filtra, ki je enak razmerju M/D .

4.6.6 Polifazno prevzorčenje

Ker so zgornji algoritmi časovno zahtevni, se seveda poraja vprašanje o optimizaciji. Če dobro pogledamo interpolaciji, vidimo, da več vzorcev kot dodajamo v vhodni signal, več ničel vstavljamo v signal. Poraja se vprašanje, kaj se zgodi s temi ničlami pri filtriranju v smislu konvolucije z nič. Konvolucija je operacija množenja in na koncu seštevanja dobljenih vrednosti. Ker vemo, da da množenje z nič rezultat nič, dobimo idejo, da nam na teh mestih ni treba računati vrednosti konvolucije. Če za primer interpoliramo signal za $M = 4$, naš nizkoprepustni filter pa ima 200 koeficientov, pomeni, da lahko za en izhod filtra izračunamo samo $200/4$ vrednosti. Torej 50 vzorcev. To pa je ogromen prihranek pri hitrosti računanja.

Primer za polifazno prevzorčenje lahko vidimo na sliki 22. Slika prikazuje interpolacijo za faktor 4, pri čemer je struktura filtra realizirana polifazno.



Slika 22. Diagram prikazuje polifazno prevzorčenje signala, organizirano kot strukturo KEO filtrov [7]

4.6.7 Polifazno prevzorčenje in analizator

Filter za prevzorčenje se v analizatorju uporablja tako, da se določi maksimalna frekvenca izhodnega signala glede na frekvenco vzorčenja in želeno resolucijo kota pri meritvi. Glede na to se izdelava KEO-filtra. Med meritvijo se interpolacijski faktor spreminja glede na to, kakšna je vhodna frekvenca signala, ki se izračuna iz odvoda dveh sosednjih kotov. Tako je frekvenca našega prevzorčenega signala ves čas enaka in potrebujemo samo en filter za prevzorčenje. To pomeni, da če je vhodna frekvenca signala manjša, interpoliramo manj vrednosti, če je frekvenca višja, interpoliramo več vrednosti.

$$t_f = \frac{t_n}{D} \quad (4.13)$$

kjer sta:

t_f = število koeficientov na fazo filtra

t_n = število vseh koeficientov filtra

4.6.8 Zaključek

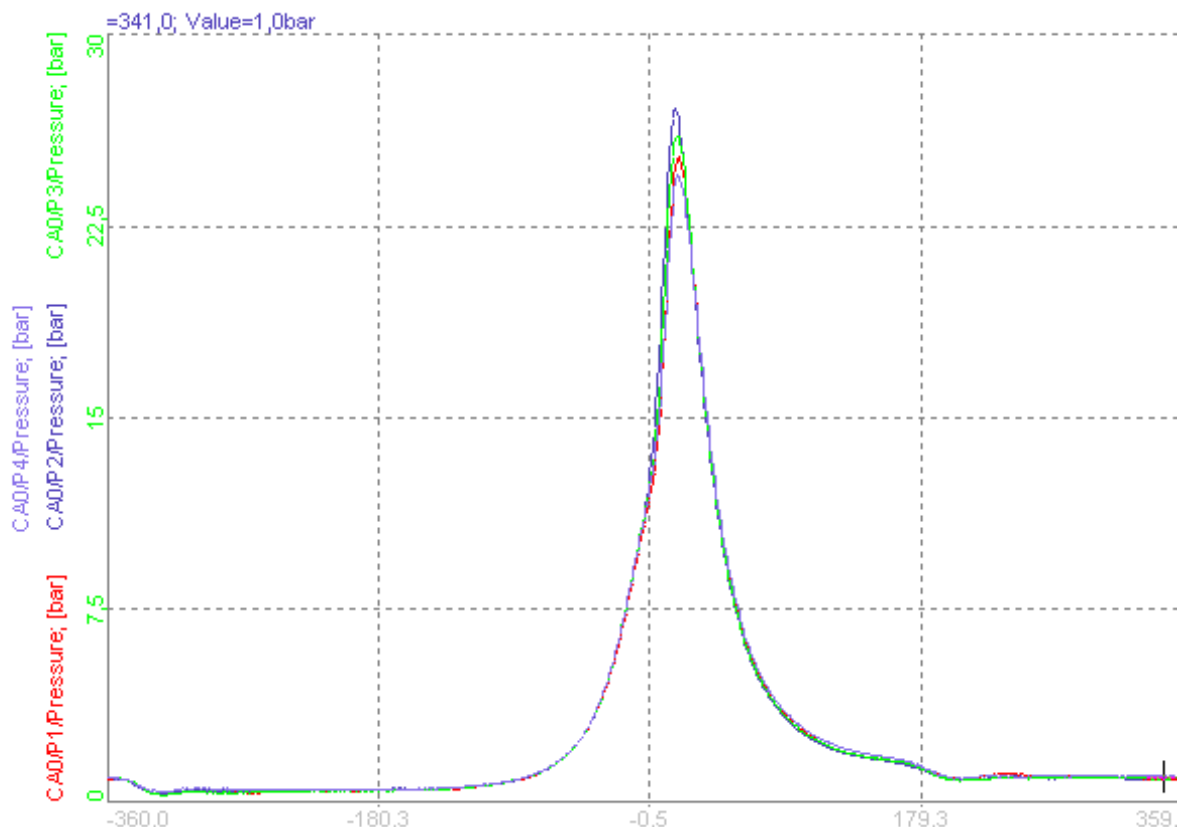
Ker lahko vidimo, da je za prevzorčenje potrebno veliko več matematičnih operacij, kot če zajemamo podatke direktno v kotni domeni, se vzorčenje z interno uro uporablja pri motorjih z nizkimi obrati oz. manjšim številom valjev in nižjo resolucijo kota. Za motorje z veliko valji, visokimi obrati in visoko resolucijo kota pa se uporablja vzorčenje z eksterno uro. Se pa tudi to počasi spreminja, ker prihajajo novi večjedrni procesorji. Intelov procesor i7 (recimo Core i7-970) ima denimo 6 jeder. To pomeni, da lahko pri motorju s šestimi valji računanje vsakega valja postavimo na svoje jedro, kar pa je veliko hitreje kot pri procesorju z enim jedrom ali dvema. Grafične kartice so še hitrejše, imajo nekaj sto jeder, zato obstaja tudi možnost, da se del računanja DSP-algoritmov prestavi na grafično kartico.

4.7 Tlak

Osnovni parameter, ki ga spremljamo za izračun najpomembnejših karakteristik motorja, je tlak v valju (slika 23). Tlak (p) je fizikalna količina, definirana kot razmerje med velikostjo ploskovno porazdeljene sile F in površino ploskve A , na katero ta sila prijema [6].

$$p = \frac{F}{A} \quad (4.14)$$

Za tlak imamo več enot (Pa, PSI, bar ...). V analizatorju se uporablja bar, ker je osnovna enota Pascal premajhna (1 bar = 100.000 Pa) za visoke tlačne vrednosti v valju in bi morali računati s prevelikimi številkami (neprijazno za uporabnika).



Slika 23. Tlak v barih za štiri valje. Na y-osi je tlak [bar], na x-osi pa je prikazan en obrat motorja [kotne stopinje]

4.7.1 Detekcija ZML

Geometrija motorja v smislu razmerja proženja batov je podana v načrtih motorja. Senzor za kot pa se lahko namesti naknadno, tako da razdalja od ničle senzorja za kot do ZML prvega bata ni znana. Če izračunamo ta odmik in imamo iz načrtov znane vse druge odmike, imamo vse podatke, da od ničle senzorja za kot izračunamo vse odmike. Ko pa imamo enkrat te podatke, lahko pravilno zajamemo podatke za vsak cikel motorja – za vse valje.

Algoritem izračuna ta odmik iz maksimalne vrednosti tlaka. Za pravilno delovanje je treba vžig v valju izklopiti, tako nastane ZML, ko je tlak najvišji, ker je volumen najmanjši. V nasprotnem primeru pa je zaradi vžiga največji tlak zakasnen in so vrednosti napačne.

Podatke zajemamo od enega impulza za ničlo, na senzorju za kot, do drugega impulza. Pri štiritalnem motorju moramo zajeti od prvega impulza do tretjega itn., ker imamo štiri delovne takte. Potem pa znotraj zajetega cikla računamo maksimalni tlak in na kateri vrednosti kota se pojavlja ter povprečimo rezultate čez več meritev. Zraven računamo tudi standardno deviacijo; če je prevelika (odvisno od zahtevane natančnosti in karakteristik motorja), moramo meritev ustaviti in pogledati, kaj je narobe z motorjem oz. merilno verigo, ter meritev ponoviti. Pozicija maksimalnega kota v kotnih stopinjah pa nam služi kot odmik senzorja do ZML prvega bata.

Pri štiritalnem motorju imamo štiri delovne takte, zato lahko naletimo na problem, da vzamemo napačne podatke, če pozicija maksimalnega tlaka ni znotraj +/- 180 kotnih stopinj.

Zato smo postavil pravilo, če je pozicija maksimalnega tlaka manjša od -180 oz večja od 180 kotnih stopinj, se prestavim za 360 kotnih stopinj naprej in spet iščemo maksimum.

Izračunani poziciji (v kotnih stopinjah) maksimalnega tlaka prištejemo še vrednost kota zaradi torzije glavne gredi in termodinamičnih izgub, ki jo uporabnik vpiše v analizator.

4.7.2 Poravnavanje tlaka na pravilno pozicijo kota

Ko enkrat dobimo pozicijo ničlo, to vzamemo za odmik prvega senzorja tlaka, vse druge pa uporabnik vpiše v polja za vsak naslednji valj. Odmiki se v analizatorju upoštevajo pri zajemu podatkov. Ker se v enem delovnem ciklu motorja proži več batov, moramo za en izračun zajeti vsaj dva cikla, da nam ne zmanjka podatkov za tiste bate, ki se prožijo zadnji. Zmanjkata nam lahko del delovnega takta in izpuh.

4.7.3 Poravnavanje tlaka na absolutno vrednost (ničenje)

Senzor za tlak da kot izhod nabožno vrednost, ki jo je treba poravnati na absolutne vrednosti tlaka. Pa tudi zaradi toplotnih vplivov se senzor za tlak v svečki greje in se iz cikla v cikel odmika od pravilne vrednosti. Ta odmik je treba kompenzirati, ker se nekateri izračuni nanašajo na to (recimo vrednost maksimalnega tlaka). Torej je pravilna vrednost tlaka za trenutni cikel:

$$p = p_m - p_c \quad (4.15)$$

kjer so:

p – popravljen tlak (tega se potem uporabi za izračune in vizualizacijo) [bar]

p_m – izmerjeni tlak [bar]

p_c – korekcijski tlak [bar]

Tlak lahko popravimo na več načinov. Implementirani so trije standardni algoritmi.

4.7.3.1 Termodinamično ničenje tlaka

Korekcijski faktor izračunamo po enačbi [6]:

$$p_c = -1 * \frac{P_2 - P_1}{\left(\frac{V_1}{V_2}\right)^k} - p_1 \quad (4.15 a)$$

p_1, p_2 in v_1, v_2 so vrednosti tlaka [bar] ter volumna [dm^3] v kompresijskem delu cikla (ker je tukaj porast tlaka odvisen samo od zmanjševanja volumna valja), k pa je politropni eksponent, ki ga vpiše uporabnik.

4.7.3.2 Ničenje iz vpisane vrednosti tlaka

Podati moramo kot in vrednost odmika v barih. Ta metoda se redko uporablja, ker je dokaj nenatančna.

$$p_c = p_n - p_u \quad (4.15 \text{ b})$$

kjer sta:

p_n = tlak iz vhodnih podatkov [bar]

p_u = vpisana vrednost tlaka [bar]

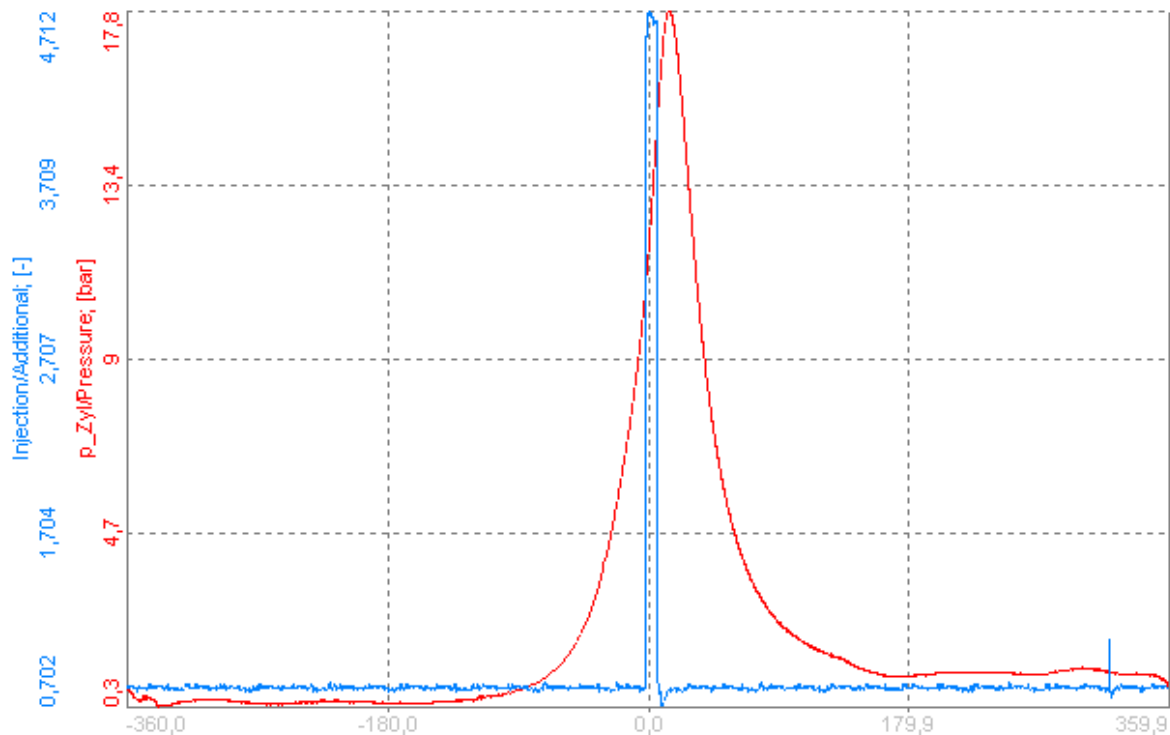
4.7.3.3 Ničenje iz izmerjene vrednosti tlaka

Ta metoda je najnatančnejša, zahteva pa dodatno merjenje tlaka na vходу (pri ventilih) v valj. Za vход moramo dodati kanal za tlak, pozicijo, od kod na tem kanalu bomo vzeli vrednost za ničenje (p_{zm}) in na kateri kotni poziciji bi radi popravili tlak na trenutnem valju (p_{mc}).

$$p_c = p_{mc} - p_{zm} \quad (4.15 \text{ c})$$

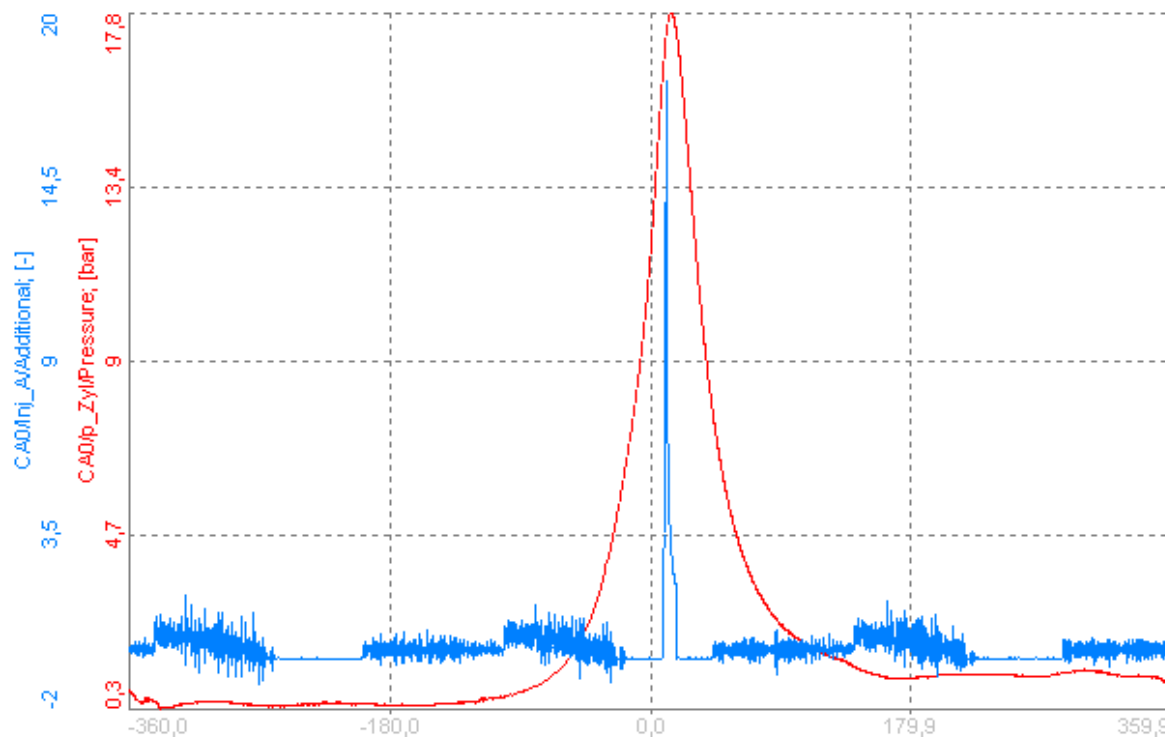
4.8 Meritev začetka in konca vbrizga goriva ter dodatni kanali

Motorji s svečko potrebujejo en injektor goriva na valj (nekateri lahko več zaradi ogrevanja na startu motorja). Obstajajo mehansko in elektronsko krmiljeni sistemi za vbrizg goriva. Ker je pomembno, na kateri poziciji bata (v kotnih stopinjah) je bil začetek vbrizga in kdaj konec, lahko vbrizg merimo z analognim sensorjem, ki da kot izhod napetost (slika 24). Ko ni vbrizga, drži konstantno vrednost, ko pride do vbrizga, pa se dvigne do določene ravni. Za konec vbrizga se spet spusti na prvotno raven. V analizatorju moramo definirati kanal in raven proženja za začetek in konec.



Slika 24. Vbrizg goriva v valj. Na y-osi je tlak [bar], na x-osi pa je prikazan en obrat motorja [kotne stopinje]

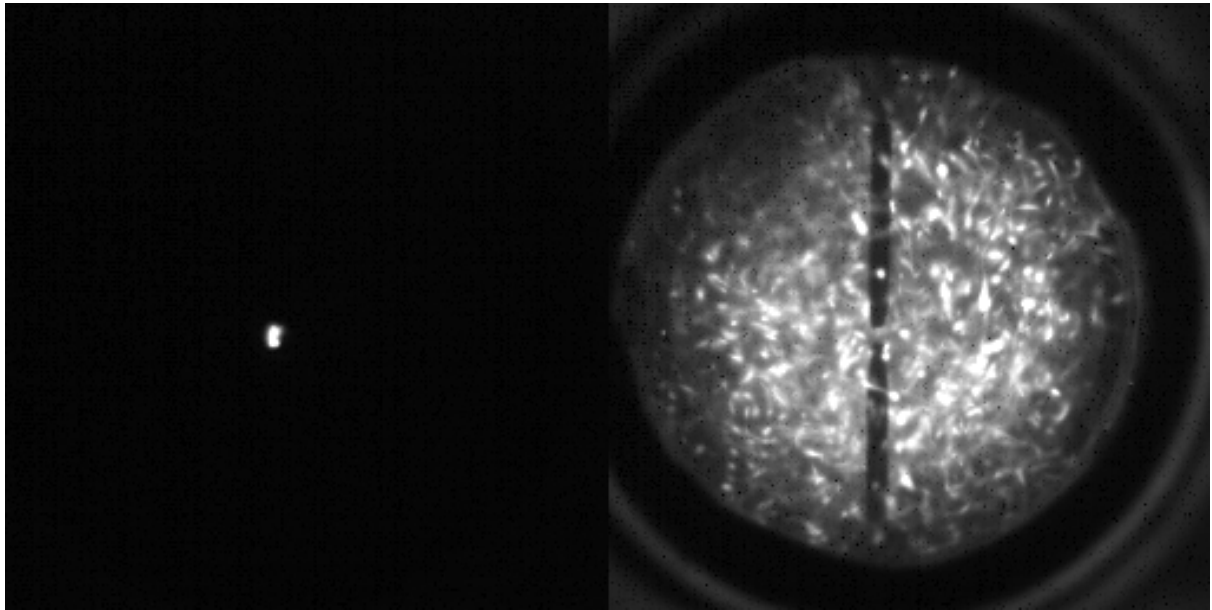
Z dodatnimi kanali lahko merimo različne temperature, vhodni tlak, vžig svečke (slika 25) itn. Te kanale določimo na vsakem valju. Lahko jih dodamo, kolikor hočemo, da procesor še zmore obdelavo. Vsak kanal se poravna na kotne vrednosti za določen valj. Na njih se ne izračunava nobena matematika. Namenjeni so samo za vizualizacijo in izvoz v kakšen programski paket za podrobnejšo analizo (npr. AVL Concerto). Večina naprednih uporabnikov si napiše svoje funkcije za izračune na teh kanalih, ker so to večinoma stvari, ki so zelo odvisne od vrste motorja.



Slika 25. Vžig svečke v valju. Na y-osi je tlak [bar], prikazan z rdečo barvo, z modro barvo pa signal svečke [V], na x-osi pa je prikazan en obrat motorja [kotne stopinje]

Kot zanimivost naj povem, da Dewesoft podpira različne hitre kamere. Za analizo izgorevanja in širitve ognja so takšne kamere zelo uporabne. Na strani valja ali pa na vrhu (če nima svečke) se zvrta v valj luknjo v katero se vgradi ognjevarno, kaljeno steklo. Na zunanji strani pa se pritrdi kamera. Dewesoft omogoča, da se shranijo videopodatki, sinhrono z analognimi in digitalnimi podatki.

Slika 26 prikazuje vžig svečke (na levi strani) in potem širjenje ognja pri vžigu (na desni strani). Slika je bila posneta na dvotaktnem bencinskem motorju (Colibri).



Slika 26. Širitev ognja v valju

4.10 Diagrami na zaslonu za nastavitve

Na zaslonu za nastavitve je možno odčitati krivuljo tlaka, pV-diagram tlaka, število impulzov senzorja za kot, hitrost vrtenja motorja, število zajetih ciklov. Prikazane so nekatere vrednosti za referenčni valj, npr. maksimalni tlak, srednji efektivni tlak. Če je zagnan algoritem za detekcijo ZML, vidimo poravnani tlak na referenčnem valju. Te vrednosti so zelo pomembne, saj uporabnik le tako vidi, da je pravilno nastavil parametre za meritev.

5. REZULTATI

Vsi rezultati so izračunani iz vhodnega tlaka in izračunanega volumna. Osnovne vrednosti so lahko vektorji ali pa posamezne vrednosti (skalarji). V Dewesoftu se za vsako osvežitev zaslonov pokličejo funkcije v naslednjem vrstnem redu:

- a) Zajem podatkov z A/D-pretvornikom
V tej fazi se zajamejo vsi novi podatki. Število novih podatkov je enako številu zajetih podatkov minus število že obdelani.
Ti podatki so lahko analogni, digitalni, GPS-koordinate, podatki iz CAN-vodila itn. Podatki se shranijo v začasne vmesnike.
- b) Izračun novih vrednosti: V tej fazi se izračunajo vsi vtičniki in matematike. Vrstni red je določen vnaprej, tako da se izračunajo moduli, ki imajo samo zajete podatke za vhode, potem pa še vsi kanali, ki imajo za vhod rezultat iz prejšnjega cikla računanja.

Opis osnovne zanke programa:

- a. Ko je kot 0 kotnih stopinj, začnemo zajem podatkov.
- b. Preberemo podatek za podatkom in jih shranjujemo v t. i. vmesni pomnilnik na vsakem valju, ki ustreza vhodnemu signalu.
- c. Če je motor dvotakten in smo našli dva prehoda skozi ničlo na kotnem senzorju oz. je motor štiritalkten in smo našli štiri prehode skozi ničlo, nadaljujemo
- d. Preverimo, ali se število podatkov ujema s toleranco, ki jo je določil uporabnik. Uporabnik lahko določi, koliko podatkov na cikel lahko manjka oz. jih je lahko preveč. To se dogaja zaradi kakšnih motenj na senzorju za kot in tako A/D-pretvornik zajame preveč oz. premalo podatkov.
- e. Če je motor štiritalkten, v primeru da vrednost maksimalnega tlaka ni med -180 in 180 kotnih stopinj, popravimo fazo. Če je bil popravek faze potreben, zajamemo še en cikel podatkov, sicer nadaljujemo
- f. Podatki so popravljene na izračun. Pokličemo funkcijo CalculateCylinders.

Psevdokoda:

```
for all new data {
    if Angle = 0 {
        read input value and store to correct cylinder buffer position

        if (engine = 2 stroke) and (zero cross = 2) or
           (engine = 4 stroke) and (zero cross = 4) {
            if (data count + tolerance < collected data) and
               (data count - tolerance > collected data) {
                If (engine = 4 stroke) and correct phase or (engine = 2 stroke)
                CalculateCylinders
            }
        }
    }
}
```

CalculateCylinders se sprehodi po vseh valjih in vseh podatkih in zavrže začetne vzorce, da doseže vpisani odmik za vsak valj. Potem pa pokliče izračune za posamezne vrednosti.

c) Shranjevanje zajetih podatkov in rezultatov v podatkovno datoteko

Vsak kanal v programskem paketu Dewesoft ima lastnost »stored«. Ta lastnost pove, ali se ti podatki shranijo v datoteko ali ne. Vsi kanali, ki imajo »stored« nastavljen, se shranijo v datoteko na disku za poznejšo analizo ali arhiv. Analizator lahko shranjuje zelo veliko kanalov. V tem primeru bi datoteke postale zelo velike. Nekateri uporabniki imajo analizator večinoma samo za opazovanje motorja, shranijo pa samo nekaj datotek na dan. Za takšne to ni problem. Drugi pa bi radi imeli celodnevni nadzor nad motorjem. Datoteke se shranjujejo skozi ves dan in se arhivirajo. Arhivi morajo ostati nedotaknjeni mnoga leta. Tukaj pa zna nastati problem s pomnilnikom. Ker smo prej povedali, da se večina podatkov izračuna iz tlaka, smo lahko v analizator dodali izboljšavo. Shranjujemo samo podatke, ki jih nujno potrebujemo za izračun, vse druge podatke pa še enkrat izračunamo, ko uporabnik naloži shranjeno datoteko. Izboljšavo lahko naredimo še tako, da nove podatke izračunavamo šele, ko se uporabnik sprehaja po datoteki. Če bi vse izračunali na začetku, bi se podatki predolgo nalagali.

d) Prikaz podatkov na zaslonu

Programski paket Dewesoft občasno osveži vse zaslone in diagrame. Uporabnik določi interval osveževanja v nastavitvah. Ko se pokliče osveževanje, vse vizualne kontrole preberejo zadnje podatke (kolikor jih pač potrebujejo) in jih narišejo. Podatki v programskem paketu Dewesoft so goli podatki iz strojne opreme, potem pa se še na vsak vnaprej določen interval izračunajo povprečja. Tako lahko ta povprečja prikazujemo in s tem spet pridobimo hitrost, ker moram izrisati manj vrednosti, kot bi jih bilo treba, če bi risali gole podatke.

Analizator se je pokazal kot velika obremenitev za CPU, zato smo znotraj analizatorja uporabili nitenje. Vsak valj dam v svojo nit oz. razdelim računanje na toliko procesorjev, kot jih je na voljo. Naslednja izboljšava je, da računam podatke bolj poredko in vmesne podatke zavržemo. Tako lahko analizator pohitrimo. Postopek pa ima eno veliko slabost, in sicer, da je izvedljiv samo takrat, kadar se podatki ne shranjujejo, ampak se uporabljajo samo za prikaz.

5.1 Diagrami podatkov v odvisnosti od kota

Na diagramih so predstavljeni podatki v vektorski obliki (npr. tlak za en cikel motorja). Na y-osi je amplitudna vrednost, na x-osi pa so kotne stopinje: za dvotaktni motor od -180 do 180 , za štiritalaktni motor pa od -360 do 360 . Korak (v kotnih stopinjah) določi uporabnik v nastavitvah analizatorja.

5.1.1 Tlak

Tlak je zajet direktno iz senzorja. Popravljen je samo za skalirni faktor in v večini primerov poravnan na absolutno vrednost.

5.1.2 Povprečni tlak

Ker se tlak iz cikla v cikel spreminja tudi zaradi šuma različnega izvora, je treba vektorje tlaka povprečiti. Ker so podatki v kotni domeni, točno vemo, na kateri poziciji je kakšna vrednost, zato lahko izračunamo povprečni tlak čez celo meritev. Seveda pa potrebujemo tudi trenutni tlak, ker se pri povprečenju izgubljajo visokofrekvenčne komponente, ki pa so zelo pomembne za realno oceno klenkanja. Vrednosti, kot je recimo maksimalni tlak, pa so veliko stabilnejše na povprečnih vektorjih tlaka.

5.1.3 Volumen valja

Volumen valja ni zajet vektor, ampak je izračunan v programu iz vhodnih parametrov geometrije. Implementirana je tudi možnost, da uporabnik sam vpiše direktno enačbo za izračun volumna. To pa zato, ker je kar nekaj motorjev, ki imajo geometrijo drugačno od standardne (recimo Wanklov motor, GoEngine).

5.1.4 Odvod tlaka

Odvod tlaka (slika 27) nam pove maksimalen narastek glede na resolucijo meritve. Iz odvoda je razvidno klenkanje in glede na to, kdaj je začel tlak rasti, lahko določimo vžig delovnih tekočin. Odvajal sem po metodi nedoločenih koeficientov [8, 9]. To pomeni, da odvod funkcije (f') aproksimiramo kot linearno kombinacijo njenih vrednosti v sosednjih točkah.

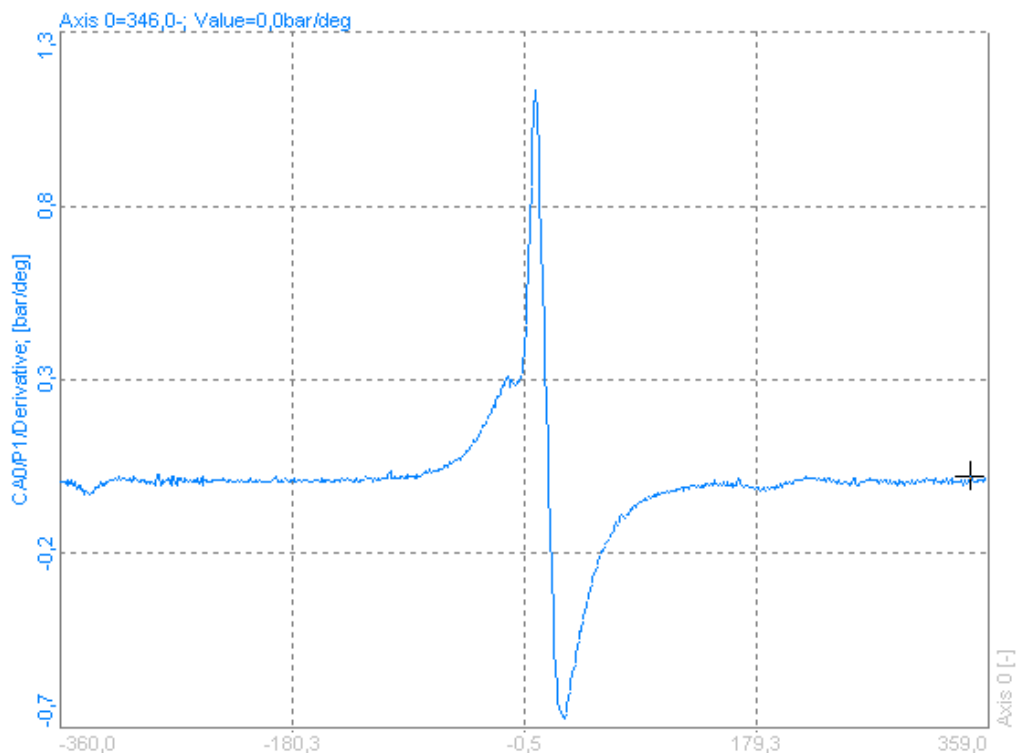
$$f' = \frac{f(x+h) - f(x-h)}{2h} \quad (5.1)$$

Pri čemer smo za korak (h) izbrali 1 vzorec.

Napaka takšne formule je enaka $\frac{-h^2}{6 f'''(x_0)}$. Zato je ta metoda natančnejša od direktne in obratne metode.

Problem pri zgornji formulaciji sta prvi in zadnji vzorec. Če jih damo kar na nič, pokvarimo drugi odvod (funkcija ni več gladka). Maksimalni narastek tlaka se večinoma zgodi po ZML, zato prvi in zadnji vzorec nista preveč pomembna. Problem smo rešili tako, da prvi in zadnji vzorec pred izračunom odvoda podvojimo. Druga, boljša rešitev bi bila, da zajamemo en podatek več na začetku in koncu cikla ter zatem izračunamo odvod.

Na vhodnih podatkih o tlaku se velikokrat pojavi šum. Ko računamo odvod, ta šum še ojačamo, zato lahko maksimalen narastek tlaka zaradi šuma nastopi, preden bi bilo to realno pričakovati. Dodali smo dve polji, in sicer za začetek in konec računanja tlaka v kotnih stopinjah. Tako se tudi rešimo težave s prvim in zadnjim vzorcem.



Slika 27. Odvod tlaka. Na y-osi je tlak $[\frac{\text{bar}}{\text{deg}}]$, na x-osi pa je prikazan en obrat motorja [kotne stopinje]

5.2 Skalarnе vrednosti

Skalarnе vrednosti so za razliko od vektorjev predstavljeni kot številka iz množice realnih števil. Te vrednosti se računajo za vsak cikel. V kanale se shranijo kot asinhroni časovni podatki, in sicer za vsak valj. Asinhroni kanali imajo številsko in časovno vrednost. Lahko imajo poljubno vzorčenje neodvisno od vzorčenja A/D-pretvornika.

5.2.1 Povprečni efektivni tlak

Navor je dobra mera za delo, ki ga lahko opravi motor, vendar je odvisen od velikosti motorja. Tako se različne motorje težko primerja med sabo. Za primerjavo različnih motorjev normaliziramo delo z volumnom valja. Tako pridemo do mep-a (mean effective pressure) [1, 6].

$$mep = \int_{-n}^n \frac{p \cdot dV}{V} \quad (5.2)$$

kjer so:

p = tlak [bar]

V = delovni volumen valja [dm^3]

n = vrednost v kotnih stopinjah, odvisna od tipa motorja in vrste mep-vrednosti

Enota za mep je bar. Za dvotaktni motor se izračuna mep prek celega cikla motorja, za štiritaktni motor pa izračunamo tri različne mep-vrednosti. Tako lahko primerjamo celoten cikel motorja ali pa samo kompresijski in delovni takt.

i_{mep} = od -360 do 360 kotne stopinje

n_{mep} = od -180 do 180 kotne stopinje

p_{mep} = od -360 do -180 in od 180 do 360 kotne stopinje

Mep-vrednost za kakovosten motor je znana in je konstantna skozi velik rang motorjev. Tako lahko mep-vrednost načrtovanega in izdelanega motorja neposredno primerjamo s to normo ter tako ugotovimo, kako učinkovito je uporabljen volumen valja. Za načrtovanje motorjev lahko uporabimo iste norme, da dosežemo želeni navor oz. moč.

Tipične vrednosti so (pri 3000 rpm):

- motorji s vžigalno svečko: 8.5–10.5 bar
- motorji s vžigalno svečko (turbinski): 12.5–17 bar
- štiritaktni dizel: 7–9 bar
- štiritaktni dizel (turbinski): 10–12 bar
- dvotaktni motorji imajo primerljive karakteristike s štiritaktnimi
- veliki počasni dvotaktni motorji: 16 bar

5.2.2 Delo

Iz podatkov tlaka v valju se da izračunati prenos dela iz delovnih tekočin na bat. Delo se izračuna čez celoten cikel in je podano v joulih [6].

$$W = \sum_{-n}^n \frac{p_n * dV * 10000}{1000} \quad (5.3)$$

Včasih se delo izračuna samo med stiskalnim in delovnim taktom. V tem primeru se i_{mep} v enačbi zamenja z n_{mep} -vrednostjo iz prejšnje enačbe.

5.2.3 Navor

Navor motorja se po navadi meri z dinamometrom. Motor se pritrdi na testno mizo, glavna gred pa se pritrdi na rotor dinamometra. Rotor je elektromagnetno, hidravlično ali pa mehansko zvezan na stator, ta pa je uležajen z ležaji minimalnega trenja. Stator in rotor sta statično zbalansirana. Navor, ki ga je deležen stator pod vplivom vrtenja rotorja, se meri z balansiranjem statorja z utežmi, vzmetmi ali pnevmatskimi sredstvi. Navor nam pove, koliko dela je sposoben proizvesti določen motor.

V analizatorju pa se navor [Nm] računa iz dela [J] [6]:

$$\tau = \frac{W}{c\pi} \quad (5.4)$$

kjer je:

c = je za dvotaktni motor 2, za štiritaktni pa 4, ker se navor normalizira na en delovni takt

5.2.4 Moč

Moč nam pove, koliko dela opravi motor v enoti časa. Enota za moč je kW [6]:

$$P = \frac{w * rpm}{60 * 1000} \quad (5.5)$$

kjer je:

w = delo [J]

rpm = povprečni obrati motorja v minuti v tem ciklu (rounds per minute)

5.2.5 Podatki o meritvi

Izračunajo se še nekateri podatki o sami meritvi, ki so lahko za vsak valj ali pa za celoten motor:

a) Vrednosti za valj:

- CycleCount – števec zajetih ciklov motorja (0 ... n)
- Pulses per cycle – število impulzov na cikel
- Speed – povprečna hitrost motorja v enem ciklu (rpm)
- Time – časovni podatek (v s) za vsak podatek, dobljen iz hitrosti. Tako za vsak podatek v kotni domeni dobimo natančno informacijo o tem, kdaj se je dogodek zgodil.

Čas za interno in eksterno vzorčenje določimo glede na spodnji dve formuli:

- Interna ura $\rightarrow \text{Time} = \sum_0^n \frac{f}{Fs}$ (5.6 a)

- Eksterna ura $\rightarrow \text{Time} = \sum_0^n \frac{\text{perioda}}{1000}$ (5.6 b)

kjer so:

f – povprečna frekvenca vrtenja v Hz

Fs – frekvenca vzorčenja

perioda – perioda med dvema vzorcema v ms

b) Vrednosti za motor:

- Max P: maksimalna vrednost za tlak $\rightarrow \max_{-n \leq x \leq n} p(x)$ (5.7a)

- Max P Pos: kotna pozicija maksimalnega tlaka $\rightarrow \text{pos}(\max_{-n \leq x \leq n} p(x))$ (5.7b)

- Max D: maksimalna vrednost odvoda \rightarrow (5.7c)

- $\max_{-n \leq x \leq n} d(x)$
- Max D Pos: kotna pozicija za maksimalni odvod \rightarrow $pos(\max_{-n \leq x \leq n} d(x))$ (5.7d)
- PCorr – vrednost korekcijskega faktorja za termodinamično ničenje

5.3 Termodinamika (skalarji in vektorji)

Pod termodinamiko štejemo izračun temperature v valju, sproščanje toplotne energije in klenkanje. Rezultati teh izračunov so predstavljeni kot skalarji in kot vektorji. Izračuni pa so velikokrat bolj mišljeni kot ocene dejanskega stanja. Toliko, da razvijalec dobi občutek, kje je, ne pa da bi te številke jemal kot popoln rezultat. Težava je v tem, da nekaterih spremenljivk preprosto ne moremo uvrstiti v enačbo, nimamo metode, da bi jih izmerili. Za sproščanje toplotne energije pri izračunu denimo ne upoštevamo izgub skozi batne obročke in stene valja itn., ker se teh parametrov ne da zmodelirati na splošno, temveč so zelo odvisni od posameznega motorja.

5.3.1 Temperatura

Temperaturo (slika 28) znotraj valja lahko merimo z namenskim senzorjem in kanal s podatki izberemo kot dodatni kanal v analizatorju. Lahko pa jo izračunamo iz splošne plinske enačbe, ki velja za idealni plin. Mešanica v valju ni idealni plin, se pa temu dovolj približa. Temperatura je izražena v kelvinih.

Splošna plinska enačba

$$p * V = n * R * T \rightarrow T = \frac{p * V}{n * R} \quad (5.8)$$

kjer so:

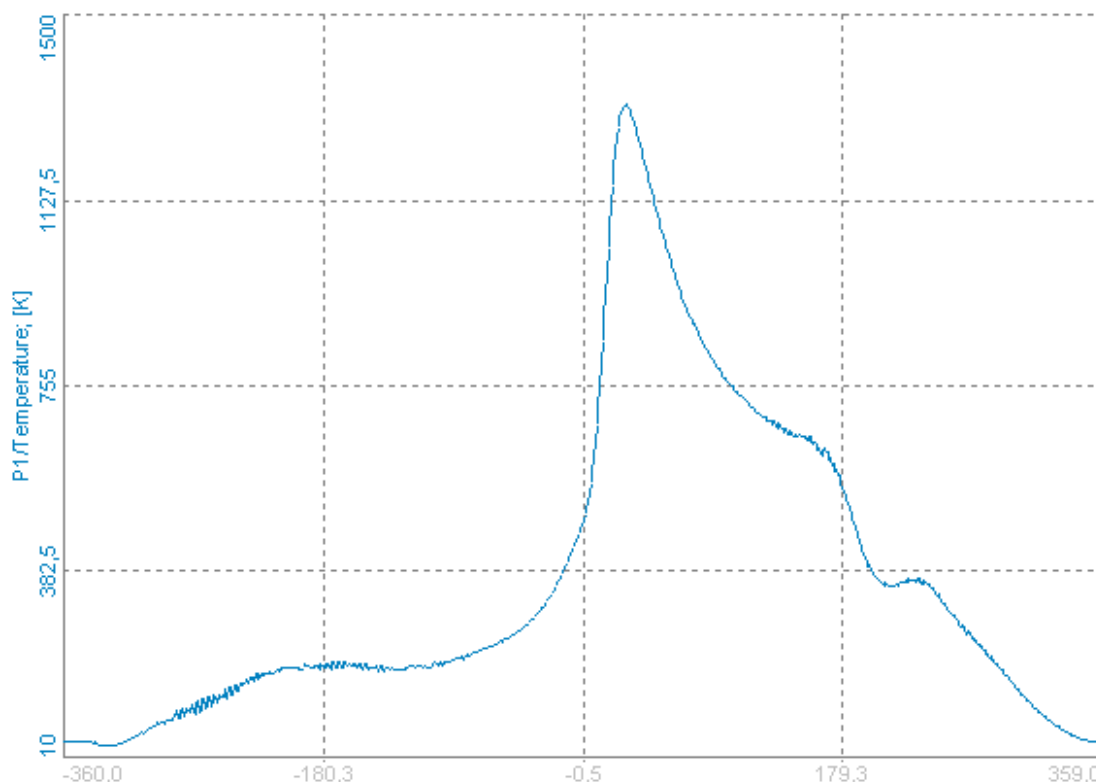
p = tlak [bar]

V = volumen [dm³]

R = splošna plinska konstanta [$\frac{J}{mol K}$]

T = temperatura [K]

n = teža plina [g]



Slika 28. Temperatura v valju. Na y-osi je temperatura [K], na x-osi pa je prikazan en obrat motorja [kotne stopinje]

Za izračun temperature mora uporabnik vpisati težo plina v valju (n) ali pa izbere možnost, da se teža izračuna iz vhodnega tlaka v valju. Za izračun teže se spet uporabi plinsko enačbo. Uporabnik mora vpisati tlak pri odprtih ventilih (okrog -180 kotnih stopinj) in temperaturo na tem mestu. Določiti mora še tako imenovani volumetrični učinek oziroma volumetric efficiency. Ta parameter je razmerje med količino goriva in zraka, ki vstopa v valj, ter realno kapaciteto valja.

5.3.2 Sproščanje toplotne energije (t. i. heat release)

Vrednost sproščanje toplotne energije nam pove, koliko toplote bi morali dodati vsebini valja, da bi dobili izmerjeni porast tlaka.

Izračuna se po prvem zakonu termodinamike (enota $\frac{kJ}{m^3 \cdot deg}$) [6]:

$$T_q = \frac{k}{k-1} * p * \frac{dv}{d\theta} + \frac{1}{k-1} * V \frac{dp}{d\theta} \quad (5.9)$$

kjer so:

k = politropni eksponent

$$dV = \text{sprememba volumna} \left[\frac{dm^3}{deg} \right]$$

$$dp = \text{sprememba tlaka} \frac{bar}{deg}$$

Iz tega lahko izračunamo integral T_i (enota je $\frac{kJ}{m^3}$):

$$T_i = \sum_{-n}^n T_q \quad (5.10)$$

Začetek in konec izgorevanja sta definirana kot 5 % in 95 % vrednosti T_i .

Psevdokoda algoritma:

```

for i = stevilo_vzorcev_v_ciklu do
{
  Curr = vrednost trenutnega vzorca v %
  Next = vrednost naslednjega vzorca v %
  for J = 0 to 1 do
    if (MaxTIVal[j] >= Curr) and (MaxTIVal[j] <= Next) and (MaxTIPos[j] = -1) then
    {
      if (Next - Curr) = 0 then
        MaxTIPos[j] = -2
      else
        MaxTIPos[j] = (i + (MaxTIVal[j] - Curr) / (Next - Curr)) * R;
    }
  }
}

```

R = kotna resolucija

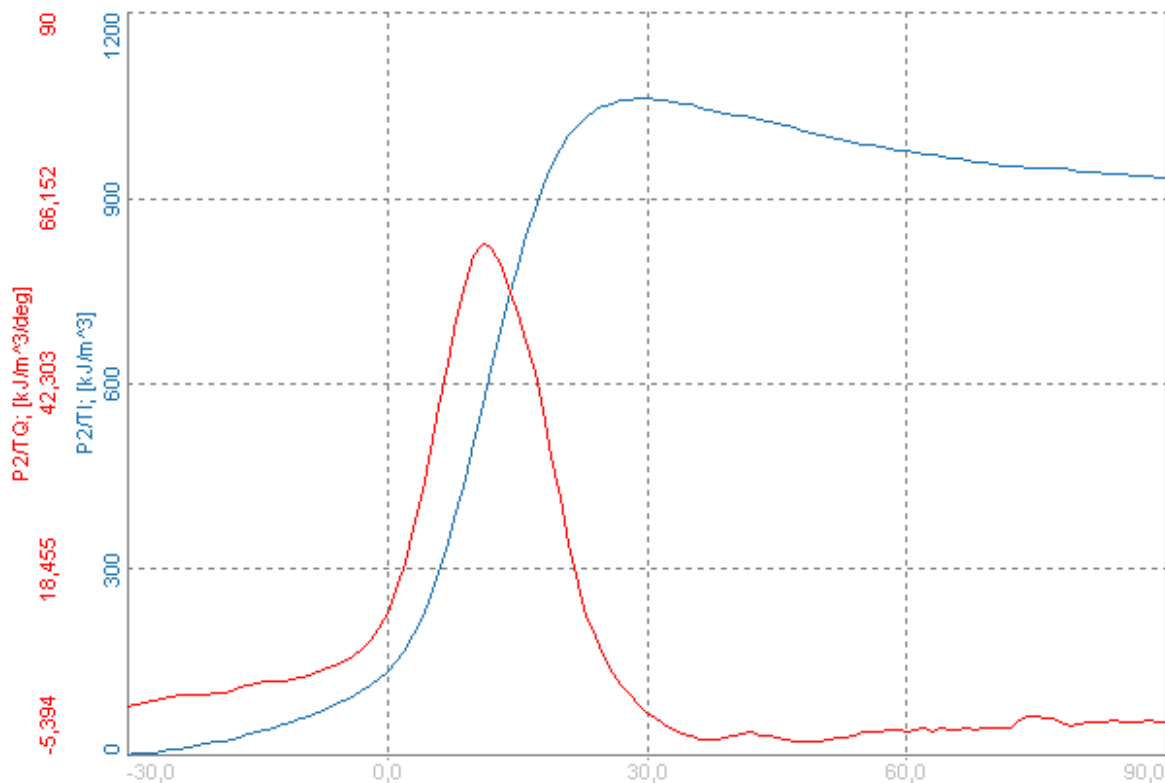
MaxTIPos = -1

MaxTIVal[0] = Max(Ti) * 0.05

MaxTIVal[1] = Max(Ti) * 0.95

Kakor vidimo iz zgornje psevdokode, se pozicija interpolira zaradi še večje natančnosti izračuna (neodvisne od frekvence vzorčenja). Za motorje z direktnim vbrizgom in dizelske motorje se stvar malo zakomplicira, ker gre T_i , tik preden se vzpne, v negativno vrednost. Kot začetek za iskanje pa se vzame prehod skozi ničlo po prehodu iz negativne vrednosti.

Slika 29 prikazuje sproščanje toplote (modra krivulja) od -30 do 90 kotnih stopinj in integrirano vrednost za sproščanje toplote (rdeča krivulja).

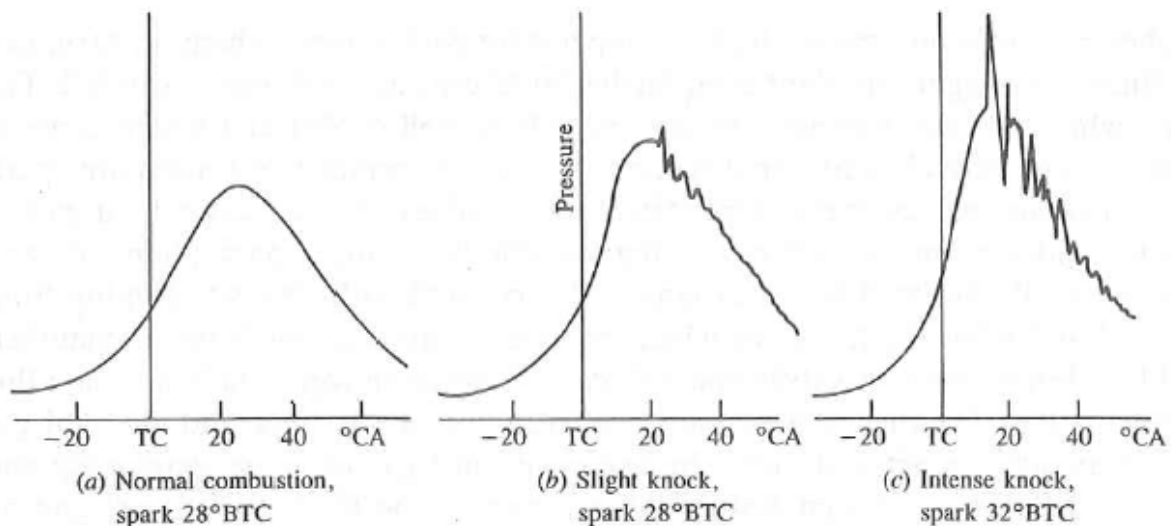


Slika 29. Na y-osi sta prikazana $Tq \left[\frac{kJ}{m^3 \cdot deg} \right]$ in $Ti \frac{kJ}{m^3}$. Na x-osi je prikazan en del obrata motorja [kotne stopinje]

5.3.3 Klenkanje

Iz prejšnjih poglavij vemo, da termični izkoristek krožnega procesa raste s povečanjem kompresijskega razmerja. Spričo tega bi morali stremeti, da na koncu kompresijske faze dosežemo čim višje tlake, če ne bi v praksi obstajale določene ovire. Pri Ottovih motorjih, kjer stiskamo mešanico goriva in zraka, ne smemo dopuščati samovžiga niti pojava klenkanja. Klenkanje je v glavnem odvisno od goriva ali, boljše, od odpornosti goriva proti klenkanju [2]. Slika 30 prikazuje primer krivulje tlaka brez klenkanja in z njim.

Bencinski motor klenka takrat, kadar poleg vžiga zaradi iskre z vžigalne svečke nastane tudi samovžig zmesi goriva in zraka. Samovžig, ki sproži sočasen vžig na več mestih, povzroči prehitro (sunkovito) zgorevanje, plamen pa se razširi z veliko hitrostjo in tlak postane previsok (slika 30), sledi močan sunkovit zvok, ki ga imenujemo klenkanje. To povzroči večjo mehansko in toplotno obremenitev ročičnega mehanizma, obenem pa se zmanjša tudi moč [6].



Slika 30. Klenkanje [6]

Klenkanje pri pospeševanju nastane predvsem takrat, kadar pospešujemo ob polnem plinu in nizki hitrosti v visoki prestavi. Največkrat ga povzročata gorivo s prenizkim oktanskim številom in nepravilno nastavljen vžig.

Klenkanje pri visoki hitrosti največkrat nastane v zgornjem območju vrtilnih frekvenc in predvsem pri polni obremenitvi. Zaradi hrupa v notranjosti vozila ga največkrat ne zaznamo pravočasno. Tudi tukaj je največkrat vzrok gorivo.

Drugi vzroki klenkanja: poleg uporabe neprimerne goriva so lahko razlogi za klenkanje tudi prezgodnji predvžig, neenakomerna porazdelitev mešanice in slabo odvajanje toplote.

Žarni vžig povzročajo žareči deli v zgorevalnem prostoru, še preden pride do normalnega vžiga mešanice zraka in goriva z iskro z vžigalne svečke.

Klenkanje bencinskih motorjev se pozna kot visokofrekvenčni signal, superponiran na osnovni krivulji tlaka. Pojavlja se v rangi od 10 do 15 kHz, zato moramo imeti dovolj visoko vzorčenje (več kot 30 kHz). Ker je klenkanje zelo podobno šumu, ga moramo nekako razločiti od šuma. Zato primerjamo pravilno filtrirano krivuljo tlaka v območju, kjer pričakujemo klenkanje, z nefiltrirano krivuljo. Dewesoft uporablja za detekcijo klenkanja Siemensov algoritem.

Algoritem:

1. Za izračun moramo vedeti točno pozicijo maksimalnega tlaka, zato najprej z nizkoprepustnim filtrom filtriramo krivuljo tlaka, da se znebimo klenkanja in dobimo referenčni signal. Če ne bi filtrirali signala, bi bila pri močnem klenkanju maksimalna vrednost tlaka na napačni poziciji.

$$p_{ref} = Lp(p) \quad (5.11)$$

2. Z visokoprepustnim filtrom filtriramo signal, da dobimo samo klenkanje.
3. Nastavimo velikost okna, v katerem bomo računali klenkanje. Treba je določiti okno za referenčni signal in signal, ki vsebuje klenkanje.

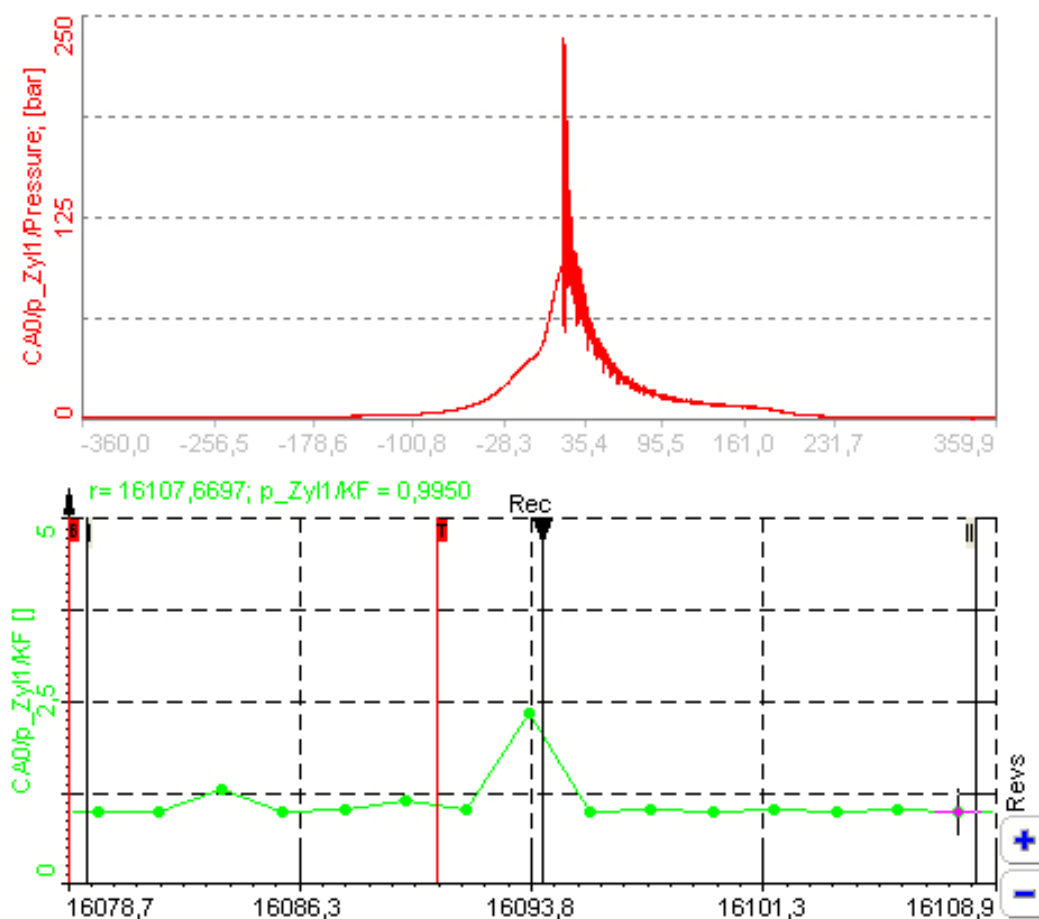
$$p_{knock} = Hp(p) \quad (5.12)$$

4. Integriramo oba signala in izračunamo razmerje za vsako pozicijo kota. Pri integraciji ignoriramo vrednosti, ki so pod ravno šuma, ki jo vpiše uporabnik. Tako pridemo do vrednosti, imenovane faktor klenkanja oziroma kf (knocking factor). Vrednosti pod 1 ne pomenijo klenkanja, vrednosti nad 3 pa pomenijo močno klenkanje.

$$kf = \frac{\int p_{ref}}{\int p_{knock}} \quad (5.13)$$

5. S tem ko se spreminja hitrost vrtenja motorja, se spreminja tudi maksimalna vrednost tlaka. Če poznamo karakteristiko motorja, jo lahko upoštevamo in vpišemo vrednosti v dveh točkah. Točke vmes pa se linearno interpolirajo.

Slika 31 prikazuje močno klenkanje na tlaku. Faktor klenkanja znaša 2,5.



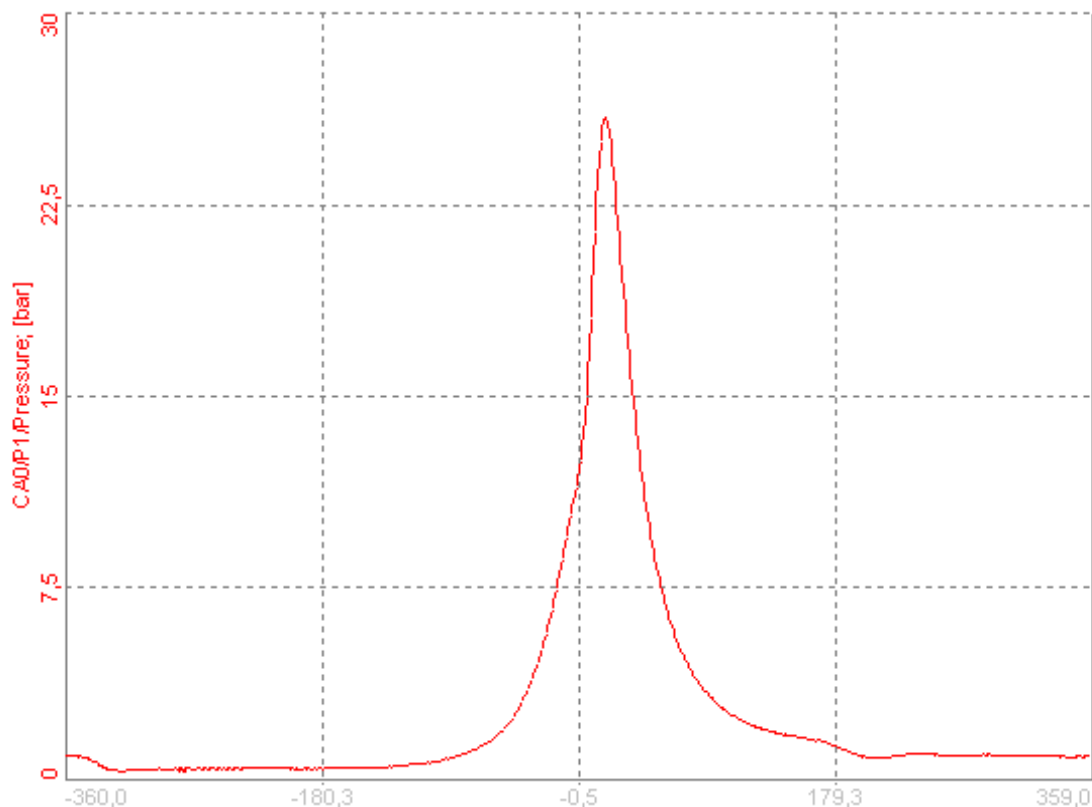
Slika 31. Zgoraj je prikazan tlak [bar] s klenkanjem na en obrat motorja [kotne stopinje], spodaj pa faktor klenkanja glede na obrate motorja [obrat]

6. PRIKAZ REZULTATOV

Pravimo, da slika pove več kot tisoč besed. S tega vidika je seveda zelo pomembna pravilna predstavitev podatkov. Analizator lahko prikazuje podatke v številski obliki ali pa kot vektorje. Iz termodinamike (kot veje fizike) so znani diagrami, ki najučinkoviteje prikažejo želeno informacijo (npr. pV-diagram). Za te prikaze se je dalo uporabiti nekaj standardnih grafov v Dewesoftu, nekatere pa je bilo treba še dodati.

6.1 CA Scope

CA Scope (slika 32) ima na y-osi amplitudo, na x-osi pa kotne stopinje. ZML (torej 0 kotnih stopinj) je na sredini diagrama. Na tem grafu se prikazujejo vsi vektorji v kotni domeni (tlak, odvod tlaka itn.).



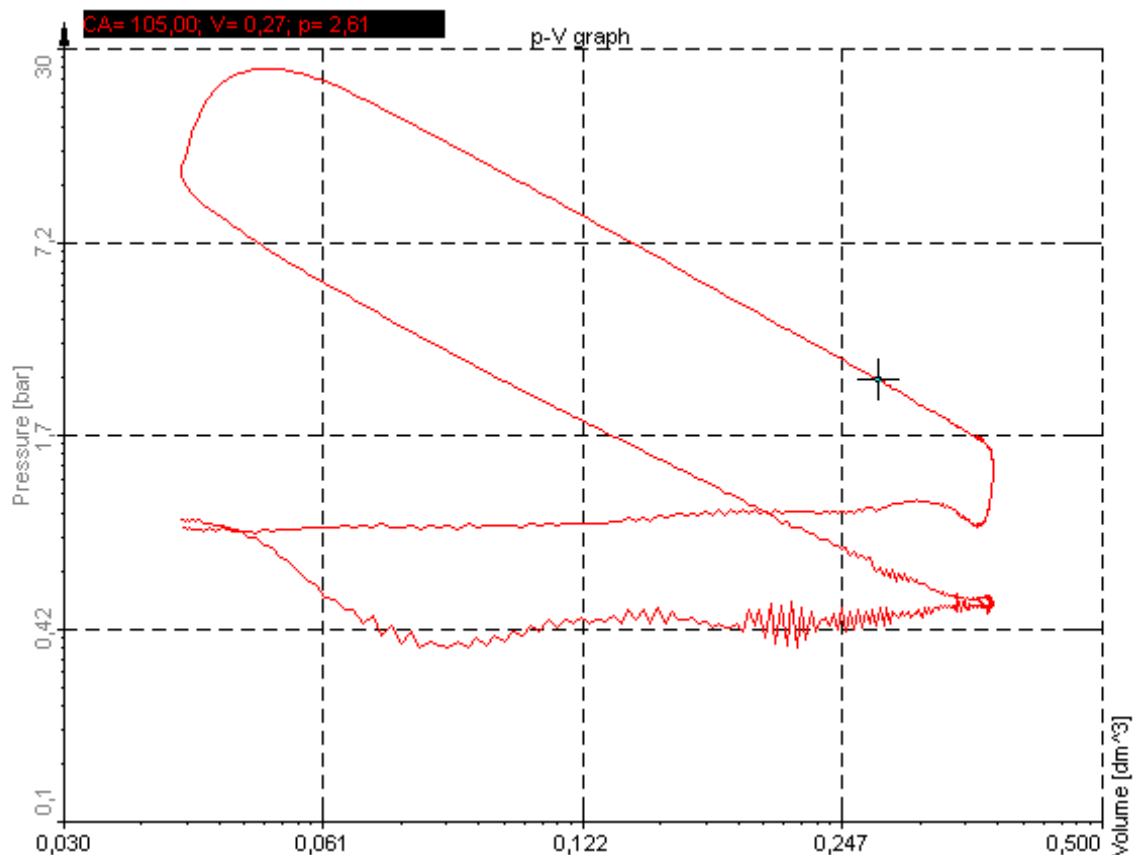
Slika 32. CA Scope prikazuje rezultate v kotni domeni

6.2 Delovni diagram

Delovni diagram (slika 33) prikazuje idealiziran Ottov cikel. Sesalni takt je izobara, stiskanje je adiabatna sprememba, delovni (vžig delovnih snovi) takt je izohora. Cikel se konča z izohornim procesom in izobarno kompresijo v izpuhu [6]. Na y-os se nariše tlak, na x-os pa volumen. Iz tega diagrama se da neposredno izračunati delo, tako da se integrira območje omejeno z grafom krivulje.

$$W = \int_n^n \frac{p}{dV} \quad (6.1)$$

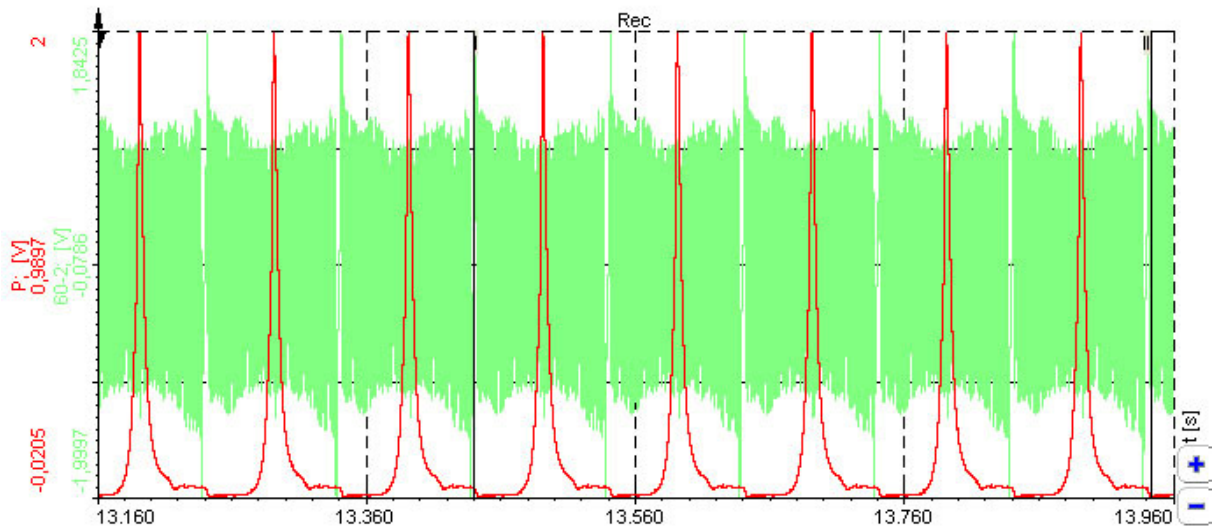
Diagram nariše tudi točki za začetek oz. konec izgorevanja. Ker pa je $p * V^n = c$ eksponentna funkcija, graf podpira tudi logaritemsko merilo. V tem primeru sta kompresija in delovni takt prikazana linearno, kar velja za bolj standarden prikaz diagrama.



Slika 33. Delovni oziroma pV-diagram. Na y-osi je tlak [bar], na x-osi pa je prostornina valja [dm³]

6.3 Prikaz v obliki rekorderja

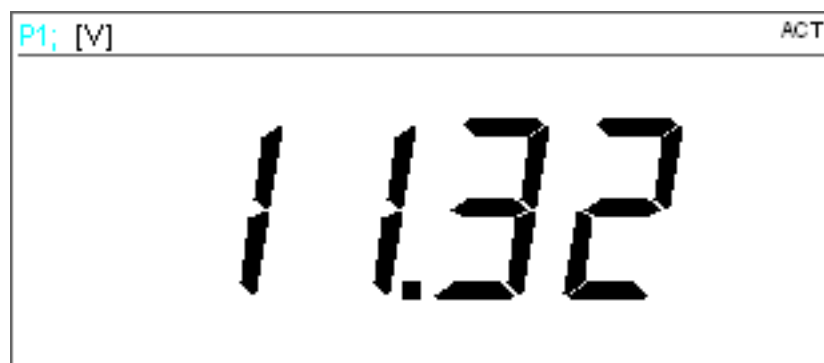
Na rekorderju je na y-osi amplituda, na x-osi pa čas oz. obrati (za zunanjo uro). Rekorder je dobil svojo podobo po starih analognih rekorderjih. Prikazuje zgodovino časovnih podatkov.



Slika 34. Prikaz v obliki rekorderja. Na y-osi sta tlak [bar] in senzor 60-2 [V], na x-osi pa je čas [s]

6.4 Prikaz v obliki digitalnega merilnika

Digitalni merilnik (slika 34) prikazuje samo en podatek, in sicer trenutno vrednost. Po videzu je ima izgled električnega multimetra.



Slika 35. Digitalni prikaz

7. ANALIZA

Uporabniki shranijo podatke iz dveh razlogov:

- a) konstantni nadzor motorja in opazovanje, kaj se dogaja na dolgi rok;
- b) kratke datoteke z različno nastavljenimi parametri motorja, da bi ugotovili, kateri parametri se obnesejo najbolje.

V obeh primerih pa pride čas, ko potrebujejo podatke za analizo. Kot smo že omenili, se podatki zaradi prostora na pomnilniškem mediju izračunajo v realnem času.

Obstaja pa še druga plat analize. Uporabniki bi radi spreminjali parametre analizatorja in ponovno izračunali rezultate. To je v programskem paketu Dewesoft podprto. Lahko spremenimo npr. politropni eksponent in ponovno izračunamo rezultate.

8. IZVOZ PODATKOV

Podatki iz analizatorja se lahko izvozijo v različne formate: Flexpro, Excel, Diadem, Matlab, UFF, Famos, Text.

Izvozimo lahko:

- ciklične podatke
- podatke enkrat na cikel
- povprečje ciklov

Večina formatov podpira en ali drug tip izvoza. Nobeden od teh formatov pa ne podpira nadaljnje analize podatkov. Zato je bilo treba podpreti format, v katerem je možno tudi naprej analizirati podatke. Takšno orodje je AVL Concerto.

8.1 Izvoz podatkov v AVL Concerto

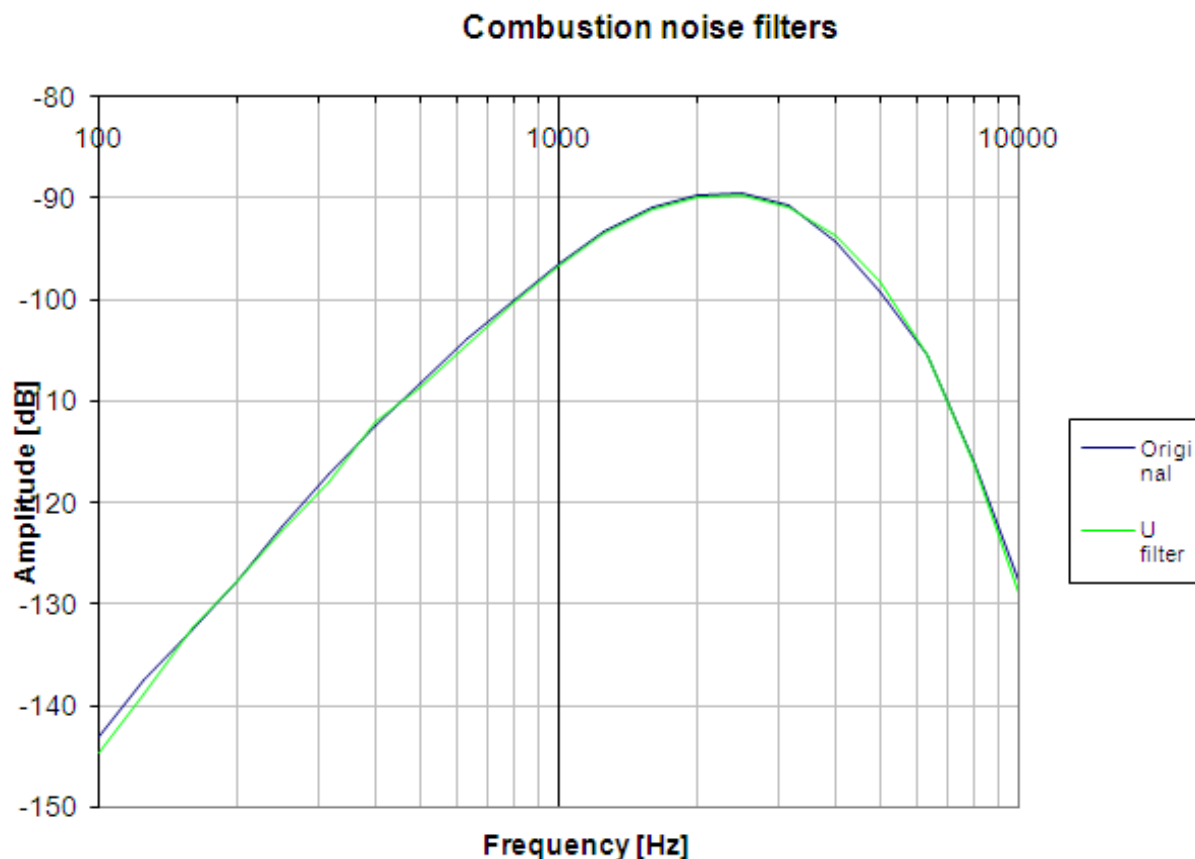
AVL Concerto je generično orodje za postprocesiranje podatkov, ki se najbolj specializira za analizo podatkov iz analizatorjev motorjev z notranjim izgorevanjem. Velja za enega najpogosteje uporabljenih programskih paketov za analizo podatkov takšne vrste. Uradni format za AVL Concerto je iFile.

9. POVEZAVA S TESTNO MIZO

Motor se lahko testira v avtomobilu med samo vožnjo. Pogosteje pa ga postavijo v posebno sobo na t. i. testno mizo. Tako lažje pridemo do vseh delov motorja in dodamo različne senzorje. V programskem paketu Dewesoft je napisan vtičnik, ki ves čas pošilja osnovne podatke (npr. maksimalni tlak, hitrost) prek lokalnega omrežja testni mizi. Če pride do napačnega delovanja motorja, recimo prevelike hitrosti ali obremenitve, testna miza namreč sama ustavi motor in opozori uporabnika, da je prišlo do napake.

10. CA NOISE

CA noise je matematični modul, ki se uporablja za oceno hrupa znotraj avtomobila, glede na merjeni tlak v valju. Uporablja se poseben filter, ki ga je razvil AVL, imenovan U-A-filter (slika 36). To je NEO-filter in se računa v časovni domeni. Izhod je hrup v decibelih.



Slika 36. U-Filter. Na y-osi je amplituda [dB], na x-osi pa je prikazana frekvenca [Hz]

Filter je sestavljen iz dveh filtrov, in sicer A in U. U-filter je prenosna karakteristika med motorjem in kabino s potniki, A pa je utežna krivulja. Utežna krivulja predpostavlja, da je človeško uho manj občutljivo za nizke kot za visoke frekvence. Tako podatki, uteženi z A-krivuljo, prikažejo realnejše amplitudne vrednosti zvoka, ki lahko poškoduje človeški sluh.

11. PRIMERJAVA S PROGRAMOM CONCERTO (AVL)

Analizator sem primerjal z AVL-ovim paketom za analizo motorjev z izgorevanjem, Concertom. Concerto je industrijski standard za analizo motorjev. Uporabljen je bil dizelski dvotaktni motor.

a) Geometrija motorja:

Število valjev: 2 (1 je bil uporabljen za primerjavo)

Hod bata: 73 mm

Premer bata: 76 mm

Ojnica: 131 mm

Kompresijsko razmerje: 9

Minimalni volumen valja: $0,04 \text{ dm}^3$

Maksimalni volumen valja: $0,37 \text{ dm}^3$

Gorivo: dizelsko

b) Nastavitve za meritev kota:

Vzorčenje: interna ura

Senzor za kot: 60-2 (60 zob in 2 manjkajoča)

Resolucija kota: 1 kotna stopinja

c) Drugi parametri za primerjavo:

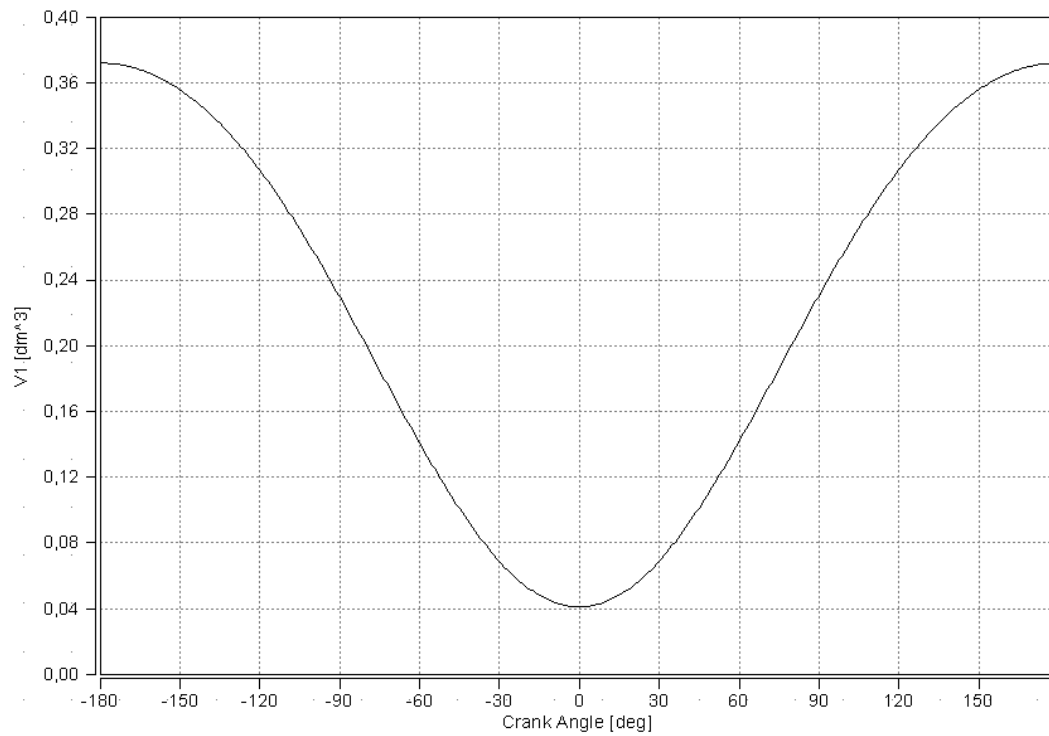
Politropni eksponent: 1,32

Masa plina: 0,3 g

Ničenje tlaka: Termodinamično ničenje

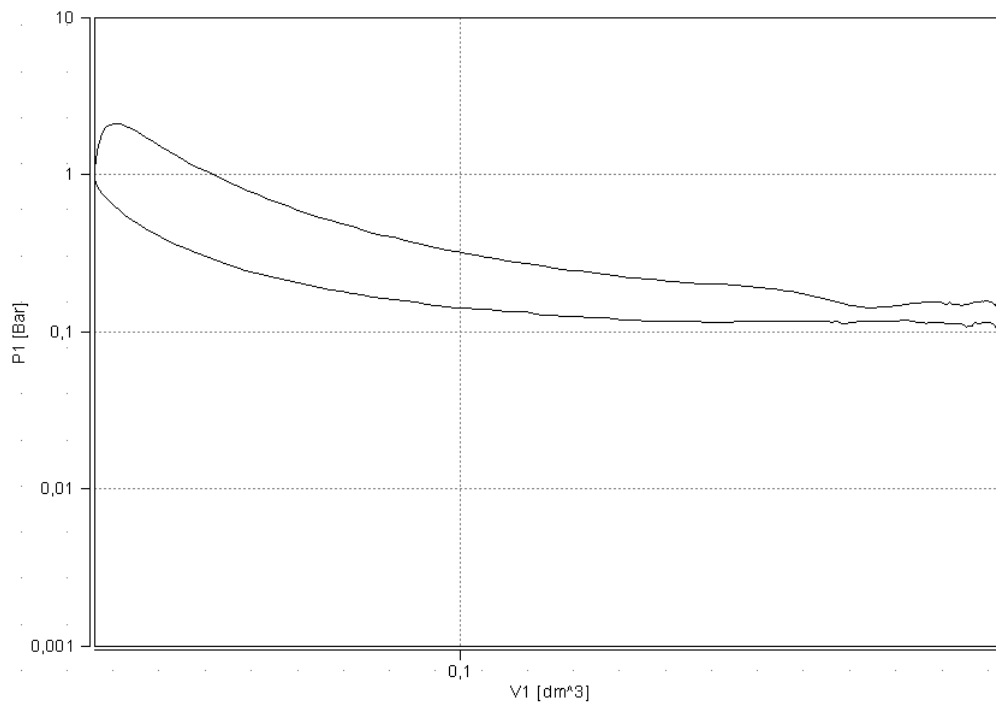
Vsi spodnji diagrami in tabele so narejeni v Concertu.

11.1 Volumen



Slika 37. Odvisnost volumna [dm^3] od kota [kotne stopinje] pri dvotaktnem motorju.

11.2 pV-Diagram



Slika 38. Diagram pV za dvotaktni motor. Na y-osi je tlak [bar], na x-osi pa je prostornina valja [dm^3]

11.3 Posamezne vrednosti

ADMax1	Deg	4,0000
ADMax1_Concerto	deg	4,5000
APMax1	Deg	8,0000
APmax1_Concerto	deg	8,0000
DMax1	Bar/deg	0,20388
DMax1_Concerto	Bar/deg	0,20768
I51	Deg	1,7915
I51_Concerto	deg	3,3000
I901	Deg	5,8554
I901_Concerto	deg	7,4500
IMEPn1	Bar	0,14542
IMEPn1_Concerto	Bar	0,14545
PMax1	Bar	2,1225
PMax1_Concerto	Bar	2,1225

Slika 39. Tabela primerjave rezultatov za Dewesoft in Concerto.

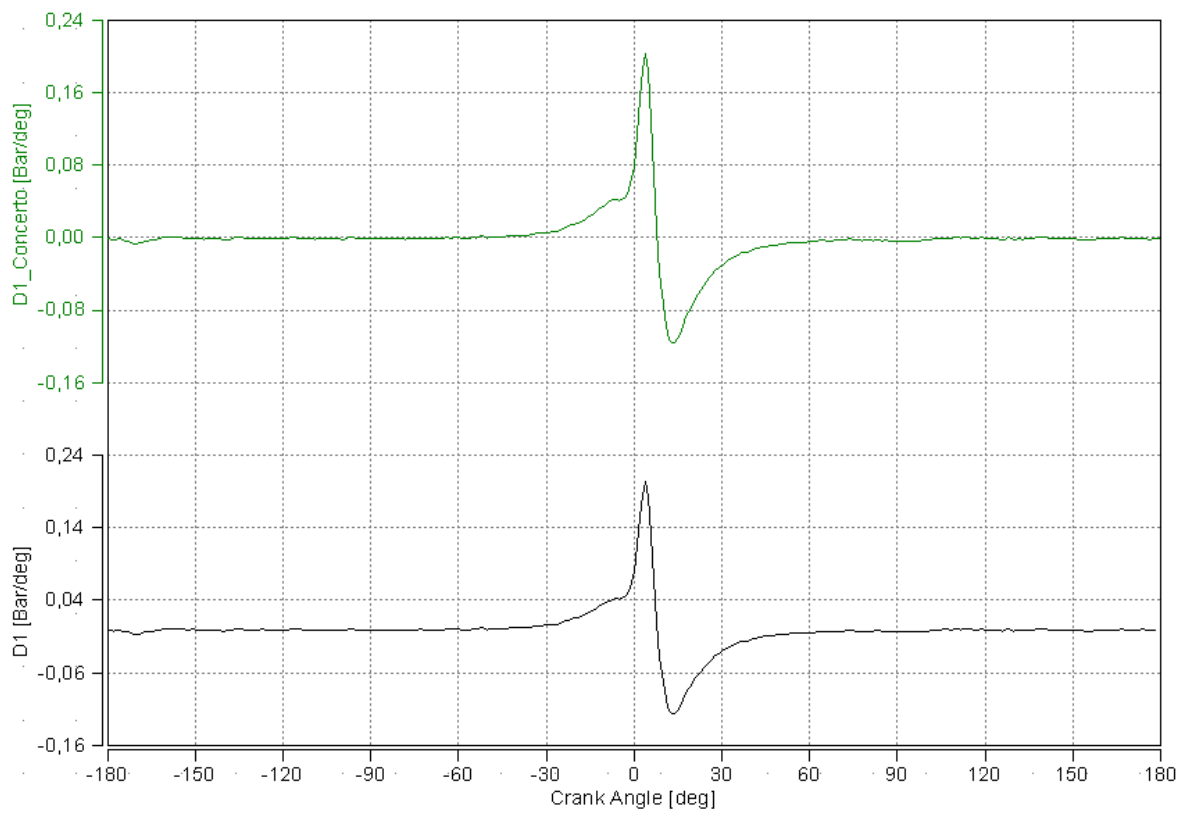
Iz slike 38 vidimo, da se vrednosti dobro ujemajo. Največja težava je pri vrednosti za sproščanje toplotne energije. Rezolucija meritve je 1 kotna stopinja, napaka pa znaša za I5 1,5085 in za I90 1,5946 kotne stopinje. Ker je možno to vrednost izračunati na več načinov, je precej verjetno, da je v Concertu uporabljen drugačen algoritem. Napaka pa niti ni tako kritična, saj znaša okrog 0,5%. V praksi je samo iz tlaka zelo težko izračunati sproščanje toplote zelo natančno. Napake lahko glede na model znašajo tudi do 10%. Pri formuli za odvod se prav tako vidi odstopanje. Ker je vrednost za pozicijo maksimalnega odvoda med dvema polnima kotnima stopinjama, sklepam, da uporabljajo neko drugo formulo. Glede na to, da je vrednost točno na sredini resolucije meritve, sklepam, da uporabljajo metodo končnih diferenc:

$$f'(x) = \frac{f(x+h) - f(x)}{h} \quad (11.1)$$

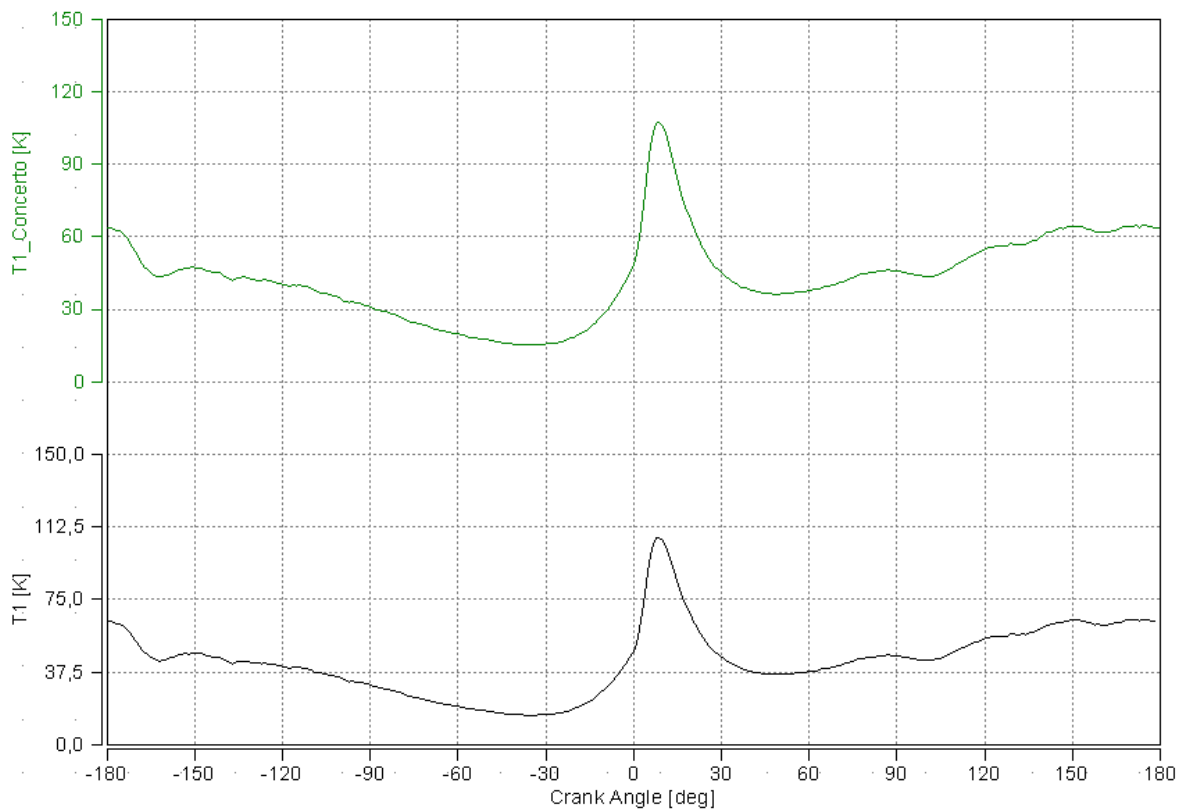
Rezultat pa premaknejo za 1 / resolucija / 2 kotne stopinje.

11.4 Vektorji

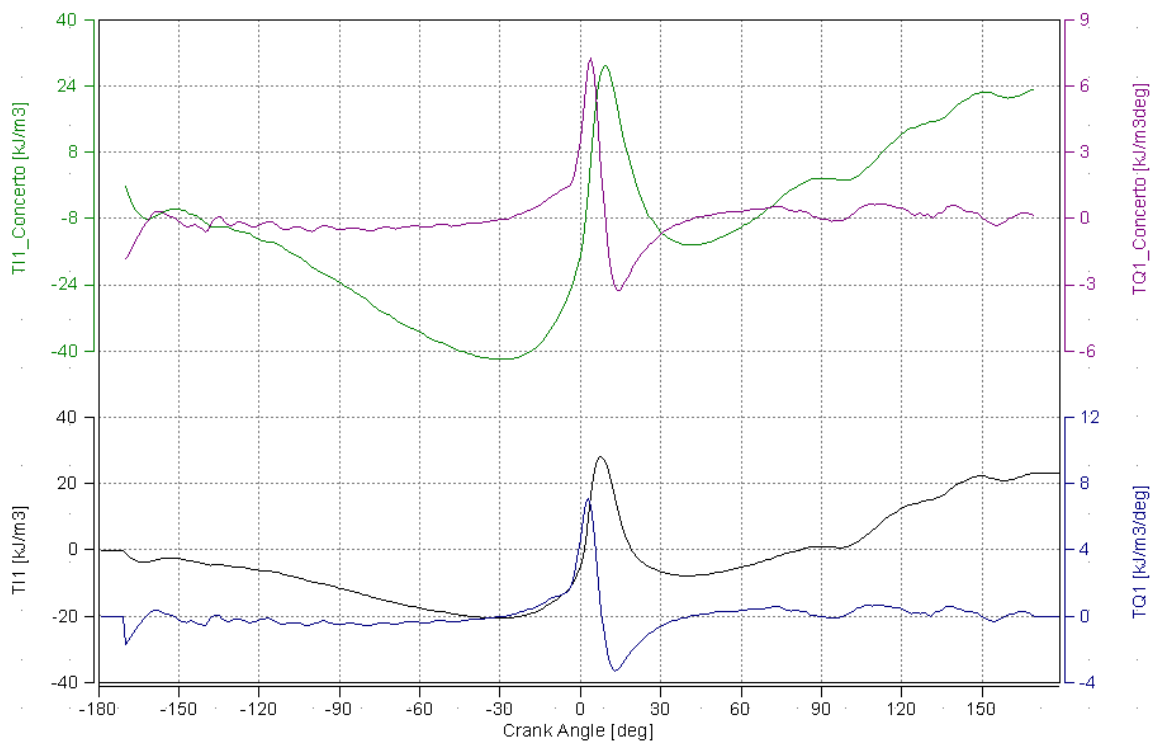
Tukaj so prikazani še vektorji, pri katerih se lepo vidi, kako se rezultati ujemajo. Na zgornji polovici vsakega grafa so prikazane vrednosti, izračunane v Concertu, na spodnji pa vrednosti, izvožene iz Dewesofta. Vse drugo se tako kot pri posameznih vrednostih dobro ujema, pri sproščanju toplotne energije pa vidimo nekaj razlike.



Slika 40. Primerjava odvoda tlaka med orodjema Dewesoft in Concerto. Na y-osi je tlak $\left[\frac{\text{bar}}{\text{deg}}\right]$, na x-osi pa je prikazan en obrat motorja [kotne stopinje]



Slika 41. Primerjava temperature v valju med orodjema Dewesoft in Concerto. Na y-osi je temperatura [K], na x-osi pa je prikazan en obrat motorja [kotne stopinje]



Slika 42. Primerjava $Tq \left[\frac{kJ}{m^3 \cdot deg} \right]$ in $Ti \frac{kJ}{m^3}$ na y-osi (med orodjema Dewesoft in Concerto). Na x-osi je prikazan en del obrata motorja [kotne stopinje]

12. SKLEPNE UGOTOVITVE IN MOŽNOSTI ZA NADALJNI RAZVOJ

Analizator motorjev z notranjim izgorevanjem se je v praksi izkazal zelo dobro. Najbolj težaven del je predstavljala izdelava filtra za prevzorčenje signala. Osnovna implementacija filtra je zelo počasna, zato je bilo potrebna dobra optimizacija v smislu same izdelave filtra in tudi kodiranja algoritma. Ker je filter potreben samo za vzorčenje z interno uro (velika obremenitev CPU-ja), te težave pri interni uri ni bilo. Pri vzorčenju z eksterno uro je obremenitev CPU-ja bistveno manjša, vendar se pojavijo težave z višjimi harmoniki (višji od $f_s/2$). Te težave se praktično ne da odpraviti. Če primerjamo obe metodi so rezultati praktično enaki. Ko obe metodi primerjamo z standardnim programskim paketom za meritve na področju motorjev (AVL Concerto) so rezultati primerljivi.

Naslednja težava je bil iz izračunom sproščanja toplote. Model za sproščanje toplote je zelo splošen in veliko spremenljivk se samo z meritvijo tlaka ne da upoštevati (npr. izgube pri batnih obročkih). Potrebno je bilo predelati enačbo tako, da so rezultati praktično uporabni in čim bližje že obstoječim programskim paketom.

Pri izdelavi standardnih diagramov za prikaz podatkov iz analizatorja ni bilo večjih težav. Potrebno je bilo dobro definirati kako se bodo podatki prikazovali. Potem pa izdelati minimalno število diagramov, ki lahko prikažejo vse tipe rezultatov.

Končna rešitev je vsebovala skoraj vse standardne funkcije iz področja analize motorjev. Dodali smo tudi vzorčenje z interno uro, kar velja na tem področju za najsodobnejši pristop. S tem, da je programski modul integriran v programski paket Dewesoft pridobimo to, da lahko poleg osnovne meritve motorja, spremljamo tudi napetosti, tokove, uporabljamo video kamere, CAN signale, itn.

Za analizator je še veliko možnosti za nadaljnji razvoj. Tukaj bom naštel najbolj bistvene, ki bi po izkušnjah analizatorju dodale največjo vrednost:

- a) **povprečenje tlaka in dodatnih kanalov:** V trenutnem analizatorju je mogoče dobiti podatke za trenutni tlak in za povprečno vrednost za celo meritve. To velja tudi za dodatne kanale. Velikokrat pa je zaželeno, da bi se povprečile vrednosti za zadnjih N ciklov, pri čemer je N večje od nič in manjše od števila vseh ciklov. To pride v poštev, kadar imajo direktni podatki s senzorjev vsebujejo veliko šuma. Na takšen način ne pokvarimo podatkov, kakovost signala pa močno izboljšamo.
- b) **poljubne formule (predloge za formule in urejevalnik):** Analizator uporablja veliko parametrov za definicijo geometrije in drugih nastavitvev. Vsebuje tudi veliko različnih formul. Vendar pa je uporabnikov veliko in vsi so med sabo različni, vsak potrebuje druge formule in izračune. Smiselno bi bilo, da bi se te stvari predstavile iz analizatorja v XML-predloge, tako da bi se lahko parametri dodajali po želji. Glede na to pa bi bilo dobro, če bi se dale formule poljubno definirati. Na začetku bi bile uradne predloge za dvotaktni in štiritalaktni motor z vsemi standardnimi formulami, uporabnik pa bi lahko prek urejevalnika dodajal svoje formule in parametre.
- c) **formule za tlak in dodatne kanale:** Smiselno bi bilo, da je mogoče definirati formule za tlak ter dodatne kanale, in sicer za:

- trenutne vrednosti
 - povprečne vrednosti za N ciklov
 - povprečne vrednosti za celotno meritev
- d) **detekcija ZML:** V algoritem za detekcijo ZML bi bilo dobro dodati, da se lahko algoritem uporabi na vseh valjih (ne samo referenčnem). Tako bi dobili odmike vseh valjev, ne samo prvega, in s tem preverjali izmerjeno geometrijo motorja s specifikacijo motorja. Ker je gorivo drago, veliki počasni motorji (npr. za ladje) pa porabijo v eni uri ogromno goriva, bi bilo dobro, če bi detekcija kotov lahko izračunala vse kote naenkrat. Ker pa to velikokrat ni možno, ker ne sme biti zgorevanja v delovnem taktu, mora biti dana možnost, da lahko preverjamo enega po enega.
- e) **izboljšanje resolucije za meritev z eksterno uro:** Odmik prvega valja od ničle senzorja je omejen z resolucijo meritve. To omejuje nekatere izračune, kot sta npr. MEP ali pa pozicija maksimalnega tlaka. Realno gledano pa lahko iz detekcije ZML zvezmo točen kot maksimalnega tlaka.

Vzemimo primer, da meritev izvajamo z resolucijo 1 kotna stopinje. ZML detekcija nam da kot rezultat 10,5 kotne stopinje. To vrednost lahko razbijemo na dve vrednosti, in sicer na zaokroženo vrednost resolucije, ki jo bomo uporabili za zajem podatkov, in na ostanek, ki ga bomo uporabili za računanje volumna, pozicij in označitev osi na grafih. Torej:

novi_odmik = TruncToResolution (ZML_odmik)
ostanek_odmika = ZML_odmik – novi_odmik

pri čemer TruncToResolution poišče najbližjo vrednost v kotnih stopinjah resolucije meritve, ki je nižja od ZML_odmik.

Podatke za valje še vedno zajemamo z vrednostjo novi_odmik. Popravimo pa vrednosti za:

- volumen začnemo računati na $-180 + \text{ostanek_odmika}$ za dvotaktni motor in $-360 + \text{ostanek_odmika}$ za štiritaktni motor. Tako bodo vse enačbe (MEP, sproščanje toplotne energije, delo itn.), ki uporabljajo volumen, izračunane natančneje.
- osi na grafih prestavimo za ostanek_odmika v levo in dobimo točnejši prikaz krivulje tlaka in dodatnih kanalov.
- izračune maksimalnih pozicij popravimo za ostanek_odmika, s tem pa dobimo še točnejši rezultat za npr. pozicijo maksimalnega odvoda, tlaka itn.

VIRI

- [1] Haim H. Bau, *The thermodynamics of the internal combustion engine*. Dostopno na: <http://www.scribd.com/doc/29382348/Eng-Therm>
- [2] (2002) BP lubricants and fuels, *Why does my car knock?* Dostopno na: http://www.bp.com/liveassets/bp_internet/australia/corporate_australia/STAGING/local_assets/downloads_pdfs/f/Fuel_news_why_does_car_knock.pdf
- [3] (2011) Dewesoft, *DEWESoft v7.0.3, data acquisition, processing, analyzing and storage software, User manual*. Dostopno na: http://www.dewesoft.com/download?file=UserManual_703.pdf
- [4] (2008) Dewetron, *DEWE-ORION-1616-10x, Technical reference manual*. Dostopno na: http://www.dewamerica.com/downloads/manuals/adcards/ORION-1616-100_manual.pdf
- [5] Dewetron, *DEWE-5000, Instruments*. Dostopno na: http://www.dewetron.com/fileadmin/download.dewetron.com/products/dewe-5xxx/brochures/english/lowres/dewetron_trg_dewe-5000_de_b100401e.pdf
- [6] John B. Heywood, *Internal Combustion Engine Fundamentals*, Singapore, McGraw-Hill, 1988
- [7] Richard G. Lyons, *Understanding Digital Signal Processing (2nd Edition)*, Prentice Hall PTR, 2004
- [8] Bojan Orel, *Osnove numerične matematike*, Ljubljana, Založba FE in FRI, 1997
- [9] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, *Numerical recipes in C (Second Edition)*, Cambridge University Press, 1988
- [10] Steven W. Smith, *Digital signal processing. A practical guide for engineers and scientists*, Newnes, 2003