



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem**  
Villamosmérnöki és Informatikai Kar  
Méréstechnika és Információs Rendszerek Tanszék

# **Jelterjedés-vizsgálat általános abszolút nyomásszenzor ASIC áramkörében**

TUDOMÁNYOS DIÁKKÖRI KONFERENCIA

*Készítette*

Kovács Nóra Orsolya

*Tanszéki Konzulens*

Scherer Balázs

*Külső Konzulens*

Bartakovics Tamás

2013. október 25.

# Tartalomjegyzék

<b>Összefoglaló</b>	<b>3</b>
<b>Abstract</b>	<b>4</b>
<b>1. Nyomá szenzorok bemutatása</b>	<b>5</b>
1.1. A nyomá szenzorokról általánosságban . . . . .	5
1.1.1. A szenzor karakterizálása . . . . .	6
1.1.2. Az érzékelő pontossági tartománya . . . . .	7
1.1.3. A nyomá szenzorok fizikai felépítése . . . . .	8
1.2. A mikromechanikus rendszer . . . . .	9
1.2.1. A nyomá sérzékelők csoportosítása - abszolút és relatív nyomá szenzor . . . . .	11
1.3. ASIC . . . . .	11
1.3.1. A hőmérsékletmérés . . . . .	12
1.4. A kommunikáció . . . . .	12
1.4.1. Az SPI protokoll leírása . . . . .	13
1.4.2. A SENT protokoll leírása . . . . .	14
<b>2. A jelterjedés bemutatása</b>	<b>15</b>
2.1. A MAP szenzor . . . . .	15
2.2. A kiértékelő áramkör bemutatása . . . . .	16
2.2.1. A bemeneti áramkör feladata . . . . .	17
2.2.2. Az órajel előállítása . . . . .	17
2.2.3. Az analóg-digitális átalakító (ADC) . . . . .	18
2.2.4. A decimális szűrő feladata . . . . .	18
2.3. Digitális jelfeldolgozás (DSP) . . . . .	19
2.3.1. A dióda-hőmérséklet jelútja . . . . .	19
2.3.2. Az NTC ellenállás jelútja . . . . .	20
2.3.3. A nyomá sérték jelútja . . . . .	20
2.4. Karakterizálás . . . . .	20
<b>3. A mérési környezet felépítése</b>	<b>21</b>
3.1. A klímakamra . . . . .	22
3.2. A nyomá szabályozás . . . . .	23

3.3.	Az Aufnahme . . . . .	23
3.4.	A mérőelektronika . . . . .	25
3.4.1.	Unibox . . . . .	25
3.4.2.	PXI Rack . . . . .	25
3.4.3.	A számítógép feladata . . . . .	25
<b>4.</b>	<b>Szoftver</b>	<b>27</b>
4.1.	A szoftverfejlesztési feladat bemutatása . . . . .	27
4.2.	A parancs leírása . . . . .	28
4.2.1.	A fejlesztési környezet - CVI . . . . .	30
4.3.	Az adatfeldolgozási módszerek . . . . .	31
4.3.1.	A TDMS struktúra bemutatása . . . . .	31
4.3.2.	A MARS program bemutatása . . . . .	32
4.4.	A programkód dokumentálása . . . . .	35
4.4.1.	Az SVN verziókövető szoftver bemutatása . . . . .	35
4.4.2.	Dokumentáció létrehozása a programkódból - Doxygen . . . . .	37
<b>5.</b>	<b>Összefoglalás</b>	<b>38</b>
5.1.	Jövőbeli tervek . . . . .	38
	<b>Ábrák jegyzéke</b>	<b>40</b>

# Összefoglaló

Napjainkban egyre nagyobb hangsúlyt kap a szenzortechnológia. Számos szenzortípus létezik a gyorsulásszenzoroktól kezdve a mágneses szenzorokon át a fényszenzorokig, amelyeket a mindennapi élet legkülönbözőbb területein is megtalálunk. Nyomás-szenzort alkalmaznak a robotikában, autópárhán és számos más pneumatikus érzékelést megkövetelő felhasználási területeken. A nyomásszenzorok eltérhetnek technológiájukban, funkciójukban, működésükben valamint előállítási költségben, de közös tulajdonságuk hogy a mérés és tesztelés lehetőségét biztosítani kell, akár extrém feltételek között is (például a -40 valamint 125 Celsius fokos tartományban).

Az autópárhán fejlődésével egyre kisebb, érzékenyebb és bonyolultabb kialakítású szenzorok kerültek forgalomba, amelyeknél a környezeti zavarhatások ellenére az érzékelő valós jelének el kell jutnia a gépjármű különböző elektronikai kiértékelő moduljaiba vagy a központi elektronikába. Ezért fontos a szenzor karakterizálása, amelyen a szenzor jeleinek mérését és kiértékelését értjük. A karakterisztikának jeleznie kell, ha valamely paraméter az átlagos vagy biztonságos működéstől eltér, hiszen ez súlyos balesetekhez vezethet. Az ilyen szenzorok gyártása milliós nagyságrendekben történik, ezért a hiba bekövetkezésének, terjedésének és hatásának vizsgálata is fontos hangsúlyt kap a fejlesztés során.

A dolgozat bemutatja a Robert Bosch Kft. által gyártott, az autópárhán használt nyomásszenzorok karakterizálásához szükséges mérési, tesztelési és fejlesztési irányelveket, egy általános szenzortípus felhasználásával. Bemutatásra kerül az érzékelő felépítése és működési elve, kommunikációs protokollja, a jelút és egyes regiszterek vizsgálata az ASIC-en elül, valamint a tesztelési és fejlesztési környezet. Egy meglévő mérési szoftver fejlesztésének bemutatására is sor kerül, amelyben ezeket a csomópontokat különböző nyomás- és hőmérsékletértékek mellett mérjük. Végül ezeket az adatokat adatstruktúrában kerülnek tárolásra.

Jelen dolgozat célja egy általános nyomásszenzor regisztereinek kiolvasása. Ez számottevő fejlődés, hiszen eddig az ASIC struktúra kimenő és bemenő jeleinek paramétereit tudtuk mérni, viszont ezen fejlesztéssel már a különböző A/D átalakító, szűrő, átlagoló, stb. elemek előtti és utáni regiszterek értékeit is el tudjuk tárolni, majd ezeket eredményesen kiértékelni.

# Abstract

Nowadays great emphasis has been placed on sensor technology. Countless sensor types can be found in the market; acceleration and pressure sensors, magnetic sensors, optical sensors, etc.. Pressure sensors are used in the field of robotics, automotive technology, and in numerous other applications. The types of pressure sensors differ in manufacturing technology, function, operation and production cost, but generally, they have the ability to measure and evaluate their sign even in extreme environmental effects, such as extremely high or low temperature.

As the automotive technology became more and more advanced, the characteristics of the sensors became one of the most highlighted aspects in the developing process. To optimize the sign of the sensor a design was made to reduce the noise from the environmental effects, which is called characterization. This process includes the measurement and evaluation of the sign. The result must show the parameters of the sensor, which are outside of the optimal operating limit, because they can cause malfunctions and accidents. For industrial production the number of the manufactured sensors can be measured in millions, so the probability of the occurrence of a fault should be close to zero.

My thesis presents the guidelines of the measurement, testing and development in the automotive industry provided by Robert Bosch Ltd., especially in the field of the absolute pressure sensors. Construction, communication and the signal path of the sensor will be introduced as well as the developing and testing environment. An existing measurement software will be described, in which the data nodes of the signal path of the sensor can be monitored and stored.

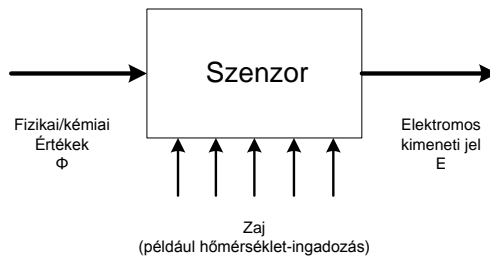
The aim of my research is to read the registers of a generally used pressure sensor. This development is significant in the present and in the future, because not only can the measurement of the input and the output parameters of the ASIC structure be made, but also the evaluation and storage of the data of the A/D converter, filter, average unit, etc. can be achieved.

# 1. fejezet

## Nyomásszenzorok bemutatása

### 1.1. A nyomásszenzorokról általánosságban

A szenzorok olyan mérés technikai érzékelők, amelyek valamilyen fizikai vagy kémiai paramétert vagy paramétereket elektromos jellé alakítanak át. Ilyen paraméter lehet például a nyomás, hőmérséklet, gyorsulás, stb. Nyomásszenzorok esetében - mivel ez a fizikai mennyiség közvetlenül nem mérhető, - szükség van egy mikromechanikus szerkezetre, aminek kimenete a mért nyomásértékeknek megfelelő elektromos jel. Biztosítani kell továbbá ennek az analóg, majd digitális jelnek a biztonságos továbbítását.



1.1. ábra. Egy általános érzékelő működési alapelve

Az átalakítás folyamatát az alábbi módon írhatjuk le, ahol az elektromos jeleket  $E$ -vel, az ezen kívüli stimulusokat (a különböző fizikai és kémiai jeleket) pedig  $\Phi$ -vel jelöljük (1.1. ábra):

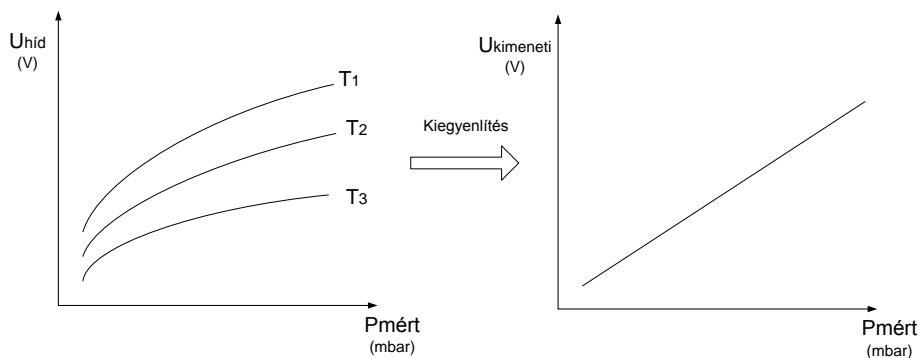
$$\begin{aligned} E &= f(\Phi, Y1, Y2, \dots), \\ \Phi &= g(E, Y1, Y2, \dots). \end{aligned} \tag{1.1}$$

Ahogy az (1.1) egyenletekben is látszódik, fontos a környezeti zavarjelek fellépéséből adódó hibaértékek számítása, amelyeket az egyenletekben az  $Y$  paraméterek jelölnek. Az  $f$  és  $g$  függvények, valamint a zavarjelek és a kimeneti jel meghatározása után, a  $\Phi$  mérési értékek további kiértékelését egy mikrokontroller-rendszer dolgozza fel és továbbítja a vezérlési elektronikának. Ez az úgynevezett intelligens, vagy smart szenzorok

feldolgozási szekvenciája. Ezek az érzékelők mára elengedhetlenné váltak a modern autógyártásban a modern szenzorhálózat komplexitása miatt [1].

A kész autóiipari szenzorok megvalósulásáig lezajló tervezési, szimulációs és mérés-technikai folyamatok bemutatásához a Robert Bosch Kft.-nél töltött gyakorlatom ad megfelelő tapasztalatot, ezért az itt alkalmazott szenzortípus és fejlesztési környezet segítségével fogom bemutatni, hogy a jeloptimalizáció milyen nagy szerepet játszik egy érzékelőtípus fejlesztésében, majd megvalósulásában.

### 1.1.1. A szenzor karakterizálása



1.2. ábra. A mért nyomásértékek eltérése különböző hőmérsékleteken

A szenzor karakterizálása, vagyis a kiegyenlítés alatt a szenzor nyers adatainak kompenzálását értjük, mivel a szenzor által mért értékek nem egyeznek meg a tényleges környezeti paraméterekkel. Nyomásszenzorok esetében a nyers, valamint a korrigált értékek jelleggörbéit a 1.2 ábra mutatja meg. Ebben az esetben a változást nagy részben a nyomás hőmérsékletfüggése okozza.

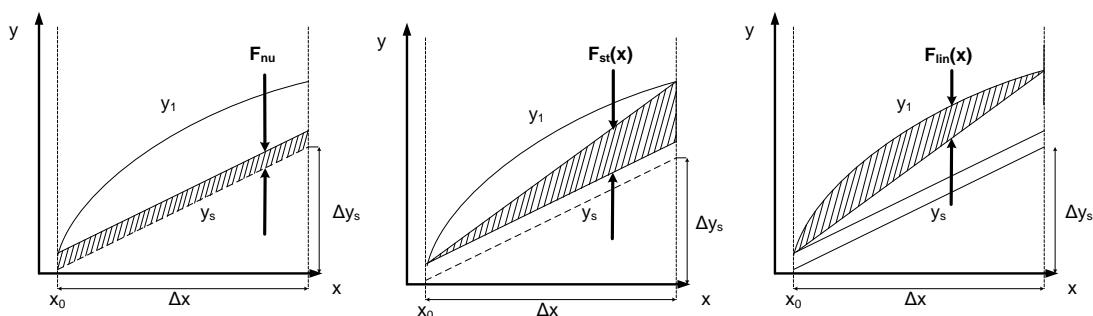
A karakterizálás során a mért jelek, és az úgynevezett referenciaszenzor jeleinek összehasonlítása történik. A kiegyenlítéshez több üzemi pont értékeinek mérése közben a pontosabb adatokat szolgáltató referenciaszenzorhoz történik a szenzor ténylegesen mért jeleinek igazítása.

A referenciaszenzor olyan eszköz, amely igen nagy pontossággal, stabilan méri a nyomásértéket, vagyis egy etalon lesz a mérési folyamat során. Ez a szenzor biztosítja, hogy a mérési bizonytalanságra vonatkozó követelmények mindig a megengedett határokon belül mozogjanak. Mivel nagy pontosságú eszközről van szó, amely egy viszonyítási alapot képez a mérés során, ezért szükség van olyan ellenőrzésekre, amely a referenciaszenzor megfelelőségét igazolja. Ilyen kötelező vizsgálat például az MGF1 mérés, amely esetén a

szenzor mért jeleinek kiértékelése történik a toleranciatartományok figyelembe vételével:

$$\frac{0,2(T_F - T_A)}{6\sigma_n} > 1,33. \quad (1.2)$$

Az (1.2)-es egyenletben a  $T_F$  jelölés a felső, a  $T_A$  az alsó határértéket jelöli, míg a  $\sigma_n$  a  $n$  számú mérés szórását adja meg. A vizsgálat sikerességének feltétele, hogy összefüggés kisebb legyen a konstans 1,33-as értéknél.



1.3. ábra. A teljes hiba felosztása

Ahogy 1.3. ábrán is látszódik, ha az ideális jelleggörbének a lineáris egyenest választjuk, akkor ettől az opcionális állapottól való eltérések a következők lehetnek: ofszet hiba (bal ábra), erősítési hiba(középső ábra) valamint linearitási hiba (jobb ábra). Az ofszet hiba vagy nullpont-eltolódás ( $F_{nu}$ ) egy eltolódás az előírt jelleggörbén az  $y$ -tengely mentén, vagyis nulla bemeneti értékek hatására a kimenet nullától különböző értéket vesz fel. Az erősítési, vagy gain hiba ( $F_{st}$ ) amikor az egyenes meredeksége eltér az előírt jelleggörbétől és a linearitási hiba ( $F_{lin}$ ) amikor a linearitási jelleg torzul.

A fent leírt hibák szisztematikusan lépnek fel, így hatásukat kompenzálni tudjuk, míg a véletlenszerű hibajelenségek (mint a környezeti zajokból és az öregedésből adódó hatások) ezekkel a korrekciós paraméterekkel már kevésbé módosíthatóak.

### 1.1.2. Az érzékelő pontossági tartománya

A bevezetőben említésre került a megbízhatósági valószínűség számítás, amely magában foglalja, hogy az adott érzékelő milyen valószínűséggel hibásodik meg 15 éves átlag élettartamra viszonyítva.



Ezeknek a megbízhatósági értékeknek az alábbi határértéken belül kell mozogniuk:

- Hibaszázalék az érzékelők esetében  $< 10$  ppm
- ASIC meghibásodási valószínűsége  $< 2$  ppm
- IC meghibásodási valószínűsége  $\ll 0,2$  ppm
- Diszkrét alkatrészek meghibásodási valószínűsége  $< 0,02$  ppm

Az érzékelők működési biztonsága többszöröse más eszközökének, erre példa, hogy mobiltelefonok esetében ez az érték körülbelül 5000 ppm.

Ezeket az értékeket meghatározni, és segítségükkel az esetleges hibalehetőségeket, és gyenge pontokat megállapítani csak olyan előzetes vizsgálatokkal lehet, amelyekkel a szenzort érő káros hatásokat szimuláljuk. Ez a szimuláció eltér személygépkocsik, vagy tehergépkocsik esetén, hiszen a gépjárműnek nem ugyanazokat a terheléseket kell elviselnie a két esetben.

### 1.1.3. A nyomásszenzorok fizikai felépítése



1.4. ábra. Az abszolút nyomásszenzor felépítése

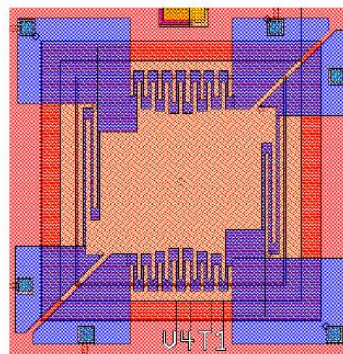
Egy szenzor magából a szenzorelemből, és az azt körülvevő házból áll, amit együttesen modulnak nevezünk. A modulban a piezo-rezisztív szenzorelem, valamint a kiértékelő áramkör általában egy külön szilícium lapkán helyezkedik el ( 1.4. ábra). A két elem jelátvitelle bondokkal, vagyis mikrohuzallal kerül megvalósításra.

Az érzékelő és a modul kommunikációs protokollja eltér egymástól, úgy mint a programozhatóságuk mértéke is. Míg az érzékelő a SPI (Serial Peripheral Interface) kommunikációs protokollt használja - amely nagymértékű programozhatóságot és fejlesztési lehetőséget biztosít, - addig a modul SENT (Single Edge Nibble Transmission) kommunikációja csupán korlátozott körű parancsok használatát engedélyezi.

Funkciójukat tekintve a szenzorelem méri a fizikai nyomásértéket, amit feszültségáramnyosan megjelenít a kimenetén. Ezt a jelet használja fel a továbbiakban egy alkalmazás specifikus integrált áramkör (ASIC - Application-Specific Integrated Circuit), amely a jelet a kiegyenlítés folyamata után SENT protokoll segítségével továbbítja.

## 1.2. A mikromechanikus rendszer

A nyomás mérése több féle módon történhet, mint például közvetlen nyomásméréssel vagy membrán szerkezetek alkalmazásával. Közvetlen nyomásmérést leggyakrabban nagynyomású szenzorok esetében alkalmaznak, az autóiparban túlnyomó többségében a membrán szenzorok terjedtek el, amelyek a kis- és közepes nyomásértékek mérésére szolgálnak. Ennek a membránszerkezetnek a segítségével történik a jelképzés, vagyis a fizikai nyomásérték elektromos jellé alakítása közbülső mechanikai fokozat segítségével. Kisnyomású rendszerek például a szívócső-, illetve a töltőnyomás érzékelők (1 bar-5 bar), közepes nyomású rendszerekben 10 bar-on működnek a féknyomás érzékelők, nagynyomású szenzorokra példa a lengéscsillapító rendszerekben használt érzékelők, amelyeket 200 bar-os tartományú értékek mérésére alkalmazunk.



1.5. ábra. A Wheatstone-híd szerkezete a membránon

Az erő-, nyomaték- és nyomásmérés szenzorainak működése három fő mérési elven történik: nyúlásmérő bélyeges, piezoelektromos, valamint kapacitív módon.

Nyúlásmérő alkalmazásakor a membrán deformáció következtében megváltozik a félvezető nyúlásmérő bélyegek elektromos ellenállása. A piezoelektromos fólia a mechanikai változást mérhető elektromos jellé alakítja. Ebben az adalékolt félvezető kristályban a

deformáció folytán a szabad töltéshordozók száma megváltozik, vagyis az elektronok gátoltabb vagy szabadabb áramlását teszi lehetővé, ami fajlagos ellenállás-növekedést vagy csökkenést okoz. Kapacitásváltozás mérésénél a membrán kismértékű behajlása a kapacitás megváltozását okozza, vagyis a változás a membránok közötti távolsággal fordítottan arányos [2].

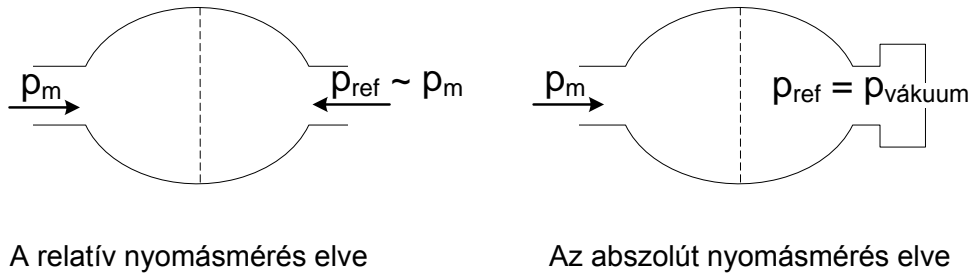
A membránszerkezet kialakítását meghatározza a mérendő nyomás nagysága, valamint a mérendő közeg kémiai és fizikai jellemzői is, hiszen a nyomásszenzorok közvetlen kapcsolatban vannak a mért közeggel. Ezért a kapacitás mérésének elvén alapuló szenzorok nem alkalmazhatók nyomásmérésre - az anyag dielektromos jellemzői torzítanak a mérési eredményeket. Mivel nagy nyomású érzékelőknél csak nagy vastagságú és kis átmérőjű membránok használhatóak, ezért ezeknél a távolságmérő módszerek nem jöhetnek szóba a mikrométeres behajlás miatt. Ez a mérés technikai elvet ezért kizárólag kis-és nagy nyomású szenzoroknál található meg. A Wheatstone-hídat felépítő ellenállások a membrán megnyúlásának hatására megváltoznak, és ezzel együtt módosul a feszültségarány és az  $U_m$  mérőfeszültség is. Ez az elrendezés sokkal érzékenyebb rendszert hoz létre, mintha a feszültség kiértékelése egyenként történne az ellenállásokon.

A felsorolt érvek miatt a legalkalmasabb a nyúlásmérő-szalagok alkalmazása a membrán felületén, amelyen a membrán deformációjának hatására mechanikus feszültség jön létre. Ezeket az ellenállásokat Wheatstone-hídba kapcsolva jön létre a kapcsolat a mérendő nyomás, és a híd kimenetén megjelenő feszültség között (1.5.ábra).

A közeg a nyomásszenzorokban nem csupán a levegő, illetve annak hiánya a vákuum lehet, hanem olaj, víz, ózon, por és hamu, üzemanyag, sópermet, valamint egyéb kémiai anyagok is. Ezek az anyagok folyamatosan terhelik, korrodálják a mikromechanikai rendszert, de ezeken kívül is figyelembe kell venni további szempontokat, mint a hőmérséklet vagy a rezgés és rázkódás, ami akár a gravitáció 1000-szerese is lehet.

### 1.2.1. A nyomásérzékelők csoportosítása - abszolút és relatív nyomá szenzor

A membrán hajlása a két oldalon meg. A Wheatstone-hídat fennálló nyomás különbségétől függ. Ennek megfelelően a nyomá szenzorokat három fő típusra bonthatjuk fel: abszolút nyomás-, referencianyomás-, és differencia-nyomá szenzorok.



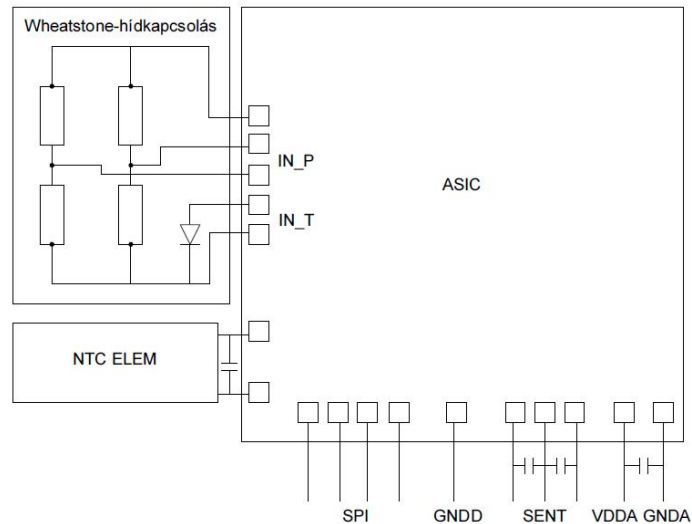
1.6. ábra. A nyomá szenzor mérési elvei

Az abszolút és relatív nyomásérzékelő közötti fő különbség a mért nyomásérték viszonyításában van. Amíg abszolút-nyomásmérésnél a referencianyomás a vákuum, addig a relatív nyomásmérésnél az a légköri nyomás, ideálisan  $10^5$  Pa, vagyis 1 bar. A differenciális szenzor esetében a membrán két oldalán lévő nyomásértékek egymáshoz képesti eltérését mérjük (1.6. ábra).

### 1.3. ASIC

A program specifikus integrált áramkör feladata a szenzor nyomás specifikus kiértékelése (1.7. ábra). Ehhez az elektromos mérőegységhez egy rezisztív Wheatstone-hídat tartalmaz. Továbbá a hőmérsékletméréshez elengedhetetlen az ASIC áramkörbe épített hőmérsékletmérő dióda, valamint a külsőleg hozzá kapcsolt opcionális használatú NTC rezisztív hőelem.

Itt található a szenzor kiértékelő egysége, amely a Wheatstone-hídról érkező különböző feszültség értékeket kompenzálja, majd átkonvertálja a digitális jelekké. A hőmérsékletfüggés minimalizálásához figyelembe veszi a termikus dióda jelét is, valamint opcionálisan felhasználja hőmérséklet adatot az NTC ellenállásból.



1.7. ábra. A szenzorelemek elhelyezkedése az integrált áramkörön belül

### 1.3.1. A hőmérsékletmérés

Ahogy már korábban is említésre került, két lehetőség is fennáll a hőmérséklet mérésénél. Az egyik közvetlenül a szenzorelemen elhelyezkedő dióda, a másik egy külső NTC (Negative Temperature Coefficient) ellenállás. Az NTC a negatív hőfoktényezőjű hőmérséklet-ellenállást, vagyis a hőmérséklet csökkenésével nő az elem ellenállása. A dióda hőmérsékletmérő tulajdonsága a p-n átmenet hőmérsékletfüggésére vezethető vissza, amely a feszültségváltozással fejezhető ki.

A dióda adatait minden esetben felhasználja a kiegyenlítő algoritmus, mivel ez közvetlenül a mérés pontjában szolgáltat adatokat. A külső NTC ellenállás használata akkor kerül implementálásra, ha egy másik közeg hőmérsékletadataira is szüksége van a rendszernek.

### 1.4. A kommunikáció

Ebben az alfejezetben két kommunikációs interfészt fogok bemutatni: az SPI-t (Serial Peripheral Interface), és a kevésbé ismert SENT (Single Edge Nibble Transmission) protokollt. Ezekon kívül még több soros kommunikációs protokoll is megtalálható a szenzorok kommunikációjában, mint a PSI (Peripheral Sensor Interface) valamint a CAN (Controller Area Network), LIN (Local Interconnect Network) esetleg a FlexRay.

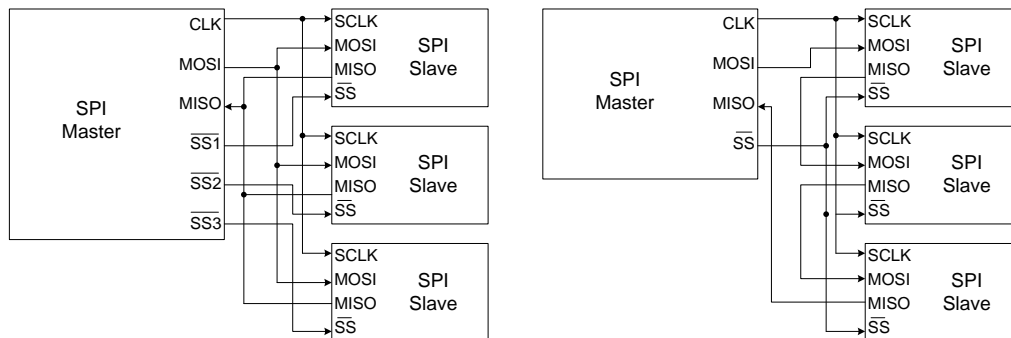
Ezek a járművekben a központi vezérlő egység és egyéb egységek közötti információtovábbításra szolgálnak, és átlagosan 3-5 egymástól független kommunikációs hálózat is működhet egy autóban.

### 1.4.1. Az SPI protokoll leírása

Az SPI a Motorola által fejlesztett kommunikáció, amely kis hatótávolságú (max. 0,2 méter), nagy sebességű Master-Slave kapcsolat, normális működési tartománya  $-0,3V \dots 3,3V$  között van.

Legfontosabb jelei a MOSI (Master Out Slave In), MISO (Master In Slave Out), SCK (Serial Clock): minden órajelre egy bitet küldünk illetve fogadunk a buszon valamint a SS (Slave Select): mindig csak az az eszköz adhat, amelyet a master kiválaszt a SS jellel.

Az átvitel full duplex módban történik, egy masterhez több slave eszköz is tartozhat, amelyek száma korlátlan. A master egység adja az órajelét a buszra, és ő határozza meg, hogy melyik slave-el történik a kommunikáció, ezt pedig a SS jellel határozza meg. Ha kiválasztjuk valamelyik slave-et, akkor az az egység a MOSI jelén keresztül a master által küldött adatokat fogadja, valamint a MISO-n keresztül az adatait az SCK órajel periódusaiban a buszra rakja. Az SPI modulokat különböző topológiák szerint helyezhetjük el, ilyen a Daisy Chain elrendezés, vagy a csillagkapcsolás.



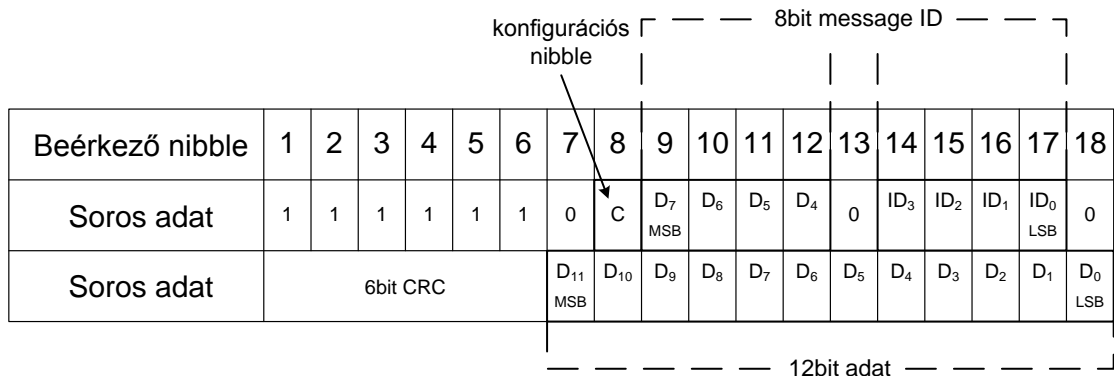
1.8. ábra. SPI csillag- (balra) és Daisy Chain topológia (jobbra) [3]

A 1.8. ábrán láthatjuk, hogy épül fel az SPI busz csillag kapcsolás 3 slave és 1 master egység esetén. Ebben az esetben az SCK, MISO és MOSI közös jelvezetéseken kommunikálnak minden slave egységgel, viszont minden slave egységhez egy külön Slave Select jel tartozik, mellyel az éppen használni kívánt slave egységet választhatjuk ki.

Daisy Chain topológia esetén csupán a SCK és a SS van csak közösítve. A master által kiadott MOSI jel a legelső slave egység MOSI bemenetéhez kapcsolódik. Ennek az egységnek a MISO jele a második slave egység MOSI jelével van összekötve és a többi így tovább. Az utolsó slave egység MISO jele a master egység MISO jelének továbbítja az adatokat [4].

### 1.4.2. A SENT protokoll leírása

A SENT protokoll speciálisan az érzékelő és a vezérlő közötti kommunikációra lett kialakítva. A fogadó modul több soros csatorna adatainak fogadására és továbbküldésére képes, amelyet a szenzor közvetít felé. Ez a modul a SAE J2716 ("SENT - Single Edge Nibble Transmission for Automotive Applications") szabványon alapul, és amely egy nagyfelbontású szenzoradat kommunikációt határoz meg az ECU (Engine Control Unit) felé. A szenzorjel lefutó élvezérelt sorozatos impulzusokként kerül továbbításra.



1.9. ábra. A SENT üzenet felépítése

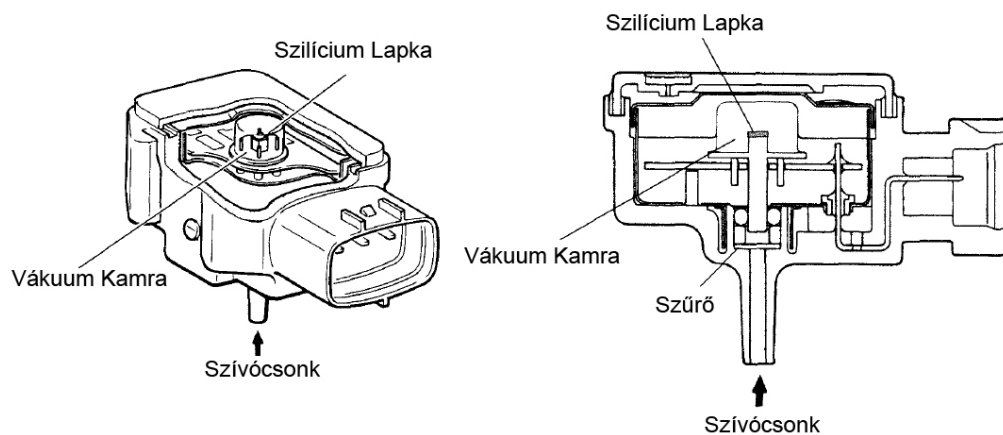
A SENT kommunikáció aszinkron, egyirányú feszültség interfész, amely a szenzortól a vezérlő felé továbbítja az adatokat. Ehhez három jelet használ: egy jelvezeték (alacsony jelszint < 0,5V, magas jelszint > 4,1V), egy tápfeszültség vonalat (5 V) és egy földelő vezeték [5]. Az adatok 4 bites egységekben kerülnek továbbításra (4 bits = 1 nibble) az állandó feszültség amplitúdójú modulált jel két lefutó éle között. Az adatkeret 24bit adatot tartalmaz (3 nibble 2 csatornához, ami nyomásszenzorok esetén a nyomás és a hőmérséklet): 6 nibble a CRC üzenet megadására, 1 nibble a konfiguráció, vagyis a kommunikációs státusz megadására [6].

## 2. fejezet

# A jelterjedés bemutatása

A jelút vizsgálatának feladata abszolút nyomá szenzoron került megvalósításra. Ebben a fejezetben az érzékelő típusának, valamint a szenzor ASIC szerkezetének felépítését mutatom be.

### 2.1. A MAP szenzor



2.1. ábra. A MAP szenzor felépítése

A MAP (Manifold Absolute Pressure) szenzor esetében a mérendő nyomás a szívócsonkon és szűrőn keresztül a szilíciumlapka egyik oldalára fejt ki a hatását. A lapka másik oldalán a referencianyomás található, amely lehet vákuum, vagy egy előre beállított nyomásérték (2.1. ábra). A mérendő nyomás megváltozásával a szilíciumlapka deformálódik, és ez a már 1. fejezetben ismertetett módon mérhető a Wheatstone-híd, és a jelfeldolgozó modul segítségével.

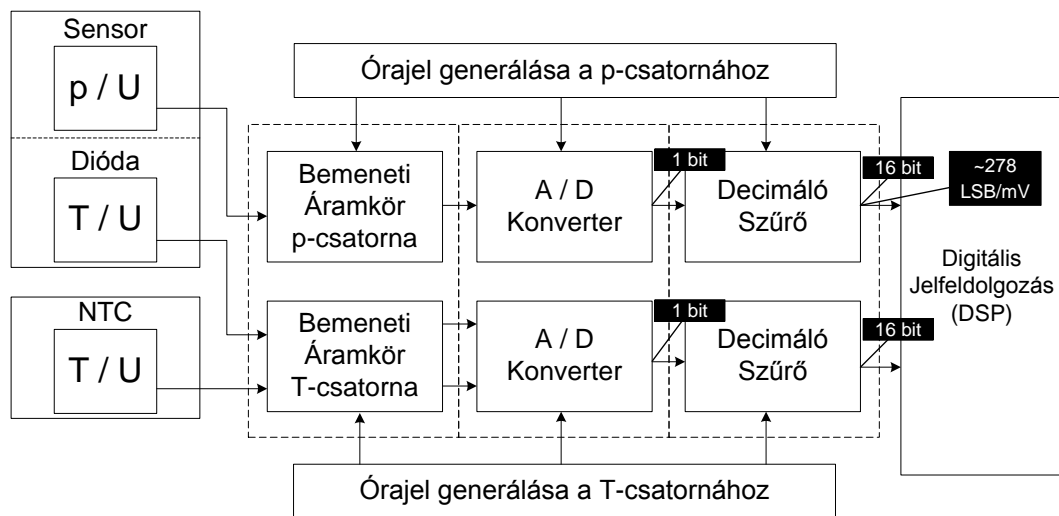


A mérés során felhasznált szenzortípus néhány jellemző paramétere:

- Tápellátás  
3.3 V lineáris szabályozó külső tároló kapacitással. A digitális tápellátást egy belső lineáris szabályozó biztosítja, amely 1.8 V -on üzemel.
- Bekapcsolási folyamat  
A bekapcsolási folyamat a feszültségszabályozók szekvenciális bekapcsolását, az oszcillátor indítását, az adatok OTP memóriából RAM-ba olvasását, a bondok hibadetektálását, a jelút ellenőrzését, valamint a jelillesztést az analóg és a digitális modulok között foglalja magába.
- Hőmérséklet- és nyomásmérés  
A korábban ismertetett módon. A nyomásérték mintavételi frekvenciája 625 kHz. A szenzor működési tartománya  $-40\text{ }^{\circ}\text{C} \dots 125\text{ }^{\circ}\text{C}$ .
- Jelút  
Digitális úton történik a jelfeldolgozás, az analóg feszültségértéket egy A/D átalakító segítségével alakítjuk digitális jelekké. A jelkésleltetési idő tipikusan 1ms. A DSP (Digital Signal Processing) modulban RAM és ROM memóriák is megtalálhatóak.
- SPI és SENT interfész  
Az SPI órajele 8 MHz-en működik, a 0.3 V... 3.3 V-os működési feszültségtartományban. Különböző SPI protokollok léteznek, mint például a BOSCH SPI Protocoll, vagy a Conti-mode SPI Protocoll.
- NVM (Non-Volatile Memory)  
Az NVM olyan memóriaegység, amely a tápellátás megszűnése után sem felejt el a beleírt adatokat (mint például az EEPROM, Flash vagy esetleg FRAM). A szenzorhoz programozható 256 bites OTP (One-Time Programmable) memória is tartozik, ekkor SPI-on keresztül biztosít írási és olvasási lehetőséget.

## 2.2. A kiértékelő áramkör bemutatása

A 2.2. ábrán látható a kiértékelő áramkör funkcionális felépítése. A nyomás méréséhez a Wheatstone-híd feszültségértékét kell átszámítani a mért nyomással arányos digitális jelekre, amit a nyomás mellett a hőmérsékleti adatok is befolyásolnak. A mindenkori hőmérséklet mérése egy a Wheatstone-hídon található dióda segítségével történik, amely nyitófeszültsége egy meghatározott diódaáramnál hőmérsékletfüggő. Ahogy már a korábbi fejezetekben is szó volt róla, a hőmérséklet további mérésére is lehetőség van, akár az ASIC-en (vagyis az áramkör belső hőmérsékletének figyelése), akár egy külső NTC ellenállás integrálásával.



2.2. ábra. A kiértékelő áramkör felépítése

### 2.2.1. A bemeneti áramkör feladata

A feladatspecifikus bemeneti áramkör (Input Circuit) feladata a jelek - amelyeket a Wheatstone-hídtól, a hőmérsékletmérő diódától, valamint a külső NTC ellenállástól kap, - fogadása valamint kompenzálása. A híd feszültségmérése arányosan történik a nyomással, és ugyanez igaz az NTC ellenállás hőmérsékletviszonyára is, viszont dióda esetében ez az összefüggés nem áll fent. A dióda mérési eredményeinek linearizációjára is ebben az áramköri modulban kerül sor.

Ezeknek a feszültségértékeknek a kiegyenlítéséhez szükség van a nyomásértékeken kívül hőmérsékletadatokra is, valamint a jelkondicionáláshoz elengedhetetlen az ofszet és erősítési tartalék szabályozása. A bemeneti adatok szinkron módon, az órajelek segítségével idő multiplexelt módon kerülnek fogadásra.

A bemeneti áramkör egy szűrő, amelynek feladata, hogy a nagyfrekvenciás zavarjeleket, - amelyek az A/D konverterek (a továbbiakban ADC) működését torzítják - kiszűrje, vagyis az elektromágneses kompatibilitást növelje. A szenzor működési tartománya  $-40\text{ °C}$  és  $125\text{ °C}$  között van.

### 2.2.2. Az órajel előállítás

Az órajel előállítása külön történik a p-csatornán (nyomás) valamint a T-csatornán (hőmérséklet). Ennek a jel integritásának megőrzése, valamint az órajelcsúszás megakadályozása szempontjából van nagy szerepe. Mindkét órajel generátor a 20 MHz-es master órajelből hozza létre az egymást nem átlapoló órajeleket, amelyek  $20\text{ MHz}/32 = 625\text{ kHz}$ -

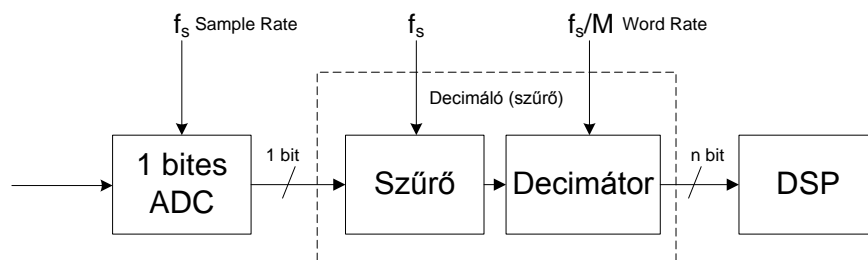
en működnek, vagyis ez lesz a mintavételezési frekvencia. Az órajel generátor biztosítja továbbá a további vezérlőjeleket a szűrő, bemeneti áramkör felé is.

### 2.2.3. Az analóg-digitális átalakító (ADC)

Az ADC feladata, hogy a beérkező analóg jeleket digitális jelekké alakítsa át. A hőmérsékleti csatolna (T-csatolna) esetében nem egy jelről beszélünk, mivel a szenzorelem diódától és az áramkör külső hőmérsékletmérése során is kap adatokat. Hogy ezeket a megfelelő módon dolgozza fel, szükség van idő-multiplexelt rendszerre. A két különböző hőmérsékletjelet különböző órajel-periódusban dolgozza fel.

### 2.2.4. A decimális szűrő feladata

A decimáló szűrő feladata, hogy lassabb jelváltozásnál, ha az ADC többször ugyanazt az értéket alakítja át, vagyis újramintavételezi a jelet, akkor minden M-edik mintát eldobunk, így az utána következő digitális jelfeldolgozó egység DSP nem lesz úgy kiterhelve. Mivel nincs szükségünk ilyen jelsűrűségekre, ezért az enyhe "túl"mintavételezést ritkítja a szűrő, a hasznos  $f_B = (f_s/M)/2$  sávot kiemelve, ahol  $f_B$  a sáv szélesség,  $f_s$  a mintavételi frekvencia (sample rate). A digitális szűrő kimenetén, a redundancia elkerülésére (és a DSP terhelés csökkentésére), az n bites numerikus minták gyakorisága az elegendő  $f_s/M$  gyakoriságra (word rate) ritkítható.

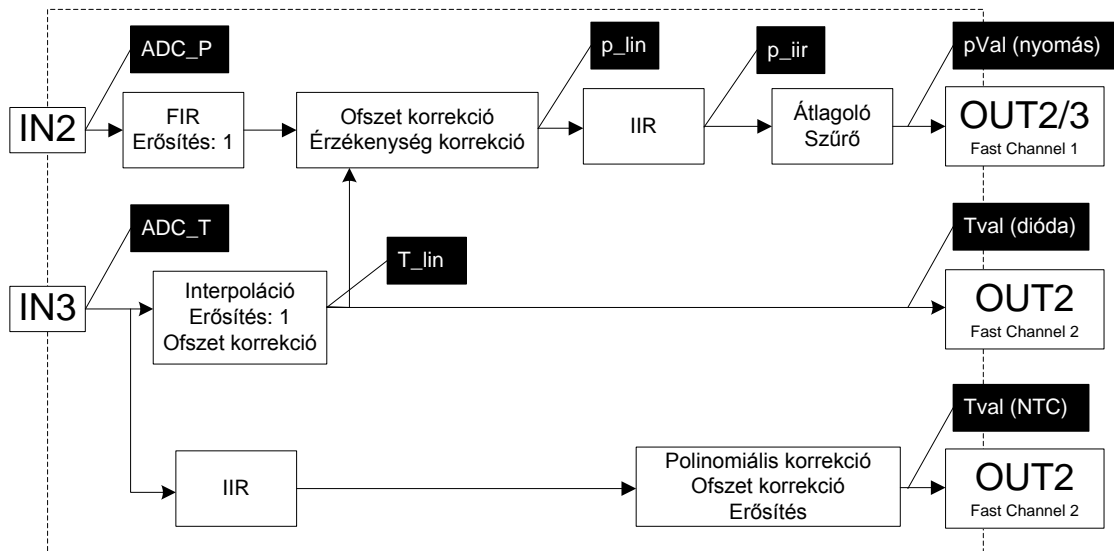


2.3. ábra. A decimáló szűrő felépítése

A digitális szűrés és mintaritkítás praktikusán egyesíthető, ez lesz a decimáló szűrő (2.3. ábra), amely segítségével lehet hatásosan illeszteni eltérő sáv szélesség igényekhez egy adott digitalizálót (front-end) [8].

## 2.3. Digitális jelfeldolgozás (DSP)

A szenzor jelútjában olyan csomópontok, vagyis regiszterek találhatóak, amelyek kiolvasása szoftveres úton megvalósítható. Itt találhatóak a különböző szabályozástechnikai kompenzációk, mint például az ofsztet, az erősítési tartalék, zavarjelek, szűrők, linearizáció.



2.4. ábra. A fontosabb regiszterek elhelyezkedése

A 2.4. ábrán láthatóak a nyomásszenzor ASIC szerkezetén megtalálható fontosabb csomópontok, valamint a jelút. A fontosabb regiszterek az ábrán feketével vannak megjelölve, ezek az ADC\_T; ADC\_P; T\_lin; p\_lin; p\_iir; p\_val; T\_val.

Az ADC\_T valamint ADC\_P a beérkező digitális, 16 bites hőmérsékletértéket jelölik, majd ezen bemeneti értékektől a jelterjedés folyamata balról jobbra történik a kimeneti csatornák felé, a p\_val, T\_val csomópontok irányába (OUT2/3 valamint OUT2).

### 2.3.1. A dióda-hőmérséklet jelútja

A dióda hőmérséklet csatornájának bemeneti jele az IN3 bemenet idő-multiplexelt értéke. A dióda jelútja egy egyszerű lineáris kompenzációt tartalmaz. Mivel a feldolgozott hőmérsékletjelet a nyomásérték hőmérsékleti kompenzációhoz alkalmazzuk, szükség van egy interpolációs szűrőre. Ezzel a 75 Hz körül mozgó bemenet jelből mintasúríttással egy 12 bites 10 kHz-es jelet kapunk.

### 2.3.2. Az NTC ellenállás jelútja

Az NTC hőmérsékleti csatorna a második órajel szerint multiplexelt értéket használja fel, ami az IN3 bemenetről érkezik. A bemeneti jel 75 Hz-es, 16 bites. A jelkompenzálás-hoz az erősítés, ofszet és polinomiális korrekción kívül egy IIR (Infinite Impulse Response, végtelen impulzusválaszú) aluláteresztő szűrő is tartozik.

### 2.3.3. A nyomásérték jelútja

A nyomásértékek jelútja komplex jelfeldolgozást és kompenzálást igényel. A teljes adatcsatorna 10 kHz-es jelekből áll, ahol a szintén 10 kHz-re interpolált hőmérsékletértékek is részt vesznek a szabályozásban. A FIR (Finite Impulse Response, véges impulzusválaszú) aluláteresztő szűrő, ofszet és érzékenység korrekció után a 16 bites jelek az IIR aluláteresztő szűrő és átlagoló szűrő segítségével kerülnek beállításra.

Az átlagoló szűrő egyszerű számtani átlagot számol. Az nyomásértékek bekérése és az átlag kiszámolása után a szűrő alaphelyzetbe kerül, így lehetséges adatvesztés nélkül a 10 kHz-es adatok ciklikus olvasása.

## 2.4. Karakterizálás

Az így kapott mérési eredmények megfelelőségének kiértékelésére szolgál a karakterizáció, amely során a felhasználó/vevő által igényelt specifikációkkal vetik össze az adatokat. Az eltérések mértékének vizsgálatával lehet következtetni a működés megfelelőségére, vagy a hiba okára.

Az ebben a fejezetben a jelfeldolgozó egységben bemutatott szűrő, kompenzáló modulok, szabályozók, valamint paraméterek, amelyek a hőmérséklet és nyomáscsatorna jeleinek beszabályozását végzik, és mind kiolvashatóak a szenzor működése közben. Ezeket az adatoknak a kiértékelésére további szenzor specifikus szoftverek állnak rendelkezésre. Ilyen a MARS program, amelyről bővebben a 4. fejezetben lesz szó.

## 3. fejezet

# A mérési környezet felépítése

A mérőkörnyezet támogatást nyújt az új szenzorok fejlesztéséhez. Ez azt jelenti, hogy a mérőhely kialakításának olyannak kell lennie, hogy az adott új típusra rugalmasan és gyorsan átalakítható legyen. További feltétel az egyszerű kezelhetőség, ami például a szoftveres mérőprogramban az olyan kezelőfelület, amely különböző beállításokat kínál fel fejlesztésre jogosult személynek, és egy tesztelőnek. Továbbá a megbízhatóság is nagy szerepet játszik, vagyis hogy a mérés csak kalibrált, - vagyis megfelelő jogosultságú cég által hitelesített eszközökkel - fusson le.



3.1. ábra. A mérőhely felépítése

A mérés típusa szerint lehet aktív vagy passzív. Aktív mérésnél stressz közben mérik az érzékelő jeleit, passzív mérés esetében pedig csak a stresszhatás után. A stressz lehet több száz, akár ezer órás ciklus, amely során az érzékelők extrém környezeti hatásokat kapnak, mint például a magas-alacsony ( $-40\text{ °C} \dots 120\text{ °C}$ ) hőmérséklet ismétlődő változtatása, vagy a sópermetes stresszhatás, amely a szenzor tokozásának korrózióját szimulálja. A stresszhatások a bond és a ház élettartalmát vizsgálják, például a Weibull analízis segítségével. Egy-egy stresszvizsgálat után a szenzort visszamérve megállapít-

ható, hogy biztosítva van-e a működése 15 év után is.

A mérőhely 230 V-os vagy 400 V-os hálózatról kapja a tápellátását. A mérőműszerek egy szekrényben helyezkednek el amely a mechanikai tartásért és érintésvédelmi biztonságért is felelős. 400 V-os táplálású rendszer esetében egy beépített kapcsoló biztosítja a műszerek 230 V-os tápfeszültségét, valamint ekkor biztonsági kapcsolókkal is fel van szerelve az adott mérőhely.

Egy mérőhely felépítése (3.1. ábra):

- Klímakamra
- Nyomásszabályozó
- AUFNAHME (felvétel): a mérendő szenzorokat tartalmazza
- Tápegység: a szenzor tápellátásáért felelős
- PC
- PXI Rack (PXI tároló): a PC kiterjesztése, ebben található a digitális multiméter, függvény generátor, mátrixkártya a megfelelő szenzor kiválasztásához, RS232-es soros port, stb.
- Unibox: a méréstechnikai fejlesztés eszköze, az adatátvitelben van nagy szerepe

### 3.1. A klímakamra

A kamra biztosítja, hogy különböző hőmérsékletviszonyok között mérhessük az érzékelőt. A kommunikáció a kamra és a PC között soros porton (RS232) vagy TCP/IP kapcsolattal valósítható meg, de lehetőség van szabályzására a kamra saját vezérlőegységével is.

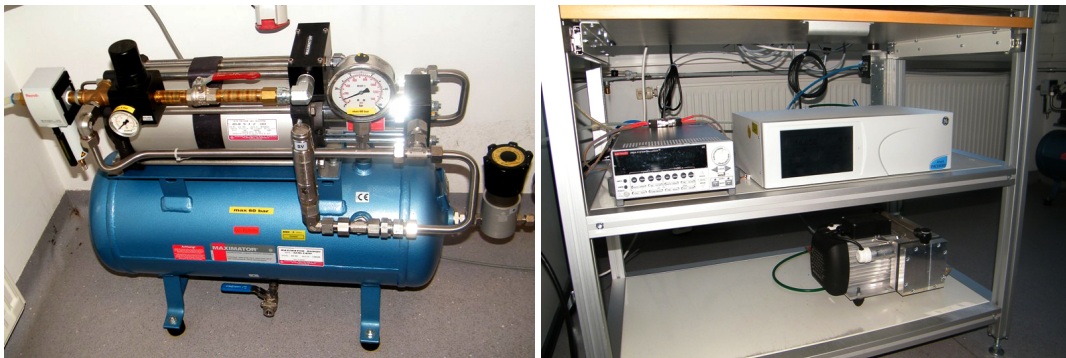


3.2. ábra. A klímakamra

A 3.2.ábrán a bal oldali képen látható a klímakamra nyitott ajtóval, benne - ahogy a jobb oldali kép mutatja - a SUBD50-es csatlakozókkal és nyomáscsővel. A 10 darab 50 pólusú csatlakozóval kialakítástól függően maximum 100 szenzor mérhető.

### 3.2. A nyomásszabályozás

A nyomásértéket a vákuumpumpa és szabályozó segítségével állítjuk be az alábbi módon: az előzetesen nagy nyomásértéket a vákuumpumpa és a Maximátor segítségével állítjuk be, a szabályzó tehát a 7 bar-ból; illetve vákuumból állítja elő a kívánt nyomást. A rendszerben megtalálható egy buffertartály is, amelynek feladata, hogy a szabályozó ellátása egyenletes legyen. Például más mérőhelyet rákapcsolva a szabályozónak legyen elég ideje és kapacitása a kívánt nyomásérték megtartásához.



3.3. ábra. A nyomásszabályozó rendszer felépítése

A 3.3.ábra jobb oldalán a központi buffer nyomásszabályozó látható. Ez biztosítja a nyomás folyamatos és egyenletes megtartását. A jobb oldali képen látható a nyomásszabályozó és alatta a vákuumpumpa, amely általában 100-200mbar nyomásértékre szabályoz, a szabályozó keveri ki a pontos nyomásértéket ebből és a légköri nyomásból. A bal oldali képen látható a Maximátort amely a buffertartály. Ilyen nyomásértékek mellett a sűrített levegőt műanyag csővezetéken keresztül vezetik a szenzorokhoz. A cső a kamrában elszenvedett káros hatások miatt szintén korrodálódik, öregedik, ezért ennek megfelelőségét minden mérés megkezdése előtt automatikusan ellenőrzi a program.

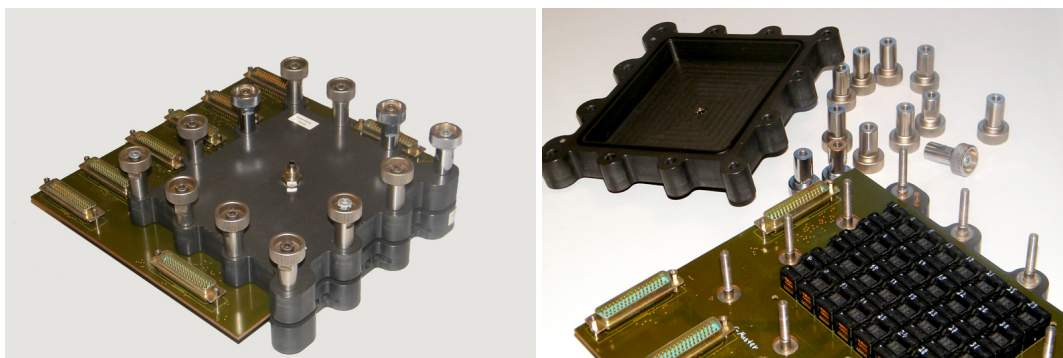
### 3.3. Az Aufnahme

Az Aufnahme feladata, hogy a mérendő szenzorokat elektronikusan és mechanikusan rögzítse és elektronikusan kontaktálja., vagyis a szenzor illesztésében van jelentős feladata. Az elektronikus kapcsolat SUBD50 kábellel (50 pólusú csatlakozó) valósul meg.



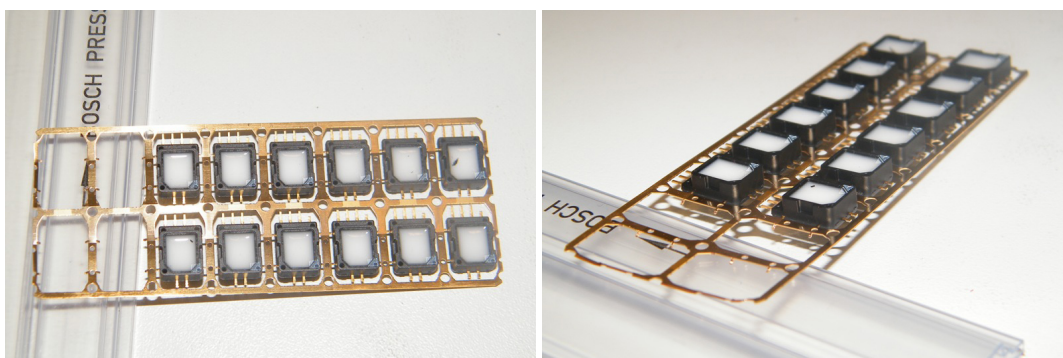
Aufnahme fajtái:

- Shuttle (3.4.ábra): a szenzor méréstechnikai kibővítése, a bondok ki vannak vezetve.



3.4. ábra. A shuttle és ház kialakítása

- Sockel(3.5.ábra): a szenzorok egy úgynevezett Leadframe-be (vezetőkeret) vannak foglalva, több szenzor található egyszerre ebben a keretben.



3.5. ábra. A sockel-be foglalt szenzorok

- Tűk segítségével a szenzor nyitott állapotában előfordulhatnak olyan esetek, főleg relatív szenzortípus esetén, hogy a két előbb bemutatott Aufnahme-típussal nem valósítható meg a mérés. Ilyenkor a szenzor kivezetéseihez tűk érintkeznek, ez valószínűsíti meg a kontaktust.

Az első két esetben szenzorspecifikus foglalatról beszélünk, ezért egy új típusú szenzor megjelenésénél új elrendezésű Aufnahme-t kell gyártani, ami idő és költségigényes folyamat. Ahogy a 3.4.ábrán is látható, az Aufnahme-t egy vastag fém ház veszi körül, ami nagyon kis térfogatot hagy a nyomás szabályozására. Ez azért előnyös, mert a kis szabályozandó térfogattal pontosabb mérési feltételeket lehet megteremteni.

### **3.4. A mérőelektronika**

A beépített mérőelektronika a szenzor kipróbálásához az extrém környezet megteremtéséért: a tápfeszültség, nyomás, hőmérséklet, elektromos jelek szimulálásáért és visszaméréséért felelős.

#### **3.4.1. Unibox**

Az Unibox felelős a mérőkártyák és a mérendő eszközök (jelen esetben a szenzorok) összekapcsolásáért, valamint a kisebb feszültségek (5V, 10V, 14V, 40V) előállításáért. Ezzel az előállított feszültséggel látja el az Aufnahme-n az áramköri elemeket, de a szenzor tápellátásáért nem ez az eszköz a felelős hanem egy kalibrált Agilent tápegység. Itt találhatóak a szenzorok közötti átkapcsolást megvalósító relékártyák (3.6.ábra), amik a jelutatókat kapcsolják. Ezek a CAN buszon keresztül kommunikálnak.

A SUBD50-es csatlakozóval összekapcsolt szenzorok és relékártyák minden egyes szenzor külön vezérelhetőségét megvalósítják meg, vagyis pontosan meg lehet határozni melyik érzékelő vagy érzékelők, illetve azok áramköri elemei kapjanak tápfeszültséget, és ezen kívül pontosan milyen jelek kerüljenek továbbításra mindkét irányban. A mérés szekvenciális, vagyis egyszerre csak egy szenzort jelei mérhetőek.

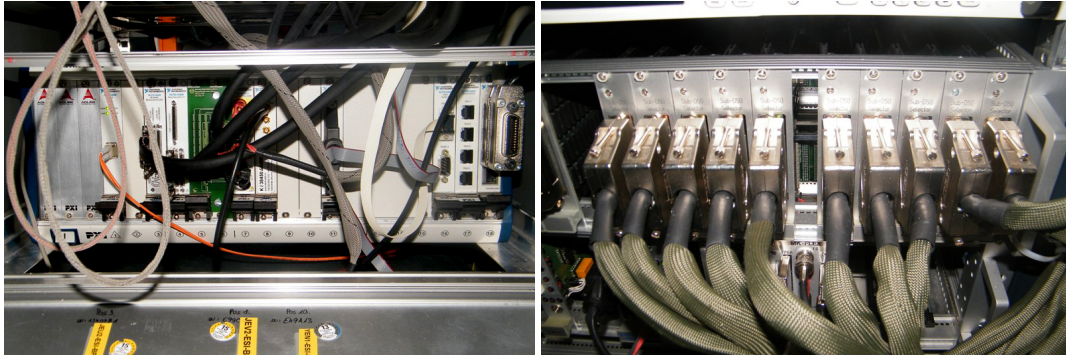
#### **3.4.2. PXI Rack**

A Rack-ben található kártyák feladata az adatok továbbítása, a mérőjelek generálása és visszamérése, a vezérlő jelek kiküldése. Ide tartozik a digitális multiméter, függvény generátor, mátrix-, valamint FPGA kártya (3.6.ábra). A PXI Rack-ben lévő kártyák egy sínparba illeszkednek, amelyben a kommunikációjuk megvalósított, így nem szükséges külön GPIB (General Purpose Interface Bus) kábel az információcseréhez ellentétben külön eszközök használatánál. Ezek a mérő-, stimuláló- és csatoló kártyák így gyorsan cserélhetővé válnak.

#### **3.4.3. A számítógép feladata**

Maga a mérés folyamata, vezérlése szoftveres támogatással, számítógéppel folyik. A PC bemeneti jelei közé tartoznak a soros és párhuzamos portok bemenetei, valamint a PXI Rack-ben található mérőkártyák, amelyek optikai kábelen keresztül kommunikálnak. A vezérlő és monitorozó jelek a már korábban leírt kommunikációs protokollok (CAN, SPI, PSI, stb.) szerint érkeznek a számítógépbe, majd onnan vissza a kártyák felé.

Ebben a mérési fázisban már a SUB50-es kártyára kivezetett lábak mérése és vezérlése szoftveres úton is kezelhető. A szenzor kiválasztása a programban történik, és egy előre



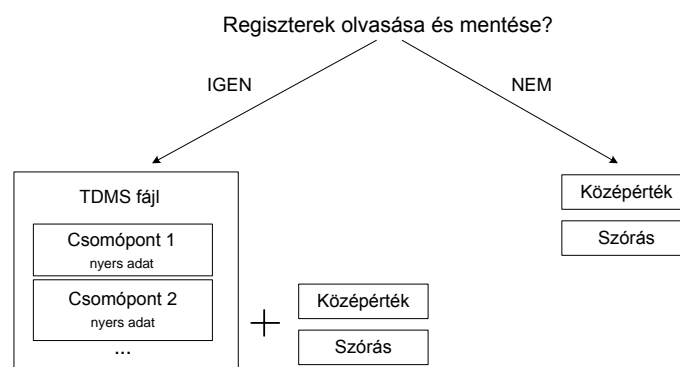
3.6. ábra. A PXI rack felépítése és a mátrixkártyák

definiált profil szerint fut le a az érzékelők szekvenciális mérése is. Erről részletesebben a 4. fejezetben lesz szó.

## 4. fejezet

# Szoftver

A szenzor ASIC jelútjának karakterizálásához a digitális adatok mérése, tárolása és kiértékelése szükséges. Eddig ez a folyamat csupán a mért két kimeneti paraméter - a nyomás és a hőmérséklet - középértékére és szórására terjedt ki. A cél az volt, hogy a különböző regiszterek értékeit is ki lehessen olvasni, majd ezek tárolása is biztosítva legyen, jelen esetben egy TDMS formátumú fájlban (4.2. ábra). Ezzel a merőben új módszerrel nem csak a hiba fellépésének tényét lehet megállapítani, hanem az előfordulás pontos helyét a jelútban, a pontos időpontot és környezeti feltételeket. Ez az SPI protokoll szerinti kiolvasás pontosabb kiértékelést tesz lehetővé a mérési folyamat során.



4.1. ábra. Az feladat során megvalósított újítás

### 4.1. A szoftverfejlesztési feladat bemutatása

A mérési szoftver eddigi működése során csupán a ki és bemeneti regiszterek olvasása volt alkalmas. A feladat egy olyan funkció implementálása, amely az ASIC szerkezet jelútjában található összes regiszter értékének kiolvasására képes. A program további működésében, és a mérési összeállításban változtatás nem történt. A szoftver működése nem szenzor specifikus, így a később megjelenő szenzortípusok mérésére is használható.

A szenzortípusok közötti váltáskor csupán a memóriacímek kiosztását kell változtatni.

A programnak képesnek kell lennie a különböző paraméterek, mint például az áram, hőmérséklet, nyomás, stb. szabályozására és visszamérésére. A szenzor jeleit fogadnia kell és eltárolnia azokat a későbbi kiértékeléshez. Mivel ez a funkció egy meglévő, megfelelően működő programba kerül integrálásba, ezért a program további működését nem szabad befolyásolnia. A hardveres mérési összeállításában változtatás nem történt a feladat implementálásakor.

A programfutás szekvenciális, vagyis egyszerre csak egy szenzort mér. A csomópontok kiolvasásának sorrendje a programban előre meghatározott sorrendben történik, a kezelői felületen ennek megváltoztatására nincs lehetőség. Az eredeti specifikációban ez a sorrend a bemeneti regiszterektől a kimeneti regiszterek felé történik. Mindegyik csomópontnak profilvezérelten kiolvashatónak kell lennie. Az adatokat MARS-ban, a nyers adatokat MATLAB-ban lehessen kiértékelni.

A mérés indítása és vezérlése profilvezérelt módon történik. A profilok olyan parancsok, amelyek megadják milyen feladatokat végezzen el a szoftver a mérés során. Ilyen parancsok a szenzor valamint mérőhely inicializálása, a hőmérséklet és nyomásértékek megadása valamint a kiolvasandó adatok megadása. Ezeket a parancsok egy külső .txt alapú fájlban kerülnek eltárolásra. Ezzel a mérés indítása sokkal átláthatóbb, valamint egyszerűbb.

## 4.2. A parancs leírása

A profilban szereplő parancsok az programozható paraméterekkel kell rendelkeznie:

<Parancs neve> <Param 1> <Param 2> <Param 3> <Param 4> <Param 5>

<Param 1>: Hányszor olvasson ki egy regisztert

<Param 2>: Ciklusszám, vagyis hogy a <Param 1> ciklust hányszor futassa le az adott csomópontra

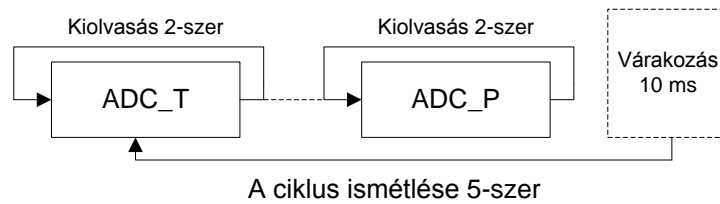
<Param 3>:

0: A csomópontok nyers adatait nem kell elmenteni

1: A csomópontok nyers adatainak elmentése TDMS formátumba időbélyeggel

<Param 4>: Várakozási idő

<Param 5>: A kiolvasandó regiszterek neve



4.2. ábra. A feladatom során megvalósított kiolvasási szekvencia

Példa:

```
<READ_NODES> 2 5 1 10 ADC_T ADC_P
```

- 1.lépés: ADC\_T(adatok kiolvasása 2x), ADC\_P (adatok kiolvasása 2x), 10 ms várakozás,
- 2.lépés: ADC\_T(adatok kiolvasása 2x), ADC\_P (adatok kiolvasása 2x), 10 ms várakozás,
- 3.lépés: ADC\_T(adatok kiolvasása 2x), ADC\_P (adatok kiolvasása 2x), 10 ms várakozás,
- 4.lépés: ADC\_T(adatok kiolvasása 2x), ADC\_P (adatok kiolvasása 2x), 10 ms várakozás,
- 5.lépés: ADC\_T(adatok kiolvasása 2x), ADC\_P (adatok kiolvasása 2x).

A mérések Windows7 alatt futottak, amely nem valós idejű operációs rendszer, ezért a időbélyegek időbélyeg csak informatív célokat szolgálnak.

A továbbiakban egy olyan mérési eredmény kiértékelését fogom bemutatni, ahol két szenzor p\_val valamint T\_val csomópontjait mértem az alábbi profil segítségével:

```
TEMP 23
PRESS 1000
READ_NODES> 5 3 1 500 pVal Tval
```

```
TEMP 23
PRESS 2000
READ_NODES> 5 3 1 500 pVal Tval
```

```
TEMP -40
PRESS 1000
READ_NODES> 5 3 1 500 pVal Tval
```

```
TEMP -40
PRESS 2000
READ_NODES> 5 3 1 500 pVal Tval
```

```
TEMP 125  
PRESS 1000  
READ_NODES> 5 3 1 500 pVal Tval
```

```
TEMP 23  
PRESS 1000  
READ_NODES> 5 3 1 500 pVal Tval
```

#### 4.2.1. A fejlesztési környezet - CVI

LabWindows/CVI (C for Virtual Instrumentation) egy ANSI C fejlesztési környezeten alapuló mérési- és tesztkörnyezet, amelyet a National Instruments fejlesztett ki. A Microsoft Windows platformon működő programban a felhasználói felület grafikus úton hozható létre. A felület minden eleméhez olyan függvények (Call-back) rendelhetők, amelyeket a program rögtön végrehajt az adott elem aktiválásakor (például egy gomb megnyomásakor). A LabWindows/CVI a hagyományos fejlesztési irányok mellett az eseményvezérelt alkalmazások fejlesztését is támogatja. A programfejlesztést automatikus kódgenerálási és hibakeresési szolgáltatások segítik. A CVI alkalmas önállóan futtatható programok, DLL-ek és ActiveX-ek létrehozására, és egyben kiváló eszköz műszermeghajtó szoftverek fejlesztésére is.

A CVI néhány jellemzője:

- Grafikus felhasználói felület (GUI) létrehozása  
Egyszerűen, grafikusan kezelhető GUI alakítható ki, Call-back függvényekkel definiálható gombokkal, csúszkákkal, számlálóval, bemeneti adatok megadásával és kimeneti eredmények megjelenítésével.
- ANSI C fejlesztői környezet
- Mérésadatgyűjtő hardverek kezelése  
Széles spektrumú eszközök és kártyák implementálására van lehetőség a különböző hardverek - GPIB, USB, soros, Ethernet kommunikáció segítségével - való vezérlésére. A programban beépített I/O könyvtárak segítségével az eszközök vezérlése már előre implementálva van, az adatgyűjtés folyamata és az interaktív kapcsolat automatikus kódgenerálással megvalósított.
- Jelanalízis függvények  
A szoftverben a különböző jelanalízishez szükséges algoritmusok és funkciók előre definiálva vannak. Több száz beépített, specifikus funkció áll rendelkezésre, hogy a beérkező nyers adatok kiértékelését kezelni lehessen [10].

### 4.3. Az adatfeldolgozási módszerek

A csomópontok kiolvasásának megvalósítása előtt csupán a ki- és bemeneti regiszterek középértékének és szórásának az olvasására volt képes a programkód. Azzal, hogy az egyes regiszterek szekvenciálisan kiolvashatóvá váltak több ezer adat keletkezett, aminek tárolását és kiértékelését meg kellett oldani.

Adattárolásra a TDMS (Technical Data Management Streaming) struktúra kerül kiválasztásra, mivel ez nagy mennyiségű adat tárolására és gyors lokalizálására. Az adatok kiértékelése a Boschban használt MARS programmal történt, amely a TDMS fájlok kezelésére is képes.

#### 4.3.1. A TDMS struktúra bemutatása

TDMS adatszerkezet generálásakor két specifikus fájl keletkezik. A .tdms kiterjesztésű fájl tartalmazza a jól strukturált adatokat, a .tdms\_index pedig az adatok indexelését tárolja el. A bináris TDMS fájl struktúrája három szakaszra bontható. A gyökérhez (Root) tartoznak a csoportok (Group), a csoportokhoz a csatornák (Channel), a csatornákhöz pedig a tárolt adatok. Mindegyik szekcióhoz megadhatóak egyedi tulajdonságok, mint például a dátum, idő, a létrehozó email címe, neve, stb. A szerkezeti elemek és adatok mennyisége egyik hierarchiai elemben sincs korlátozva. Az TDMS fájlt Microsoft Excel programmal is meg lehet nyitni, egy ingyenes, a National Instruments által elérhetővé tett Add-In installálásával. Az Excel fájl első munkalapján láthatóak [11].

	A	B	C	D	E	F
1	Root Name	Title	Author	Groups	Date	
2	Aufnahmenr.: 10145	Knotenpunktwerte	kvn2bp		2 09-18-2013	
3						
4	Group	Channels	Description			
5	ChipIndex_0		6			
6	ChipIndex_1		6			
7						
8	ChipIndex_0					
9	Channel	Datatype	Unit	Length	Description	
10	pVal_Time	DT_STRING	hour:min:sec	90	Measured Time	
11	pVal_Data	DT_DOUBLE	n.a.	90	Measured Raw Data	
12	Tval_Time	DT_STRING	hour:min:sec	90	Measured Time	
13	Tval_Data	DT_DOUBLE	n.a.	90	Measured Raw Data	
14	Pressure PT100	DT_DOUBLE	mbar	90	PT100 Pressure	
15	Temperature	DT_DOUBLE	Celsius	90	Measured Temperature	
16						
17	ChipIndex_1					
18	Channel	Datatype	Unit	Length	Description	
19	pVal_Time	DT_STRING	hour:min:sec	90	Measured Time	
20	pVal_Data	DT_DOUBLE	n.a.	90	Measured Raw Data	
21	Tval_Time	DT_STRING	hour:min:sec	90	Measured Time	
22	Tval_Data	DT_DOUBLE	n.a.	90	Measured Raw Data	
23	Pressure PT100	DT_DOUBLE	mbar	90	PT100 Pressure	
24	Temperature	DT_DOUBLE	Celsius	90	Measured Temperature	
25						
26						

4.3. ábra. A TDMS struktúra megjelenítése

A TDMS fájlt a CVI környezet egyszerűen le tudja generálni két előre beépített könyv-



	A	B	C	D	E	F	G
1	pVal_Time	pVal_Data	Tval_Time	Tval_Data	pOut_Time	Temperature	
2	12:18:15.883000	1339	12:18:16.482000	806	12:18:17.041000	22,944153	
3	12:18:15.958000	1339	12:18:16.558000	806	12:18:17.107000	22,944153	
4	12:18:16.033000	1339	12:18:16.633000	806	12:18:17.183000	22,944153	
5	12:18:16.108000	1339	12:18:16.707000	806	12:18:17.258000	22,944153	
6	12:18:16.183000	1338	12:18:16.773000	806	12:18:17.333000	22,944153	
7	12:18:29.090000	1338	12:18:29.714000	807	12:18:30.314000	22,930929	
8	12:18:29.164000	1338	12:18:29.798000	807	12:18:30.379000	22,930929	
9	12:18:29.238000	1339	12:18:29.873000	807	12:18:30.456000	22,930929	
10	12:18:29.304000	1338	12:18:29.956000	807	12:18:30.530000	22,930929	
11	12:18:29.381000	1338	12:18:30.031000	807	12:18:30.596000	22,930929	
12	12:18:42.007000	1339	12:18:42.574000	807	12:18:43.149000	22,952406	
13	12:18:42.082000	1339	12:18:42.649000	807	12:18:43.222000	22,952406	
14	12:18:42.157000	1339	12:18:42.722000	807	12:18:43.297000	22,952406	
15	12:18:42.222000	1339	12:18:42.797000	807	12:18:43.372000	22,952406	
16	12:18:42.299000	1339	12:18:42.872000	807	12:18:43.447000	22,952406	
17	13:01:58.074000	2614	13:01:58.919000	800	13:01:59.886000	23,01454	
18	13:01:58.128000	2614	13:01:58.979000	800	13:01:59.936000	23,01454	
19	13:01:58.194000	2614	13:01:59.036000	800	13:02:00.003000	23,01454	
20	13:01:58.245000	2614	13:01:59.103000	800	13:02:00.053000	23,01454	
21	13:01:58.310000	2614	13:01:59.153000	800	13:02:00.119000	23,01454	
22	13:02:18.219000	2615	13:02:19.081000	800	13:02:19.906000	23,022581	
23	13:02:18.280000	2614	13:02:19.138000	800	13:02:19.972000	23,022581	
24	13:02:18.339000	2614	13:02:19.198000	800	13:02:20.030000	23,022581	
25	13:02:18.393000	2613	13:02:19.255000	800	13:02:20.089000	23,022581	
26	13:02:18.448000	2615	13:02:19.314000	800	13:02:20.143000	23,022581	
27	13:02:39.937000	2614	13:02:40.796000	800	13:02:41.779000	22,933405	

4.4. ábra. A TDMS struktúra megjelenítése

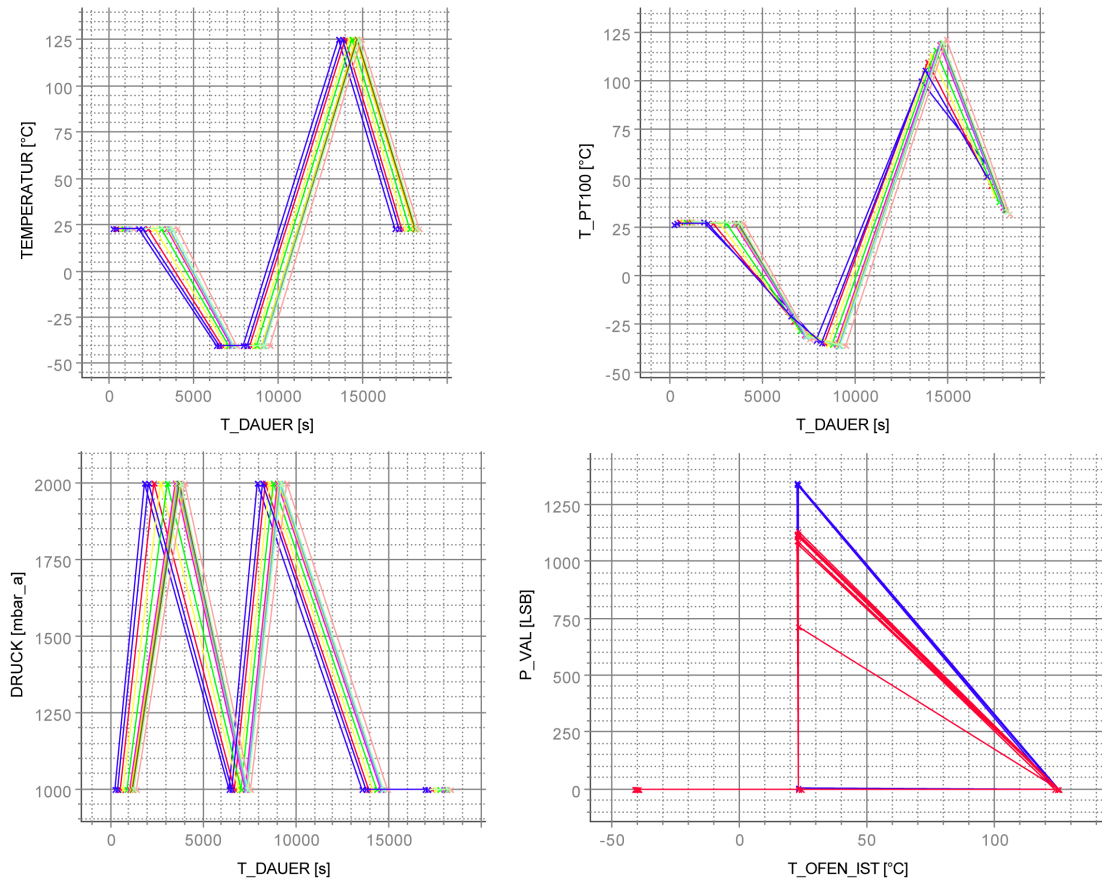
tár segítségével. A 1.2-es fejezetben leírt mérés lefutása után a 4.3. ábrán látható struktúrát kaptam. A két szenzor azonosítója ChipIndex\_0 és ChipIndex\_1, ezek a TDMS fájl csoportjai. Mindegyik csoport hat csatornát tartalmaz, ezek a pVal\_Time, pVal\_Data, Tval\_Time, Tval\_Data, Pressure PT100 valamint a Temperature. Mindegyik csatorna 90 adatot tartalmaz:  $(5 \text{ <csomópont olvasása>})(3 \text{ <ciklus ismétlése>})(6 \text{ <mérés>}) = 90$  (4.4. ábra).

#### 4.3.2. A MARS program bemutatása

A MARS egy olyan egyedülálló, a Robert Bosch mérés technikai fejlesztési részlege által fejlesztett szoftver, amely a különböző mérés technikai feladatok eredményeinek kiértékelésére szolgál. A programban meg lehet adni különböző szűrőket, így a felesleges adatokkal a későbbiekben már nem is kell foglalkozni. A MATLAB környezetnek köszönhetően különböző függvényeket lehet az adatokra illeszteni.

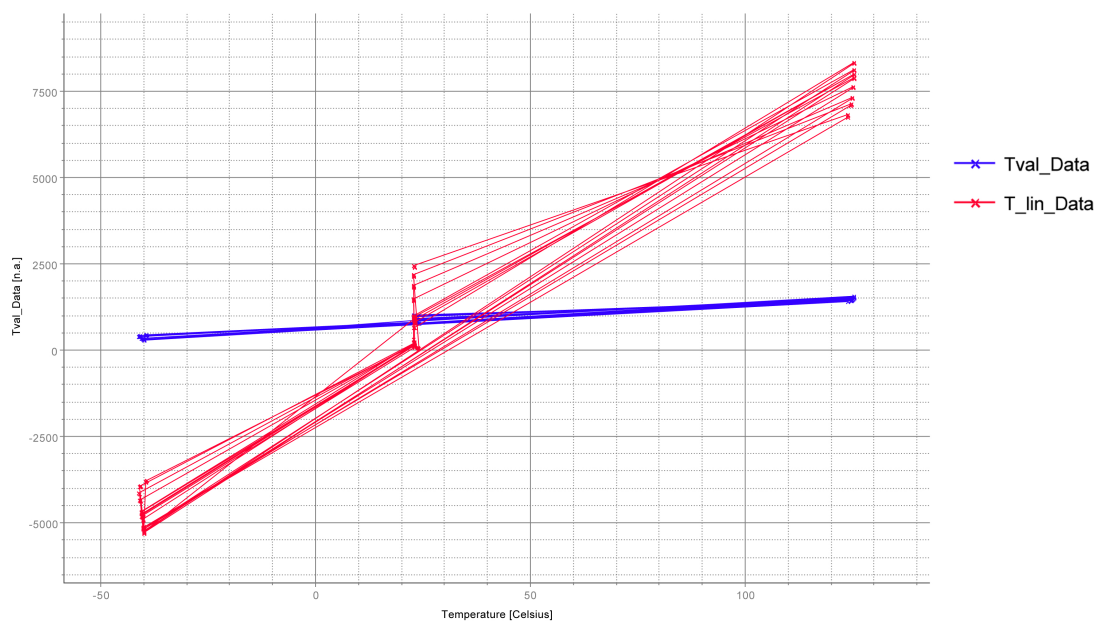
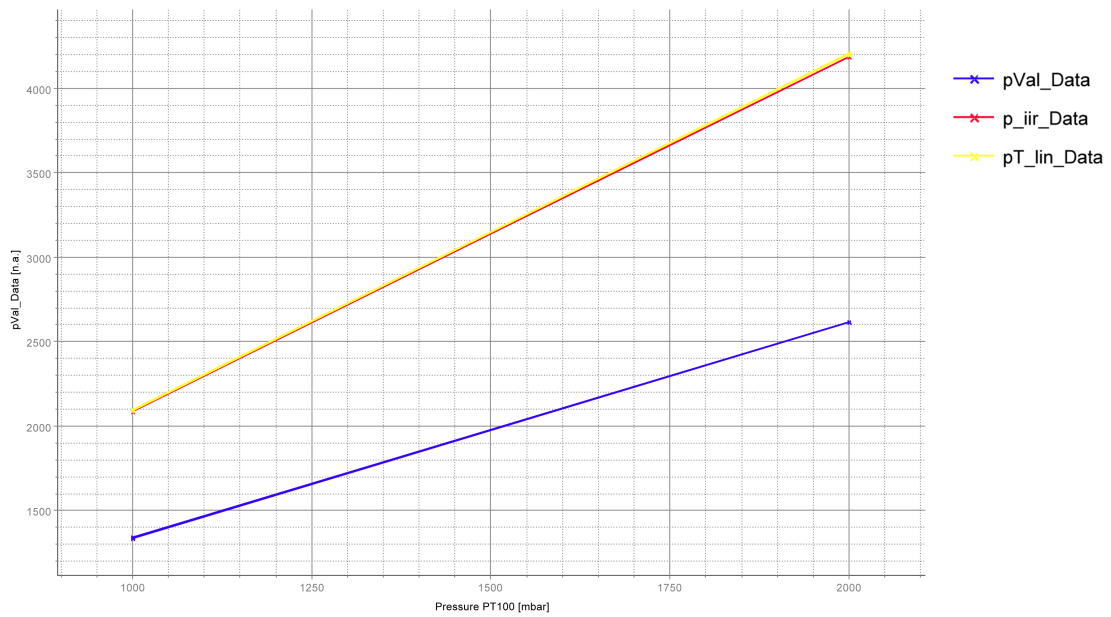
Főbb jellemzői:

- MATLAB környezetben alapuló program
- Optimalizált adatkezelés: az összes használatban lévő adatfájl-típus egyszerű kezelése
- Kiértékelési sablonok a könnyebb kezelhetőség érdekében
- A szoftver javításának, fejlesztésének lehetősége



4.5. ábra. A fontosabb regiszterek elhelyezkedése

A 4.5. ábra jobb felső grafikonja az általam beállított hőmérsékletértékeket jeleníti meg a szenzorok esetében, viszont a tényleges (TEMPERATUR), dióda (P\_T100) által mért jelek a jobb ábrán látszódnak az idő (T\_DAUER) függvényében. A szenzorok különböző nyomásértékeket (DRUCK) mérnek a kamrahőmérséklet (T\_OFEN\_IST) inhomogenitásából adódóan. Az alsó grafikonokon azok a nyomásértékek láthatóak amelyeket az időtől (bal ábra) és a kamrahőmérséklettől (jobb ábra) függően sikerült ábrázolnom.



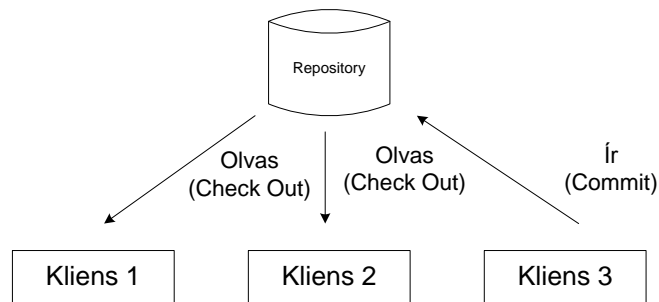
4.6. ábra. A regiszterek hőmérséklet és nyomásértékei

A másik két kiértékelést, amelyek a (4.6) ábrán láthatóak, a TDMS fájl kiértékelése során kaptam. A fenti ábrán látható a p\_Val, p\_iir, p\_lin regiszterek értékei a hőmérséklet függvényében. Az alsó ábrán látható, hogy - szintén a hőmérséklet függvényében - a linearizáció utáni T\_lin értékek lineáris, de nagy szórású egyeneseket alkotnak, de a kimeneti értékek már egy egyenesen helyezkednek el.

## 4.4. A programkód dokumentálása

Egy fejlesztési feladat megoldása során több fejlesztő is dolgozhat ugyanazon a programkódon. Előfordul, hogy a két változtatás nem konzisztens egymással, a programkód frissítése során a program nem fordul le. Ennek elkerülésében segítenek a verziókövető rendszerek.

### 4.4.1. Az SVN verziókövető szoftver bemutatása



4.7. ábra. A verziókövető rendszerek alapfogalmai

Az SVN (Subversion) egy verziókövető rendszer, amelynek alapja egy adott project összes változásának nyilvántartása. Egy verziómenedzsment rendszer nyilvántart minden egyes file-on végrehajtott változtatást, valamint a könyvtárstruktúrát érintő minden változást. A felhasználónak lehetősége van megtekinteni a project vagy egy file állapotát egy adott pillanatban, vagy megtudhatja, hogy ki, mit és mikor változtatott az adott projecten.

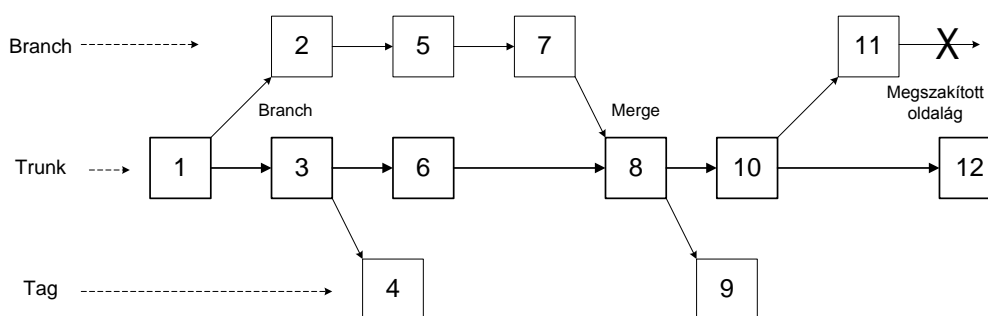
A verzió követés alapfogalmai (4.7. ábra):

- Repository (raktár): Központi nyilvántartás az adatoknak vagy projectnek (a master copy).
- Client (kliens): Felhasználó, aki dolgozni kíván a projecten.
- Working copy (használatban lévő másolat): Egy Kliens által a projectből létrehozott munkaváltozat, amit szabadon változtathat.

A verziómenedzsment rendszerekkel kapcsolatban felmerülő első kérdés általában, hogy hogyan támogatja a felhasználók együttműködését. Ilyen stratégia nélkül könnyen előfordulhat, hogy egy file-t vagy projectet egyszerre többen módosítanak, majd felülírják egymás módosításait (a módosítások nem tűnnek el, de nem is kerülnek bele az új verzióba).

Az SVN a Copy-Modify-Merge megközelítést alkalmazza, amely jellemzői:

- Egyszerre több fejlesztő is ki „Check out”-olhatja ugyanazt a fájlt, vagy is készíthet róla egy másolatot.
- Mindenki a saját Working copy-ját használja, amely a Repository (vagy annak egy részének) saját gépen található leképezése. Ezzel a koncepcióval egyszerre több fejlesztő is dolgozhat ugyanazon a kódon.
- A létrejövő új vagy módosított fájlokat „Commit” paranccsal olvasszák a főágban futó programba és így hoznak létre egy új verziót.
- Ha már a fő program lett módosítva, akkor a fejlesztő nem tudja módosítani a Repository tartalmát amíg nem „Merge,-eli (olvasztja össze) az újabb verzióval a kódját. Ezzel az esetleges konfliktusok kiderülnek, viszont ezeknek a megoldásáról a fejlesztőknek kell dönteniük.



4.8. ábra. A Törzs ág, Címkézett ágak, és Elágazások vagy Trunks, Tags és Branches

Egy komplex project verziókövetése során nem csak az egyes file-ok verzióit, hanem az egész project konfigurációját nyomon kell követni (4.8. ábra). Egy project esetében három ehhez kapcsolódó fogalmat kell megjegyezni: ez a Trunks: az ág, amely tartalmazza a project törzsét, tehát a hivatalos fejlesztési irányt. A Branch az elágazások a project fejlesztési irányától kicsit eltérő funkciók kipróbálására, megvalósítására szolgál. A lényege az, hogy ilyenkor a fejlesztés eredeti menete ne bonyolódjon, hanem egy saját leágazást lehessen nyúzni. Az esetek többségében az elágazások visszaintegrálódnak az eredeti projectbe. A Tags, vagyis a címkézett ág felel meg az egyes stabil és működőképes release-eknek. Egy project fejlesztése során időről időre szoktak ilyen release verziókat létrehozni. Ez azért fontos, mert a Trunk ág folyamatos fejlesztésben van, és így általában nem tartalmaz komplett és jól működő verziót. Van úgy, hogy a Trunk ág éppen el sem tud indulni, vagy nem is fordul le [9].

#### 4.4.2. Dokumentáció létrehozása a programkódból - Doxygen

A fejlesztési feladatom végső stádiuma egy dokumentáció létrehozása volt, amely tartalmazza a programkód specifikációját, működési elvét és használatának leírását.

A programkód dokumentálására egyszerű megoldást nyújtott a Doxygen program. Ez az eszköz a programkódból egy komplex kóddokumentáció generálására képes. Ha a kód kommentezésénél speciális, a Doxygen program által elvárt karaktersorozatok közé illesztett magyarázatok szerepelnek, akkor ezeket fel tudja dolgozni és egy dokumentációt hoz létre. Bár a megszokottól némileg eltérő kommentezési stílust kell alkalmazni, de többféle stílus is támogatott.

A közvetlenül a programkódból generált fájl képes eltárolni fájlok, osztályok, függvények, struktúrák, változók adatait és hivatkozásait egymásra. Többek között megadható a név, a fejlesztő neve, a létrehozás dátuma valamint egy rövid leírás. A program képes továbbá vizuálisan, grafikonok segítségével felvázolni a kapcsolatokat a különböző elemek között grafikonok, diagramok segítségével. Ezeket a funkciókat a konverzió előtt beállítva a szoftver automatikus végrehajtja őket. A Doxygen kimenete HTML, MS-Word, PDF formátumú is lehet [12].

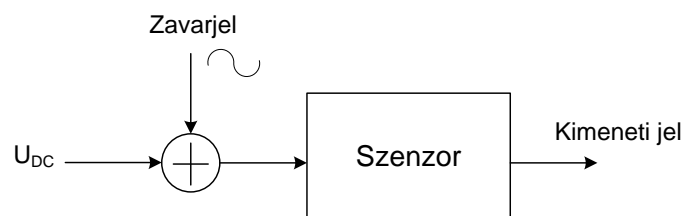
## 5. fejezet

# Összefoglalás

A TDK dolgozatomban egy olyan mérési elv és mérőrendszer felépítésével foglalkoztam, amely képes a nyomásmérő szenzorok ASIC struktúrájában található regiszterek szekvenciális kiolvasására, valamint ezeknek az adatoknak az eltárolására és kiértékelésére. Megismertem továbbá a különböző kommunikációs protokollokat, önállóan megterveztem és egy már meglévő, komplex mérőprogramba implementáltam egy adatfogadást elvégző szoftvermodult.

Munkám során betekintést nyerhettem az autóiipari nyomásszenzorok, mérési elvek és mérőhelyek működésébe, elsajátítottam egy, a mérnöki gyakorlatban is alkalmazott komplex mérési szoftver használatát az alkalmazott mérési környezettel, valamint az adatmentés, adatkiértékelés és dokumentálás folyamatát. Alkalmaztam a párhuzamos programfejlesztés és az új, módosított szoftver verifikálásának menetét.

### 5.1. Jövőbeli tervek



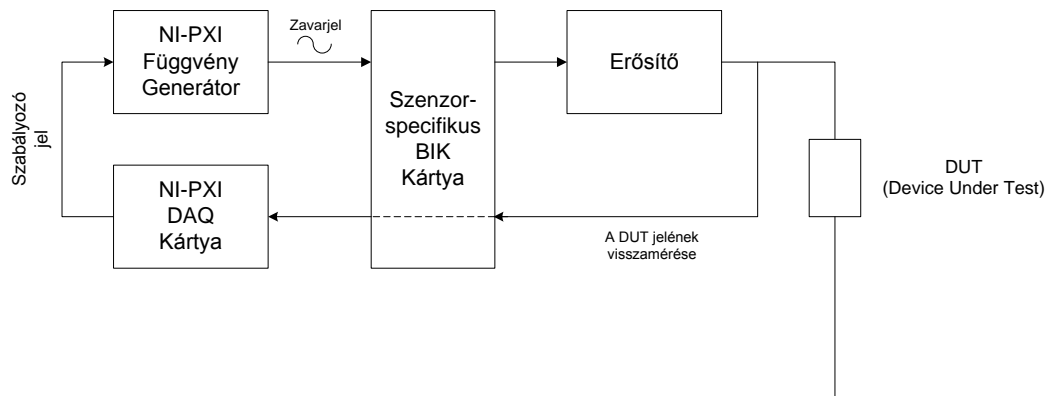
5.1. ábra. A jövőben megvalósítandó feladat elve

A következő feladatomban az, hogy egy olyan mérést tegyek lehetővé, ahol a tápfeszültségre exponálódó zajok szimulálása és mérése lehetővé válik. Ahogy a 5.1. ábrán is látha-

tó, az érzékelő bemenetére nem csak a tápfeszültség értéke, hanem különböző zavarjelek is kerülhetnek, amelyek megjelennek a szenzor kimenetén.

A project célja, hogy az eddig implementált mérési funkciók változatlanok maradjanak, de a fent említett funkció is elérhető legyen a mérések során. Ehhez a mérési szoftverben kell azt létrehozni, hogy a szinuszos zavarjelek is megjelenjenek a szenzor bemenetén, vagyis egy stimulusként lehessen vezérelni ezeket a jeleket. A szenzorok mérése szekvenciális sorrendben kell hogy történjen, a zavarfeszültség tipikus értéke  $100\text{ mV}_{pp}$ , frekvenciája  $1\text{ kHz} - 1\text{ MHz}$  értékek között kell hogy szabályozható legyen.

A mérési elrendezés a 5.2. ábrán látható. A szabályozási rész egyik fontos pontja a függvénygenerátor kimenetét megjelenő jel szabályozása az eszköz bemeneti jelének visszamérése és alapján.



5.2. ábra. A tervezett mérési elrendezés

Az ebben a dolgozatban bemutatott feladat is integrálásra kerül ebbe a más jellegű rendszerbe. A nyers adatok a bináris TDMS formátumban kerülnek eltárolásra. Ezzel az új, eddig ismeretlen mérési megoldással lehetővé válik a szenzort érő zavarjelek szimulálása és az érzékelő viselkedésének precíz vizsgálata.



# Ábrák jegyzéke

1.1.	Egy általános érzékelő működési alapelve . . . . .	5
1.2.	A mért nyomásértékek eltérése különböző hőmérsékleteken . . . . .	6
1.3.	A teljes hiba felosztása . . . . .	7
1.4.	Az abszolút nyomá szenzor felépítése . . . . .	8
1.5.	A Wheatstone-híd szerkezete a membránon . . . . .	9
1.6.	A nyomá szenzor mérési elvei . . . . .	11
1.7.	A szenzorelemek elhelyezkedése az integrált áramkörön belül . . . . .	12
1.8.	SPI csillag- (balra) és Daisy Chain topológia (jobbra) [3] . . . . .	13
1.9.	A SENT üzenet felépítése . . . . .	14
2.1.	A MAP szenzor felépítése . . . . .	15
2.2.	A kiértékelő áramkör felépítése . . . . .	17
2.3.	A decimáló szűrő felépítése . . . . .	18
2.4.	A fontosabb regiszterek elhelyezkedése . . . . .	19
3.1.	A mérőhely felépítése . . . . .	21
3.2.	A klímakamra . . . . .	22
3.3.	A nyomá szabályozó rendszer felépítése . . . . .	23
3.4.	A shuttle és ház kialakítása . . . . .	24
3.5.	A socket-be foglalt szenzorok . . . . .	24
3.6.	A PXI rack felépítése és a mátrixkártyák . . . . .	26
4.1.	Az feladat során megvalósított újítás . . . . .	27
4.2.	A feladatom során megvalósított kiolvasási szekvencia . . . . .	29
4.3.	A TDMS struktúra megjelenítése . . . . .	31
4.4.	A TDMS struktúra megjelenítése . . . . .	32
4.5.	A fontosabb regiszterek elhelyezkedése . . . . .	33
4.6.	A regiszterek hőmérséklet és nyomásértékei . . . . .	34
4.7.	A verziókövető rendszerek alapfogalmai . . . . .	35
4.8.	A Törzs ág, Címkézett ágak, és Elágazások vagy Trunks, Tags és Branches	36
5.1.	A jövőben megvalósítandó feladat elve . . . . .	38
5.2.	A tervezett mérési elrendezés . . . . .	39

# Irodalomjegyzék

- [1] Bosch Sárga füzetek sorozat, *Szenzorok a gépjárművekben*, ISBN 963900583, Maróti könyvkiadó, Budapest, 2008.
- [2] [http://www.ob121.com/hard\\_sensor.html/piezorezisztiv\\_erzekelo](http://www.ob121.com/hard_sensor.html/piezorezisztiv_erzekelo), 2013. szeptember 28., 19:14.
- [3] [http://de.wikipedia.org/wiki/Serial\\_Peripheral\\_Interface](http://de.wikipedia.org/wiki/Serial_Peripheral_Interface), 2013. szeptember 30., 17:32.
- [4] Kupó Csaba, *Általános illesztő egység tervezése autóiipari szenzorokhoz*, Szakdolgozat, 2013.
- [5] [http://en.wikipedia.org/wiki/SENT\\_\(protocol\)](http://en.wikipedia.org/wiki/SENT_(protocol)), 2013. szeptember 28., 20:10.
- [6] <http://www.design-reuse.com/articles/32208/challenges-in-validation-of-communication-protocols.html>, 2013. szeptember 28., 20:22.
- [7] <http://sine.ni.com/np/app/main/p/docid/nav-110/lang/hu/fmid/6324/>, 2013. október 12., 9:38.
- [8] Dr. Pápay Zsolt, <http://http://www.hit.bme.hu/papay/edu/Acrobat/conv2A.pdf>, 2013. október 15., 12:49.
- [9] Scherer Balázs, *Rendszertervezés (Embedded System Design)*, BME MIT: MSc beágyazott információs rendszerek szakirány, Óravázlat, 2010.
- [10] <http://sine.ni.com/np/app/main/p/docid/nav-110/lang/hu/>, 2013. október 23., 10:02.
- [11] <http://www.ni.com/white-paper/3727/>, 2013. október 23., 19:27.
- [12] <http://www.stack.nl/dimitri/doxygen/>, 2013. október 24., 8:43.