



Høgskulen
på Vestlandet

BACHELOROPPGAVE:
B019E-21 ULTRALYD VÆSKENIVÅ OG
VÆSKETETTHET

Frode Løvbrekke Aase
Øyvind Dahle Ådland

30. mai. 2019

Dokumentkontroll

<i>Rapportens tittel:</i> B019E-21 Ultralyd væskenivå og væsketetthet	<i>Dato/Versjon:</i> 30.Mai 2019
	<i>Rapportnummer:</i> B019E-21
<i>Forfatter(e):</i> Frode Løvbrekke Aase Øyvind Dahle Ådland	<i>Studieretning:</i> EEL16 EAU16
	<i>Antall sider m/vedlegg</i> 47
<i>Høgskolens veileder:</i> Mathias Christian Mathiesen	<i>Gradering:</i> Åpen
<i>Eventuelle Merknader:</i> Vi tillater at oppgaven kan publiseres.	

<i>Oppdragsgiver:</i> HVL	<i>Oppdragsgivers referanse:</i>
<i>Oppdragsgivers kontaktperson(er) (inkludert kontaktinformasjon):</i> Mathias Christan Mathiesen <i>Epost adresse: Mathias.Christian.Mathiesen@hvl.no</i>	

Forord

Vi er to bachelorstudenter fra to forskjellige studieretninger som har valgt å skrive sammen. Én fra elektronikkstudie og én fra automasjonsstudie. Det har vært spennende å jobbe sammen med tanke på det brede fagområdet vi til sammen dekker. Denne oppgaven ble valg fordi vi syntes det så ut som en spennende utfordring som bydde på mye nytt for oss begge. Å skrive bacheloroppgave innebærer mye arbeid, men også mye hjelp fra gode veiledere som har pekt oss i riktig retning.

Vår veileder, Mathias Christian Mathiesen har gitt oss gode råd og vært til stor hjelp. Vi takker for et godt semester og god veiledning. Takk til både Kjell Eivind Frøysa og Eivind Vågslid Skjæveland for lån av utstyr over lengre tid.

Vi vil spesielt også takke Sven Ole Hoff fra Holger Hartmann AS for at vi fikk låne to transdusere i så lang tid.

Sammendrag

Denne bacheloroppgaven er en praktisk designoppgave som går ut på å designe og produsere en batteridreven håndholdt enhet. Enheten skal kunne brukes langs siden på et Corneliusfat, som er et fat for oppbevaring og tapping av øl. Den er på størrelsen av en dykker sin oksygentank. Enheten skal kunne finne væsknivået i fatet. Den skal også kunne si noe om hvor langt ølet er kommet i bryggeprosessen ved bruk av tetthetsmålinger. Det skal brukes ultralyd for å få utført disse målingene. Denne enheten skal ligne en stendersøker i både form og funksjon.

Opgaven er gitt av en av foreleserne ved Høyskolen på Vestlandet, Mathias Christan Mathiesen, som også underviser i høyskolens ølbryggingsfag. Han har var studentene sin veileder gjennom denne oppgaven.



Rapporten inkluderer en løsning til hvordan å behandle signalene fra en ultralydstransduser brukt i sammenheng med Corneliusfatet. Det er også forslag til løsning for hvordan å drive transduseren med et eget batteridrevet system. Transduserdrift ble ikke produsert, men forslaget til løsningen er testet og bekreftet til å fungere fra et annet dokument som er referert til i oppgaven.

Løsning til databehandling er basert på enkel analog elektronisk krets som skaper digitale pulser for hver stor hendelse som er fanget opp av ultralydstransduseren. Disse pulsene blir behandlet av en mikrokontroller som tar tiden mellom utsendt ultralyds puls og mottatt ekko. Tidsdata blir sendt over internettet til en Android mobil applikasjon, hvor dataen blir presentert grafisk. Kodingen av mikrokontrolleren, bruken av database og Android applikasjonen blir også forklart i detalj.

Denne rapporten vil også dekke hvordan å velge en transduser for det aktuelle stålfatet, og hvordan å drive den ved bruk av DPR300 ultrasonic pulser fra JSR ultrasonics. Målerdesignet har per nå ikke stort markedspotensial hos private hjemmebryggere, ettersom prisen på transduseren er veldig høy. Den fungerer allikevel, og står klar til når billigere komponenter kan anskaffes.

1 Innhold

Dokumentkontroll	2
Forord	3
Sammendrag	3
1 Innledning	6
1.1 Oppdragsgiver	6
1.2 Problemstilling	6
1.3 Hovedidé for løsningsforslag	6
2 Kartlegging av transduser funksjon	7
2.1 DPR300	7
2.2 Valg av transduser	9
2.2.1 Bølgeinterferens	9
2.2.2 Piezo-elektrisk transduser	9
2.2.3 Konklusjon for valgt transdusertype	12
2.3 Frekvensvalg	13
2.3.1 Akustisk impedans	13
2.3.2 Kalkulasjoner	14
2.3.3 Konklusjon for transduser	16
3 Kravspesifikasjon	16
4 Analyse av problemet	16
4.1 Utforming av mulige løsninger	17
4.1.1 Kick generering	17
4.1.2 Likeretting og ADC	18
4.1.3 Detektore hendelser	18
4.1.4 Arduino koding	19
4.1.5 Mobilapplikasjon	19
4.2 Konklusjon for valgt løsning	20
5 Realisering av valgt løsning	20
5.1 TPD	20
5.1.1 Likeretter	20
5.1.2 Schmitt-trigger	21
5.2 Android mobilapplikasjon	24
5.2.1 Arduinokode for WiFi modul	25
5.2.2 Android studios	26

5.2.3	ThingSpeak	27
5.3	Arduino kode	28
5.3.1	Koden.....	29
6	Testing	32
7	Diskusjon	34
7.1	Fremdrift.....	34
7.2	Endringer i planen	34
7.3	Forslag	34
7.3.1	Logikk og FPGA	34
7.3.2	Transduserdrift	36
7.3.3	Mobilapplikasjons utvidelse	38
8	Konklusjon.....	38
9	Referanser	39

1 Innledning

1.1 Oppdragsgiver

Oppgaven kommer fra Høgskulen På Vestlandet (HVL) og er gitt av Mathias Christian Mathiesen. HVL er skolen som teamet studerer ved, og Mathias hører til HVL sitt institutt for elektrofag. Han er kontaktperson og veileder for prosjektet.

1.2 Problemstilling

Hjemmebryggere og mikrobryggerier har behov for en enkel metode for å ta målinger på ølet deres. Corneliusfat [1] er en tank av stål på størrelsen av en dykker sin oksygentank, og fatet blir brukt til oppbevaring og tapping av øl. Denne tanken kan også brukes til å brygge ølet. En har lyst å foreta nivåmålinger på tanken for å vite hvor mye øl som er igjen. Omtrent alle løsninger til nå innebærer å slippe en måler inni fatet for å kunne se væsknivået i tanken. Dette utstyret må vaskes og steriliseres for hvert bruk.

Tetthetsmålinger er brukt til å determinere om bryggeprosessen er fullført og da om ølet er ferdig. Dette gjøres med labutstyr som krever stikkprøver av ølet. Dette utstyret må også vaskes og steriliseres. Problemet er at øl kan få en infeksjon ved å komme i kontakt med uønskelige mikroorganismer. Dette ødelegger alt ølet i fatet, som da må kastes.

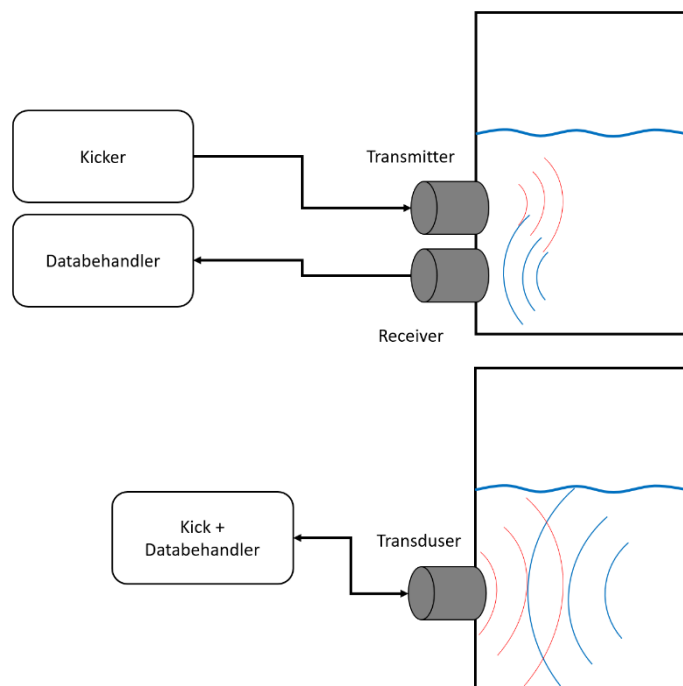
Et apparat skal designes og produseres. Det skal kunne måles tetthet og finne grensesnittet mellom gass og væske i stålfatet uten å komme i kontakt med væsken. Dette apparatet skal benytte seg av ultralyd for å utføre disse målingene.

1.3 Hovedidé for løsningsforslag

Det skal lages en måler som benytter ultralyd til å måle både nivå og tetthet fra utsiden av tanken. På denne måten kan ølet få ligge i fred. Brukeren kan da ta så mange målinger de vil uten å måtte åpne fatet, komme i kontakt med ølet eller bekymre seg om infeksjon. Måleren skal brukes langs siden av fatet, som gjør at brukeren kan få en varsling når brukeren har lagt apparatet direkte over væskegrensesnittet.

2 Kartlegging av transduser funksjon

For å få dette til, må en transmitter for å sende ut en ultralydpuls inn i corneliusfatet. Denne transmitteren trenger en form for driver som i dette prosjektet blir kalt for kicker. Kikeren sender korte pulser med høy spenning for at transmitteren skal sende ut ultralyd. Det må også brukes en receiver for å motta ultralydsignalet. Da kan dataen behandles. Dette skal altså samles til én enkel enhet som kan drive og lese informasjon fra én transduser.

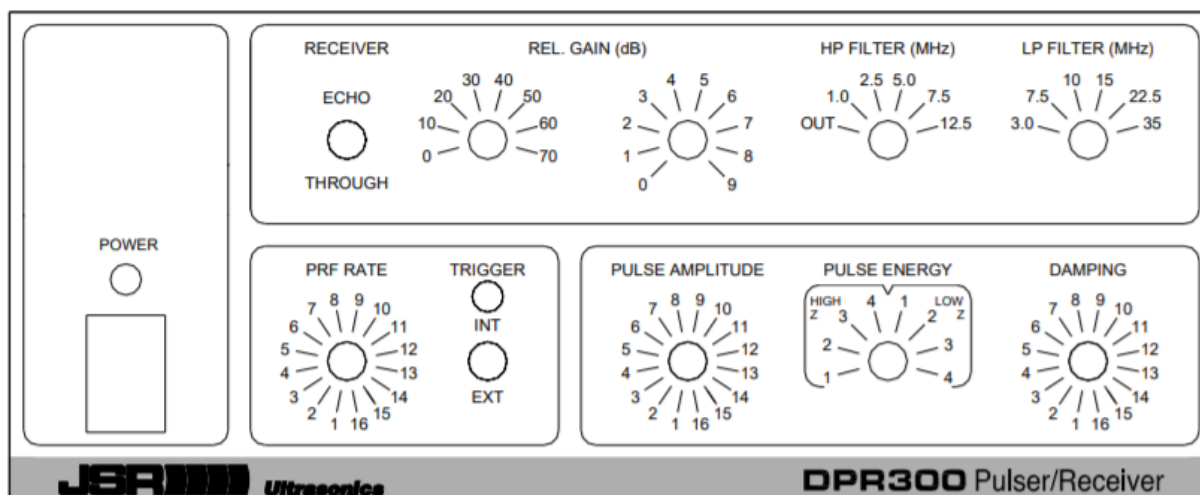


2.1 DPR300

DPR300 produsert av JSR ultrasonics er en datamaskinstyrt ultralyd pulser og mottaker med en ekstremt lav støy. Mottakeren har en båndbredde på 35Mhz, som betyr det er mulig å drive transdusere opp til 35Mhz. Pulseren har en 475 volts amplitude rekkevidde. I en typisk DPR300 applikasjon produseres det en elektrisk spenningpuls med høy amplitude, denne pulsen blir sendt til instrumentets T / R-kontakt (Through/Receive). En ultralydstransduser kobles til T / R-kontakten via en Coax kabel. Deretter konverteres den elektriske energien fra eksitasjonspulsen til en ultralydpuls som forplantes i et medium. DPR300 tilbyr fire energinivåer, seksten amplitudenivåer og to pulsimpedansverdier som gjør at brukeren kan justere egenskapene til eksitasjonspulsen. Denne eksitasjonspulsen vil videre i referert til som kick. Seksten diskrete dempningsnivåer i DPR300 tillater transduseren sin respons å bli justert over et bredt dempningsområde.

Innstillinger på DPR300

- Instrumentkontrollene inkluderer mottakerforsterkning
- høyt og lavpassfilter-cutoff frekvensvalg
- pulsenergi
- pulsamplitude
- pulsimpedans
- dempningsnivå
- Transmisjonsmodusvalg
- Pulsekko: For en transduser som skal sende og mota signal.
- Gjennom: For to transdusere der den ene sender signalet og den andre mottar.
- Pulsrepetisjon
- Pulser-triggerkildevalg for intern eller ekstern triggerkilde.



DPR300 er designet for krevende applikasjoner. Mottakeren har en rask gjenoppretting og er fullstendig beskyttet mot elektromagnetisk støy og interferens for å sikre et høyt signal / støyforhold. I tillegg kan pulser-impedansen, pulsenergien og pulsamplituden tilpasses individuelt for å optimalisere eksitasjonspulsen for en bestemt applikasjon eller transduser.

DRR300 produserer et elektrisk «kick» med høy amplitude (Volt) og lav strøm som blir tilført transduseren. Dette får det piezo-elektriske elementet inni transduseren til å vibrere som medfører at det lager ultralyd lydbølge. Når lydbølgen returnerer tilbake til transduseren begynner piezo elementet og vibrere på nytt, og det blir produsert en elektrisk puls som DPR300 tolker og omregner til spenning.

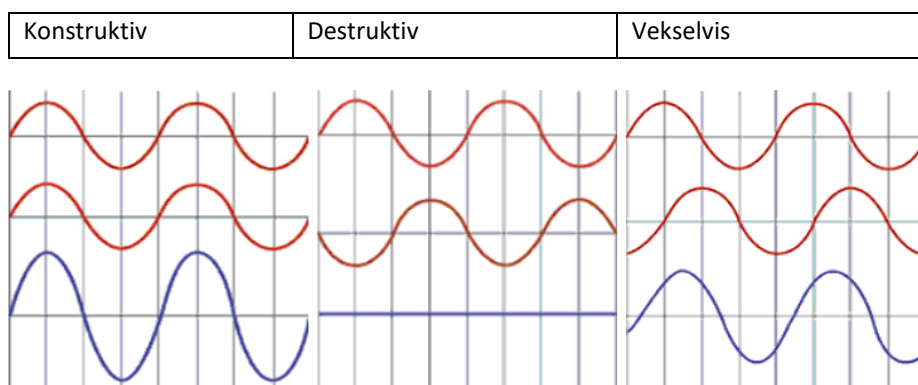
På DPR300 kan Puls raten til transduseren justeres fra 100Hz til 5Khz, det vil si den avfyres en puls hvert 10 – 0.2 millisekund. For oss er det ikke nødvendig med høy puls rate, en høy pulsrate vil trolig også kunne skape feil verdier i koden med tanke på at koden også trenger tid til å kjøres. En Høy pulsrate vil også føre til unødvendig slitasje på selve transduseren.

2.2 Valg av transduser

En Transduser er et elektrisk eller elektromekanisk apparat som har i oppgave å konvertere en type energi til en annen. I dette tilfelle blir det å omforme fra elektrisk energi til ultralydsignal.

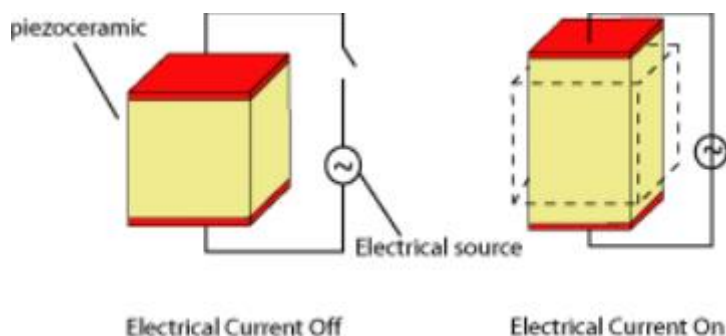
2.2.1 Bølgeinterferens

For å kunne begrunne valget av transduser må bølge interaksjons prinsipp forstås. Lydbølger propagerer ikke bare fra ett punkt på overflaten til transduseren, men fra flere punkter. Dette resulterer i en lyd felt der flere bølger samhandler og skaper interferensen. Bølgene overlapper hverandre og amplituden av lyd trykket på et gitt punkt blir summen av amplitudene til bølgene som interfererer, som gir opphav for utrykkene destruktiv og konstruktiv interferens. To individuelle bølger i samme fase kombineres til en forskyvning av amplituden lik det dobbelte av de to samhandlende bølgene, dette kalles konstruktiv interferensen. To individuelle bølger med komplett ulike fase kombineres slik at amplituden nulles helt ut og resultatet blir et flatt signal, dette kalles destruktiv interferens. To individuelle bølger som verken er helt i fase eller helt ute av fase resulterer i en amplitude som er lik summen av bølgenes amplitude på et gitt punkt, kalt vekselvis interferensen



2.2.2 Piezo-elektrisk transduser

Omdannelsen fra en elektrisk puls (kick) til mekanisk vibrasjon, og omdannelsen fra ekkoet sitt mekanisk vibrasjon til elektrisk energi er selve basisen for ultralyd testing. Det aktive element piezo elektriske elementet er selve hjerte til transduseren. Her blir elektrisk energi omdannet akustisk energi, og omvendt. Det aktive elementet er i utgangspunktet et stykke polarisert materiale (dvs. noen deler av molekylet er positivt ladet mens andre deler av molekylet er negativt ladet) med elektroder festet til to av sine motsatte flater. Når et elektrisk felt påføres over materialet, vil de polariserte molekylene justere seg med det elektriske feltet, noe som resulterer i induerte dipoler innenfor molekylær eller krystallstruktur av materialet. Denne justeringen av molekyler vil føre til at materialet endrer dimensjoner. Dette fenomenet kalles elektrostriksjon. I tillegg vil et permanent

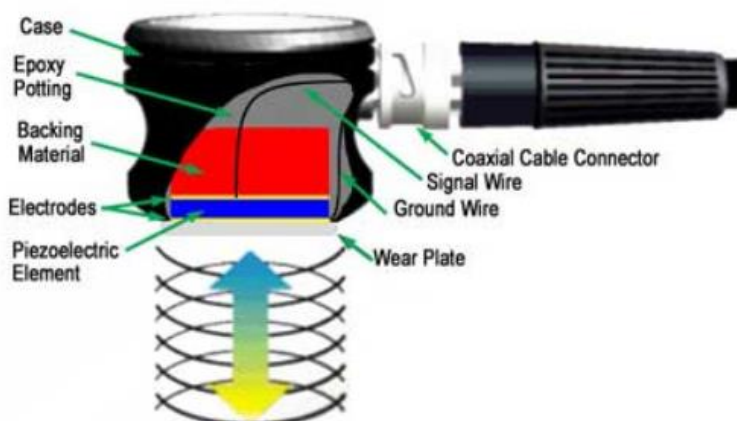


polarisert materiale som kvarts (SiO_2) eller bariumtitanat (BaTiO_3) produsere et elektrisk felt når materialet endrer dimensjoner som følge av en pålagt mekanisk kraft. Dette fenomenet er kjent som den piezo elektriske effekten.

Tykkelsen til det aktive elementet bestemmes av frekvensen transduseren ønsker å produsere. Et tynt element vibrerer med en bølgelengde som er to ganger dens tykkelse. Derfor kuttet piezo elektriske krystaller til en tykkelse som er halvparten av ønsket radiert bølgelengde. Jo høyere frekvens transduseren produserer, desto tynnere blir det aktive elementet.

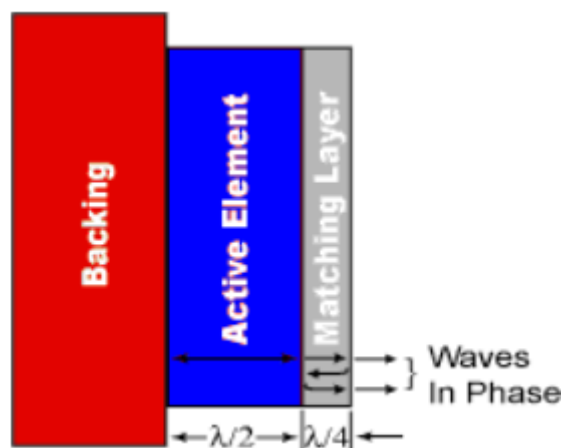
2.2.2.1 Karakteristikk til piezo-elektrisk ultralydstransduser

Transduseren er en svært viktig del av ultralydinstrumentet. Som omtalt på forrige side, inneholder transduseren et piezo-elektrisk element som omdanner elektriske signaler til mekaniske vibrasjoner (overføringsmodus) og mekaniske vibrasjoner i elektriske signaler (mottaksmodus). Mange faktorer, inkludert materiell, mekanisk og elektrisk konstruksjon, og de eksterne mekaniske og elektriske belastningsforholdene, påvirker oppførselen til transduseren. Transduseren sin mekanisk konstruksjon inkluderer parametere som strålingsoverflate, mekanisk demping, innkapsling, koblingstype og andre variabler av fysisk konstruksjon.

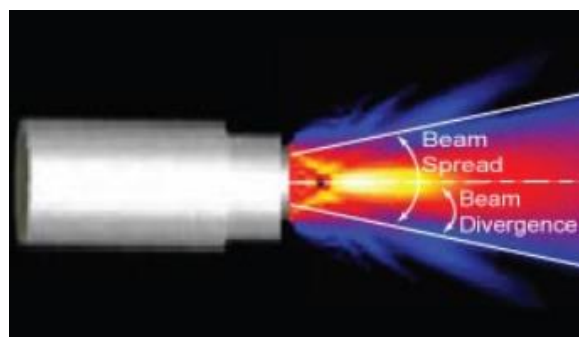


Et snitt av en typisk kontakt transduser er vist ovenfor. Det ble tidligere nevnt at det piezo-elektriske elementet er kuttet til $1/2$ ønsket bølgelengde. Dette for å få så mye energi ut av transduseren som mulig, en impedans matchende lag er plassert mellom det aktive elementet og transduserens utvendige overflate. Optimal impedans sammenstilling oppnås ved å dimensjonere det matchende laget slik at tykkelsen er $1/4$ av ønsket bølgelengde. Dette holder bølger som ble reflektert i matchende lag i fase når de forlater laget (som illustrert i bildet til høyre). For kontakttransduserer er matchingslaget produsert av et materiale som har en god akustisk impedans mellom det aktive element og stål. Immersjonsgivere har et matchende lag med akustisk impedans mellom det aktive elementet og vannet. Kontakt transduserer har også en sliteplate for å beskytte matching laget og det aktive elementet mot riper.

Støttematerialet (Backing material) som støtter opp under det aktive elementet har stor innflytelse på dempningsegenskapene til en transduser. Ved å bruke et støttemateriale med en impedans som ligner på det aktive elementet, vil det produsere den mest effektive dempingen. En slik transduser vil ha en bredere båndbredde som resulterer i høyere følsomhet. Økt ulikhet i impedansen mellom det aktive elementet og støttematerialet fører til økt materialpenetrasjon, men transduser sin følsomhet reduseres.



Noen transdusere er spesialfremstillet til å være mer effektive sendere og andre til å være mer effektive mottakere. En transduser som fungerer bra i en applikasjon, vil ikke alltid produsere det ønskede resultatet i en annen applikasjon. For eksempel er følsomheten for små defekter proporsjonal med produktet av effektiviteten til transduseren som en sender og en mottaker.

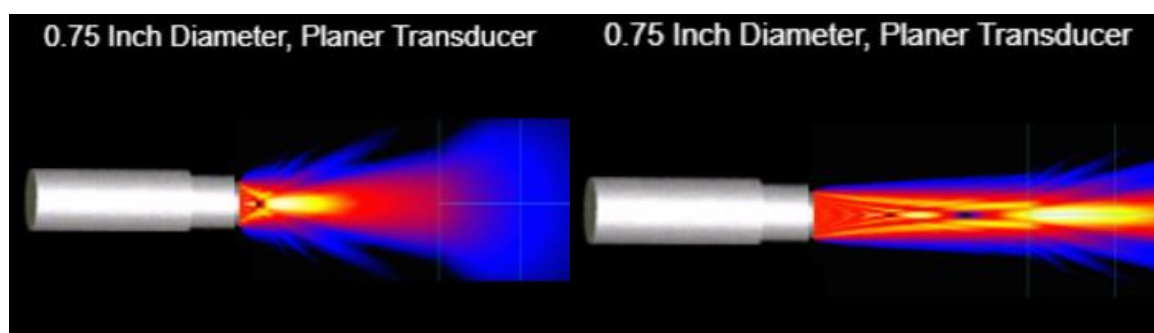


Oppløsning er evnen til å lokalisere defekter nær

overflaten eller i nærheten av materialet og krever en svært dempet transduser. Det er også viktig å forstå båndbreddebegrepet, eller frekvensområdet, som er forbundet med en transduser.

Frekvensen på en transduser kalles sentralfrekvensen og avhenger hovedsakelig av støttematerialet. Høyt dempede transdusere vil reagere på frekvenser over og under sentralfrekvensen. Det brede frekvensområdet gir en transduser med høy oppløsningsevne. Mindre dempede transdusere vil vise et smalere frekvensområde og dårligere oppløsningsevne, men større penetrasjon. Sentrale frekvensen vil også definere evnen til en transduser. Lavere frekvenser (0.5MHz-2.25MHz) gir større energi og penetrasjon i et materiale, mens høyfrekvente krystaller (15.0MHz-25.0MHz) gir redusert penetrasjon, men større sensitivitet for små diskontinuiteter. Høyfrekvente transdusere brukt med riktig instrumentering, kan forbedre feiloppløsning og tykkelsesmålingskapasiteter dramatisk.

Transdusere er konstruert for å motstå noe misbruk, men de bør håndteres nøye. Misbruk, slik som et fall kan føre til sprekk i den utvendige sliteplaten, det aktive elementet eller støttematerialet.



1MHz

5Mhz

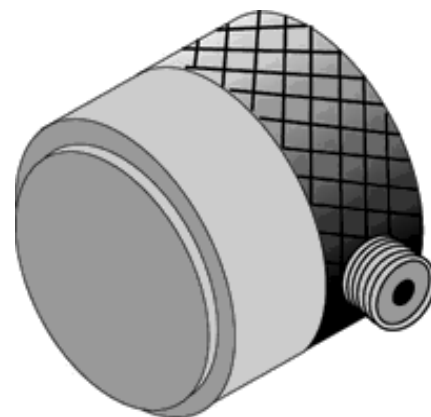
Strålespredningen bestemmes i stor grad av sentralfrekvensen og diameteren til transduseren. strålespredningen er større ved bruk av en lavfrekvent transduser enn ved bruk av en høyfrekvent transduser. Når transduktorens diameter øker, vil strålespredningen reduseres.

Stråle vinkel er et viktig å ta hensyn til i valg av transduser av et par grunner. For det første reduserer strålespredningen amplituden til refleksjoner siden lydfeltene er mindre konsentrerte og dermed svakere. For det andre kan strålespredning føre til flere vanskeligheter med å tolke signaler på grunn av refleksjoner fra de innvendige veggene av testobjektet. Karakteristikken av lydfeltet generert av en transduser er en forutsetning for å forstå det observerte signaler.

2.2.3 Konklusjon for valgt transdusertype

Det er viktig å velge transduser som har rett ønsket frekvens, båndbredde og som fokuserer på å optimalisere inspeksjonsevnen. For applikasjonen blir en kontakt transduser det rette valget.

Kontakttransdusere brukes til direkte kontaktinspeksjoner. Elementene er beskyttet i en robust innkapsling for å motstå glid friksjon i kontakt med en rekke materialer. Disse transduserne har et ergonomisk design slik at de er enkle å gripe og bevege seg langs en overflate. De har ofte utskiftbare sliteplater for å forlenge levetiden. Ett kontakt medium av vann, fett, oljer eller kommersielle materialer brukes til å fjerne luftgapet mellom transduseren og komponenten som blir inspisert. Under testing ga vaselin det beste resultatet, siden vaselin har en akustisk impedans nær rustfritt stål. Kontakt sensorene har en akustisk impedans som matcher de fleste type metaller. Frekvens området ligger en del høyere enn på de mer kommersielle transduserne, dette er gjort med tanke på at de brukes i kontakt med metall. Med høyere frekvens er det også en høyere oppløsning og sensitivitet som er viktig for avlesningen av selve ekkoet som blir returnert. Ulemper med valg av en høy frekvens er at forplantnings avstand blir mindre, men dette er ikke et reelt problem for oss med tanke på diameteren til Corneliusfatet. Høy frekvens vil også gi en smalere stråle (mindre vinkel) som er en fordel med tanke på at det ikke ønsket et ekko fra vannoverflaten når transduseren er plassert over vannoverflaten. Piezo-elementet i kontakt sensorene kan produsere longitudinale og laterale lyd bølger. For best mulig resultater i applikasjonen er det viktig at bølgene er retnings bestemt, derfor har er det valgt å gå for et piezo-element som produserer longitudinale bølger. Tidsoppløsningen er et systems egenskap til å gjenkjenne to punkter som separerte. Den longitudinale aksens oppløsning øker med økt frekvens og minsket lengde av pulsen ut fra transduseren. Den laterale oppløsning avtar med økt forplantningsavstand, derfor gir korte forplantningsavstander best resultat i den laterale oppløsningen. Transduseren må klare å skille mellom inngangs og utgangs bølger. Med tanke på den lave forplantningsavstanden på 30 cm blir dette problematisk med lave frekvenser. Grunnet kravspesifikasjonene er det kun kontakt transdusere som er et aktuelt valg for oppgaven. Dette ble en stor utfordring for oss med tanke på markedsprisen til disse sensorene som starter på halvparten av prosjektets budsjett på 10 00kr. Dette medførte at en revurdering av kravspesifikasjonene måtte gjennomføres sammen med veileder. I prosessen av å kjøpe inn en transduser ble Holger Hartmann AS i Oslo kontaktet. De var villig til å låne vekk to kontakttransdusere uten kostnad.



2.3 Frekvensvalg

Kontakt sensorene produseres i et bredt spekter av frekvenser, alt fra 200 KHz til 10 MHz. Under testingen av HVL sine piezo-elektriske transdusere kom det tydelig fram at ikke alle frekvenser er egnet for applikasjonen. Prosedyren ble litt baklengs da testing ble gjennomført først, for å så understreke resultatene med beregninger fra fysikken som ligger bak. Det er to fysiske prinsipper som er spesielt relevant for oss med tanke på valg av frekvens, Akustisk Impedans og halvbølge resonans.

2.3.1 Akustisk impedans

All materie har en akustisk impedans. Forskjeller i karakteristiske akustisk impedans bestemmer hvor mye av en bølge som blir transmittert og reflektert når en "plan bølge" når et plant grensesjikt mellom to medier. Når tetthet til materialet øker vil også lydbølgene i materialet sin hastighet øke fordi partiklene er tettere på et molekylært nivå, så energien blir overført kjappere.

$$z = p/u.$$

ρ er massetettheten til mediet (kg/m³)

u er lydhastigheten i mediet (m/s)

z_0 materialavhengig og måles i (Ns/m³)

Det skilles mellom karakteristisk akustisk impedans og akustisk impedans. Akustisk impedans er ikke bare avhengig av materialet, men også av fysisk utforming. Størrelsen forteller hvor stort lydtrykk som genereres som følge av vibrasjon av molekyler i et akustisk medium ved en gitt frekvens og for en bestemt gjenstand eller geometri. Den akustiske impedansen Z er oftest betydelig mer frekvensavhengig enn den karakteristiske akustiske impedansen Z_0 . Akustisk impedans er en svært nyttig størrelse innen akustikk, f.eks. når det skal regnes på hvor mye effekt som må til for å få et visst lydnivå i en konsertsal. Det er imidlertid aller størst forskjell mellom karakteristisk akustisk impedans for luft og metallet på utsiden av Corneliusfatet. Det er nødvendig å få lyd effektivt inn og ut av fatet, derfor er det nødvendig med et smørbart materiale mellom transduseren og fatet. Dette materialet bør ha omtrent samme karakteristiske akustiske impedans som metallet ultralydbølgene skal gå inn i. Lydbølgene vil etter refleksjon i grenseflater mellom ulike impedanser kunne oppfanges som et ekko. Dette er forutsatt at den initiale lydimpulsen allerede er avsluttet før ekkoet kommer tilbake. Bli ekkoet analysert som en funksjon av tid, vil avstanden kunne bestemmes. Dersom lyden sendes i vel definerte retninger, vil det også kunne danne et bilde av det som er inne i Corneliusfatet.

2.3.2 Kalkulasjoner

$$\text{Rustfritt stål: } Z = 7700 \frac{\text{kg}}{\text{m}^3} \times 5800 \text{ m/s} = 44,66 \times 10^6 \text{ (Ns/m}^3\text{)}$$

$$\text{Vann: } Z = 1000 \frac{\text{kg}}{\text{m}^3} \times 1498 \text{ m/s} = 1,498000 \times 10^6 \text{ (Ns/m}^3\text{)}$$

$$\text{Luft: } Z = 1.225 \frac{\text{kg}}{\text{m}^3} \times 343 \text{ m/s} = 420 \text{ (Ns/m}^3\text{)}$$

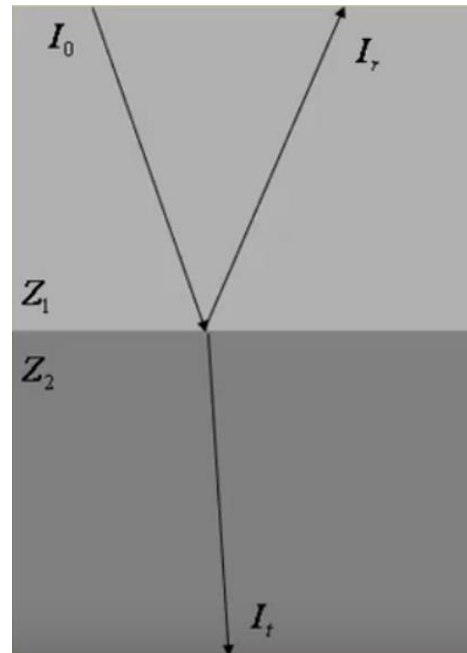
R: Refleksjonskoeffisient (%)

Fra rustfritt stål til vann

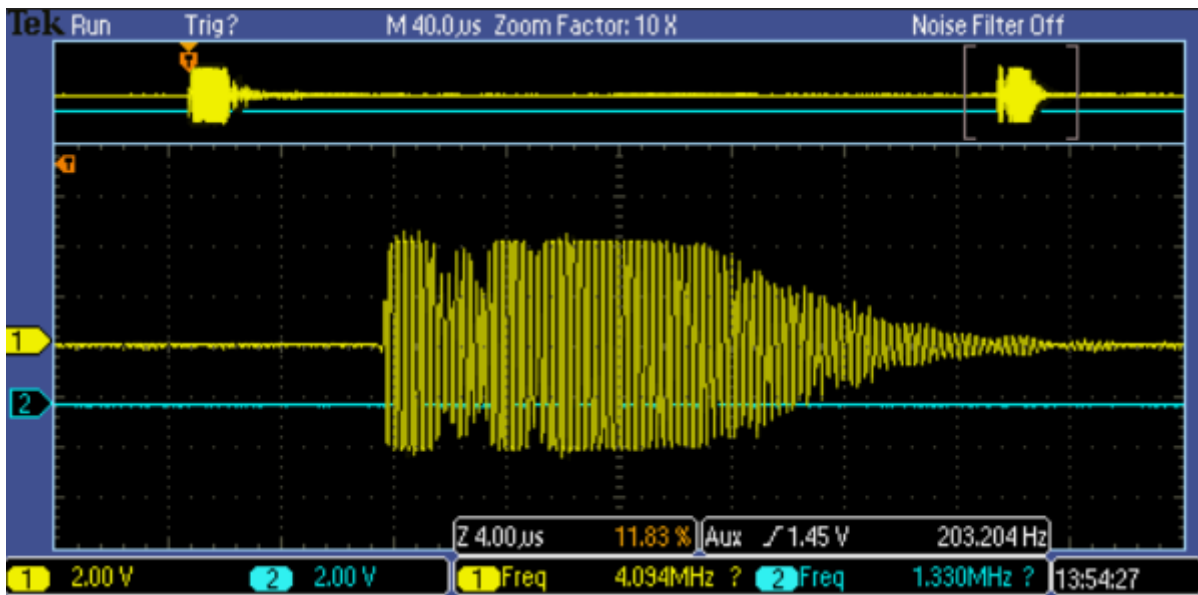
$$R = \frac{(z_2 - z_1)^2}{(z_2 + z_1)^2} = 87,44\%$$

Fra rustfritt stål til luft

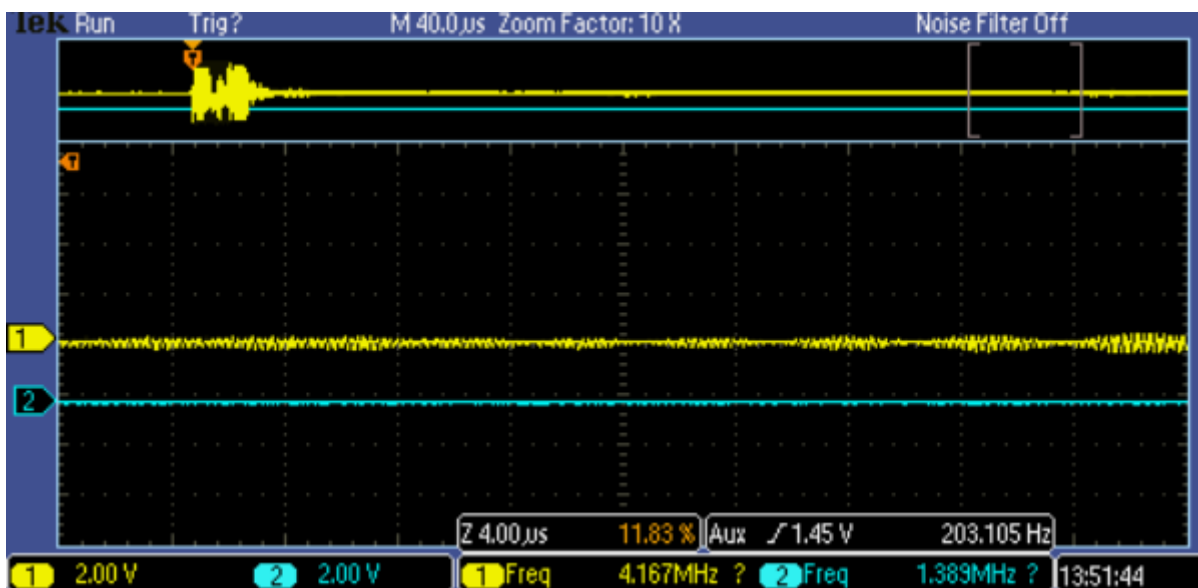
$$R = \frac{(z_2 - z_1)^2}{(z_2 + z_1)^2} = 99,99\%$$



Siden så godt som all av energi blir reflektert tilbake fra stål til luft (99,99%) vil ikke transduseren registrere et ekko grunnet den store forskjellen i akustisk impedans. Dette er til fordel for applikasjonen da det indikerer at lydimpulsen blir sendt ut over væskespeilet og det ikke lengre er behov for å prøve å se etter størrelse forskjell i amplituden og tid tur/retur mellom ekko fra vann og luft. Fra stål til vann blir 12,64 % av energien overført til væsken noe som med første øyekast virker for lavt. Halv bølge resonans øker energien som blir tilført vannet som blir forklart under. Oscilloskopet og Arduino klarer å registrere ekkoet som blir returnert



Måling med 5MHz immersjons transduser ved væske.

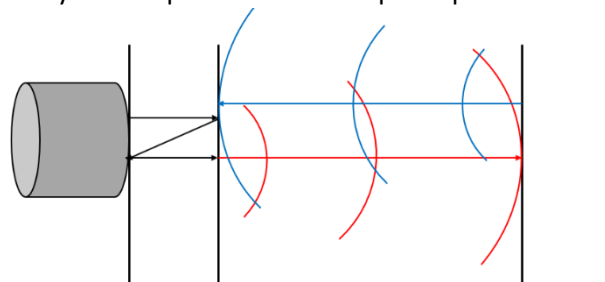


Måling med 5MHz immersjons transduser over væsknivå.

På bilde nummer en er transduseren plassert over vannoverflaten. Det kommer frem at oscilloskopet ikke registrer noe ekko, noe som er ønskelig. Trigger pulsen har høyere amplitude som en funksjon av tid i forhold til bilde nummer 2 der transduseren er plassert på vannoverflaten. Dette kan begrunnes med beregningene over. Nesten all energien blir reflektert tilbake fra metallet til transduseren som gir en økt styrke på amplituden til trigger pulsen. Oscilloskopet registrer ikke energien som blir reflektert tilbake fra metallet som et ekko.

2.3.3 Konklusjon for transduser

Corneliusfat har en typisk tykkelse på stålveggen som ultralyden må passere. Man må passe på å velge en frekvens som ikke blir utsatt for destruktiv interferens internt i stålveggen. Ideelt er målet at frekvensens bølgelengde i rustfritt stål skal være det dobbelte av tykkelsen til veggen.



Dessverre er stålveggen sin tykkelse ukjent, og det finnes ikke en god bestemt standard foruten at den pleier å være under 1mm.

Trolig skulle 5MHz kontakttransduser ha en bedre respons, men dette var ikke tilfellet ved testing. Signalformen som ble målt var ujevn og det var en del støy. Under testing av alle tilgjengelige transduser, ble det oppdaget at 5MHz immersjonstransduseren hadde desidert den beste responsen for den tanken som var til disposisjon.

3 Kravspesifikasjon

Oppdragsgiver har presisert noen krav til hva dette apparatet skal være og hvordan den skal fungere:

1. Det er ønskelig med en liten håndholdt enhet. Denne skal likne en stendersøker.
2. Enheten må være batteridreven.
3. Hovedmålet er å finne grensesnittet mellom væske og gass, uten at den kommer i kontakt med væsken inni stålfatet.
4. Nivåmålingen må være nøyaktig innen $\pm 1\text{cm}$.
5. Bruker må få tilstrekkelig intuitivt varsel på detektering av grensesnitt.
6. Det er ønskelig at apparatet skal kunne finne tettheten til væsken. Dette kravet ble redusert til en måling av tiden en måleprøve tok, for så å bruke tiden som en indikator på tetthet.
7. Tiden skal vises på en skjerm som brukeren kan lese av.
8. Ultralyd skal benyttes. Hensikten er å kunne foreta målinger fra utsiden at stålfatet.
9. Overkommelig kostnad, da målgruppen for endelig produkt er for privatmarkedet.

Prototypen ble laget kun med tanke på kjerneegenskapene som måtte testes. Kravet ble da redusert til ett oppsett med det grunnleggende utstyret som trengs. Oppsettet skal kunne utføre nivåmålinger ved spesifisert nøyaktighet og varsle om dette, samt tetthetsmålinger som gir oss måletiden. Det trenger ikke å være håndholdt eller batteridrevet, så lenge det blir laget et oppsett som utfører hovedkonseptet.

4 Analyse av problemet

Det er fire hovedutfordringer som må løses i denne oppgaven, disse kan deles inn i fire blokker:

Den ene blokken er transduserdrift. Det må finnes ut av hvor høy spenningsamplitude og energimengde transduseren behøver for å driftes optimalt. Designet må kunne konsistent levere dette kicket i ett bestemt intervall.

Den andre blokken er avlesning. Signalene fra transduseren må kunne mottas / fanges opp. I dette tilfellet trengs det ganske høy frekvens på transduseren. Målingene viser at et lydsignal ikke varer mye lenger enn 16µs og har en amplitude på 2V ved bruk av DPR300. Signalet er altså veldig kort.

Den tredje blokken er da hvordan signalet behandles. Denne delen må ta avgjørelser basert på signaler som er mottatt. Denne blokken står for nivåvarsling og tidsmålingen.

Den fjerde blokken står for brukergrensesnittet. Det skal varsles til brukeren på intuitiv og lett forståelig måte at en har funnet væskegrensesnittet. Den må også kunne enkelt presentere tidsmålingsdata.

I tillegg til dette, er målet at produktet skal kunne selges på privatmarkedet. Det må derfor gjøres så billig som mulig.

For å oppsummere:

1. Transduser drift
 - Skal kunne drive transduseren så den kan gi konsistente målinger.
 - Må drives i et bestemt tempo.
2. Avlesning
 - Må kunne oppdage og fange signal med kort varighet fra transduseren.
 - Må sørge for at blokk 3 kan bruke signalet.
3. Data behandling
 - Må bestemme tiden signalet tok fra transduseren, ut i væsken og tilbake.
 - Må oppdage om man måler under eller over væskegrensesnittet.
4. Varsling og brukervennlighet
 - Må gi god varsling når grensesnitt er funnet.
 - Må vise tidsmåling på en lettvinnt måte til brukeren.

Krav for alle disse punktene er at det skal være billig, ta liten plass og vær batteridrevet.

4.1 Utforming av mulige løsninger

Microkontrollere er veldig nyttig for blokk 3 og blokk 4. For prototyping og generell utprøving, er «Arduino development boards» [2] billige, liten i størrelse og lett programmerbare. Dette gjør den ideell for prosjektets nivå og bruk. Dessverre er den noe treg. Det må designes noe som kan fange opp signalet fra transduseren, og presentere dataen på en slik måte at mikrokontrolleren kan tolke og bruke signalet.

Noe må også legges til for å verne arduinoen og elektronikken fra den høye spenningen som blir generert fra blokk 1 (transduser drift). Blokk 1 må skape en kort puls med høy spenning. Denne pulsen må være med en viss amplitude, og ha en viss energimengde for å gi best resultat.

4.1.1 Kick generering

Fra kartleggingen tidligere i oppgaven leser man at for å drive en transduser, trengs det en kort puls med høy spenning og lav strøm. Dette «kicket» er det som får transduseren til å sende ut en lydbølge i dens egen gitte frekvens. For å designe dette leddet, og konstruere det for én spesifikk transduser, kan en gå ut fra en løsning foreslått av «AIP Conference Proceedings 1927, 030018 (2018)» [3]. Her

brukes det timere for å sørge for godt intervall mellom kickene og pulsbredde for hvert kick. I tillegg brukes det MOSFET transistor og andre komponenter for å drive spenningsdelen som leverer kicket.

For høyspenningsmodul som en kan drive fra batteri, så foreslås EMCO HV DC-DC konvertere i A serien [4]. Spesifikt EMCO A10 ser ut som et godt valg at DC-DC konverter.

4.1.2 Likeretting og ADC

I mottakeren kan det brukes en likeretter og resistor-kondensator (RC) ledd som «strekker» ut signalet. Det skal vare lenge nok til at det kan tas en analog avlesning med arduinoen sin Analoge Digital Converter (ADC). Denne spenningsamplituden på den analoge avlesningen skal kunne si noe om hvor måleren er plassert på grensesnittet mellom gass og væske, eller om det ble målt på væsknivå i det hele tatt. Dessverre så er arduinoen såpass treg at signalet dør ut før arduinoen plukker den opp. En rask mikrokontroller trengs med denne metoden.

4.1.3 Detektore hendelser

Arduinoen sine digitale avlesningsegenskaper er mye raskere enn analoge. Så det ønskes derfor å tilpasse signaler fra transduseren til en digital. Da kan dataen leses av raskere. Det gode med dette er at et utsendt lydsignal eller respons fra væsken kan tolkes som enkle hendelser. Disse enkle hendelsene kan da bli representert med hver sin digitale puls. Måten en kan gjøre om en hendelse til en digital puls er å bruke en likeretter. Dette er etterfulgt av et RC-ledd for utglattung av den nye DC spenningen. Da brukes det en Schmitt-trigger for å gjøre én hendelse om til én klar puls. Denne elektroniske løsningen kan kalles for en «Transduser Puls Detektor» (TPD).

4.1.3.1 TPD alene

Det er en god mulighet at TPD kan justeres på en slik måte at den kan kobles opp direkte til arduinoen. Arduinoen sin digitale inngang tar cirka 5µs til å registrere en digital verdi. Dette medfører at pulsen skapt av TPD bør ha en varighet på opptil 3 til 4 ganger denne avlesingstiden. Da kan det garanteres at arduinoen oppfatter pulsen, og leser den av riktig. Tidtelleren i denne løsningen blir litt unøyaktig.

Her er også noen forslag til hva som kan brukes etter TPD trinnet for å assistere med signalavlesning:

4.1.3.2 Ekstern logikk mottak

Mikrokontrolleren kan avlastes ved å bruke en ekstern digital logikk. Denne kan telle antall klokkepuls mellom kick og respons som mikrokontrolleren da kan bruke til å regne ut reisetiden til den utsendte lydsignalet. Denne logikken kan også vise om respons ble mottatt eller ikke. Denne blir svært nøyaktig, men gammeldags.

4.1.3.3 JK vippe mottak

Antall komponenter kan reduseres i forhold til ekstern logikk ved å bruke bare en enkel JK vippe etter TPD. Arduinoen kan lese av den logiske verdien på denne for å bedømme om det ble mottatt en respons. I og med JK vippet kan lagre på en verdi, egner dette seg godt for den trege Arduinoen. Ulempen her er at tidtelleren blir litt unøyaktig. JK vippet blir bare brukt for å ha hensyn til tregheten til arduinoen.

4.1.3.4 FPGA

En Field Programable Gate Array (FPGA) er en programmerbar hardware enhet. En FPGA programmeres med kode som da kobler FPGA brikken til en definert digital logisk krets. Denne kan

da programmeres til å bli lik det eksterne logikk forslaget som ble nevnt tidligere. Da blir den lett å sette opp, enkel å endre og svært nøyaktig ettersom klokkehastigheten kan være oppi 50MHz i noen tilfeller. FPGAer kan også vær små i størrelse, som gjør dem ideelle for denne oppgaven. Siden denne er en hardware enhet, opererer denne med høyere hastighet enn arduinoen som er en software basert enhet.

4.1.4 Arduino koding

Under utvikling av Arduino ble koden testet med og uten JK vippe som en del av kretsen. Uten JK vippe registreres det en digital puls fra hver hendelse (Utsendt puls og returnert ekko). Med JK vippe vil disse to kombineres til en større digital puls fra kicket sin positive flanke til ekkoet sin positive flanke. Teoretisk sett virker det smartere å inkludere JK vippet i kretsen, men det byr på noen utfordringer. JK vippet har behov for å bli resatt mellom hver måling for å være sikkert på at den er i takt. Hvis en resett blir oversett vil hele JK vippet sin utgang bli speilvendt og dataen blir da uleselig for arduinoen.

Arduino koden må i begge tilfeller ta tiden fra den initiale utsendte pulsen til det returnerte ekkoet for hver måling. Når Arduino registrer et gyldig ekko vil verdiene blir lagret i en tabell som Arduino regner ut gjennomsnittet av etter x antall målinger. Dette blir gjort for å ha et mål på endring i tettheten til væsken. Kretsen har en WiFi modul som laster opp verdien av gjennomsnittet til en database. Dette gir brukeren muligheten til å grafisk kunne se endringen i tetthet mellom målingene på en mobil applikasjon som henter data fra databasen.

Arduino koden må også i begge tilfeller varsle om detektert væske. Dette gjøres ved hjelp av et audioelement som lager et hørbart pip, denne hendelsen blir ikke lastet opp til databasen.

4.1.4.1 Arduino program for JK vippe

I store trekk blir koden sin oppgave å registrere første positive flanke fra den utsendte pulsen og starte en timer kalt start. Start timeren har høy oppløsning og vil fortsette å telle helt til Arduino registrerer at den initialen pulsen returnerer til en digital null, og timeren stopper. Resatt blir aktivert når den positive flanken varer i mer enn 500 mikrosekund, som er en indikator på at ekkoet ikke er blitt funnet.

4.1.4.2 Arduino program uten JK vippe

Koden sin oppgave blir å registrere initiale trigger pulsen og starte timeren kalt start. For at koden skal fortsette må en negativ flanke registreres, hvis ikke vil koden loope på nytt å resett timeren. Når ekkoet fra vannoverflaten registreres vil Arduino tolke dette som en digital høy, og timeren vil bli stoppet. Om timeren teller i mer enn 500 mikrosekund, vil arduinoen tolke dette som en ugyldig hendelse ettersom dette ikke er et ekko.

4.1.5 Mobilapplikasjon

En mobil applikasjon kan bli brukt til å lagre og presentere data og vil være intuitivt og trolig føre til en god bruker opplevelse. For å kunne gjennomføre dette trenges det i å integreres en WiFi modul i kretsen for å kunne laste opp dataen direkte til en mobil applikasjon eller via en database. Under design av mobilapplikasjon og den eventuelle databasen kan dataen fremstille grafisk i form av et 2D plott der y aksene representerer tetthet og x aksene representerer tid.

4.2 Konklusjon for valgt løsning

Arduinoen er meget god for sin pris, og lett å bruke. Analoge avlesninger er fortsatt for tregt, derfor brukes det digital avlesning i kombinasjon med TPD. Siden TPD bruker hysteres med hysteresegrenser som kan justeres, kan den brukes som en god erstatning til arduinoens analog avlesning. Ved å justere TPD slik at den gir ganske brede pulser, tilfredsstiller dette tidskravet til arduinoen. FPGA blir ikke benyttet ettersom arduinoen kan brukes direkte med TPD. Litt unøyaktighet i tidsmålingene blir da godtatt.

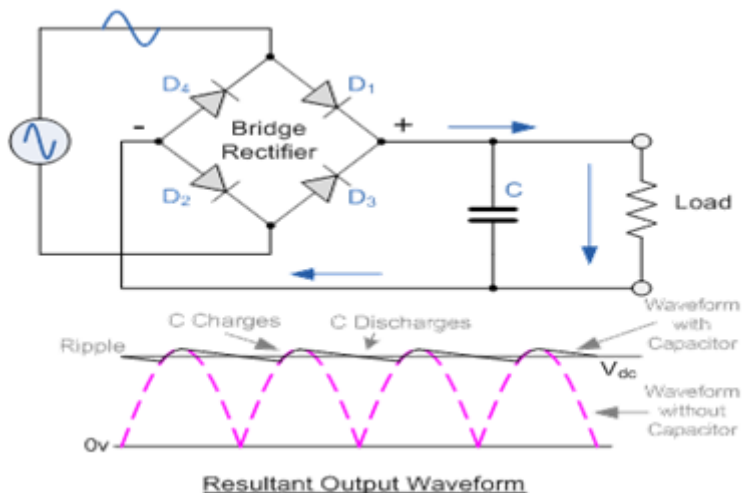
5 Realisering av valgt løsning

5.1 TPD

I første omgang ble DPR300 ultrasonic pulser brukt for å ha noe som fungerer helt godt og forutsigbart, mens den andre halvdelen av løsningen utvikles. Fokuset er på å få arduinoen og signalmottak til å fungere. All elektronikk på koblingsbrettet krever en maksimumspenning på 5V. Spenningen kan hentes fra arduinoen.

5.1.1 Likeretter

Første trinnet som signalet fra transduseren vil møte, er en likeretterbro. Denne har til hensikt å inverttere de negative bølgehalvdelenene til signalet slik at det ikke mistes noe signal. All spenning blir da positivt rettet. Etter dette trinnet glattes signalet ut med et RC-ledd. Hovedgrunnen til dette er at da oppnås en jevn spenning. Uten det jevne spenningsnivået, ville Schmitt-triggeren ha skiftet utgangsverdi flere ganger på den ene responsen. Dette er på grunn av at spenningen gjentakelig går ned til 0V og opp til sin maks amplitude, som vist på bilde nedenfor. RC-leddet glatter ut signalet, som gir en fin spenning som omtrent følger bølgeetoppene på inngangsspenningen, mottatt fra likeretteren. Dette kan også sees på bildet nedenfor.



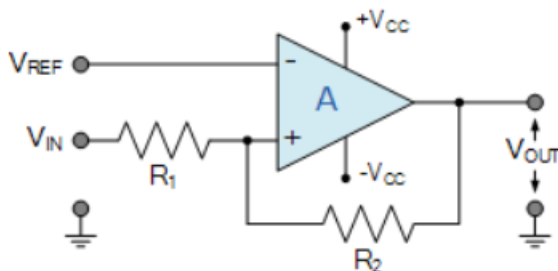
Likeretter oppsett og RC ledd [5]

For å bevare så mye spenning som mulig, brukes det lavspennings Schottky dioder. Disse har som regel en lav forward voltage (V_f) og rask reaksjons tid. Det brukes Schottky dioder med forholdsvis lav V_f .

RC-leddet ble bestemt med bruk av prøv-og-feil metode. For stor tidskonstant betyr at kondensatoren ikke utlader seg fullstendig i tide. Ved for liten tidskonstant, blir ikke varigheten til utladningen lang nok til at det tilfredsstillende kravene. $R=10k\Omega$ og $C=5.6nF$ viste seg å fungere veldig bra. Motstanden over kondensatoren når den er koblet til resten av kretsen ble målt til å være $9.5k\Omega$. Tidskonstanten « τ » blir da $56\mu s$.

5.1.2 Schmitt-trigger

Schmitt-triggeren skal lage digitale pulser med lav spenning på 0V og en høy spenning på 5V. Det blir brukt en 5V forsyning til Operasjons forsterkeren (op-ampen) og derfor velges det en Rail-to-rail op-amp for at utgangen skal kunne oppnå nivået til forsyningsspenningen. Op-ampen må også ha god evne til å endre sin utgangsspenning fort, derfor ble høy slew rate lagt i fokus. Denne op-ampen skal altså være rask, og skal kunne utnytte hele sitt forsyningsområde. Det ble derfor valgt TLV2772CP [6] operasjons forsterker fra Texas Instruments.



Ikke inverterende Schmitt-trigger [7]

$$\frac{V_{IN} - V_{Ref}}{R_1} = \frac{V_{Ref} - V_o}{R_2}$$

KCL likning for [7]

Schmitt-triggere fungerer med bruk av hysteresis. Den positive feedback-loopen tvinger utgangen i metning etter et visst vippepunkt. Ved å koble +Vcc til 5V og -Vcc til jord, defineres det to mulige tilstander til Vout. Løs likningen for Vin og finn at:

$$V_{IN} = -\frac{R_1}{R_2}V_{out} + V_{Ref}\left(\frac{R_1}{R_2} + 1\right)$$

Schmitt trigger grunnlikning

Vout kan ha to tilstander, lav utgang på 0V og høy på 5V. De blir tildelt begrepene Vol og VoH. Bruken av disse i grunnlikningen gir to forskjellige verdier på Vin som fungerer som grenser eller vippepunkter. Om inngangsspenningene krysser vippepunktene, slår utgangen over til den andre tilstanden. Disse to verdiene på Vin er hysteresisgrensene. Videre arbeid på likningen gir:

$$V_{InH} - V_{InL} = \Delta V_{IN} = -\frac{R_1}{R_2}(V_{oH} - V_{oL})$$

Hysteresis område likning

Denne likningen beskriver avstanden på den øvre og nedre hysteresisgrensen, ΔV_{in} . Det ønskes at øvre grense skal være på $V_{inH}=4V$ og nedre på $V_{inL}=3V$. Dette tilsvarer område på $\Delta V_{in}=1V$ mellom

grensene. Bruken av likningen gir et forholdstall mellom R1 og R2. Motstandene R1 og R2 blir bestemt ut fra dette forholdet. R1=6.8kΩ og R2=33kΩ+1kΩ ble avgjort som en god nok tilnærming.

Disse verdiene brukes i grunnlikningen for å bestemme Vref. Løs grunnlikningen for Vref og sett inn grenseverdien som er korrekt for tilstanden. Nedre grense (VinL) gjelder for den høye utgangstilstanden (VoH). Øvre grense (VinH) er aktuell for den lave utgangstilstanden (VoL)

Likningene for Vref blir da:

$$V_{Ref} = \frac{V_{INL} + \frac{R_1}{R_2} V_{oH}}{\frac{R_1}{R_2} + 1} = \frac{V_{INH} + \frac{R_1}{R_2} V_{oL}}{\frac{R_1}{R_2} + 1}$$

Likninger for å finne Vref

Legg merke til at Vref er lik for begge tilstandene. Dette er på grunn av at det blir bestemt hvor hysteresen blir plassert ved å sette spenning på Vref. Størrelsen på hystereserområde ble bestemt med hystereserområde likningen. Vref fungerer altså som en hystereserområde offset.

Vref er kalkulert til å bli 3.3V. Dette realiseres med en enkel spenningsdeling fra 5V kilden. R1 = 12kΩ og R2 = (1.8kΩ + 15kΩ) blir koblet i serie. Spenningen over R2 blir brukt til Vref.

Hystereserområde dimensjoneringen og plassering tar utgangspunkt i både signalformen mottatt fra likeretteren og signalstyrken fra transduseren plassert på grensesnittet. Fullt signal styrke fra transduseren oppnås ved å foreta en måling helt under væskegrensesnittet, der det er full energioverføring fra transduser til væsken inni fatet. Øvre grense er altså plassert med hensyn på hvor stor spenning den første responsen har når transduseren er plassert midt på væskegrensesnittet.

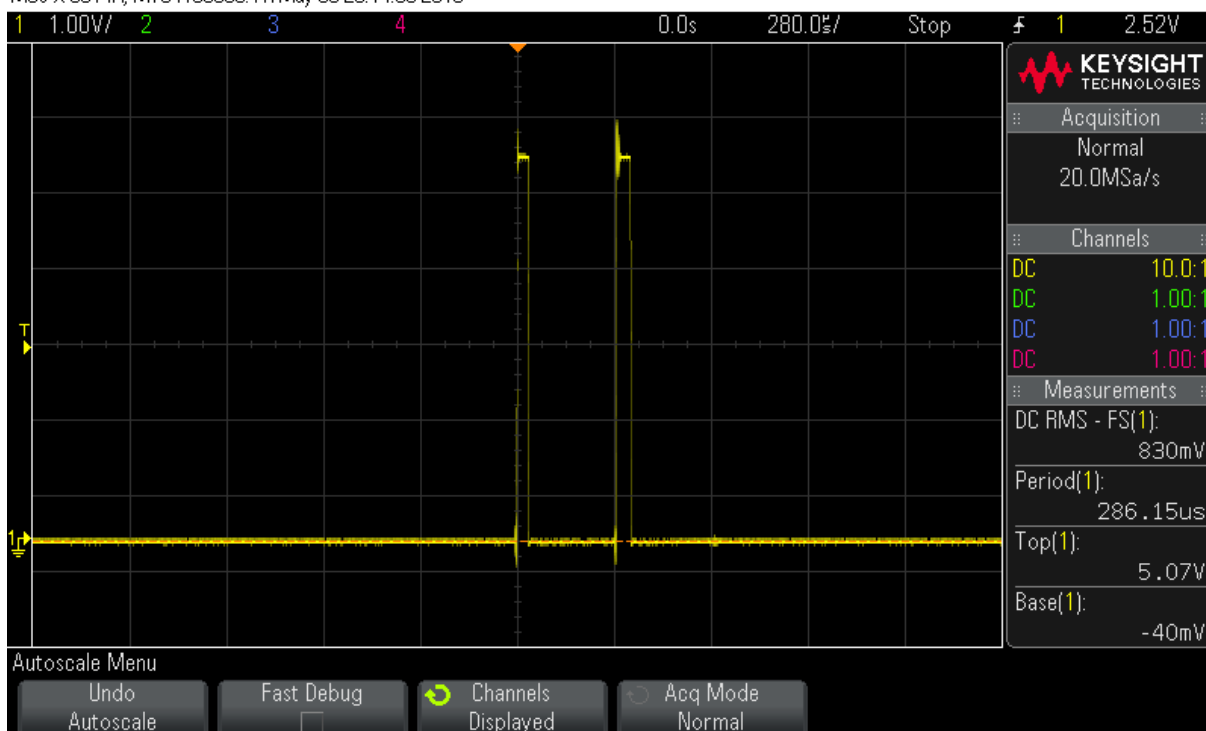
Si, for enkelhetens skyld, at gjennomsnittlig god energioverføring til væsken gir 2V utslag som respons. Det ble gjort en antagelse at ved å plassere transduseren nøyaktig på midtpunktet på grensesnittet, måles det cirka halve spenningsutslaget, 1V. Hvis øvre grense blir satt til å reagere på 1V utslag, så vil Schmitt-triggeren komme i høy tilstand nøyaktig på grensesnittet, og arduinoen kan varsle om væsknivå.

Den nedre grensen blir plassert med hensyn på formen til signalet skapt av likeretter og RC-leddet. Kondensatorens utladning er sakte, og strekker seg ut langt. Ved å plassere nedre grense langt nede, holdes Schmitt-triggeren i høy tilstand så lenge signalet fra transduseren varer, pluss cirka litt til grunnet utladningen til RC-leddet. Signalet fra transduseren varer omtrent 15μs. Ved å justere på nedre hystereserområde grense og RC ledd, får pulsen som blir generert en varighet på omtrent 40μs.

Pulsene som blir generert av Schmitt-triggeren skal representere en oppdaget hendelse. Kick og ekko/respons er de to gyldige hendelsene som da blir detektert. Signalet som kommer fra transduseren ligger på 0V helt til den sender ut og mottar signal. Kickene og responsene gir da klare signaler som vist i den første kartleggingsdelen. Resultatet av dette i kombinasjon med kretsen, er en klar digital puls med 40μs varighet for hvert kick, og for hver respons som er sterk nok.

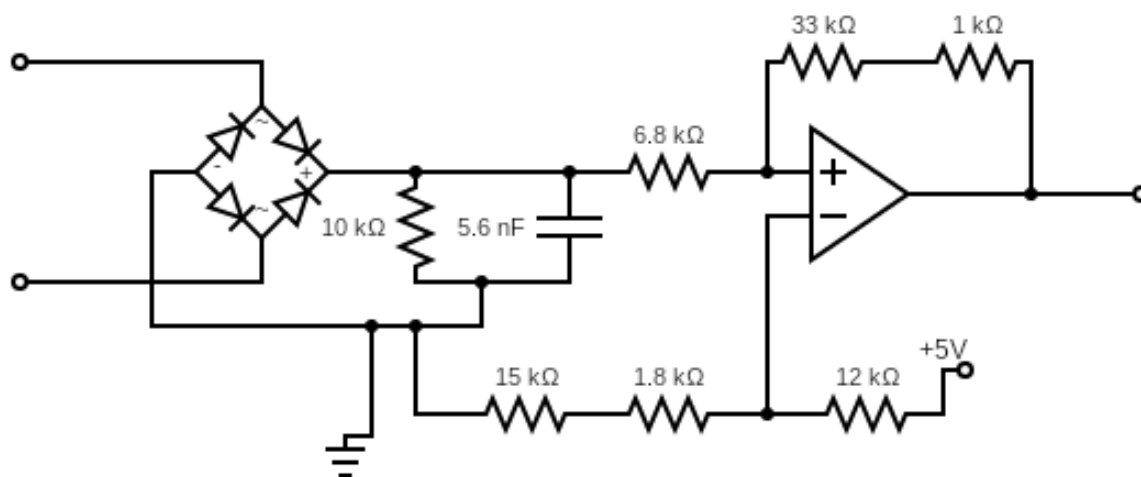
For målinger som bruker vann som væske i fatet, er tiden mellom kick og respons målt konsistent til å være 280μs. Siden øl er har større tetthet enn vann, vil denne tiden bli mindre med øl som væske.

MSO-X 3014A, MY54100395: Fri May 03 20:14:08 2019



Målt utgang av Schmitt-trigger ved måling under Væske/gass grensesnitt. Det første pulsen er fra kicket, og den andre pulsen er fra responsen/ekkoet. Cirka 280 μ s mellom hverandre.

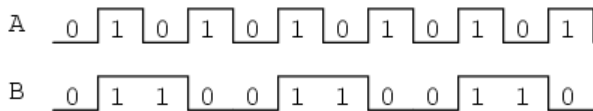
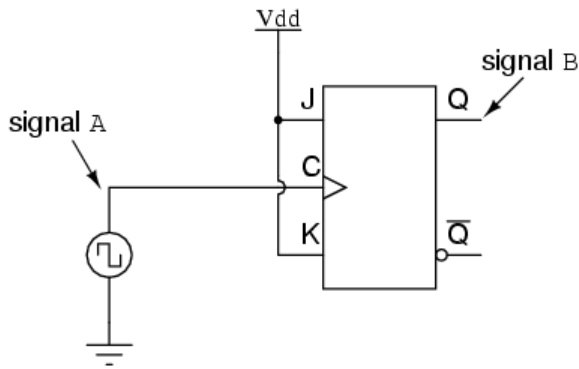
Hele denne eksterne elektronikkdelen kalles for «Transduser Puls Detektor» (TPD). Det er to motstander både i tilbakekoblingen og spenningsdelingen for å tilnærme utregnede verdier. De ekstra 1k Ω og 1.8k Ω motstandene er unødvendige støykilder, men de påvirker ikke utgangen på noe betydelig vis. Kretsen i sin helhet ser slik ut:



Transduser Puls Detektor (TPD)

5.1.2.1 JK-Vippe

Om dette blir for kort, og arduinoen ikke klarer å lese dette av raskt nok, kan én enkel JK-vippe (CD4027BE) [8] brukes. Denne blir fast koblet til «toggle-mode». Toggle-mode vil si at utgangen til vippen endrer digitalverdi etter hver høy digital flanke på klokkeinngangen sin. Da oppnås det ett digitalt nivå som varer fra kick til respons, som typisk har en tid cirka 280µs.



Kobling til JK [9]

Ved å koble J og K fast høy, og koble utgangen til TPD på klokkeinngangen, så bytter JK-vippen verdi bare når det kommer en puls fra triggeren. Dette gir arduinoen god tid til å lese av og registrere statusen til vippen. Arduinoen kan bestemme når en måling har begynt, når den er ferdig, om det ble registrert en respons og hvor lang tid den responsen tok.

5.2 Android mobilapplikasjon

For å presentere og lagre data ble det besluttet å lage en android mobilapplikasjon.

Mobilapplikasjonen er koblet opp mot en Thingspeak.com kanal, som er en gratis tjeneste for å lage servere. Det ble først foreslått å koble Arduino sin WiFi modul (ESP8266) direkte til Android applikasjonen via TCP/IP kommunikasjon uten server mellomledet. Grunnen til det ble valgt å implementere en server mellom Arduino sin WiFi modul og Android applikasjonen er bedret tilgjengelighet og brukervennlighet. Med hjelp av server kan en eller flere brukere hente måledata fra hvor som helst i verden. Det er heller ikke behov for at WiFi modulen og applikasjonen er på samme nett når data blir opplastet og nedlastet. Thingspeak.com tilbyr en restful API som bruker http forespørsler til å laste opp, laste ned og slette data, via http metodene get, put og delete. Dette er et stort pluss om applikasjonen på et senere tidspunkt skal videreutvikles. Arduino koden lager en http-forespørsel som inneholder «GET» metoden, API nøkkelen til ThingSpeak, et nummer som indikerer hvor dataen skal lagres, og verdien til tetthetsvariabelen som blir utregnet tidligere i koden. ThingSpeak registrerer data og klokkeslett for opplastingen av tetthets data, og lagrer det i et linjediagram med registrert dato og verdi. Når en ny måling på et senere tidspunkt lastes opp vil linjediagramme oppdatere seg, slik blir det visuelt enkelt å sammenlikne data og følge endringen i tetthet over tid. Prosjektet sin ThingSpeak kanal er ikke tilgjengelig for allmenheten, dette for å beskytte bruker og utvikle en form for personvern. Når Arduino lager http forespørselen må kanalen sin API nøkkel bli inkludert for å laste opp.

5.2.1 Arduinokode for WiFi modul

En ESP8266 WiFi modul må integreres i kretsen for å kunne kommunisere med ThingSpeak. ESP8266 trenger en forsyningsspenning på 3.3 volt for å drives. Arduino UNO kortet har en 3.3 volts utgangs pin, men det viser seg at strømstyrken på utgangen er for lav til å drive WiFi modulen. Derfor ble det kjøpt inn en ekstern strømforsyning med 9 volts batteri for å drive den. WiFi modulen kommuniserer via en RS232 seriell-port. Her oppsto på enda en utfordring, siden serieporten allerede brukes til å kommunisere med Arduino for datamaskinen, og den kan ikke støtte begge formål samtidig. For å løse dette inkludertes SoftwareSerial.h biblioteket i Android koden. Biblioteket gir mulighet til å emulere RS232 på to av Arduino kortet sine IO-pinner. I koden opprettes et objekt av klassen SoftwareSerial og gir IO-pinne 6 og 7 som argumenter for å opprette en ny serieport. IO-pinne 6 for mottak av seriell data, og 7 for å sende seriell data.

```
SoftwareSerial Serial1(6, 7);
```

For at WiFi modulen skal kunne koble seg opp mot internett trenger den nettverksnavn og tilhørende passord. For dette laget en «char» tabell for nettverksnavn og en for passord (bilde 1). Et objekt av klassen WiFiESPClient kalt client blir deklart

```
char ssid[] = "HUAWAIP20";           [Bilde 1]
char pass[] = "0498834316f1";
```

I koden sin «Setup» blir baudrate hastigheten til WiFi modulen satt til 9600 og baudrate hastigheten til datamaskinen satt til 115200 (bilde 2)

```
Serial.begin(115200);                [Bilde 2]
Serial1.begin(9600);
```

For å kontrollere at WiFi modulen er koblet opp rett, og er tilgjengelig brukes en funksjon som stopper underliggende kode å kjøre før modulen er tilgjengelig. Funksjonen skriver ut en melding til datamaskinen for å informere brukeren. Når WiFi.status() blir ulik WL_NO_SHIELD blir betingelsen i if-løkken usann og programmet forsetter (bilde 3).

```
if (WiFi.status() == WL_NO_SHIELD)
{
    Serial.println("Finn ingen WiFi-modul");
    while (true);
}
}                                     [Bilde 3]
```

For å kontrollere at WiFi modulen er koblet opp mot det aktuelle nettverket brukes en while- løkke. Når koden først kjøres er betingelsen i while-løkken sann, og koden inni vil loopes helt til betingelsen er usann. Status meldinger blir sendt til datamaskinen for å holde oss oppdatert. Variabelen Status

blir satt lik funksjonen `WiFi.begin(ssid, pass)` som har nettverksnavn og passord som argumenter. `Wifi.begin()` initialiserer WiFi bibliotekets nettverks innstillinger og returnerer statusen `WL_CONNECTED` for tilkoblet nettverk, og statusen `WL_IDLE_STATUS` når tilkoblingen mislykkes. Blir `WL_CONNECTED` returnert blir betingelsen til `while` løkken usann, og statusmelding «tilkobla» blir skrevet til datamaskinen (bilde 4).

```
while (status != WL_CONNECTED)
{
    Serial.print("koblar til nettverk: ");
    Serial.println(ssid);
    status = WiFi.begin(ssid, pass);
}
Serial.println("tilkobla");
```

[Bilde 4]

Metoden `printWifiStatus()` skriver ut nettverksnavn og IP adresse på den serielle overvåkeren. Denne metoden er ikke nødvendig for brukeren og er kun for debugging av koden. Når tilkoblingen til nettverket er fullført og tetthets data er målt og utregnet kalles funksjonen `sendThingspeak()` som tar inn tetthets data som argument. For å kunne sende http forespørselen må funksjonen `connectSSL` brukes og returnere sann. Funksjonen tar inn `ThingSpeak` sin URL som ligger lagret i variabelen `server`, og utgangsport nummer som argument. Statusmeldinger til datamaskinen blir skrevet ut, etterfulgt av selve http-forespørselen. Objektet `client` skriver meldinger serielt til `ThingSpeak` via `println()` funksjonen. Http metoden `Get` blir satt i begynnelsen av melding for å indikere at det skal lastes opp data. Etterfulgt av API nøkkelen, lokasjonen hvor dataen skal lagres, og selve verdien. For å avslutte kommunikasjonen med `ThingSpeak` sendes en tom. Etter opplasting aktiveres metoden `Client.available()` som skriver ut en melding fra `ThingSpeak` på den serielle overvåkeren. Denne funksjonen er kun for debugging av programmet og kommer ikke til å være en del av koden som levers til brukeren (bilde 5)

```
void sendThingspeak(long value)
{
    if (client.connectSSL(server, 443))
    {
        Serial.print("kobla til ");
        Serial.println(server);
        client.println("GET /update?api_key=" + String(APIKEY) + "&field1=" + long(value) + " HTTP/1.1");
        client.println();
        Serial.println("sendt til thingspeak.com");
    }
}
```

[Bilde 5]

5.2.2 Android studios

For best mulig grafisk presentasjon av data ble det besluttet å inkludere `ThingSpeak` sine egne biblioteker. `ThingSpeakChannel` tar seg av hele http forespørselen for å laste ned dataen.

```
tsChannel = new ThingSpeakChannel( channelId: 767809, readApiKey: "8CETFAU2Y0BZDPS7");
```

[Bilde 6]

ThingSpeakChart er et bibliotek for å vise ulike 2D plot av nedlastet data. Brukergrensesnittet har ingen visuell knapper som brukeren kan aktivere, og det er heller ikke noe behov for det. Brukeren skal kun kunne se et 2D Linje plott der x aksene representerer tid, og y aksene representerer tetthet. Et objekt av ThingSpeakChannel kalt tsChannel blir initialisert med kanal id og api nøkkel for lesing som argumenter (bilde 6)

Objektet tsChannel har en metode som heter setChannelFeedUpdateListner som oppdateres automatisk når nye verdier blir lastet opp til ThingSpeak. Et objekt av ThingSpeakChart kalt tsChart blir initialisert med kanal id og feltet data skal lastes ned fra. Hensikten med dette objekt er formatering av dataen som skal presenteres. Dette inkluderer farge på grafen, navn på aksene, antall verdier som skal inkluderes, format på data og klokkeslett. For å holde brukergrensesnittet oppdatert bruker metode setListener på tsChart objektet (bilde 7).

```
tsChart = new ThingSpeakLineChart( channelId: 767809, fieldId: 1 );
tsChart.setNumberOfEntries(17);
tsChart.setValueAxisLabelInterval(300);
tsChart.setFilled(true);
tsChart.setYAxisName("Density");
tsChart.setXAxisName("Date");
tsChart.useSpline(true);
tsChart.setDateAxisLabelFormat("HH:mm");
tsChart.setLineColor(Color.parseColor( colorString: "#7a1818" ));
tsChart.setAxisColor(Color.parseColor( colorString: "#70afea" ));
```

[Bilde 7]

5.2.3 ThingSpeak

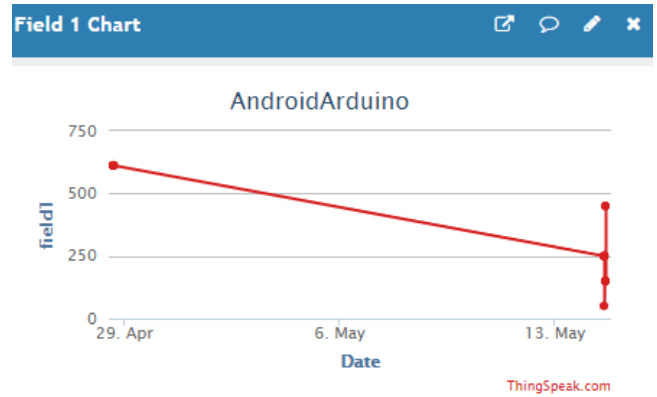
ThingSpeak er en «open-source internett of things» applikasjon med api for å lagre og hente data fra elektronikk som er koblet opp mot internett via http protokollen. Siden ThingSpeak er en gratis tjeneste for å opprette en egendefinert server kanal trengs det ikke å programmere i SQL. Dataen lagres i et felt kalt feild1 som målerverdier lastes ned og opp til. Kanalen heter ArduinoAndroid. Alle kanaler har unike api nøkler for lesing og skriving. Nøkkelen for å laste ned data til Android er vist under (bilde 8). Skal da løsningen videreutvikles har ThingSpeak egen kalender funksjon, alarm varsling og muligheten til å analysere dataen i Matlab.

Read API Keys

Key	<input type="text" value="8CETFAU2Y0BZDPS7"/>
Note	<input type="text"/>

[Bilde 8]

Det er mulig for brukeren å sjekke måledataen direkte på nettsiden til ThingSpeak. Et bilde av feltet «field1» er vist under (bilde 9)



[Bilde 9]

5.3 Arduino kode

Datatyper og variabler

- `int btn = 5;`
 - Navn for IO-pinne 5
- `int teller = 0;`
 - For å angi posisjon til tabellen
- `int JK = 4;`
 - for IO-pinne 4
- `int Buzzer = 8;`
 - for IO-pinne 8
- `long initial_Pulse;`
 - for start initiale puls teller
- `long echo;`
 - for registrert ekko teller
- `long timelapse;`
 - for differanse mellom `initial_Pulse` og `echo`
- `long mySensVals [10] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};`
 - for tabell med 10 plasser for godkjente måleverdier
- `long tetthet;`
 - for gjennomsnittet av dataen fra `mySensVals` som skal lastes opp
- `long lastConnectionTime = 0;`
 - for sist registrert opplasting
- `long postingIntervall = 60000;`
 - for hvor ofte det ønsker å laste opp
- `boolean flagg = false;`
 - for å forsikre at programmet ikke registrer falske verdier
- `boolean Upload = false;`
 - for å starte opplasting til ThingSpeak når knappen er trykket
- `boolean btn_WiFi = digitalRead(knapp);`
 - for å kontrollere at knappen er trykket
- `volatile boolean Rising = false;`
 - for å kontrollere at en stingende flanke er registrert
- `volatile boolean Falling = false;`
 - for å kontrollere at en fallende flanke er registrert

Innganger og utganger

- IO-pinne 8: Utgangssignalet fra den aktive piezo buzzeren.
- IO-pinne 5: Inngangssignalet fra grønn trykknapp.
- IO-pinne 4: Inngangssignalet fra JK vippen.
- IO-pinne 3: Inngangssignalet fra attachInterrupt funksjonen for fallende flanke.
- IO-pinne 2: Inngangssignalet fra attachInterrupt funksjonen for stigende flanke.

Void setup og loop

I setup blir IO-pinnene deklart og tilkobling til WiFi blir opprettet. Void loop repeterer seg selv håndterer resten av koden med unntak av fire andre funksjoner som blir referert til fra void loop.

Disse er:

- void SensorFall ()
- void SensorRise ()
- void printWifiStatus ()
- void sendThingspeak (long tetthet)

5.3.1 Koden

Arduino har en innebygget telle funksjon som har en oppløsning på en halv mikro sekund, noe som er en god nok indikator på tetthetsmål for brukeren. Ved hjelp av Interrupt blir målingene mer presise, dette har blitt sjekket opp mot Oscilloskopet. Interrupt stopper koden som blir kjørt når den registrerer et stigende eller fallende fra JK vippen. Koden vil da umiddelbart kjøre funksjonen som er relatert til det enkelte interruptet. Her er det viktig at minst mulig kode skjer inne i interrupt funksjonen siden det har en påvirkning på tiden koden bruker på å utføres. Interrupt funksjonen kalt SensorRise setter en boolsk variabel kalt Rising til sann, og setter den boolske variabelen Falling til usann. Det finnes en tilsvarende funksjon som heter SensorFall som gjør det motsatte, Rising blir usann og Falling blir sann. Utgangssignalet fra JK vippen går høy ide en ultralyd puls blir avfyrt, dette plukkes opp av interruptet for stigende flanke og SensorRise funksjonen starter og setter Rising til sann, og Falling blir satt til usann. Koden forsetter deretter ekskviteringen i Void loop. For at koden skal starte timeren initial_Pulse må

Rising være satt til sann, som er betingelsen i første if-løkken (Bilde 10).

Når betingelsen er oppfylt startes en teller. En boolsk variabelt kalt flagg er i starten av programmet initialisert til usann, denne settes til sann inne i if-løkken. Dette for å forsikre oss om at den initiale pulsen fra transduseren er registrert først.

```
if (Rising == true) {
    initial_Pulse = micros();
    flagg = true;
}
```

[Bilde 10]

TPD utvider det initiale signalet til digital høy i 40ms. Etter dette vil signalet fra JK vippen bli digital lav før et eventuelt ekko registreres. Det må registres en digital lav verdi fra JK vippen mellom den initiale pulsen og ekkoet for at resten av koden skal utføres, og attachInterrupt for fallende flanke må registrere en endring i signal fra lav til høy for å aktivere funksjonen SensorFall som setter Falling til sann og Rising til Usann. Alt dette blir testet i en ny if-løkke (bilde 11).

```
if (flagg && (Falling == true)) {           [Bilde 11]
    flagg = false;
}
```

Når et ekko returneres og registreres av transduseren oppfatter attachInterrupt på IO-pinne 2 hendelsen, og SensorRise funksjonen starter på nytt. En if-løkke tester om betingelsen Rising er satt til sann, og timeren Echo starter. Timelapse er summen av Initial_Pulse minus echo (bilde 12). Dette gir oss en tidsintervall fra den Initiale Pulsen fra transduseren til ekkoet returnerer. Som nevnt tidligere er tiden til variablene timelapse en indikator på tettheten til væsken. Under ølets brygge prosess vil tettheten til ølet minske, og lyden vil bevege seg tregere i væsken. Dette resulterer i at verdien til timelapse er mindre ved starten av bryggingen enn ved slutten. For å finne væske nivået på innsiden av tanken er det ikke nødvendig med tidsmålinger. Så lenge ekkoet blir returnert vil arduinoen kunne identifisere væske nivået, dette grunnet den store forskjellen i akustisk impedans mellom stål og luft.

```
if (Rising == true) {                       [Bilde 12]
    long echo = micros();
    long timelapse = initial_Pulse - echo;
```

For å forsikre oss om at tidsintervall verdien timelapse lagrer ikke kommer fra en ny puls tester en dette i en ny if-løkke (bilde 13). Corneliusfatet som ble brukt for testing har en innvendig diameter på 30 cm. Benyttes en annen tank med større innvendig diameter vil tiden mellom den initiale pulsen og ekkoet øke. Øvre lovlig tidsgrense ble økt for at applikasjonen skal fungere på tanker med litt større innvendig diameter. Fra testingen med vann i Corneliusfatet vet en at ekkoet blir registrert nøyaktig 280 mikro sekunder etter den initiale pulsen blir sendt ut. For å indikere at væskegrensesnitt er funnet, gir den aktive piezo-elektriske buzzeren fra seg en lav-volum lydvarsling.

```
if ((0 < timelapse) && (timelapse < 500)) {
    mySensVals[teller] = timelapse;
    teller++;
    digitalWrite(Buzzer, HIGH);           [Bilde 13]
}
```

En tabell variabel kalt mySensVals og variabelen teller blir deklart, begge av datatypen long. Variabelen teller blir satt lik 0 i begynnelsen av programmet. Hensikten til teller variabelen er å indikere hvilken posisjon tetthets i tabellen tids verdi skal lagres i. Første gang tids verdien til timelapse er innenfor det godkjente intervallet blir verdien lagret i 0 posisjon i tabellen og variabelen teller blir inkrementert. Grunnen til at det ble valgt å opprette tabellen mySensVals i stedet for å sende en enkelt tidsmåling over WiFi, er at tidsmålingen i gjennomsnitt varierer med 5-20 mikro sekunder. Når 10 unike måleverdier er blitt lagret i tabellen tas gjennomsnittet av alle verdiene lagres i variabelen sum og sendes til ThingSpeak over WiFi. For å laste opp dataen over WiFi må brukeren selv trykke på den grønne knappen på utsiden av boksen. Knappen er koblet til IO-pinne fem som en input. Når Knappen trykkes settes den boolske variabelen btn_WiFi til digital høy. En if-løkke øverst i void loop () sjekker om knappen er trykket, og setter en annen boolsk variable Upload til sann. Når Upload er satt til sann og det er mer enn ett minutt siden sist WiFi opplasting til ThingSpeak startes Arduino funksjonen sendThingspeak () med gjennomsnittet av måledataen i tabellen som argument (bilde 15).

```
if ((millis() - lastConnectionTime > postingIntervall) && (Upload == true))
{
  sendThingspeak(tetthet);
  lastConnectionTime = millis();
}
```

[Bilde 15]

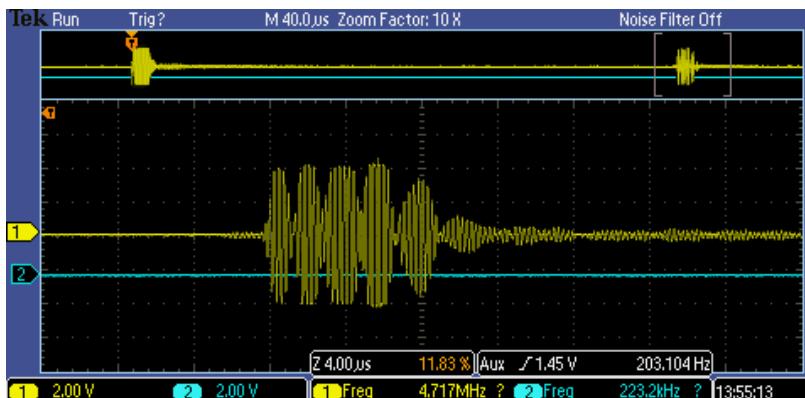
Når verdiene er sendt over WiFi blir Tabellen mySensVals posisjoner overskrivet og de tidligere verdiene settes lik null. Variabelen teller settes også lik null. Puls raten på DRP300 er satt til den laveste innstillingen, og sender en puls hver 10 millisekund. Det er ikke interessant for brukeren å få gjennomsnittet av tidsverdier lastet opp til databasen i underkant av hvert sekund, derfor er det satt en begrensning på hvor ofte data kan lastes opp til ThingSpeak. Når data har blitt sendt til ThingSpeak vil koden hindre at ny data blir lastet opp før det har gått ett minutt. Programmet blir uresponsivt når WiFi modulen sender data til ThingSpeak. Den aktive Buzzeren lager lyd kontinuerlig helt til opplastingen er fullført.

6 Testing

Utover prototyperealisering måtte det foretas testing for å finne feil og gjøre de rette endringene. Disse testene gikk ut på prøving og feiling for å komme frem til en løsning som ble vurdert til å være tilstrekkelig. Mye av testingen gikk ut på å teste hele oppsettet i fullskala, med bruk av måleutstyr for å feilsøke. Oscilloskop ble brukt for kretsen, og datamaskin ble brukt for å overvåke og debugge arduinoen. Dette kan gjøres på grunn av at dette er et lite system som ikke er farlig å jobbe med, verken for mennesket sin del, eller for utstyret sin del. All form for høyere spenning ligger igjen inni DPR300, som filtrerer vekk den høye spenningen fra signalet.

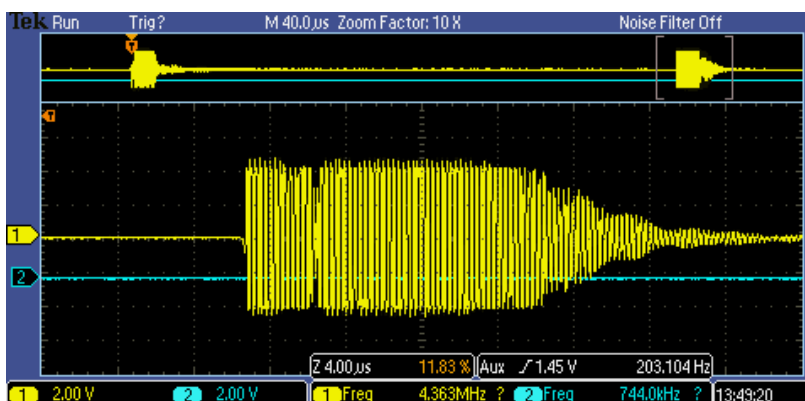
Alle tilgjengelige transdusere ble testet med forskjellige innstillinger på DPR300, for å se hvilken transduser med hvilke innstillinger som gav best respons. Disse ble testet på Corneliusfat med et godt lag med vaselin smurt mellom transduseren og tanken. Det var denne serien med tester som ledet til bruken av 5MHz immersjonstrasduseren fra Olympus, ettersom den hadde den beste responsen. Skjermbilde fra alle transduser testene kan sees i appendiks D.1.

Responsen ble også testet ved forskjellige posisjoner på tanken. Inni Kornelius fatet er det et metallrør som er brukt til å hente væsken ut av fatet. Dette røret er plassert rett ved en sveis som går vertikalt opp hele siden på fatet. For å teste om dette hadde noen påvirkning ble røret fjernet, og responsen ble målt direkte på sveisen og område rundt. Resultatet var at responsen ble godt svekket i område rundt sveisen. Responsen på sveisen så slik ut:



Responsmåling over sveis

Sammenliknet med full respons når en måler 90° vekk fra sveisen på fatet:



Responsmåling fullrespons

Om det blir målt på et område som pleier å gi god respons, for så å plassere røret inntil veggen til stålfatet foran transduseren mens den måler, flater signalet helt ut. Når røret ligger i fatet pleier den ikke å ligge inntil veggen, så dette er ikke så stort problem. Det viktigste er å unngå sveisen. Hvis man tenker at fatet er en klokke, der sveisen er plassert på klokken 12, burde en foreta målinger på klokken 3 og klokken 9 for best resultat.

Under testing av TPD ble det oppdaget et konstant høyt nivå på 5V på utgangen av triggeren. Dette hadde sin årsak i at spenningsresponsen ble større enn forventet fra den første transduser respons testen. Dette holdt det generelle spenningsnivået langt over den originale øvre hysteresegrensen. Det generelle spenningsnivået ble målt til å ligge på 1.77V. Øvre hysteresegrense var satt til 1.35V. Løsningen til dette var å endre hysteresen til 3V nedre grense og 4V øvre grense. Dette løste problemet på flere måter. Transduseren sin spenningsrespons var fortsatt like sterk, men basespenningen ble redusert til omtrent 200mV.

JK vippe ble originalt tatt med i løsningen, og var med i den første kretsdesign og oppkobling. JK trinnet hadde ingen problemer med sin grunnfunksjon. Under testing av systemet sammen med arduinoen, viste det seg at det ble mer problemer enn goder med å bruke JK vippen. Vippen må resettes ved visse punkt for å sørge for at arduinoen kan lese av informasjonen fra vippen riktig. Hvis JK kom i utakt, ville hele bølgeformen til utgangen av TPD inverteres. Informasjonen blir da uleselig for arduinoen. Å holde kontroll på reset, samt kunne lese av riktige tider fra TPD, ble sett på som en upraktisk metode. JK ble derfor sløffet i testing fasen.

Løsning ble da å bare bruke TPD og koble denne direkte opp til arduinoen. TPD sine utsendte pulser måtte da gjøres bredere. Kondensatoren ble endret til 5.6nF for å øke utladningstid, noe som gav 40µs brede pulser. Dette er tilstrekkelig for arduinoen som trenger i underkant av 5µs for å foreta en digital avlesning. Om en vil gjøre pulsen fra TPD bredere, kan nedre hysteresegrense senkes, men dette er ikke nødvendig.

Testing av arduinoen ble gjort med prøv-og-feil metoden. Forskjellige løsninger og koder ble forsøkt, med forskjellig vinkling til problemstillingen. Det var denne testfasen som gjorde at JK vippen ble sløffet. Debuggingen ble gjort med skrivefunksjoner i koden, som skrev tekst til seriell vinduet. Dette gir oss en indikator på hvor langt koden er kommet, hvor den stopper opp og hvor ofte koden havner i visse løkker.

For å sjekke nøyaktighet til nivå målingen, ble det brukt et målebånd som strakte seg fra toppen av fatet og ned til vannivået. Denne målingen ble da merket med tusj på utsiden av fatet. Nivået på denne streken ble sammenliknet med nivået der måleren sluttet å varsle om væske. Målingen var godt innenfor ±1cm av denne streken. Måling av den reelle tiden lyden tar fra transduseren, til andre siden av fatet og tilbake, ble sett på oscilloskopet under testing av TPD. Disse 280µs dukket opp i hvert eneste trinn i TPD helt konsistent. Verdien på tidsmåling fra arduinoen var vanligvis cirka 40µs mindre enn dette. Dette kan være fordi arduinoen starter telleren ved negativ flanke til kicket, og stopper ved positiv flanke ved respons. Én puls tid går da tapt, som vanligvis ligger på 40µs.

7 Diskusjon

7.1 Fremdrift

Det ble satt noen få større tidsfrister for når noen større faser skulle vær ferdig. For eksempel fristen for de siste store bestillingene av deler, og en frist for når grunn-design skulle være ferdig. Disse ble holdt ganske bra. Halvveis i semesteret, ved en midtveis presentasjonen av fremgangen til prosjektet, var teamet én uke bak skjema. Målet for å få prototypen operasjonell var 9 mai. Prototypen ble fullført 5 dager senere, den 14 mai. Underveis ble designavgjørelser og veiledermøter dokumentert og arbeidstimer estimert.

7.2 Endringer i planen

Mye av tiden gikk til grunnleggende design, finne ut hvordan å håndtere problemstillingen og hvilken innfallsvinkel som var best. Mye av arbeidet gikk til løsninger som ble totalt sløpfet når fordelene og ulempene ble veiet, eller når et stort problem med løsningen ble oppdaget.

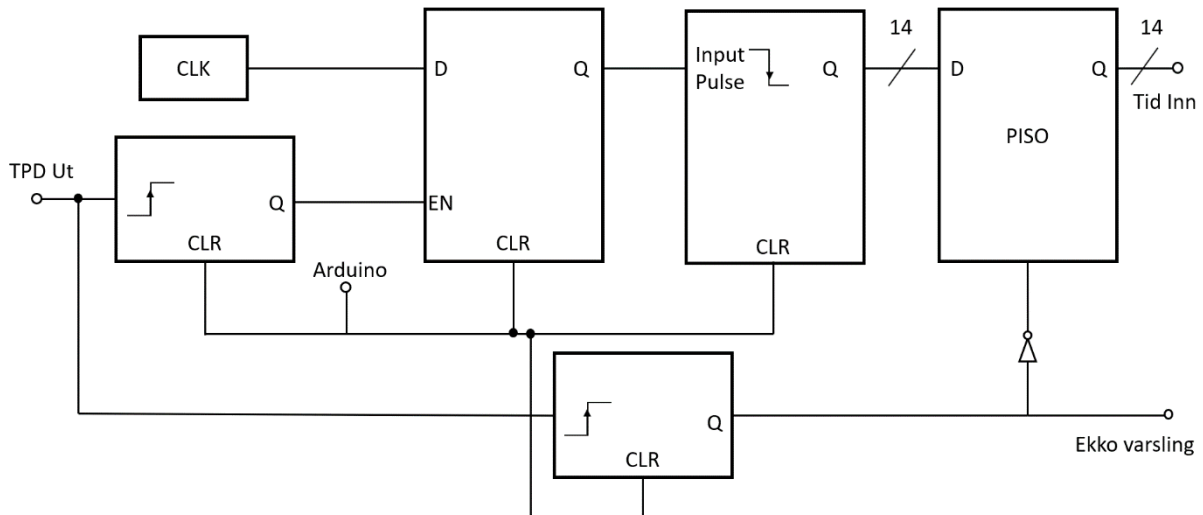
Endringer i design ble i tillegg gjort under implementeringsprosessen. Ekstern logikk-løsning ble sløpfet når delene som var bestilt inn ble sett, og det ble oppdaget hvor upraktisk hele oppsettet var. Det ble da bestemt å bare bruke TPD og JK vippe. Senere ble JK vippet sløpfet, da det bare ble problematisk for arduinokoden.

Det ble også bestemt på et punkt at det var nok arbeidsmengde for to personer å få oppsettet til å fungere med bruk av DPR300. Å bygge egen transduserdriver ble da nedprioritert. I stedet for å bygge denne, ble det bestemt å gjøre grunnarbeidet til denne delen. Det ble utforsket nøye hvordan denne eventuelt kunne bli bygget, så designet kunne tas med i denne rapporten.

7.3 Forslag

7.3.1 Logikk og FPGA

Det ble forsøkt å bruke ekstern logikk for å tolke pulsene fra TPD for å bestemme om det var mottatt en respons etter kicket og hvor lang tid det gikk før responsen ble mottatt. Denne benyttet seg av JK vipper i toggle modus, en D-latch, en puls teller som kunne telle binært opptil 14 bit, et PISO register og en egen klokkegenerator. Denne kretsen ble laget slik at den kunne telle kontinuerlig i 3.27ms fra kick, med en oppløsning på 0.2 μ s. Denne skal også kunne måle på tanker med diameter på 2.42m, hvis en ser vekk fra svekkelse i lydintensitet over propageringsdistanse.



Logikk konsept skisse

Ideen bak skissen er at når en puls blir gitt inn på inngangen, vil begge toggle-vippene få høy utgangsverdi. For D-latchen sin del, betyr det at klokken (CLK) sin verdi konstant går videre til telleren, som teller for hver negativ pulsflanke. Hvis det ikke er noe respons, vil «Ekko detect» sin verdi være høy og arduinoen vil tolke det som at den ikke måler på væsknivå, og da resette logikken for sikkerhetens skyld.

Om det kommer en respons, så vil toggle-vippene gå ned til lav verdi, dette har flere effekter. Først og fremst vil D-latchen skille CLK og telleren. Enable inngangen til PISO registeret gå fra lav til høy, og registeret vil da hente inn data fra telleren. I tillegg vil «Ekko detect» være lav og arduinoen tolker det som væsknivå måling. Arduinoen kan da hente inn dataen fra registeret serielt. Deretter vil arduinoen resette logikken.

For å telle tiden til ekkoresponsen brukes det en standard pulsteller. Denne kan variere med hvor mange bit den kan telle. En annen ting som kan variere er klokkefrekvensen som blir brukt til å få telleren til å telle. Valget av klokkefrekvens er vesentlig. Om det blir valgt en høy klokke og lavt antall bit, kommer telleren til å telle seg til maks verdi og resette seg før ekkoet har nådd tilbake til transduseren. Blir det valgt lav frekvens, blir det dårligere oppløsningen i tellingen. Med vann i stålfatet ble det målt en ekkotid på $280\mu\text{s}$. Lavere frekvenser gir høyere periodetid, og en periodetid på for eksempel $20\mu\text{s}$ gir oss maks 14 klokkeperioder å telle på innen $280\mu\text{s}$. Dette langt ifra nøyaktig nok.

Først ble det bestemt at 200ns , 5 klokkeperioder hvert mikrosekund er kjempegodt, dette tilsvarer en 5MHz klokke. Dette påvirker minimum krav til antall bit på telleren, som deretter påvirker maksimum diameter til tank som kan måles på.

For å finne utav hvor lenge en teller kan telle før den resetter seg selv, kan det regnes slik:

$$\text{Periode Tid} * 2^{\text{antall bit}} = \text{Telle tid}$$

Dette er tiden som er til disposisjon. Lyden er tregere i ølet enn i vann, så det kan beregnes maksimum anbefalt diameter på tanken som en kan måle på. Lydens fart i vann er omtrent 1480m/s. Derfor:

$$\frac{\text{Telle tid} * 1480\text{m/s}}{2} = \text{Maks diameter på tank}$$

Ved bruk av en 14-bits teller og 5 MHz klokke, så kan tanken teoretisk sett ha en diameter på 2.42m. Sannsynligvis vil den reelle diameteren vær mye mindre på grunn av svekkelse i lydintensitet over den distansen.

7.3.1.1 Arduino MKR vidor 4000

Dette konseptet vil føre til nøyaktig tidsmåling, fordi telleren har god oppløsning og den benytter seg av hardware for å holde styr på tiden. Dette konseptet kan programmeres på en FPGA. Med denne løsningen er det enklere å rette på eventuelle feil, dersom det ikke skal konstrueres en krets med bruk av individuelle innkjøpte deler. En kan bruke et eget FPGA brett, og koble det man trenger videre til arduinoen for dataprosessering. Man kan også velge å bruke «Arduino MKR vidor 4000» [10], som er en arduino med innebygget FPGA modul. Arduino vidor har også til fordel at den er mindre enn de fleste FPGA kortene. Den er også mindre enn arduino UNO. Så dette vil være et kjempeflott valg videre.

7.3.2 Transduserdrift

Målet med å lage en egen transduserdriver, er å imitere én innstilling fra DPR300 som fungerer best. Grunnen er helt enkelt at den tar mindre plass og krever ikke mye forsyningspenning, og kan derfor drives av et lite batteri. I dette tilfelle med 5 MHz immersjonstrasduser fra Olympus, ble det brukt følgende innstillinger:

PRF: 1

Pulse amplitude: 11

Pulse energy: Low Z, 4

Damping:

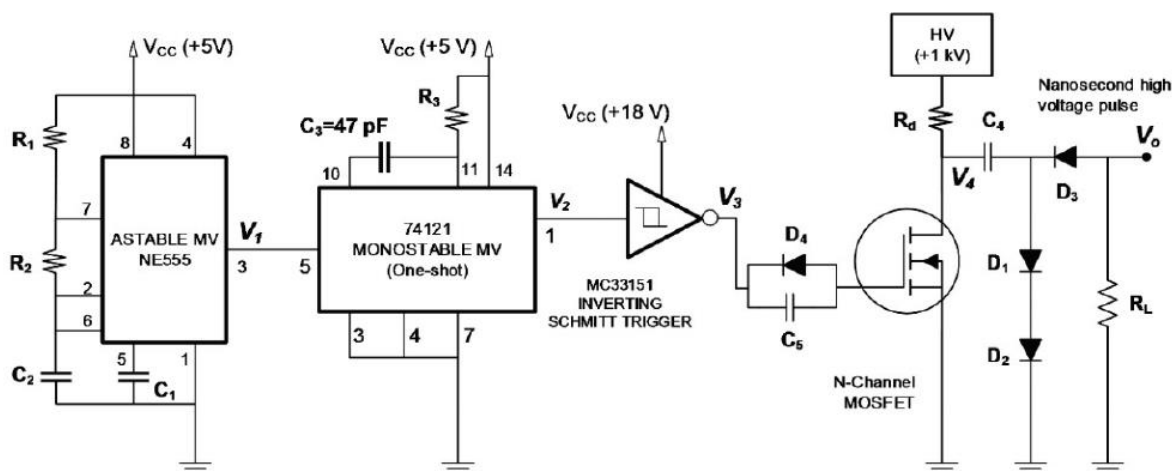
Gain: 55db

LP filter: 35

HP filter: OUT

En må undersøke hva disse innstillingene egentlig betyr, så replikere dette. For eksempel gir PRF rate 1 et mellomrom på 10ms mellom hvert kick. PRF rate 2 gir 5ms mellom hvert kick. Dette kan en bruke oscilloskop til for å kartlegge helt. Datablad for DPR300 finner en på nettet [11]. Det er anbefalt å ha lav PRF, grunnet at en da gir transduseren tid til å gi fra seg all energi før den byr seg på ny puls. I tillegg lar man da kondensatoren i TPD lade ut.

Et godt forslag til krets for transduserdrift er gitt av «AIP Conference Proceedings 1927, 030018 (2018)» [3]. Den ser slik ut:



Denne kretsen består av en astabil og monostabil multivibrator for timing av pulsen. Schmitt-triggeren er egnet til å drive MOSFET transistoren som skal tappe fra høyspenningskilden. Timingene i korte trekk fungerer slik: Den astabile MV blir justert slik at den imiterer PRF, hvor ofte en vil ha kick. Bredden på pulsene skal være så korte som mulig, fordi det er bare viktig med dens positive flanke. Den monostabile MV blir da dimensjonert til å justere pulsbredden / utladningstiden til kicket. Kicket blir en negativ spik som eksiterer transduseren og skaper en lydimpuls. Dokument med detalj om hvordan å velge de rette verdiene på komponentene, og hvordan kretsen fungerer finner du lenken til i referanselisten [3].

For at denne skal fungere på et lite batteri og være en håndholdt enhet, trenger man en løsning som kan konvertere den lille batteri spenningen om til en høyere spenning, 300V-500V DC. En modul som kan fungere er EMCO HV DC-DC konvertere i A serien [4]. EMCO A10 kan være et godt forslag. Disse er på størrelsen av en seks siders terning, og kan gjøre den ønskede transformasjonen. Det er usikkert om den kan levere effekten, og dermed energimengden som trengs, på den korte tiden for å drive transduseren riktig.

Etter dette må en også trolig legge til en eller annen form for beskyttelse mod den høye spenningen før TPD for å verne elektronikken, arduino og op-amp. Men all data om utsendt lydsignal og respons må fortsatt slippes forbi, akkurat som DPR300 gjør. Ved bruk av oscilloskop for å måle respons på transduseren drevet av DPR300, kan man ikke se noen antydning til den høye spenningen, den er fjernet, men en ser effekten den hadde på transduseren.

DPR300 er også utstyrt med en forsterker, LP og HP filter. Forsterkeren må lages i dette prosjektets tilfelle med 55dB forsterkning, før TPD. HP og LP filter må vurderes om er nødvendig.

7.3.3 Mobilapplikasjons utvidelse

Til nå er mobilapplikasjonen kun tilgjengelig for brukere med mobiltelefoner som kjører Android sitt operative system. En mobilapplikasjon som også er tilgjengelig for Apple er en relevant utvidelse. Mobilapplikasjonen viser til nå endring i tetthet grafisk, men den gir ikke en pekepinn på hvor mange dager som gjenstår før brygget er ferdig. For å kunne estimere antall gjenstående dager er det behov for å samle inn tetthetsdata over en lengre periode. Android applikasjonen gir til nå kun grafisk tetthetsdata for en Cornelius fat om gangen. Om brukeren ønsker å måle på flere tanker samtidig kan flere felt legges til i ThingSpeak og kildekoden til Arduino og Android Studios må endres. Dette er litt tungvint, så en applikasjon som tilbyr grafisk tetthetsdata på opptil flere Cornelius fat samtidig er en aktuell utvidelse.

8 Konklusjon

Dette er altså en fungerende løsning for behandling av tetthetsdata ved bruk av tidsmålinger, og et veldig godt system for varslings av væsknivå. Løsningen består av enkel og billig analog elektronikk, én arduino med WiFi modul og en 5MHz immersjons transduser. Systemet er testet, nøyaktigheten på nivåmåleren er målt, og det er funnet at den ligger godt innenfor $\pm 1\text{cm}$ kravet. Løsningen tar også liten plass, og den kan med enkle grep pakkes tettere sammen. Ved eventuell bruk av Arduino vidor, som har FPGA, mikrokontroller og WiFi modul innebygd i ett kort, kan løsningen bli enda mer kompakt. I det tilfelle bare trenger vidoren, TPD, transduserdriver og batteri.

Tidsmålingene kunne vært mer nøyaktig. Dette kan forbedres enten ved bruk av FPGA i stedet for arduino mikrokontrollere, forbedringer på arduino koden, eller ved vidoren som er en kombinasjon av FPGA og mikrokontroller. Maksimal tankbredde med denne løsningen er uvisst. Kodemessig kan tiden programmet har til å vente på responsen utvides. Dermed utvides maksimal tankbredde. Den egentlige begrensingen ligger i lydintensitet over en viss reisedistanse. Det er usikkert hvor stor distanse en trenger for at TPD ikke skal kunne registrere responsen. Denne løsningen kan også trolig svikte dersom stålfatet er skadet eller deformert på noe vis.

Hele systemet må kunne drives av et lite batteri, og transduserdriver må legges til for å fullstendig oppfylle kravspesifikasjonen. og så få hele systemet batteridrevet. Videre foreslås det å undersøke muligheten for forskjellige kontaktmedium i stedet for vaselin mellom transduser og stålfatet.

Løsningen i selv tilfredsstillende de fleste kravene satt for oppgaven. Det er et problem som foreløpig stopper nivåmåleren fra å komme ut på privatmarkedet. Det trengs en transduser som må oppfylle to krav. Det ene er at ultralyden må kunne propagere gjennom metall/væske, og det andre kravet er en frekvens som er kompatibel med tykkelsen av veggen i stålfatet. Disse to kravene øker prisen på transduseren i stor grad. En god transduser som oppfyller kravene til å fungere oppimot et væskefylt Corneliusfat er cirka kr 4000,-. Den endelige prisen på produktet kommer sannsynligvis til å bli altfor høy for den vanlige hjemmebryggeren til å ville kjøpe produktet.

9 Referanser

- [1 [Internett]. Available: https://en.wikipedia.org/wiki/Cornelius_keg.
]
- [2 [Internett]. Available: <https://www.arduino.cc/>. [Funnet 24 05 2019].
]
- [3 S. N. S. S. a. B. M. W. Darmawan Hidayat, «MOSFET-based high voltage short pulse generator for ultrasonic transducer excitation,» AIP Conference Proceedings 1927, 030018 (2018), 09 02 2018. [Internett]. Available: <https://doi.org/10.1063/1.5021211>. [Funnet 2019 05 24].
- [4 «digikey.com,» Power xp, [Internett]. Available: <https://www.digikey.com/en/product-highlight/x/xp-power/dcdc-converters-emco?fbclid=IwAR3MlfigLnWs2nZCA9Xmbp8jxrkreRjqoHg9OG08zPo91QirfxnncDeZXLg>. [Funnet 24 05 2019].
- [5 [Internett]. Available: <https://www.elprocus.com/bridge-rectifier-basics-application/>. [Funnet 1
] Mai 2019].
- [6 Texas Instruments, *Datasheet TLV2772CP*.
]
- [7 [Internett]. Available: <https://www.electronics-tutorials.ws/opamp/op-amp-comparator.html>.
] [Funnet 2 Mai 2019].
- [8 Texas Instruments, *JK flip flop CD4027B datasheet*.
]
- [9 [Internett]. Available: https://www.ibiblio.org/kuphaldt/electricCircuits/Digital/DIGI_11.html.
] [Funnet 1 Mai 2019].
- [1 Arduino, «Arduino MKR vidor 4000,» [Internett]. Available: <https://store.arduino.cc/mkr-vidor-0> 4000. [Funnet 27 05 2019].
- [1 JSR Ultrasonics, February 2011. [Internett]. Available:
1] <http://www.jsrultrasonics.com/documents/DPR300OpManual.pdf>. [Funnet 26 05 2019].
- [1 [Internett]. Available: https://l.facebook.com/l.php?u=https%3A%2F%2Fwww.nde-2ed.org%2FEducationResources%2FCommunityCollege%2FUltrasonics%2Fcc_ut_index.htm%3Ffbclid%3DIwAR0mCRRhrDFPF4NZsmu1Dvk2v_tF1e3rksaplxS-ORbt-wd6x2HJkaPXsiY&h=AT1KZFsB7b7_sJuBrd798Z6rtofhAeNc_B6qn54axmOaQlbz. [Funnet 30 05 2019].

- [1 International Conference on Theoretical and Computational Acoustics (4th : 1999 : Trieste, Italy),
3] Theoretical and computational acoustics '99, World Scientific, 2004.

Appendiks A Forkortelser og ordforklaringer

Her skal relevante forkortelser og ordforklaringer komme i alfabetisk rekkefølge.

FPGA	Field Programable Gate Array
Kick	Høy spennings puls som eksiterer transduseren, som fører til utsendt ultralydsignal
PRF	Pulse Repetition Frequency
Respons / Ekko	Mottatt ultralydsignal som har reist tilbake til transduseren
TPD	Transduser Puls Detektor
Vf	Forward voltage
Vin	Spenningen på operasjonsforsterkeren sinn inngang
Δ Vin	Differansen mellom Schmitt-triggeren sin øvre og nedre hysteresegrense
Vout	Operasjonsforsterkerens utgangsspenning
Vref	Referansespenningen, spenningen på operasjonsforsterkeren sin negative inngang

Appendiks B Prosjektledelse og styring

B.1 Prosjektorganisasjon

Teamet består av to personer. Dermed har mye av arbeidet blitt gjort sammen, ettersom det ble behov for begge sin samlede kompetanse for å drive med testing i full skala. Når det ble mulighet, ble arbeidsoppgavene fordelt jevnt for å effektivisere. Men ingen offisielle roller ble tildelt.

B.2 Fremdriftsplan

				Uke	w2	w3	w4	w5	w6	w7	w8	w9	w10	w11	w12	w13	w14	w15	w16	w17	w18	w19	w20	w21	w22	w23	w24	w25	
				Man	7/1	14/1	21/1	28/1	4/2	11/2	18/2	25/2	4/3	11/3	18/3	25/3	1/4	8/4	15/4	22/4	29/4	6/5	13/5	20/5	27/5	3/6	10/6	17/6	
				Fre	11/1	18/1	25/1	1/2	8/2	15/2	22/2	1/3	8/3	15/3	22/3	29/3	5/4	12/4	19/4	26/4	3/5	10/5	17/5	24/5	31/5	7/6	14/6	21/6	
#	Aktivitet	Start Dato	Slutt Dato																Påske										
1	Forstudie arbeid	7/1	31/1																										
2	Krav spesifikasjon	16/1	31/1																										
3	Research	16/1	31/1																										
4	Forstudie innlevering	31/1						◆																					
5	Løsnings design	31/1																											
6	Ferdig design av løsning	8/3										◆																	
7	Midtveis presentasjon	29/3																											
8	Siste Innkjøp frist	15/3																											
9	Implementering	18/3	8/4																										
10	Testing	8/4	3/5																										
11	Mål for ferdig oppgave	9/5																											
12	Bachelor oppgave seminar/ Siste UV dag	10/5																											
13	Bachelor oppgave innlevering	31/5																											
14	Bachelor oppgave presentasjon	12/6																											

Hendelse / milepæl ◆

Appendiks C Brukerdokumentasjon

C.1 Brukerdokumentasjon

Boksen er utstyrt med et hengslet topplokk av gjennomsiktig plexiglass, for å enkelt kunne bytte ut 9 volts batteriet. Et lys på ESP8266 WiFi modulen lyser når den får tilført strøm fra batteriet.

DPR300 sitt signal output utgang kobles igjennom hullet i etuiet til signal inngangen markert med navnet «transduser inn». Bruk en overgang mellom coax og multimeterkabel. DPR300 sin T/R kobles til transduseren via Coax kabel.

Den aktive buzzeren vil kontinuerlig lage lyd ferm til WiFi modulen er tilkoblet til ThingSpeak. Når buzzeren stilner er måleren klar for bruk.

Et godt lag med vaselin må påføres mellom transduser og Corneliusfatet. For best mulig resultat skal transduseren ikke føres over sveisen.

Transduseren føres ned over Corneliusfatet sin utvendige side. En kort hørbar varsling indikerer at vannoverflaten er funnet.

Hold transduseren i ro i cirka 2 sekunder for å beregne et gjennomsnitt av 10 tetthetsmålinger.

Trykk på trykknappen for å laste opp måledataen over WiFi. Buzzeren vil da gi fra seg en hørbar varsling under dataopplastingen. Når lyden opphøres er opplastingen ferdig.

C.2 Drift og installering

For å ta i bruk utstyret må en ThingSpeak kanal opprettes, noe kildekode i Android studios og Arduino må også utfylles.

- **ThingSpeak**
 - Første steget er å lage seg en bruker hos på ThingSpeak.com, som er en svært enkelt prosess. Etter at bruker er blitt registrert gis mulighet om å opprette en ny kanal med valgt fritt navn. Den nye oppdrette kanalen blir gitt ett kanal nummer som senere skal lagres i Android studio koden. Under den nye oppdrette kanalen får du muligheten til å trykke på «Channel settings». Der inne skal trykkboksen field1 avmerkes, og gi den et valgt fritt navn.
 - Trykk på knappen API Keys for å notere deg ned Nøkkelen for skriving.

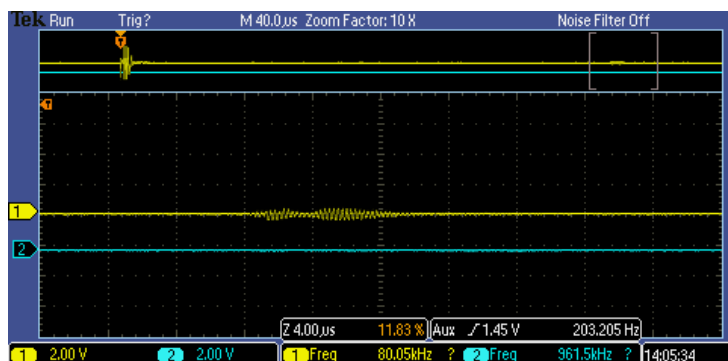
- **Arduino**
 - variablene SSID og Passord må fylles inn med brukerens eget WiFi.
 - API nøkkelen for skriving skal lagres i variabelen APIKEY [] i Arduino.

- **Android Studio**
 - API nøkkølen for lesing og Kanal id nummeret lagres i Android studio objektet tsChannel. Programmet er forhåndsinnstilt til å hente data fra field1 fra ThingSpeak.

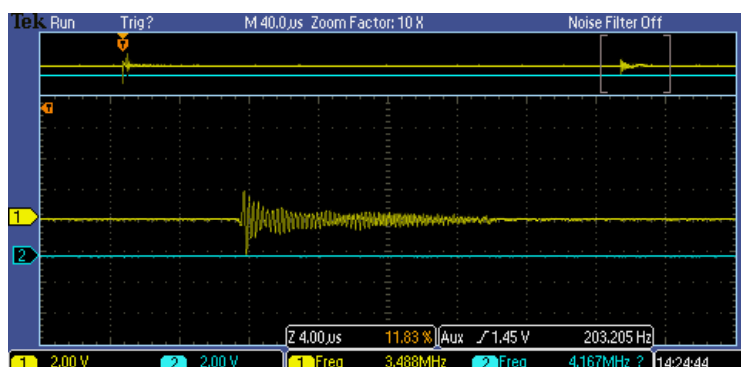
Appendiks D Oscilloskop skjermbilder

D.1 Responstest av forskjellige transdusere

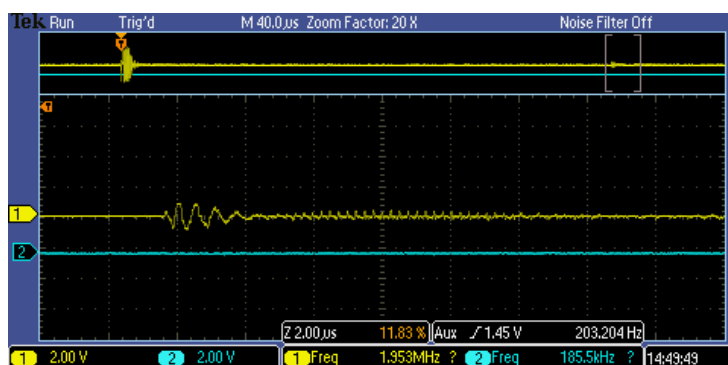
Responsen til en del transdusere ble testet oppimot fatet. Dette er en liste av responsene til de andre alternativene. Alle transdusere ble testet med DPR 300 innstilt til den på optimale innstilling for den spesifikke transduseren. Alle testene ble gjort med DPR300 mottaker del innstilt på 55dB gain, LP filter på 35MHz og HP filter på OUT. Oscilloskopet er stilt inn likt for alle målingene.



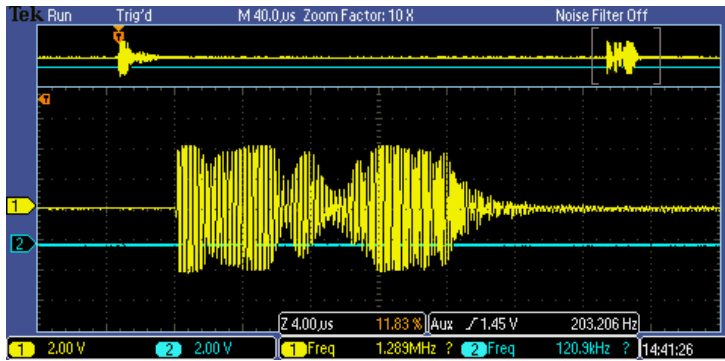
0.5MHz Olympus immersjons transduser



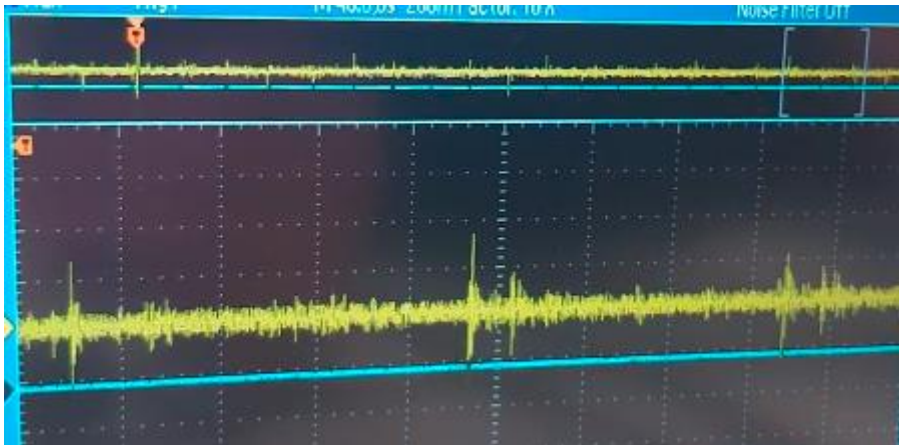
2.25MHz Olympus immersjons transduser



2MHz Olympus kontakt transduser

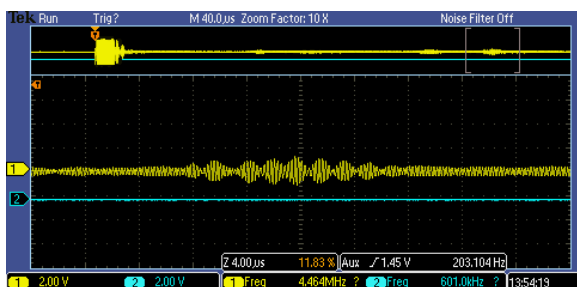
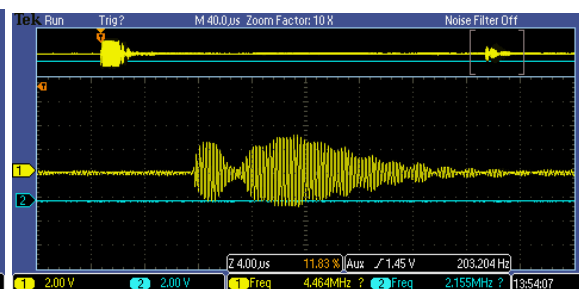
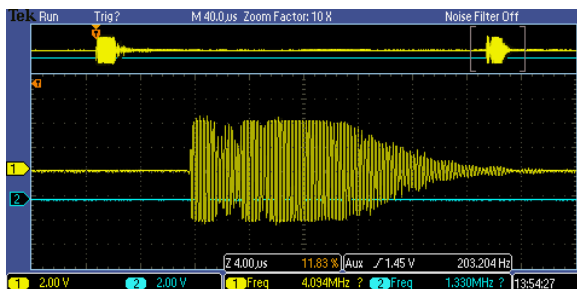


5MHz Olympus kontakt transduser



Støy målt fra 5MHz kontakttransduser

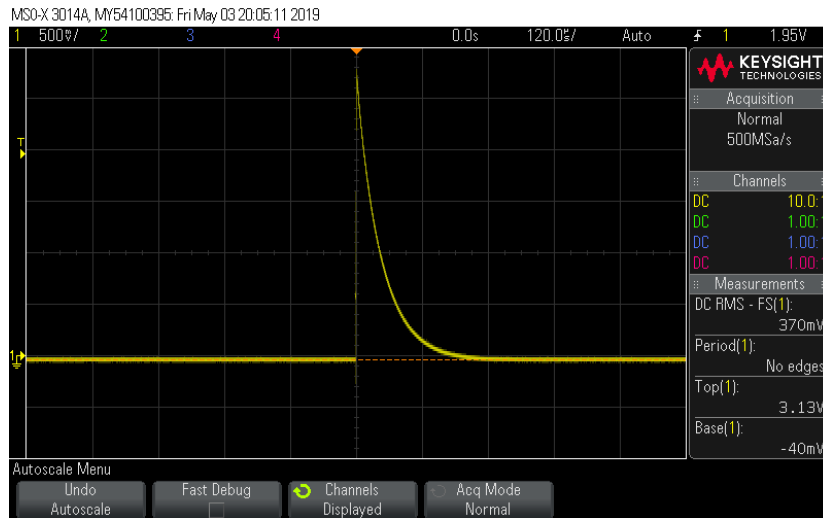
Når responsen til den valgte transduseren blir målt, ser en at signalstyrken minsker ettersom mer av transduseren ligger over væskeniået på fatet. Her er noen bilder fra testen. Man se at styrken minsker vesentlig. Merk at transduseren sin posisjon på fatet ble ikke endre mer en 5mm fra første bilde til siste.



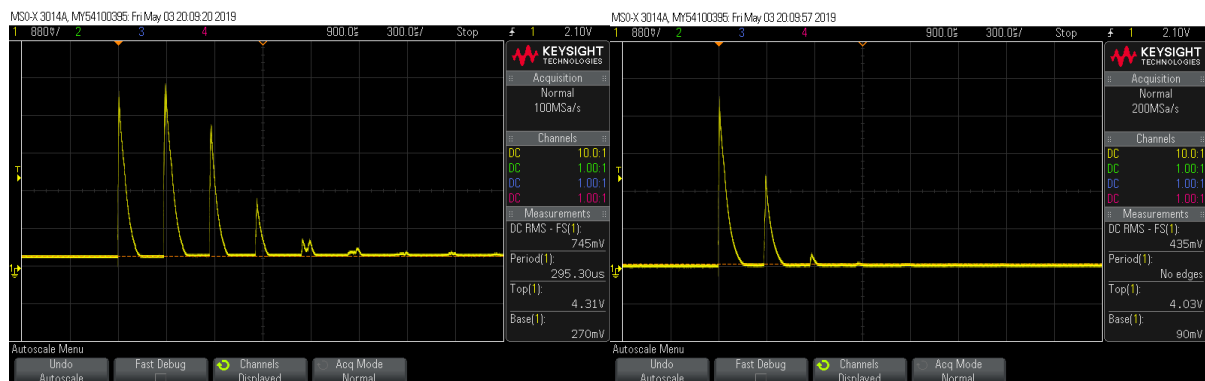
5MHz immersjons transduser deg væskegrense.

D.2 TPD drift

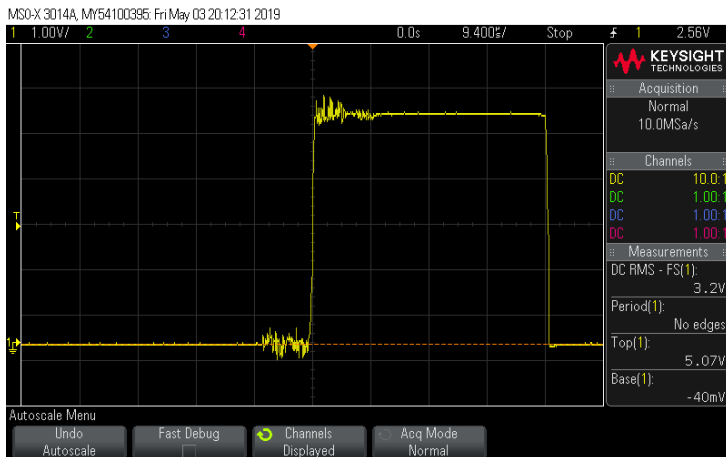
Her er en samling av målinger fra forskjellige TPD trinn når den er i drift.



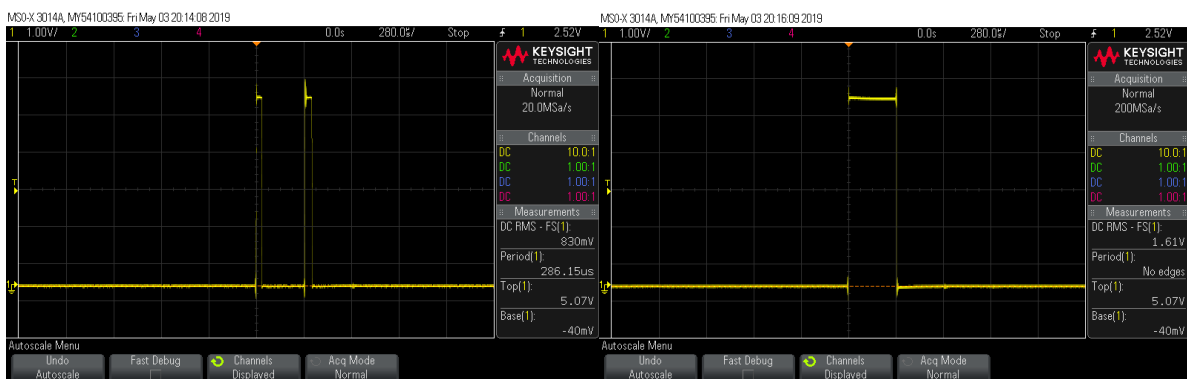
Ett enkelt kick, målt etter likeretteren.



To bilder som viser kick, og ekkoene som spretter inni fatet. Bilde nr.2 viser styrken til ekkoene når en måler på væskegrensesnittet. Målepunktet er etter likeretteren. Merk at det er cirka 280µs mellom hver hendelse. Det er tiden lyden tar fra transduseren, til andre enden av fatet, og tilbake.



Én puls generert av Transduser Puls Detektoren.



TPD utgang og dens effekt på JK-vippen. Man ser at puls én skurr på vippen, mens puls to skurr den av. Merk igjen at det er $280\mu\text{s}$ mellom første og andre puls på det første bilde, og $280\mu\text{s}$ fra positiv til negativ flanke på det andre bilde.

Appendiks E Datablad

E.1 Valgte transduser

OLYMPUSTel: 781-419-3900
www.olympus-ims.com**TRANSDUCER DESCRIPTION**PART NO.: V307
SERIAL NO.: 1020779
DESIGNATION: IMMERSION
FREQUENCY: 5.00 MHz
ELEMENT SIZE: 1 in. DIA.**TEST INSTRUMENTATION**FLAW DETECTOR: Epoch 600; S/N: 130524208
TEST PROGRAM: TP103-4
CABLE: RG-58 A/U LENGTH: 4FT**TEST CONDITIONS**PULSER SETTING: Energy: 100 VOLT; Damping: 50 OHM; Shape: Spike
RECEIVER SETTING: Gain: 20dB; Filter: DC-10.0 MHz
TARGET: 2 IN. SILICA
JOB CODE: TP200
WATER PATH: 1.014 in**MEASUREMENTS PER ASTM E1065**WAVEFORM DURATION:
-14DB LEVEL - 0.487 US
-20DB LEVEL - 0.526 US
-40DB LEVEL - 0.945 US
SPECTRUM MEASUREMENTS:
CENTER FREQ. - 4.57 MHz
PEAK FREQUENCY - 4.22 MHz
-6DB BANDWIDTH - 61.19 %**COMMENTS:**

F#: 4.6

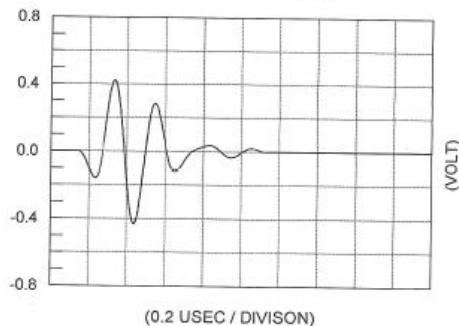
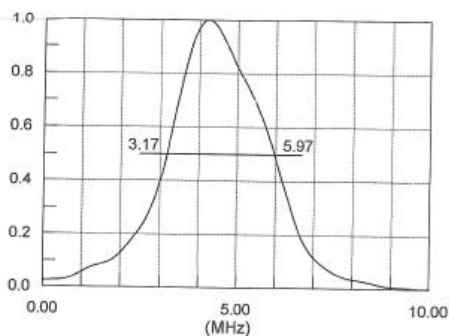
** ACCEPTED

TECHNICIAN (3)



DATE: 12-01-2015

This Certificate may not be reproduced except in full without written approval of OSSA

SIGNAL WAVEFORM**FREQUENCY SPECTRUM**

TP103 Rev. A

Dette er databladet til transduseren som ble valgt til å tas i bruk. 5MHz immersjons transduser.