

Indholdsfortegnelse:

| | |
|--|----|
| Forord:..... | 2 |
| Indledning: | 2 |
| Indledning: | 3 |
| Problemformulering:..... | 4 |
| Hardware..... | 5 |
| Følere: | 5 |
| Digital: | 5 |
| Forsyningspumper:..... | 6 |
| Råvandspumper:..... | 6 |
| Tilbageskylspumpe / Tilbageskylsføler: | 6 |
| Kompressor: | 6 |
| Analog:..... | 7 |
| Temperatur: | 8 |
| Luftfugtighed: | 8 |
| Afgangstryk: | 8 |
| Tank niveau:..... | 8 |
| Flow: | 8 |
| Interface | 9 |
| PC systemet..... | 9 |
| Klemplan:..... | 10 |
| Software: | 11 |
| Log Software:..... | 11 |
| Web præsentation..... | 13 |
| Grundside: | 14 |
| Grafisk side: | 15 |
| Indhentning i Excel: | 16 |
| Konklusion | 17 |
| Litteraturliste:..... | 18 |
| BILAG | 19 |

Forord:

Som afsluttende opgave i 2. semester, har jeg valgt at lave et projekt der omhandler Internet overvågning af Pårup vandværk.

Jeg kom på ideen da jeg snakkede med formanden for vandværket, og han fortalt om nogle problemer de havde haft fordi de ikke kunne holde øje med f.eks. pumper og andre afgørende faktor.

Jeg foreslog ham at lave et overvågnings system, hvor man kunne overvåge bl.a. drift, fejl og forbrug på værket. Han var meget interesseret i ideen, og gav mig grønt lys.

Sammen med teknikeren fra vandværket, gennemgik vi hvad han og forbruger kunne få gavn af at se på Internettet.

Principtegning vedlagt som bilag.

Grunden til jeg valgte dette emne skyldes hovedsagligt at det var et projekt som jeg fandt utrolig spændende.

Ideen i at man overvåge fysiske hændelser og målinger, ved hjælp af en computer, syntes jeg var en genial idé til netop emnet "Dataopsamling, bearbejdning og præsentation".

Og da jeg har en baggrund som elektriker, følte jeg godt at jeg kunne magte opgaven, også selv om det skulle installeres i praksis, hvilket jeg havde bestemt jeg ville gøre.

I den anledning vil jeg gerne sige tak til Pårup vandværk, for udlåning af test faciliteter, samt speciel udstyr som ultralyds føler og magnetisk flowmåler.

Uden den virkelige idriftsættelse af systemet, ville mange af de småfejl, som jeg har fundet under indkøringsprocessen, aldrig været blevet rettet.

Min motivation til opgaven har også været meget stor, da jeg fra starten har vist at det skulle bruges til noget efter min eksamen.

Også en tak til Herning Tekniske Skole EL afd. og VVS afd. , for udlåning af komponenter og viden.

Jan Kildelund Pedersen

Indledning:

I kampen for at de små virksomheder kan overleve de store udgifter til personale, der skal overvåge maskiner og andet materiel, har jeg valgt at konstruere et computer baseret system der varetage denne opgave.

Jeg vil lave et system, der kan overvåge et vandværk og informere om de aktuelle situation til teknikere og forbrugere.

Systemet skal laves så det ikke kræver nogen form for vedligeholdelse, ud over den normale vedligeholdelse der er ved en standard pc.

Som hovedpunkt skal det være muligt at se status på de faktorer der havde en afgørende betydning for vandforsyningen, så som pumper, kompressor og vandbeholdning.

Endvidere vil jeg kunne overvåge de enheder som forudsagde de fejl der forekommer.

Dernæst ville jeg kunne holde øje med vandforbruget og vandtrykket, derved er det muligt at opdage f.eks. en lækage på vandnettet.

Alt dette vil jeg lave så det er tilgængelig via en webside som jeg vil konstruere v.h.a. HTML og Java script.

Det projekt vil jeg fuldføre i praksis, ved at installere anlægget på Pårup Vandværk.

Problemformulering:

Under emnet "DATA opsamling, bearbejdning og præsentation", vil jeg kigge på problemstillingerne omkring elektronisk overvågning af et vandværk. Jeg vil i mit projekt lave elektronisk overvågning af Pårup vandværk, for at løse de problemer der forekommer under driften.

1. Manglende tilbageskyl af rensefilter, grundet manglende vand ved tilbageskylningspumpen.
2. ukendte antal pumpetimer for råvandspumper, forsyningspumper, tilbageskylpumpe, og kompressor.
3. manglende service oplysninger som luftfugtighed, vand temperatur og vandforbrug.

Formålet med dette projekt:

- At skabe et system der kan alarmere ved uønskede situationer i form af indikering eller alarm på status hjemmeside (og derved forøge den generelle kvalitet af den leverede varer).
 - For lav vandstand i hovedtanken
 - mislykkede tilbageskyl af rensefilter
- At skabe et system der kan informere administration, teknikker, såvel som forbruger om service oplysninger som f.eks.
 - Byens vandforbrug
 - Nu
 - Dag
 - År
 - Vandets temperatur
 - Nu
 - Vandets afgangstryk
 - Nu
 - Drift af maskiner og pumper
 - Kompressor (drift, starter og timer)
 - Råvandspumper (drift, starter og timer)
 - Tilbageskyls pumper (drift, starter og timer)
 - Afgangspumper (drift, starter og timer)
 - Flow i tilbageskylsrør under tilbageskyl (ekstra sikkerhed).
 - Vandstand i hovedtanken

Opgaven vil jeg løse ved hjælp af føler der registrerer de fysiske ændringer, temperaturfølere, fugtfølere, trykføler, ultralyd niveau føler, flowmåler, og digitale slutfunktioner, som via analoge og digitale ISA-interface overføres til en Pc'er.

Her opsamles data, og gemmes data af et C++ program til videre præsentation på en webside med tabeller konstrueret v.h.a. Java script

Denne computer skal så samtidig fungere som web-server der præsenterer oplysningerne, som så er tilgængelig via Internettet/Intranet (eksisterende ADSL-Forbindelse).

Hardware

Følere:

For at kunne elektronisk registrere en bevægelse, en temperatur eller en dybde, er det nødvendigt at bruge føler. I dag findes der næsten føler der kan registre alt, endda føler som kan registrere ting vi ikke kan.

I mit projekt gør jeg brug af flere forskellige typer af føler:

- Temperatur føler
- Fugt føler
- Tryk føler
- Flow føler
- Ultralyds føler
- Mekanisk hjælpe relæer

Disse føler er delt op i 2 grupper:

- Analog
- Digital



Digital:

Digitale føler er føler som definere deres måling med enten 0 eller 1 (on/off, low/high).

Disse føler virker for det meste meget simple ved af man sender et signal ud til føleren, og når føleren så registrere at tilstanden ændre sig, vil den returnere signalet igen o.s.v.

De 2 tilstande defineres i dette tilfælde med en spænding efter følgende regler:

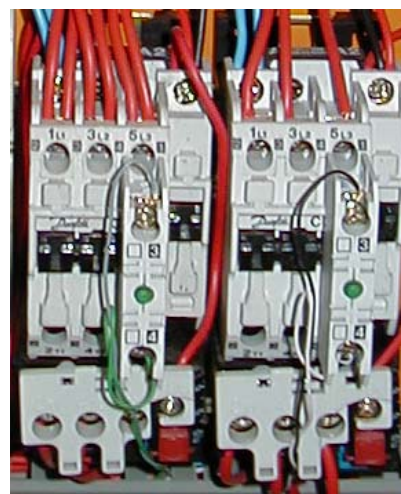
- Logisk 0 = MAX 0,8 V
- Logisk 1 = MIN 2,0 V

Digitale føler er i dette projekt brugt til registrere alle maskinernes tilstande, om de køre eller de ikke køre.

Ved at montere et mekanisk hjælperelæ på de kontaktor(relæer), som forsyner hver enkelt maskine med strøm, er det muligt at få en indikation af om den pågældende maskine er startet eller stoppet. Dette er naturligvis galvanisk adskilt, så det signal der sendes ud fra computeren aldrig bliver blandet med hverken det er aktivere kontaktor eller Maskine.

Kravene til installationen af digitale føler er normalt ikke så strenge som f.eks. analog føler, da der kun er tale om to tilstande(on/off).

De eksisterende regler i maskindirektivet(204.1) skal selvfølgelig overholdes!



Forsyningspumper:

På de 3 første Digitale indgange, sider forsyningspumpe 1,2 og 3.

Forsyningspumperne pumper vand fra beholdningstanken, og ud til forbrugeren, og holder det konstante tryk..

Pumpe 1 og 3 er endvidere styret af en VLT(frekvensomformer) for at kunne holde et konstant tryk, lige meget hvor meget forbruget svinger.

Pumpe 1 og 3 køre skiftevis, ca. 14 dage ad gangen. Dette holdes styr på manuelt, ved at vandværksteknikeren må besøge vandværket engang imellem, og tjekke timetællerne.

Dette kan nu gøres online på nettet, så han kun behøver at besøge vandværket når der skal skiftes.

Råvandspumper:

Indgang 4 og 5 overvåger de 2 råvandspumper, som pumper vand fra boringen, igennem filteret og ned i beholdningstanken.

Pumperne kør på skift, når niveauet i beholdningstanken når ned til 2.00 m, (med mindre den er i gang med at bageskylle filteret).

Overvågningen sker ved at overvåge den kontaktor der trækker motoren.

Det vil sige at hvis pumpen viser den kør, men vandstanden ikke stiger(afhængig af forbruget selvfølgelig), da er der sket en fejl med råvandspumpen.

Tilbageskylspumpe / Tilbageskylsføler:

Nu når der er blevet pumpet tilstrækkelig med vand igennem filteret, bliver det beskidt og fuld af okker. Derfor bageskyller man filteret.

Dette sker automatisk 2 gange om ugen, ved at et mekanisk ur starter tilbageskylspumpen, og vand pumpes fra beholdningstanken, gennem filteret og ud i kloakken.

Her har der været store problemer med at få gennemført tilbage skyllene, da vandet søkkede fra pumpen, og tilbageskyllene ikke blev gennemført, men den mekaniske tæller stadig talte derudaf.

Jeg har så monteret en tryk måler som indikere når trykket forsvinder fra pumpen, og giver en alarm på hjemmesiden.

Derved er det muligt at opdage fejlen i tide, inden forbrugeren får rødt vand ud af hanen.

Endvidere vil tælleren på siden kun tælle op, når der gennemføres et tilbageskyl, hvor trykket har været tilstede.

Kompressor:

På den sidste digitale indgang, overvåger jeg kompressoren.

Her er det muligt at følge med i hvor mange gange den starter, og hvor mange timer den har kørt.

Analog:

Analoge føler definere deres målinger med mere end 2 tilstande, jeg har i dette projekt valgt føler der videregiver deres måleresultater ved at variere en strøm mellem 4-20 mA.

For det meste kan man selv vælge i hvilken form man vil have udlæst sine resultater.

F.eks.

- 0-20 mA,
- 4-20mA,
- 0-5 V
- 0-10 V.

Da computers analog indgange arbejder med signaler i form $-5V$ til $+5V$ skal strømmene først omsættes til spænding.

Dette gør jeg ved at lade strømmen løbe gennem en 250Ω modstand, og så "føle" over modstanden.

Da vil spændingen over modstanden være:

- $250 * 20 \text{ mA} = 5V$
- $250 * 4 \text{ mA} = 1V$

Men konverteringen stopper ikke her. Inden signalet kan bearbejdes i computeren skal det den varierende spænding konverteres til en værdi(helt tal).

Konverteringen finder sted i "A/D konverteren" der er placere på det I/O kort som registrere alle data fra følerne.

Den "A/D konverter" som jeg bruger, er en 12 bit konverter, det vi sige at den opløser spændingen $-5V$ til $+5V$ om til en værdi fra 0 til 4096.

$$4096/10 = 409,6 \text{ (enheder pr. volt)}$$

Eks.

$$\begin{array}{ll} 20 \text{ mA} * 250 \Omega = 5V & 10 * 409,6 = 4096 \text{ (Max Værdi)} \\ 4 \text{ mA} * 250 \Omega = 1V & 6 * 409,6 = 2457,6 \text{ (Min Værdi)} \end{array}$$

Opløsning $4096 - 2457,6 = 1638$

Dette resultere desværre i at den endelig opløsning kun bliver på 1638 enheder.

For at udbedre opløsningen skulle jeg have brugt et andet I/O kort der kan måle unipolar($0-10v$) i stedet for bipolar($-5V$ til $+5V$).

Denne opløsning er nu også brugbar til dette projekt.

Eks.

Tryk føler:
 0-15 Bar
 præcision: $15 / 1638 = \text{ca. } 0,01 \text{ bar}$

Niveau føler:
 0-275 cm
 præcision: $275 / 1638 = \text{ca. } 0,1 \text{ cm}$

Her vil man så stadig betragte målingerne som præcise.

Installationen af analog føler er lidt mere krævende, da det er meget små signaler der arbejdes med, anbefales det at man benytter skærmet kabler.

Temperatur:

På den første analoge indgang er der monteret en temperaturføler, som fortæller temperaturen på vandet der forlader vandværket og ud til forbrugeren.

Denne oplysning, er ment som en service oplysning til forbruger.

Luftfugtighed:

En anden og lidt mere speciel føler, er fugtføleren.

Den fortæller hvor stor luftfugtighed er inde på vandværket.

Denne oplysning er nærmere en service oplysning til teknikker, end den er til forbruger.

Afgangstryk:

Denne måling er nok den mest relevante oplysning for både teknikker og forbruger.

Afgangstrykket fortæller nemlig hvor stort et tryk der er på vandnettet, altså om der kommer vand ud når man åbner for hanen.

Denne oplysning er yderst nyttig, hvis man som forbruger er i tvivl om det er hjemme hos en selv der er problemer med vandet eller det er på vandværket!

Websiden vil komme med både en visuel og en audio alarm, hvis vandtrykket falder til under 3 bar eller stiger over 5 bar!

Tank niveau:

Ude På vandværket var der i forvejen installeret en ultralyd niveauføler, der aflæser vandstanden nede i tanken. Den valgte jeg at koble på systemet, så det også var muligt at aflæse tankniveauet fra websiden. Derved har man hele tiden mulighed for at følge med i den aktuelle situationen.

Her er alarmen sat til < 1.50 m.

Flow:

Den nok mest interessante information fra systemet er flow. Flow er mængden af vand der pumpes ud af vandværket.

Her var jeg endnu engang heldig og finde en i forvejen installeret føler, nemlig en DANFOSS MAGFLOW 3100 med tilhørende kontrolpanel MAG5000, som også havde en analog udgang.

Udstyret var i forvejen installeret til at måle m^3/t , som er den måleenhed som bruges til afregne med kommunen. Den fik jeg omkonfigureret til at give l/s i stedet samtidig med m^3/t .

Derved var det muligt at kunne give et mere præcis resultat, hvilket ville give en bedre mulighed for at vise "live" målinger på websiden.

Målingerne bruger jeg til flere ting:

1. Måling af det nuværende flow(forbrug) i l/s, som vises på websiden, og opdateres hvert 5. sekund.
 2. Måling af det nuværende flow(forbrug) i m^3/t , som vises på websiden, og opdateres hvert 19. sekund.
 3. Måling af det totale forbrug i m^3/t
 4. Hver $\frac{1}{2}$ times forbrug skrives i en fil, der gemmes på harddisken under den aktuelle dag. •
 5. Hvert døgn's forbrug skrives i dagslog.txt på harddisken. •
- Disse logfiler der så mulig at indhente i et Excel regneark til videre præsentation som f.eks. grafer over forbruget pr. time over et døgn og pr. dag over et år.

Interface

Som tidligere nævnt er det nødvendigt af få omsat spændinger og strømme til data værdier. Dette opnås ved at benytte et I/O-kort(input/output).

Det kort som jeg har fået stillet til rådighed er et kort fra NUDAQ købt gennem Danbit

Kortet er et middeltkort hvis man kigger på kapacitet og hastighed.

Kortet har bl.a. følgende egenskaber:

- 16 Digitale indgange
- 16 Digitale udgange
- 8 Analoge indgange(12 bit)
- 1 Analog Udgang(12 bit)

Fordelene ved netop dette kort er nok at der er ”lidt af det hele” på, til en rimelig overkommelig pris. Samtidig med at det er nemt at komme i gang med, hvis man blot følger manualen, og nogle enkelte eksempler der følger med. Ud over eksemplerne, findes der også nogle test utilities på den medfølgende cd.

For at få opsamlet data'en fra indgangene, og den videre bearbejdning, har jeg konstrueret et program i Turbo C++.

Princippet i at indhente informationerne forgår ved at man:

1. Beder kortet indhente data.
2. Beder kortet konvertere data.
3. Beder kortet returnere data.

værdierne læses så ind i nogle variabler til senere brug.

PC systemet

Interfacet er installeret i en Pentium 166 med 160 mb ram.

Der er endvidere installeret et 100 Mbit netkort, et standard VGA grafikort, og naturligvis ISA interfacekortet.

På maskinen er der installeret følgende software:

- Operativsystem
 - Windows 98 SE
- WEB(port 3000) & FTP Server(intern)
 - LiteServe
- Remote Management(port 11600)
 - Remote Administrator 2.1
- Log Af I/O data
 - Vand.exe (af: Jan Pedersen)



Klempplan:

| Navn | Connector / Farve | Pin | Sub-D 25 Pin | Klem nr. | Bemærkninger: |
|-----------------|-------------------|---------|--------------|----------------------------|--------------------|
| | CN1 | | | | |
| Analog ind 0 | Grøn | 1 | 9 | 21 | Temperatur Føler |
| Analog ind 1 | Sort | 2 | 10 | 23 | Fugt Føler |
| Analog ind 2 | Blå | 3 | 11 | 25 | Tryk Måler |
| Analog ind 3 | Gul | 4 | 12 | 27 | Niveau Måler |
| Analog ind 4 | Brun | 5 | 13 | 29 | Flow Måler |
| Analog ind 5 | Orange | 6 | 14 | 31 | Disponible |
| Analog ind 6 | Hvid | 7 | 15 | 33 | Disponible |
| Analog ind 7 | Grå | 8 | 16 | 34 | Disponible |
| GND(fra analog) | Lilla | 20(-27) | 8 | 19,22,24,26,28,30,32,34,35 | |
| +5V | Rød | 19 | 7 | 2,4,6,8,10,12,14,16,18,20 | |
| | CN2 | | | | |
| Digital in 0 | Blå | 1 | 1 | 1 | Defekt |
| Digital in 1 | Gul | 2 | 2 | 3 | Forsyningspumpe 1 |
| Digital in 2 | Brun | 3 | 3 | 5 | Forsyningspumpe 2 |
| Digital in 3 | Orange | 4 | 4 | 7 | Forsyning pumpe 3 |
| Digital in 4 | Hvid | 5 | 5 | 9 | Råvandpumpe 1 |
| Digital in 5 | Grå | 6 | 6 | 11 | Råvandpumpe 2 |
| Digital in 6 | Rød | 7 | 17 | 13 | Tilbageskyls Pumpe |
| Digital in 7 | Lilla | 8 | 18 | 15 | Tilbageskyls Føler |
| Digital in 8 | Grøn | 9 | 19 | 17 | Kompressor |



Software:

Log Software:

Når data'en står klar til læsning på I/O-kortet, er det så op til softwaren af indhente den til videre bearbejdning.

Til dette formål har jeg konstrueret et stykke software i Turbo C++ (vand.exe).

Programmets hovedopgave består i sin helhed i at opsamle data'en fra I/O-kortet, bearbejde og konverter målte værdier, og gemme dem til senere præsentation.

Ud over dette, har det følgende features:

- Konvertering til respektive værdier
- Log af
 - Aktuell tilstand
 - Starter
 - Tid
 - Mængde pr. time i dato angivet fil (kun flow)
 - Mængde pr. dag (kun flow)
- Backup af data til harddisk
- Automatisk reload fra backup, i tilfælde af strømafbrydelse
- Realtime log på serveren skærm.

Programmet er opbygget af en række funktioner som bliver kaldt fra hoved delen.

Funktionerne i programmet er:

| | |
|--|---|
| Grader, Fugt, Tryk, niveau, Flow, flowm3 | Disse funktioner, konverter standard input værdier fra følerne til værdier der svare til dem følerne registrere |
| Timeout, (omregntilmsekunder) | Timeout er en funktion, der hele tiden holder øje med om tiden x er udløbet, v.h.a. underfunktionen omregntilmsekunder. Denne funktioner er en af grundelementerne i programmet, da det starter funktionerne på det rigtige tidspunkt |
| Mindelay | Mindelay er en mindre funktion, som giver interfacet en kort pause til at bearbejde data, og udføre opgaver stillet af programmet. |
| Readdport, readdporthigh | Disse 2 funktioner indlæser alle de digitale indganges status. Dette gøres ved at starte fra det 8. bit og tjekke om de enkelte bit plads værdier er repræsenteret(1,2,4,8,16,32,64,128). Ved at tag den hentede værdi fra porten(0-255), og undersøge om den er større eller lig med bitplads værdierne. De hentede værdier gemmes i et global array (d[]) |
| Readapoer | Readaport indlæser værdierne fra de analoge porte, ved først at initialisere den ønskede kanal, og derefter trigge den. Derefter vil kortet sætte en speciel bit til 1, når data'en er klar til at blive hentet. Derefter hentes data'en ned i et global array(a[]) For at gøre indlæsning mere stabil hentes 20 målinger, og gennemsnittet gemmes. |
| dato | Funktionen laver den aktuelle dato, som bruges i dagsloggen. Ud over dette laver den også variabler som bruges når der skal laves biblioteker, navngivet efter hvilken måned, de deri gemte |

| | |
|--------|--|
| | daglige m ³ målinger er taget. Og til navngivningen af filer, hvor de daglige m ³ målinger bliver gemt |
| Tid | Tid konstruere en variable der indeholder aktuel klokken. |
| Reload | Ved genstart af programmet, vil denne funktion hente tidligere gemt backup data, fra filen c:\vandv\timedata.dbv (tider, starter, og total m ³ flow. |

Hoveddelen består af 4 watchdogs:

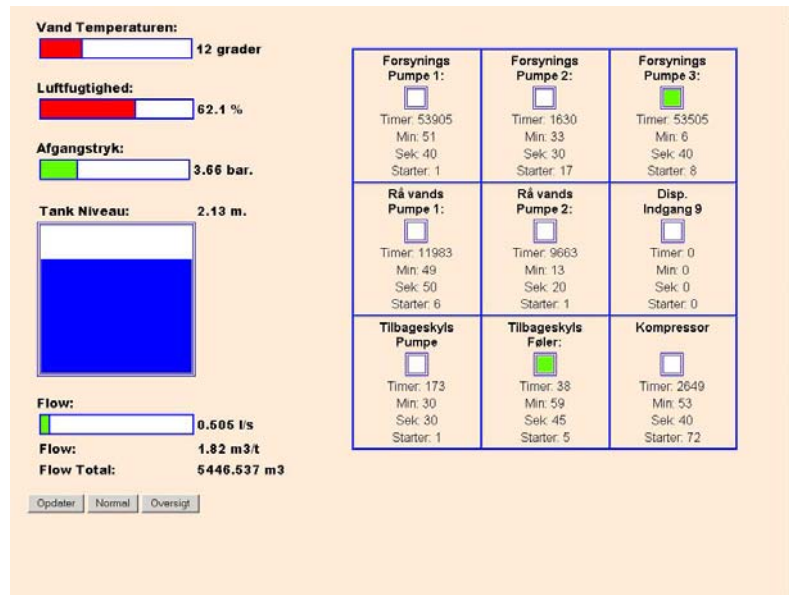
| | |
|------------------|--|
| Hver 24 timer | Hver 24 timer bliver det totale vandforbrug for det døgn skrevet i dagslog.txt i et semikolon delt formatet, så det kan hentes ind i et regneark. |
| Hver 30 minutter | Hver 30 min. Tages en backup af timedata, starter og totalflow, som gemmes i C:\vandv\timedata.dbv. Og så skrives forbruget for de sidste 30 min til c:\vandv\'månde_år\'\'dato\' .txt. Denne til er også semikolon delt, til brug i f.eks. Excel. |
| Hver 19 sekunder | Tider, starter og totalflow skrives i gdata.js, som bruges af websiden. Gdata.js er placeret på ramdrevert |
| Hver 5 sekunder | Hver 5 sekunder indhentes og udskrives aktuel status på digitale og analoge port til data.js Ud over dette bliver alle de aktive portes tider talt op med 5 sek. Omregning mellem sek./min/timer bliver fortaget. Starter bliver tjekket, "on til off" eller "off til on" Timeflow, dagflow og totalflow bliver talt op med forbruget inden for de sidste 5 sek.(dette gøres ved at tag den sidste målte værdi(l/s) og gange den med 5 sek.). Dette medføre et upræcis resultat da forbruget svinger lidt pr. sek., derfor er totalflow målingen kun vejledende. |

Hele Source koden og Main(flow) er vedlagt som bilag!

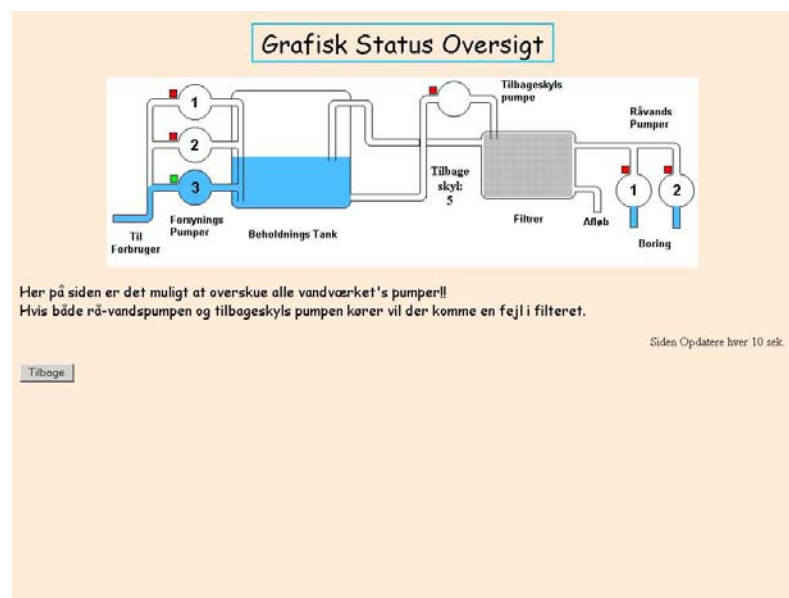
Web præsentation

For at gøre administration af vandværket nemmere, har jeg valgt at lave et WEB interface til præsentation af de registrerede data. Jeg har lavet 2 sider.

- En grundside med alle de målte værdier i form af indikator, bar og tabeller.



- Og en oversigtsside med en grafisk tegning af vandværket hvor man kan følge tilstandene på alle pumperne, og flowet i rørene.



Grundside:

For at få vist de målte værdier på siderne, kræver det det lidt basal kendskab til HTML og JAVA SCRIPT.

HTML bruger jeg til at lave siderne rent grafisk, og Java til at få indlæst variablerne fra de tidligere gemte filer data.js og gdata.js.

For at få variablerne overført til HTML/JAVA siden skal man indlæse dem først:

```
<SCRIPT LANGUAGE="Javascript" SRC="http://80.199.12.19:3000/data.js"></SCRIPT>
<SCRIPT LANGUAGE="Javascript" SRC="http://80.199.12.19:3000/gdata.js"></SCRIPT>
```

Dette medfører, at hver gang denne side bliver loadet vil de 2 filer blive læst:

f.eks.

```
D1=0;
A0=12.0;
```

Dette vil blive forstået som alm. Erklæringer af variabler, som så kan bruges i HTML/JAVA koden.

```
<SCRIPT LANGUAGE="Javascript">
  if(A0 <= 10)
  {
    document.write('<td width='+((4.5*A0))+ ' bgcolor=#66ff00>&nbsp;</td>');
    document.write('<td width='+ (198-(4.5*A0))+ ' bgcolor=white>&nbsp;</td>');
  }
  if(A0 > 10)
  {
    document.write('<td width='+((4.5*A0))+ ' bgcolor=red>&nbsp;</td>');
    document.write('<td width='+ (198-(4.5*A0))+ ' bgcolor=white>&nbsp;</td>');
    document.write('<BGSOUND SRC="alarm.wav" LOOP="infinite">');
  }
</SCRIPT>
```

Ovenfor ses hvordan jeg bruger variablerne til at lave en tabels længde, endvidere sætte jeg også parameter, der vil lave tabellen rød eller green afhængige af om den målte værdi er større eller mindre end 10.



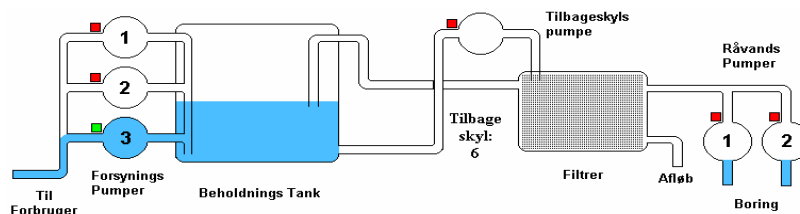
Til sidst beder jeg websiden om at afspille en .wav fil, hvis den værdien er større end 10, hvilket her betyde at der er et meget stort vandforbrug, f.eks. en rør lækage.

Grafisk side:

Fremstillingen af den grafiske oversigt var lidt mere kompliceret.

For at kunne lave den "levende" oversigt, måtte jeg først konstruere et billede, og der efter skære den op i mindre stykker..

Derefter ændrede jeg alle billederne manuelt, så der var et til hver eneste tilstand.



På HTML siden samlede jeg så alle stumperne, v.h.a. Java Script, alt efter indgangenes værdi.

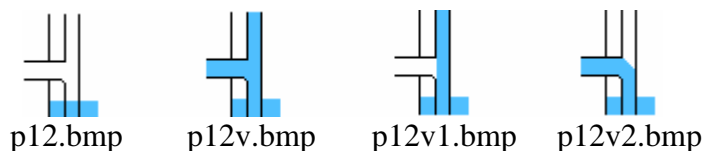
```
<SCRIPT LANGUAGE="Javascript">
  if(D1 == 1 && D2 == 0)
    document.write('<TD VALIGN=TOP ALIGN=LEFT><IMG SRC="p12v1.bmp"
BORDER=0></TD>');
  if(D1 == 0 && D2 == 1)
    document.write('<TD VALIGN=TOP ALIGN=LEFT><IMG SRC="p12v2.bmp"
BORDER=0></TD>');
  if(D1 == 1 && D2 == 1)
    document.write('<TD VALIGN=TOP ALIGN=LEFT><IMG SRC="p12v.bmp"
BORDER=0></TD>');
  if(D1 == 0 && D2 == 0)
    document.write('<TD VALIGN=TOP ALIGN=LEFT><IMG SRC="p12.bmp"
BORDER=0></TD>');
</SCRIPT>
```

Her ses hvordan der i forhold til variabelernes tilstand, ændres på den grafiske side, ved at lade Javascriptet konstruere resten af tabel HTML koden.

I eksemplet ovenfor ses en situation hvor der er tale om 4 situationer.

Der er her tale om at 2 variabler ved navn D1 og D2, som er Forsyningspumpe 1 og 2.

Nedenfor ses de 4 billeder som valget står imellem.

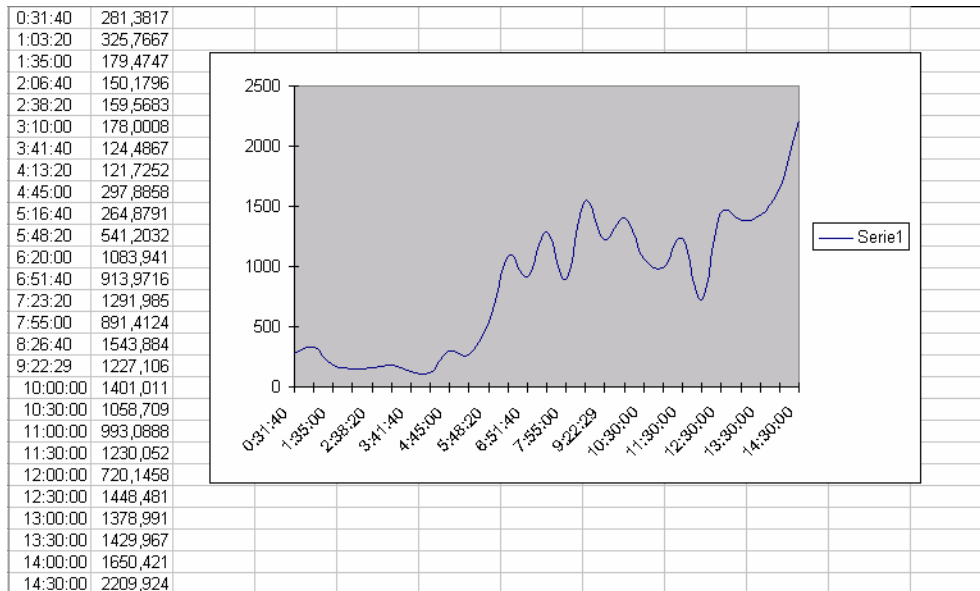


Som det ses drejer det sig om vandrøret fra vandtanken og til pumpe 1 og pumpe 2.

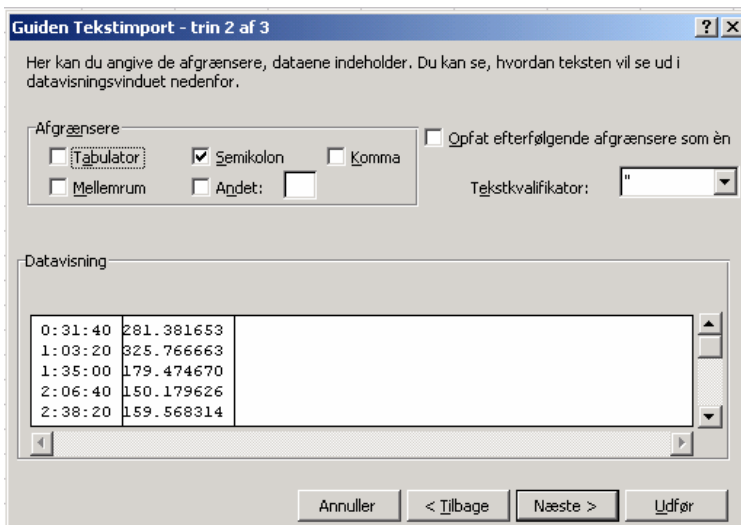
Så vil websiden blive genereret ud fra variabelernes tilstand, så billedet P12v1.bmp kun bliver indlæst når forsyningspumpe 1 køre uden forsyningspumpe 2.

Indhentning i Excel:

Som tidligere nævnt gemmes forbruget pr. ½ time og pr. dag.
Disse oplysninger kan indhentes i Excel til videre bearbejdning og præsentation.



Det eneste man skal være opmærksom på, er at man importerer filen som en semikolon separeret fil, uden noget tegn for tusindtalsseparator og med punktum som decimalseparator.



Et eks. Er vedlagt som bilag.

Konklusion

Resultatet af projektet resulterede i et simpelt stabilt system der overvåger Pårup vandværk døgnet rundt, og informere om alle relevante målinger og resultater.

Det er nu muligt at holde øje med drift af alle installerede maskiner, samt mængden af leveret vand til forbruger.

Der er fundet en løsning på vandværkets problemer med fejl ved tilbageskyl, rør-lækkager og generel mangel på informationer fra vandværket.

Endvidere har man mulighed for at indhente data om forbruget fra hver ½ time på enhver dag, samt total forbruger pr. dag. Disse data er klar til indhentning i et regneark f.eks. Excel.

Alt dette har medført en større tryghed hos teknikeren, da han hele tiden kan følge med i driften og rette kommende fejl inden de bliver kritiske.

Under arbejdet med idriftsættelsen af systemet opdagede jeg hurtigt, at min tidligere erfaring som elektriker var en stor fordel.

Systemet er nu sat i drift på Pårup vandværk, og lever indtil videre op til kravene.

De stille krav fra vandværkets side, om indholdet af overvågningen, er blevet godkendt af vandværkets teknikere.

Dog skal det siges at teknikeren allerede har afgivet ønske om en del udvidelser, som kunne gøre vandværk mere tekniker-venlig.

Disse muligheder kunne f.eks. være mulighed for at kunne resætte termorelæer, omskifte drift mellem forsynings pumper. Dog vil det betyde at sikkerheden til systemet fra internettet skal øges, eller at kommunikationen til systemet skal etableres på en anden måde.

Litteraturliste:

Forfatter: Finn Elvekjær
Title: TURBO C++
Forlag: SYSTIME
Udgave: 1. Udgave 5. oplag
Udgivelsesår: 1991
ISBN: 87 7783 008 3

BILAG