

Logging med micro:bit, en metode å integrere programmering med naturfag

A. Stormo, T. Smevik, og E. I. Romijn, *NTNU*

ABSTRACT: Praktisk arbeid er en viktig del av naturfag. Programmering blir et stadig viktigere verktøy i praktisk arbeid, også for folk som jobber med naturfag. Som en del opplæringen og for å støtte innføringa av Fagfornyelsen i skolen har Micro:bit blitt utbredt over hele Norge og alle trinn i grunnskolen, delvis også i videregående opplæring. I denne artikkelen beskriver vi et undervisningsopplegg som viser hvordan man kan bruke Micro:bit for å samle data og lage måleserier, en viktig del av naturfaglig forskning og praktisk arbeid.

1 INTRODUKSJON

1.1 Hvorfor Micro:bit?

Naturfagundervisningen i norsk skole skal sørge for at elevene får jobbet både praktisk og utforskende [1]. For å utforske fenomener som tyngdekraft eller syrerreaksjoner trenger en ikke nødvendigvis nøyaktige målinger, men hvis en skal kunne vurdere mer presise deler av naturfag, som for eksempel ohms lov eller kjemiske likevekter, er det nødvendig med mer nøyaktige data. Manuelle målinger tar tid, og i en undervisningssituasjon vil det være vanskelig å få store nok data til å kunne sammenlikne godt med enkelte av de naturfaglige modellene. Et alternativ er derfor å oppgi et eksisterende datasett som elevene kan vurdere, men å registrere data fra forsøk er en del av den grunnleggende ferdigheten «digitale ferdigheter» i naturfag [1].

Et alternativ til å manuelt ta og notere ned målinger er å automatisere prosessen ved hjelp av dataloggere og så behandle målingene på datamaskiner. Det finnes dedikert utstyr og programvare som gjør dette, men hvis en behersker programmering og har litt teknologisk kompetanse, er dette noe en kan gjøre selv. Nå som programmering er blitt en del av den norske læreplanen [1] åpner det seg opp en mulighet for å gjøre nettopp dette i klasserommet.

I tillegg til at programmering av teknologiske gjenstander er blitt en del av den norske læreplanen, er også alle norske grunnskoler tilbudt programmerbare mikrokontrollere gjennom prosjektet Super:bit [2]. Mikrokontrolleren er utviklet i samarbeid med BBC og heter Micro:bit [3]. Denne kan programmeres til å gjøre en hel del, inkludert respondere på kommandoer fra en bruker, kontrollere andre digitale gjenstander, samt samle data fra måleinstrumenter.

Når elever kan ta både manuelle målinger og automatiserte måleserier kan de også få egen erfaring av hvordan ny teknologi gir nye muligheter for å vurdere fenomener, og dermed et godt utgangspunkt for å forstå hvordan teknologisk utvikling har drevet utviklingen av vitenskapelig forståelse av naturen.

Når vi ser på læreplanen i naturfag, ser vi at det allerede på mellomtrinnet er mulig å bruke Micro:bit til å lære om både naturfaglige tema og hvordan naturvitenskap utvikler seg. Følgende kompetansemål for 7. trinn gir gode rom for denne type undervisning [1].

- gi eksempler på hvordan naturvitenskapelig kunnskap er utviklet og utvikler seg
- utforske, lage og programmere teknologiske systemer som består av deler som virker sammen
- utforske faseoverganger og kjemiske reaksjoner og beskrive hva som kjennetegner dem
- utforske elektriske og magnetiske krefter gjennom forsøk og samtale om hvordan vi utnytter elektrisk energi i dagliglivet

For å ruste kommende lærere til å gjennomføre forsøk med bruk av programmerbare mikrokontrollere har vi på lærerutdanningen utviklet undervisning der vi lærer studentene å ta automatiserte måleserier ved bruk av Micro:bit. Denne har blitt gjennomført med lærerstudenter på både grunnskoleutdanninger for 1 til 7, 5 til 10, samt lektorutdanningen for 8 til 13. og etter- og videreutdanning av lærere i Trondheim kommune.

1.2 Følgforskning

Undervisningsopplegget er en del av et prosjekt for å skrive et hefte for lærerutdannere som vil ta i bruk programmering som en del av naturfag på grunnskolelærerutdanningen. Vi håper at heftet også vil kunne være nyttig for fremtidige og nåværende lærere selv om hovedfokuset er lærerutdannere. I tillegg til at vi håper å få verdifull innsikt i hvordan man kan integrere programmering i naturfagundervisningen håper vi også å finne ut noe om hva som skal til for at framtidige lærere tar i bruk programmering, og hvordan våre studenter møter det i praksis nå. Dette er det kommentert kort på til slutt i artikkelen.

2 UNDERVISNINGSSOPPLEGG

Undervisningsopplegget tar utgangspunkt i tre forskjellige måter å lage måleserier med Micro:bit. De forskjellige metodene beskrives under hver økt. Hver type målemetode ble demonstrert med en måling studentene skulle teste.

Micro:bit har en del interne sensorer og det finnes i tillegg et stort utvalg i eksterne sensorer. I dette prosjektet har vi valgt å begrense oss til de interne sensorene, både for å ikke gjøre målingene unødvendig komplekse med å legge til en komponent til og fordi det er de sensorene man har tilgang til med en Micro:bit, altså det utstyret norske skoler har. Ikke alle har kjøpt til ekstra utstyr som for eksempel fuktighetsmålere og CO₂-målere, men for de som har så vil metodene som presenteres her også kunne brukes med eksterne sensorer.

De interne sensorene er temperatur målt i Celsius, magnetfelt målt i μT , akselerasjon målt i mG (tusendedeler av tyngdeakselerasjonen) og lysnivå. Lys måles i en skala uten enhet, fra null til 255. Med versjon 2 kom i tillegg mulighet for å måle lyd med samme type måling som for lys, en skala uten enhet fra null til 255.

2.1 Økt 1: Enkeltmålinger og lister

Den enkleste måten å ta målinger med en Micro:bit er å lage en blokk som gir målingen om man trykker på en knapp. Man må da registrere målingene med å lagre den på et annet vis, for eksempel ved å skrive den ned. Om man vil lagre målingen på Micro:bit har man muligheten til å lage variabler. Micro:bit støtter variabelformatene tall, tekststreng, liste med tall og liste med tekst. Det er egne blokker for å jobbe med lister.

I undervisningsopplegget fikk studentene presentert det som et Parsonsproblem [4], det vil si at man får blokkene som inngår i problemet og så skal man sette sammen et problem som fungerer selv. Oppgaven er illustrert i Figur 1, med løsningsforslag i Figur 2. Oppgaven ble gitt ut gjennom Micro:bit classroom. I Micro:bit classroom kan man sette opp et online klasserom som studenter/elever kan logge seg inn på. Man trenger ikke noen brukerkonto, og man kan sette det opp sånn at det ligger kodeforslag eller blokker der. Man kan også sende enkeltelever eller alle kodeforslag underveis.

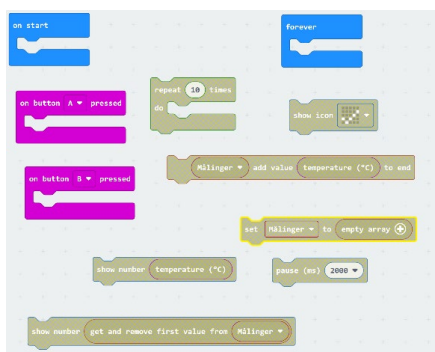


Fig. 1. Figuren viser blokkene som studentene fikk dem i Micro:bit classroom.

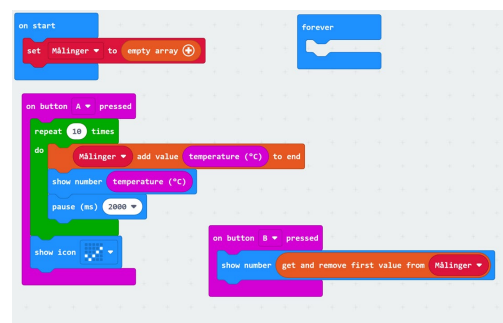


Fig. 2. Mulig løsningsforslag for loggeoppgaven i figur 1.

Den første oppgaven var å programmere Micro:bit'en til å måle temperatur når man trykker på knapp A. Den neste oppgaven er å lage et program som når man trykker på knapp A måler temperaturen ti

ganger og lagrer den i ei liste, og som leser av verdiene når man trykker på knapp B. Programmet blir sånn at man leser av en og en verdi som så fjernes, dette fordi man da kan jobbe med lister uten å bruke vilkår eller andre løkker enn «gjenta X ganger»-blokka.

2.2 Økt 2 – Del 1: Live logging

Micro:bit har ei blokk som gjør at man kan vise hvor stor en størrelse man måler uttrykt med hvor mange av LED-lysene på skjermen som lyser. Disse målingene kan man også følge i sanntid på en graf om Micro:bit'en er koblet til PCen, og man kan laste ned målingene underveis om man vil importere tallene til andre program som for eksempel Excel. Et eksempel på graf er gitt i Figur 4.

Programmeringsmessig er dette den enkleste måten, da man bare trenger to blokker, som vist i Figur 3. Dette er en fordel om måleseriene er hovedformålet, men litt begrensende om man vil ha med noe mer avansert programmering.



Fig. 3. Live logging av temperatur. Maks temperatur er satt til 30 og serial write er på sånn at man kan se temperaturen live på PCen

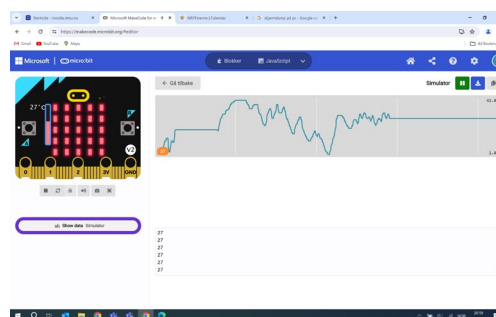


Fig. 4. Resultat av live logging av temperatur med simulator-micro-biten. Man ser data representert på tre måter: med LED-lys, med graf og med tall.

Hovedbegrensningen med metoden er at den må være knyttet til PCen, mens en stor fordel for mange forsøk er at man kan se utvikling live. Søylediagram med LED-lysene er også godt egnet til å finne endringer i for eksempel magnetfelt eller lys i et rom, for da trenger man ikke PCen og kan gå rundt omkring for å sjekke størrelsene.

2.3 Økt 2 – Del 2: Ekstern utvidelse for datalogging

Det finnes mange eksterne utvidelser til Micro:bit. Det er tre hovedkategorier, sensorer, diverse driverbrett og ting som blir styrt av Micro:bit, som robotbiler og droner. De fleste av disse programmeres med et eget sett av blokker. Man finner disse blokkene ved å trykke utvidelse, søke på det man skal programmere og så trykke på dette. Da lastes blokkene automatisk inn i MakeCode. Her finnes det også utvidelsen «datalogger» som ikke trenger noe fysisk ekstern utvidelse, men man trenger versjon 2 av Micro:bit for å bruke den. Et eksempel på kode som bruker «datalogger» er gitt i Figur 5.

Denne eksterne utvidelsen lagrer data i lister i Micro:bit'en. De kalles kolonner, og må opprettes før man kan begynne målingene. Det er en egen blokk for å lage kolonnene, og man kan lage flere kolonner så man lagre flere måleserier samtidig. Når man har gjort målingene (med Micro:bit'en frakoblet) kobler man den til PCen igjen, og finner dataene i en fil på Micro:bit'en kalt MyData. Et eksempel på hvordan disse dataene vises frem er gitt i Figur 6. Det er vesentlig at man kobler av og på Micro:bit'en ellers vil ikke MyDatafila oppdateres og man finner ikke målingene.

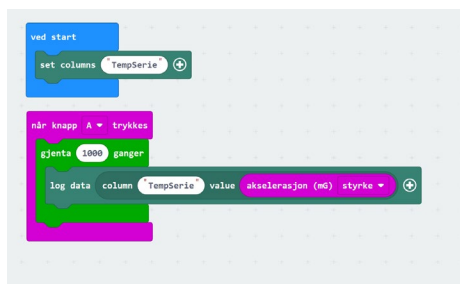


Fig. 5. Program som lager en kolonne og så logger 1000 akselerasjon i den om man trykker knapp A.

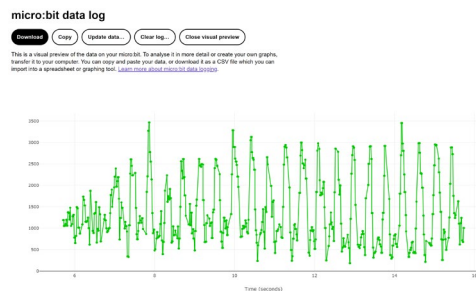


Fig. 6. Grafisk framstilling av data fra koden i figuren til venstre.

Om man ikke legger inn noe pause mellom målingene vil Micro:bit'en måle så fort den kan, så det blir da ca. 10 ms mellom hver måling. Basert på tester vi har gjort i klasserommet håndterer dataloggeren rundt 12000 målinger før den blir full.

2.4 Økt 3 åpen oppgave

I den siste økta fikk studentene en åpen oppgave: Dere skal programmere to Micro:bit slik at de tar den samme måleserien, men på forskjellige steder. De må starte samtidig, så de må kommunisere med hverandre. Studentene velger selv hvilken loggemetode som egner seg best for å få sammenlignet underveis eller etterpå.

Formålet med den åpne oppgaven var å la studentene bruke deler av det de hadde lært om igjen for at det skulle sitte bedre, og for se hvordan det fungerte å legge til noe mer programmering. Flere Micro:bit kan kommunisere med hverandre ved å bruke diverse blokker for radio. For å få to stykker til å gjøre målinger samtidig kan de enten kommunisere med hverandre eller med en tredje Micro:bit som starter dem.

Eksempler på ting som ble målt var for hvordan temperaturen sank da Micro:bit'en ble lagt i fryseren, hvordan et smell hørtes på to forskjellige steder og hvordan akselerasjonen forandret seg etter hvordan man hoppet.

3 OPPSUMMERING

Undervisningsopplegget har vist seg å kunne tilpasses forskjellige vanskelighetsnivå og studentenes forkunnskaper. Det har jevnt over fengst studentene, da det gjennom å justere vanskelighetsgraden er mulig å sørge for at alle får målt noe. Metodene gir også mulighet for å teste måleserier og få data raskt, noe som også har fengst de fleste. For den første økta som inneholder mest programmering, så kan man med fordel dele opp Parsonsproblemet for grupper som ikke har store programmeringsforkunnskaper. Vi jobber også med en liste med tips for feil som mange gjør, slik at læreren kan være forberedt på hva man sannsynligvis bør gå gjennom i plenum, samt vanlige bugs.

Rekkefølgen målemetodene presenteres ser ut til å ha en del å si for hvilke metoder studentene velger til den åpne oppgaven. Mange prøver bygge opp hele programmet på en gang og får problemer med feilsøking, noe som er enklere om man får en og en del til å fungere først. Det tyder også på at også de som har programmert en del i studiet ikke er vant til å bygge opp programmer selv.

Foreløpige funn om grunnskolelærerstudenter (1-7) sin bruk av Micro:bit til å logge data og programmering generelt forteller oss at studentene vi har intervjuet er positivt innstilt til programmering i skolen, men mener selv at de mangler kunnskap til å selv sette det i en naturfaglig kontekst. Terskelen for å ta programmering i bruk blir da høy. Dette stemmer med erfaringer fra undervisning av lærere i etter- og videreutdanning. De fleste ønsker, i tillegg til opplæring i programmeringsverktøy og metoder, å bli vist ferdige undervisningsopplegg som passer med kompetansemålene og som er rimelige klare til å brukes i skolen.

REFERANSER

- [1] Kunnskapsdepartementet, “Læreplan i naturfag (NAT01-04).” Fastsatt som forskrift. Læreplanverket for Kunnskapsløftet 2020., 2019. [Online]. Available: <https://www.udir.no/lk20/nat01-04>
- [2] Super:bit-prosjektet, “Superbit.” [Online]. Available: <https://www.superbit.no/>
- [3] The Micro:bit Educational Foundation, “Micro:bit.” [Online]. Available: <https://microbit.org/>
- [4] D. Parsons and P. Haden, “Parson’s programming puzzles: a fun and effective learning tool for first programming courses,” presented at the Proceedings of the 8th Australasian Conference on Computing Education-Volume 52, 2006, pp. 157–163.