

Arduino mikrokontroliera darbināšana Scratch

Arduino apraksts un tā programmēšanas iespējas Scratch.

Autors: Imants Gorbāns

Vietne: [E-mācības skolās](#)

Kurss: Jauno datorīku skola

Grāmata: Arduino mikrokontroliera darbināšana Scratch

Drukājis: Imants Gorbāns

Datums: trešdiena, 2014. gada 17. septembris, 02:12

Satura rādītājs

[Ievads](#)

[1. Nepieciešamais eksperimentu sākšanai](#)

[2. Gaismas diožu darbināšana ar Scratch 1.4](#)

[3. Arduino sensoru darbināšana](#)

[Pēcvārds](#)

Ievads



Visi Latvijas Universitātes Datorikas fakultātes [Jauno datoriku skolas materiāli](#) ir licencēti. Autors Imants Gorbāns šo materiālu ir licencējis ar [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported](#) licenci.

Scratch ir lieliska programmēšanas valoda un izstrādes vide pirmajiem soļiem programmēšanā. Tā izskatās vienkārša, intuitīvi apgūstama, bet dod pārsteidzoši daudz iespēju, pat vairāk kā daudzas "lielās" skolās lietotās pilnas Tjūringa valodas. Jau aplūkojam vienkārši programmās iekļaujamās, bet vispār netriviālās iespējas strādāt ar tīmekļa kameru jeb "vebkameru", mikrofonu, spēļu veidošanā izmantot blokus, kas vada objektu sadursmes, atstarošanas, virziena maiņu, ziņojumu apmaiņu utt. Lai pārsteigumu saraksts būtu pilnīgs, aplūkosim *Scratch* iespējas sadarboties ar mikrokontrolieriem (*Arduino*, *LEGO WeDo*, *PicoBoard*, *LEGO Mindstorms NXT* u.c.).

Viens no populārākajiem mikrokontrolieriem, ko plaši izmanto skolās u.c. mācību iestādādēs, tajā skaitā pasaules augstskolās, ir *Arduino*. Tas ir neliels, daudzfunkcionāls un lēti nopērkams (ebay.co.uk oriģinālais *Arduino* ar piederumu komplektu maksā ap/no 20...30 €, bet alternatīva ražotāja praktiski tāds pats mikrokontrolieris - pat zem 10 €). Neliels mīnuss ir, ka *Arduino* nav savu *Scratch* iebūvētu paplašinājumu (tādi ir *LEGO WeDo*, *PicoBoard*), priekš *Arduino* tie ir jāsameklē un jāpieinstalē, un tas viss ir izdarāms; kādēļ gan darīt vienkārši, ja var darīt sarežģīti;))

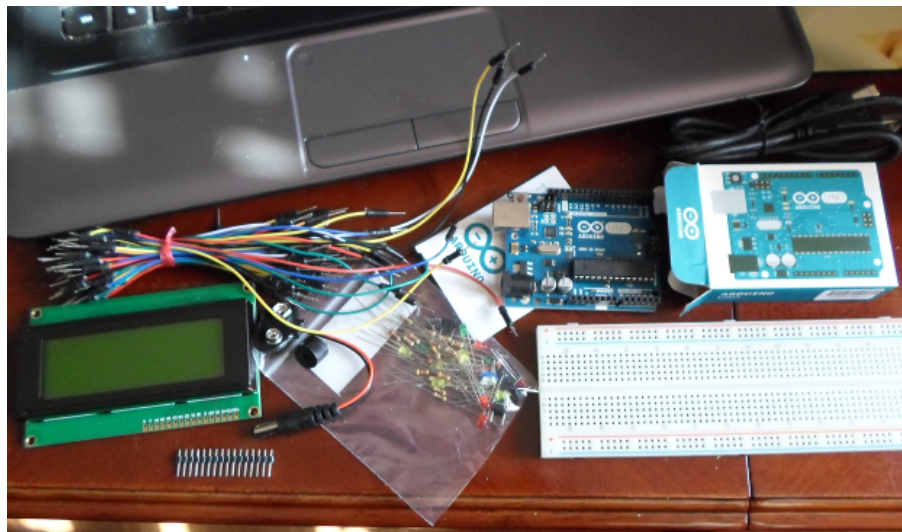
Ar *Scratch* var darbināt *Arduino* pieslēgtas gaismas diodes, displejus, detektēt sensoru datus, darbināt robotiņus ar soļu elektromotoriem, vadīt nelielas automātiskas sistēmas. Aplūkosim *Scratch* un *Arduino* sasaistes pamatprincipus un kādus piemērus, kas ļaus saprast ideju un radoši domāt, lasīt, meklēt internetā, lasīt un domāt tālāk. Te gan uzreiz jāsaprot, ka šis temats vairs nav ļoti vienkāršs, tomēr, ar zināmu izturību saņemoties, rezultāts atalgos pūles.

1. Nepieciešamais eksperimentu sākšanai

PIRMKĀRT, ir jāsapņādā šādas aparatūras jeb "dzelžu" lietas:

- Arduino UNO,
- dažas LED gaismas diodes,
- 560 Om rezistors katrai LED diodei,
- 10 kOm rezistors,
- pogas tipa slēdzis,
- izolācijas materiāla (plastmasa, stiklasķiedra utml.) plāksne visa stiprināšanai,
- nedaudz vadi,
- vēlāk varēs nopirkt arī kādu displeju, pīkstuli, soļu dzinēju.

Viens no pirkuma variantiem dots 1. att., kaut var nopirkt arī tikai vienu pašu Arduino alternatīvo mikrokontrolieri un arī pārējo sameklēt pa lēto. Arduino mājas lapa: <http://www.arduino.cc/>. Detaļu viens no komplektiem apskatāms ar skaidrojumiem <http://4tronix.co.uk/arduino/Super-Kit.php>, nopirkt var www.ebay.co.uk u.c.



1. att. Arduino mikrokontrolieris ar pārējām eksperimentiem noderīgajām detaļām.

To visu var nopirkt par dažiem desmitiem EUR eBay vai pāris Latvijas radiodetaļu specializētajos veikalos. Scratch vislabāk lietot lokālo instalāciju, var kā uz *Windows*, tā *Mac iOS* vai *Linux*. Ja lietojat *Linux*, tad jaunākajās versijās, piemēram, *xUbuntu 14.04 LTS*, vēlamams 32 bit (ja 64 bit, jāveic 32 bit atbalsta paku instalēšana).

OTRKĀRT, ir jāsaprot, kuru *Scratch* papildus programmatūru instalēsīm. *Arduino* un *Scratch* sadarbība ir iespējama vairākos veidos, bet katrā no tiem ir jāpieinstalē speciāla programmatūra

- paplašinājumi (*Extensions*). Bieži lieto paplašinājumus *A4S*, *S4A*, *S2A*, kā arī citus iesāktos projektus, iespējams, vienkāršāks kādreiz būs *Scratchbot* risinājums, ir iespēja veidot arī savus paplašinājumus, bet ar to nesāksim. Ja lietos pēdējo minēto, tad, kad tas būs gatavs, nebūs vajadzīgas nekādas papildus darbības komandrindā, nebūs arī nekādas jāielādē pašā mikrokontrolerī, kā to dara dažos citos gadījumos pa seriālo portu, izmantojot Arduino IDE.

Lai saprastu problēmas laukumu, droši vien der izlasīt http://wiki.scratch.mit.edu/wiki/Hardware_That_Can_Connect_to_Scratch, http://wiki.scratch.mit.edu/wiki/Scratch_Extension.

Uz šo brīdi var identificēt vismaz divus funkcionējošus risinājumus:

a) ja savukārt lietosiet iepriekšējo versiju - Scratch 1.4, tad paplašinājuma izvēle ir *S4A* (*Scratch 4 Arduino*). *S4A* URL: <http://goo.gl/4pU2h>. Te gan problēma ir tā, ka lai lietu GitJub lapas, uz Windows jāpieinstalē speciāla programmatūra: <https://windows.github.com/>; ja nu kas, mazāk mudžekliski būs visu lasīt šajā konspektā vai, piemēram, saitē <http://s4a.cat/>.

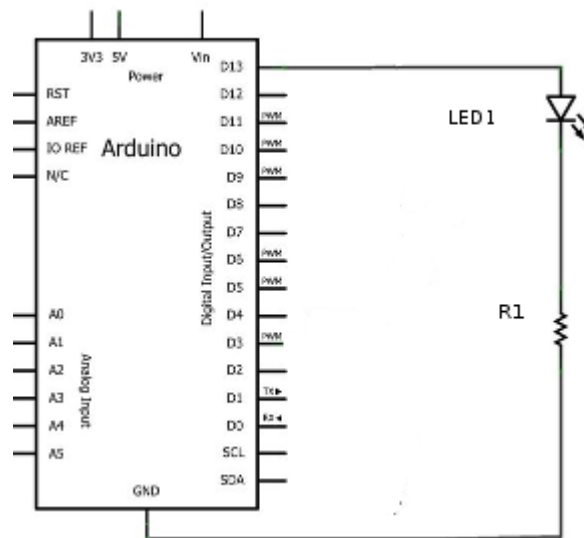
b) *A4S* (*Arduino for Scratch*), kas strādā ar Scratch 2.0 nesaistes instalāciju (*Scratch 2.0 Offline Editor*, par tā instalēšanu skatiet 1. tēmas lielāko konspektu). *A4S* saite: <https://github.com/damellis/A4S>, sekojiet instrukcijām, ja izvēlaties šo variantu;

Tālāk vispirms aplūkosim variantu a - izmantosim *S4A*, ar kuru, šķiet, viss darbojas stabili.

2. Gaismas diožu darbināšana ar Scratch 1.4

Pirmais eksperiments, ar ko parasti sāk, ir gaismas diodes (LED) ieslēgšana un izslēgšana. Idejas var smelties no vairākiem internetā brīvi pieejamiem projektiem, piemēram, ja izmantosim lokāli instalētu veco labo un joprojām lejupielādējamo *Scratch 1.4* versiju, noderīgs apraksts ir saitē: <http://www.instructables.com/id/A-Gentle-Introduction-to-Arduino-for-Scratch-Users/>. Mēs izmēģināsim dažādus variantus, bet sāksim ar šo.

Vispirms izveido slēgumu atbilstoši 2. att. dotajai shēmai, pirmajā vingrinājumā lietosim tikai gaismas diodi LED1, kas pieslēgta virknē ar rezistoru, lai pārāk liela strāva nesabojātu mikrokontrolieri.



2. att. Shēma LED pieslēgšanai pie Arduino.

Praktiski salikta šāda shēma dota 6. att. Diodi un rezistoru droši vien pareizāk ir pieslēgt darba beigu posmā, pēc visas programmatūras sainstalēšanas.

Tātad, ja lietosiet iepriekšrejo versiju - Scratch 1.4, tad paplašinājuma izvēle ir S4A (Scratch 4 Arduino). S4A saite: <http://goo.gl/4pU2h>. Te gan problēma ir tā, ka lai lietotu GitHub lapas, uz *Windows* jāpieinstalē speciāla programmatūra: <https://windows.github.com/>. Bet vienkāršāk būs saprast no saites <http://s4a.cat/>.

P.S. Der pieminēt, ka tā operētājsistēma, no kuras strādāsiet ar mikrokontrolieri, nevar būt instalēta virtuāli uz *Virtualbox* bezmaksas versijas, jo tajā noklusēti nav nepieciešamā USB

atbalsta.

DARBĪBAS NEPIECIEŠAMĀS PROGRAMMATŪRAS SAGATAVOŠANAI

1) Instalējam nesaistes *Scratch 1.4* versiju atbilstoši so JDS materiālu 1. tēmā dotajam konspektam "Programmēšanas vide jebkuram skolēnam - *Scratch* jeb datorprogrammēšanas pamati dažās stundās. E-grāmata, autors: Imants Gorbāns", saite: <http://skolas.lu.lv/mod/book/view.php?id=29857&chapterid=186>.

2) No saites <http://s4a.cat/> lejupielādējam savai operētājsistēmai atbilstošo S4A paplašinājumu: *S4A15.zip* - priekš *Microsoft Windows*, bet *S4A15.deb* - priekš *Debian* saimes Linux (*Ubuntu*, *Xubuntu* tajā skaitā, stabili atbalstītas ir vecās 12.04 LTS versijas, ne 14.04). Linux gadījumā ir papildu sarežģījums: S4A ir izstrādāts priekš 32 bitu i386 arhitektūras. Mūsdienās datori tipiski ir 64 bitu (x64 jeb x86_64 arhitektūra), līdz ar to atbilstoši <http://s4a.cat/> pamācībai, tad vispirms ir terminālī jāpieinstalē i386 atbalsts:

```
sudo dpkg --add-architecture i386
```

```
sudo apt-get update
```

```
sudo apt-get install ia32-libs
```

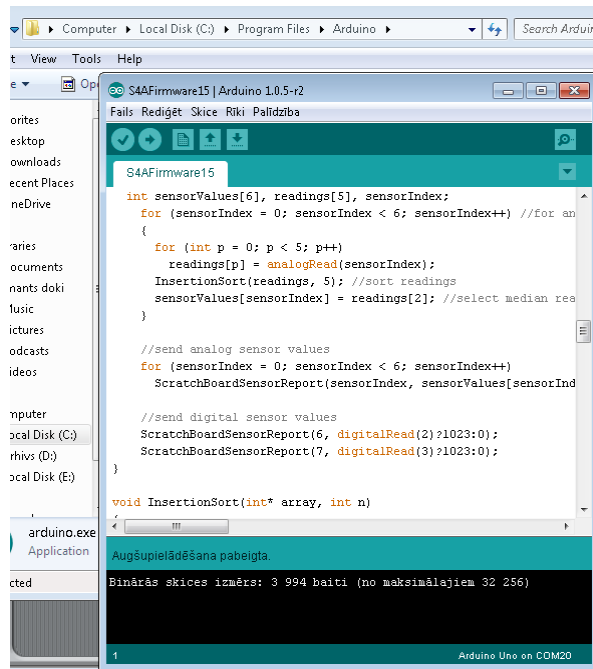
Tad uzinstalējam lejupielādēto Scratch paplašinājumu:

```
sudo dpkg -i --force-architecture S4A15.deb
```

Bet uz Windows instalēt ir vienkārši - no arhīva *S4A15.zip* palaižam *S4A15.exe*.

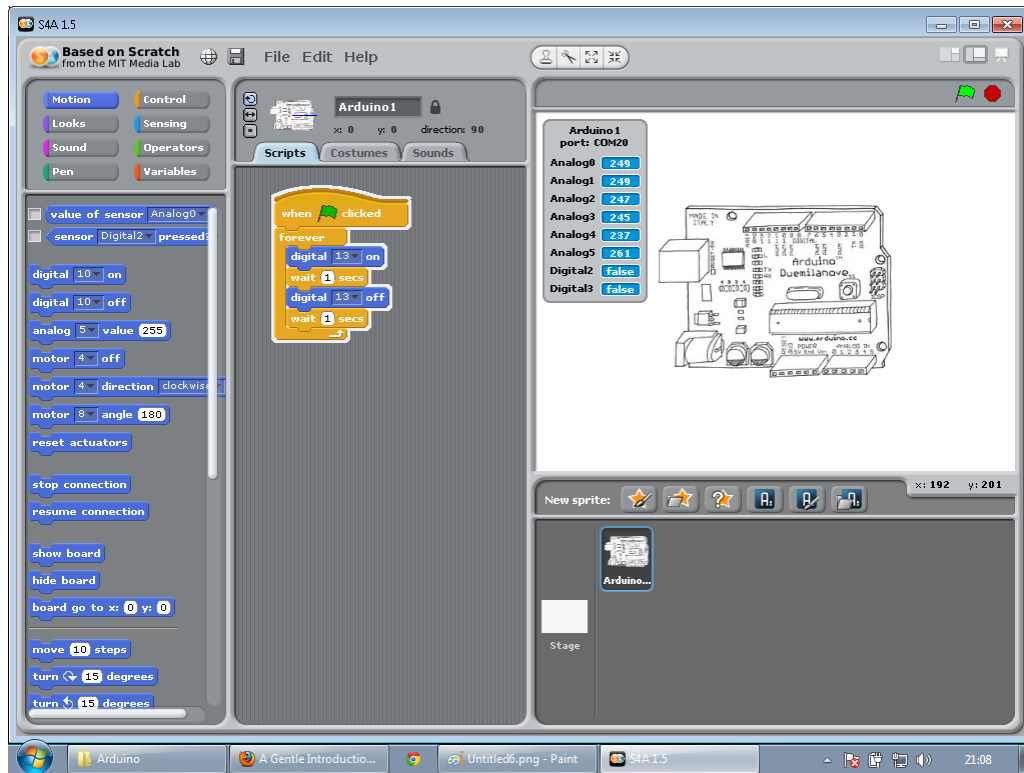
3) Tad lejupielādējam Arduino aparātprogrammatūras (*firmware*) izmainīšanas programmatūru Arduino IDE no <http://arduino.cc/en/Main/Software>, šī konspekta autors savam *Arduino Uno* izvēlējās versiju Arduino IDE 1.0.5. Tad lejupielādējam "firmvāres" failu *S4AFirmware15.ino* no <http://vps34736.ovh.net/S4A/>.

4) Tagad beidzot pieslēdzam Arduino pie datora USB porta un palaižam Arduino IDE (ja visu programmu sarakstā tas nav, tad atrodams *C:\Program Files\Arduino\Arduino.exe*). Atveram "firmvāres" failu *S4AFirmware15.ino* datorā instalētajā Arduino IDE programmatūrā, tās Tools izvēlnē izvēlamies mikrokontroliera (*Arduino Uno*) un seriālā porta (COM20, ja USB) versijas. Augšupielādējam mikrokontrolierī "firmvāri" ar *Tools > Upload*. Ja viss beidzies veiksmīgi, skats ir kā 3. att.



3. att. Arduino IDE programma pēc aparātprogrammatūras ielādes mikrokontrolierī.

5) Palaižam lokāli instalētā *Scratch 1.4* vietā tā versiju, kas papildināta ar Arduino paplašinājumu un saucas S4A. Būtībā atveras Scratch 1.4 logs, tikai darbību laukumā ir uzzīmēta Arduino grafiska bilde, bet kustību komandu kopa (zilā) ir papildināta ar jauniem blokiem, skat. 4. att.

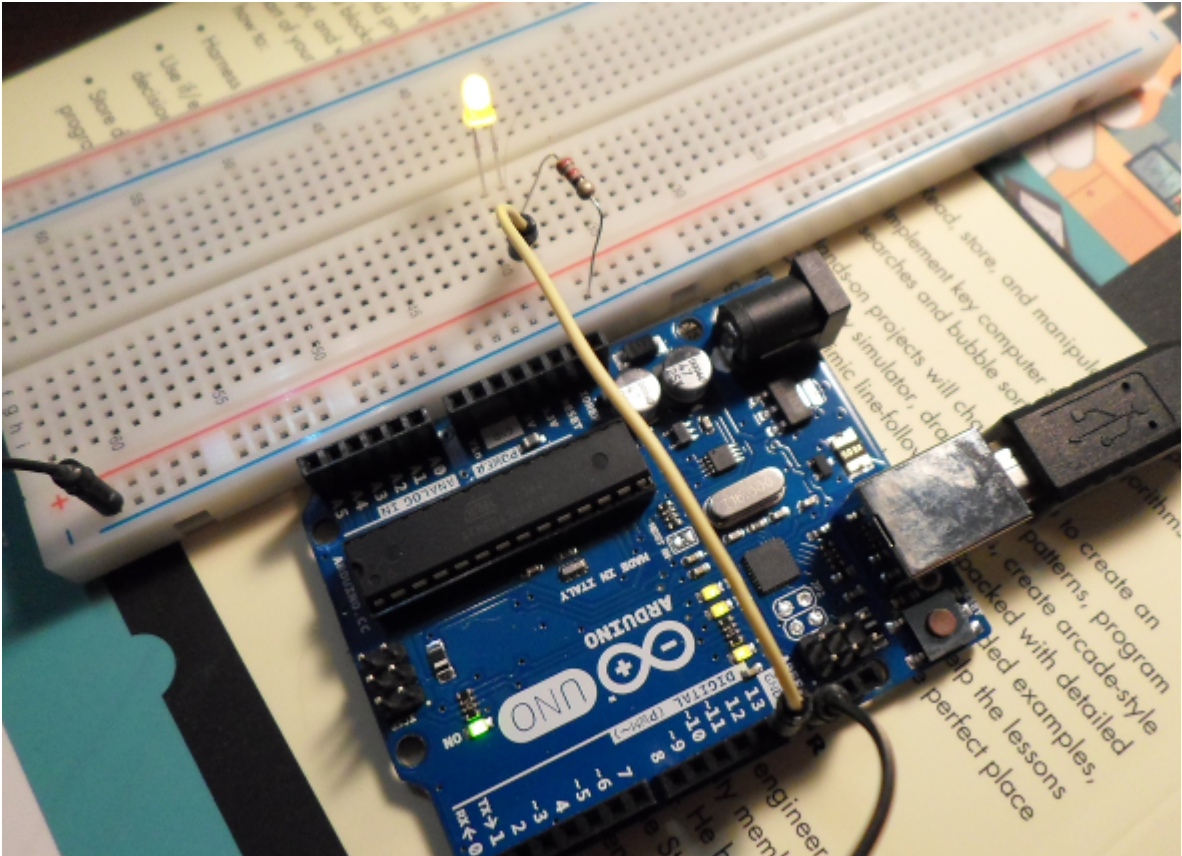


4. att. Scratch 1.4 kā S4A sagatavots darbam.

Ja iepriekšējās darbības veiktas sekmīgi, pēc zaļā karoga nospiešanas viss darbojas - *Scratch* programiņa ieslēdz un izslēdz LED diodi ik pēc 1 sek., programmas kods dots 5. att.



5. att. Scratch programma diodes ieslēgšanai.



6. att. Arduino ar gaismas diodi un rezistoru, saslēgti atbilstoši 2. att. shēmai.

Ja, nefunkcionē, pārbaudam slēguma vadus, vai ir kontakts. Diode jāiepsrauž pareizi - ar it kā novīlēto maliņu 6. att. sanāk pa labi. Caurumainajā plastmasas dēlī gar malām slēdz barošanu, bet caurumi pa 5 ir savienoti, kas arī ļauj saslēgt detaļas. Šis 1. eksperiments darbībā redzams *YouTube* video:

Šajā konspektā netika aplūkoti visi Arduino tehniskie aspekti, bet tie atklāsies pakāpeniski, līdz ar nākamajiem vingrinājumiem.

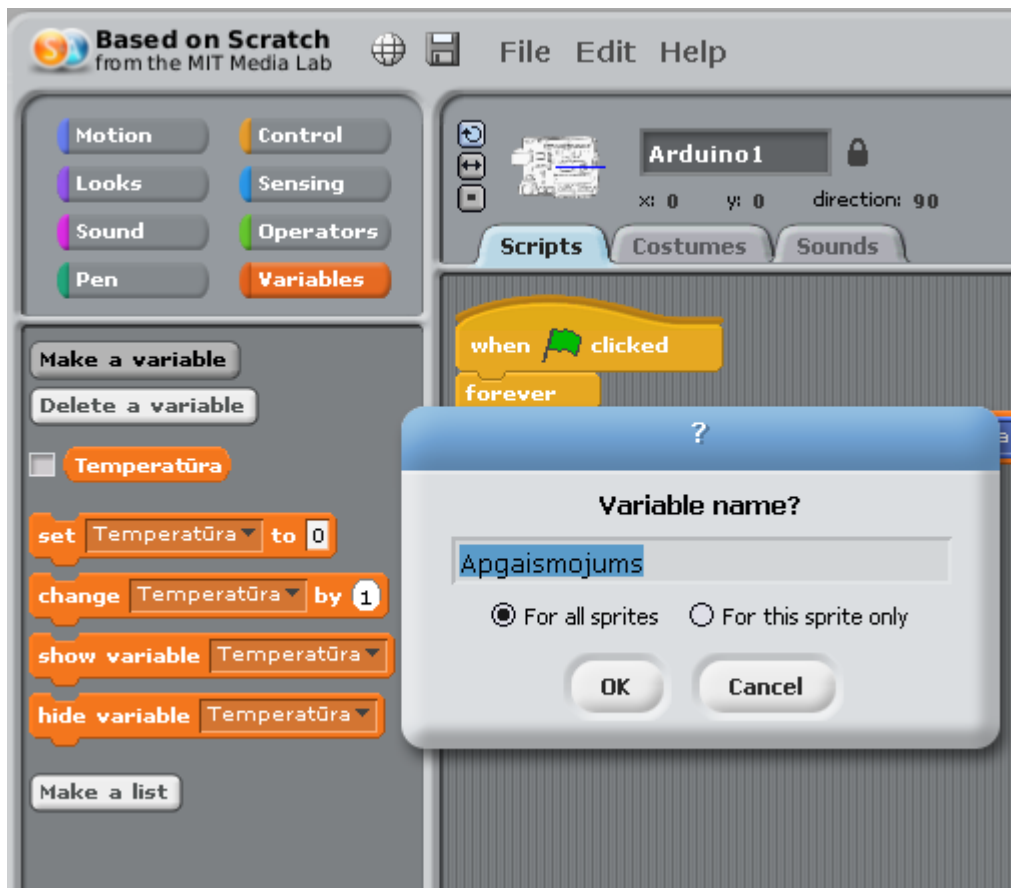
UZDEVUMI:

- a) izveidojiet shēmu un programmu ar vairākām gaismas diodēm, kas ieslēdzas secīgi ("skrejošās ugunis"),
- b) izvietojiet diodes aplī un shēmai pievienojiet pogas tipa slēdzi, kuru nospiežot mainās diožu gaismas secīgās ieslēgšanās virziens.

3. Arduino sensoru darbināšana

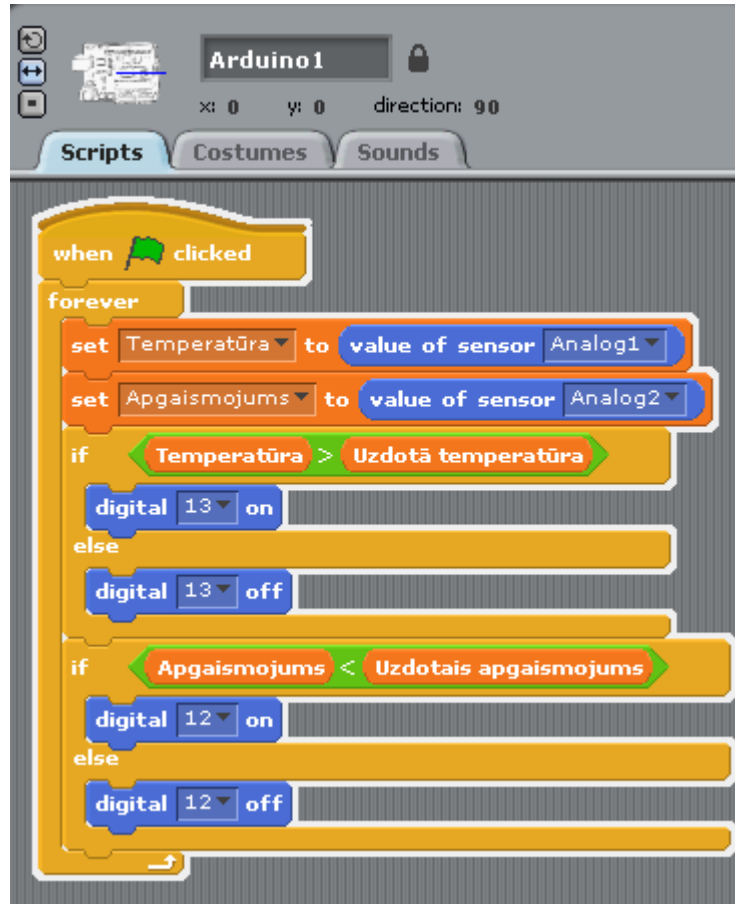
Nākamie vingrinājumi varētu būt izmantot *Arduino* pieslēgtus sensorus, elektromotorus, kā arī gaismas diožu vieto pieslēgt ierīci, kas spēj pēc nelielā signāla saņemšanas ieslēgt jau 230 V ķēdes iekārtas, piemēram, stāvlampu, ventilatoru utml.

Šajā nodaļā aplūkosim programmiņu, kas nolasa sensoru rādījumus. Par gaismas sensoru izmantosim foto rezistoru vai foto diodi, par temperatūras sensoru - termorezistoru. Tā kā sensori mēdz būt visai dažādi, tie ir jākalibrē, kas šajā gadījumā būs tikai uzdotās vērtības izvēle. Izveidosim 4 mainīgos: 2 tekošajām 2 sensoru vērtībām un 2 uzdotajām vērtībām. Uzdotās vērtības var būt piemēram, 50...100...400...800. Programmai liksim ieslēgt gaismas diodi, ja temperatūra lielāka par uzdoto (mēs pagaidām to nemēram celsija grādos, bet nosacītās vienībās), un otru diodi, ja ir apgaismojums par mazu. Izmantosim programmu *S4A* (*Scratch for Arduino* - skat. šīs grāmatas 1. nodaļu), kas ir *Scratch 1.4* apvienojums ar paplašinājumu darbam ar *Arduino*. Mainīgo izveide redzama 7. att.



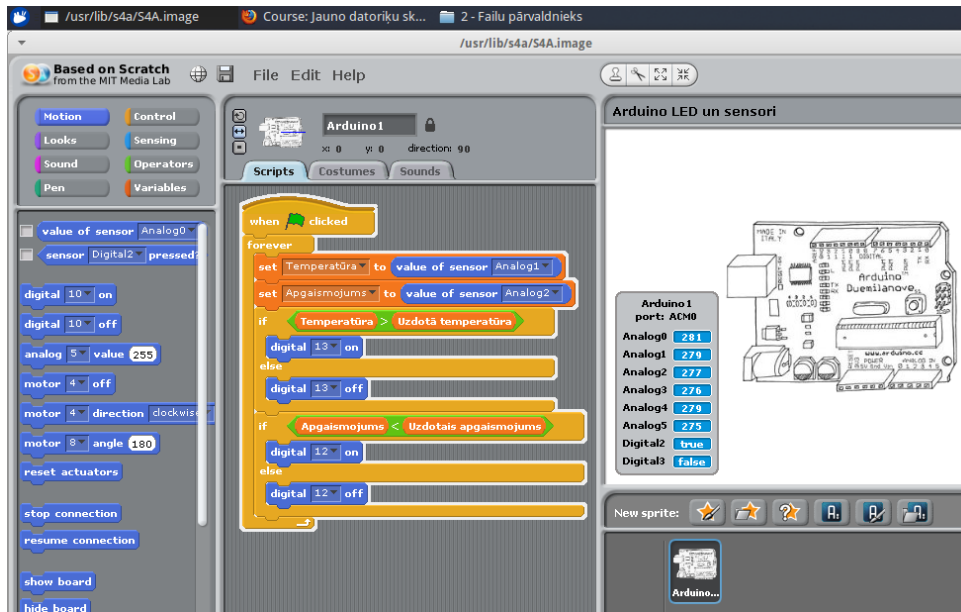
7. att. Mainīgo izveide programmā *S4A*.

Aprakstītajam scenārijam atbilstošas programmas kods ir pašsaprotams, skat. 8. att. Lai eksperiments darbotos, ir jāatrod atbilstošas vērtības, ko ievadīt mainīgajos "Uzdotais apgaismojums" un "Uzdotā temperatūra".



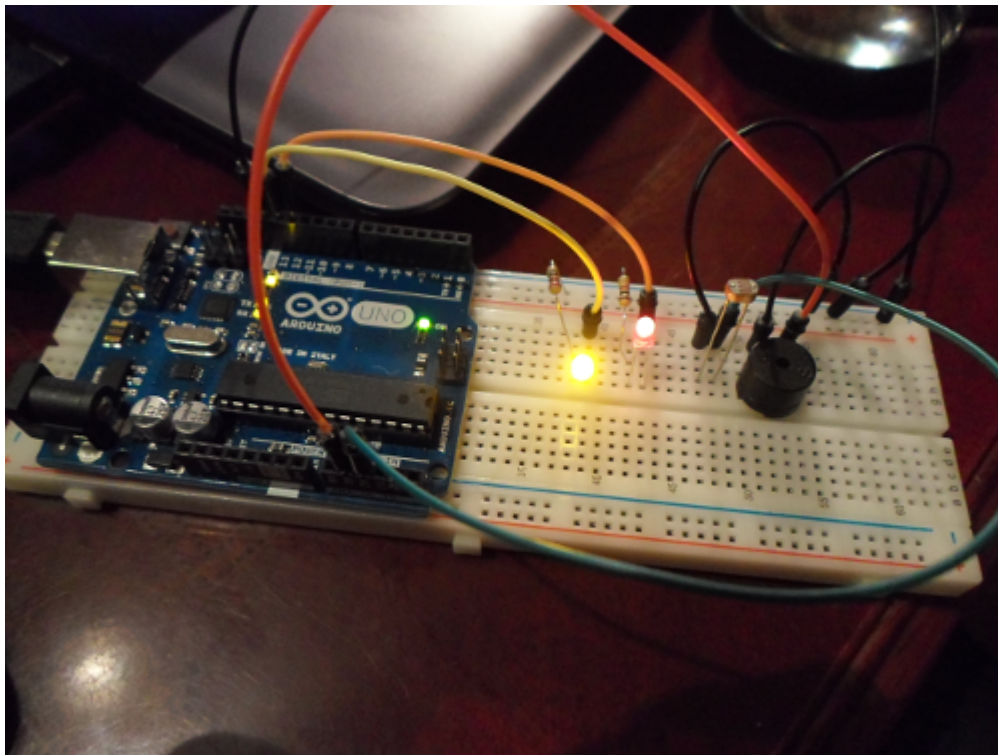
8. att. S4A programma darbam ar 2 sensoriem un 2 LED.

Dažādības un iespēju pārbaudes dēļ iepriekšējais eksperiments tikai veikts uz datora ar Windows 7 Professional, bet šis - uz Xubuntu 14.04 LTS. S4A izskats Xubuntu Linux vidē dots 9. att. Līdz ar to ir skaidrs, ka šādi eksperimenti ir tiešām lēti, jo ir veicami uz bezmaksas operētājsistēmas Linux.

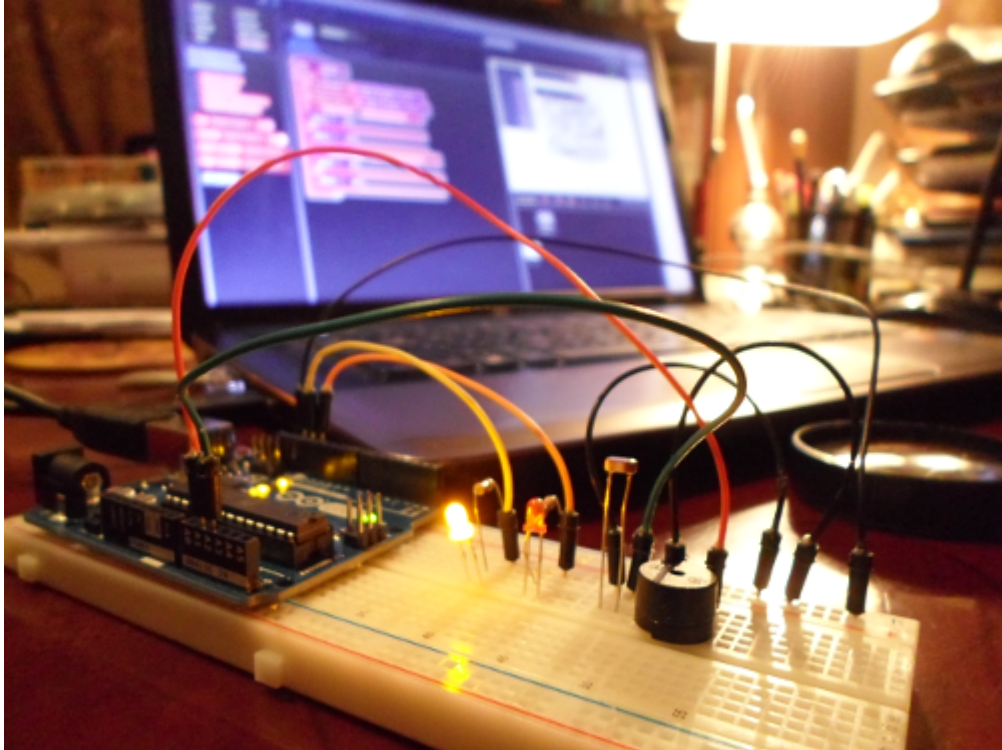


9. att. S4A strādā ar Arduino uz Linux Xubuntu 14.04 LTS.

Saslēgta shēma darbībā dota 10., 11. att. Detaļas jeb radiodetaļas var atpazīt, izmantojot skolas fizikas grāmatas vai ražotāja saiti: <http://4tronix.co.uk/arduino/Super-Kit.php>.



10. att. *Arduino* ar 2 gaismas diodēm, 2 rezistoriem un 2 sensoriem.



11. att. *Arduino* darbībā ar SA4 uz Xubuntu 14.04 LTS.

UZDEVUMI:

- a) papildiniet programmu tā, lai starta laikā tiek jautāts ievadīt temperatūru un apgaismoju, pie kura LED pārslēdzas nosacītajās vienībās,
- b) papildiniet programmu vēl tā, lai starta laikā tiek jautāts ievadīt temperatūru un apgaismoju, pie kura LED pārslēdzas, SI sistēmas mērvienībās, šim nolūkam veiciet sensoru kalibrēšanu.
- c) varbūt ir iespējams (ir, bet vajag zināšanas, detaļas un pacietību) papildināt shēmu tā, lai tā var ieslēgt galda lampu vai ventilatoru,
- d) izveidojiet datorvadāmu iekārtu skolas zvana automātiskā darbināšanai, tas varētu būt neleilas grupas nopietns projekts.

Papildliteratūra (d) uzdevumam un par radiodetaļām:

Imants Gorbāns. Dators fizikas laboratorijā skolā. Zvaigzne ABC, Rīga 2001., 144 lpp., ISBN

9984-17-795-5.

Pēcvārds

Scratch ir ne tikai vienkārša, intuitīvi apgūstama programmēšanas valoda un vide skolēniem, bet tā ir arī mūsdienīga, plašu un sarežģītu iespēju pilna. Scratch spēj atbalstīt pieslēgt šādas iekārtas (atbilstoši http://wiki.scratch.mit.edu/wiki/Hardware_That_Can_Connect_to_Scratch):

- mikrofons (kā ārējs, tā portatīvajā datorā iebūvēts),
- tīmekļa kamera jeb "vebkamera" (kā ārēja, tā portatīvajā datorā iebūvēta),
- *PicoBoard* (programmējams mikrokontrolieris),
- *LEGO WeDo* (neliels programmējams robotikas konstruktors),
- *Arduino* (programmējams lēts mikrokontrolieris),
- LEGO Mindstorms NXT (programmējams padārgs robotikas konstruktors),
- dzoistiks (ja tas kādu vēl interesē).

Pirmajiem 4 nosauktajiem ir *Scratch* iebūvēts atbalsts un var strādāt gan ar lokāli instalētu, gan pārlūkprogrammas *Scratch 2* versiju; pēdējie 3 prasa lietotājam pašam sameklēt un pieinstalēt kādu no vairākiem beta versijas līmeņa paplašinājumiem, turklāt tipiski tas darbosies tikai ar lokālo *Scratch* versiju. Pirmos 2 jau aplūkojām iepriekšējās tēmās. Mikrokontroliera izvēlē lietošanas pirmo soļu vieglums nav galvenais - svarīgāka ir cena un funkcionalitāte. Arduino var nopirkt www.ebay.co.uk un www.dx.com par sākot no 5 € (alternatīva ražotāja) līdz ~16 € (brendu); lētas ir arī pārējās detaļas eksperimentiem.

Arduino ir lētākais mikrokontrolieris, tas ir pietiekami funkcionāls arī ļoti sarežģītiem un praktiski noderīgiem projektiem, idejas skat. <http://playground.arduino.cc/projects/ideas>, <http://www.instructables.com/id/20-Unbelievable-Arduino-Projects/> u.c. Tas ir pamatojums izvēlei, kaut arī *Scratch* nav iebūvē ta atbalsta Arduino, no līdzīgām iekārtām atbalsts ir tikai *PicoBoard* un *LEGO WeDo*: šo divu paplašinājumi atrodami *Scratch* pēdējā kodu blokā "Pārējie bloki" uzspiežot uz pogas "Add an Extension".

Jūties brīvi, radiet eksperimentus, veidojiet visdažādākās programmas, koplietojiet tās Scratch portālā un pievienojiet mūsu JDS dalībnieku studijai "Jauno datorīku skolas dalībnieku kopīgote darbi": <http://scratch.mit.edu/studios/475208/>.

Veiksmi darbā un vispār!

Autors Imants Gorbāns