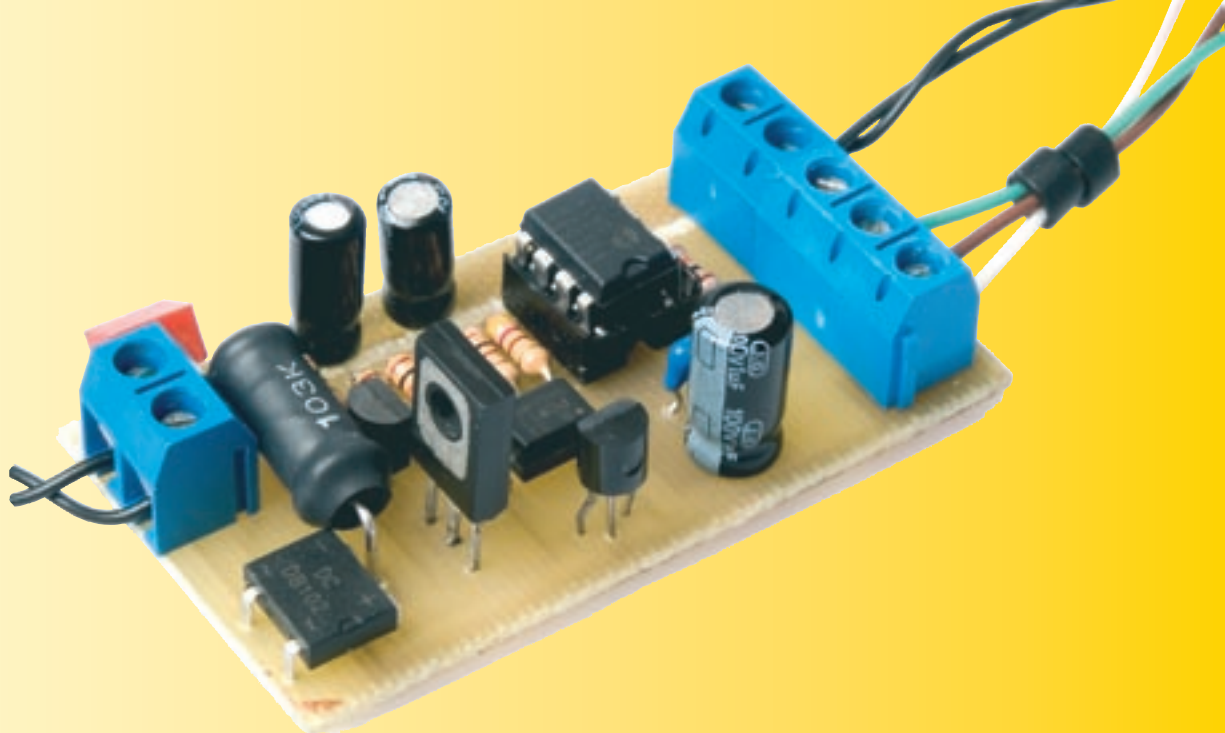


Retroketta võimalik
montaaživariant plaadil
mõõtmetega 55x30 mm



Retroketas keerleb ehk kettaga telefoni teine elu

Tänapäeva telefonid pole sageli vanematele inimestele mõeldud. Sest nagu vananedes tihti juhtub, silmad enam hästi ei seleta ja näpud tõntsid – moodsa kõnetoru tibatillukestele klahvidele on raske pihta saada. Ainus kasutatav sidevahend ehk vana kettaga «aurutelefon», mis paraku igal pool enam ei tööta. Mida teha?

Just seesuguse murega pöördus üks tuttav hiljuti minu poole, aga kuna tegelesin parasjagu oma poolelioleva programmi silumisega, ei viitsinud teemasse eriti süveneda. Ühmasin vaid, et eks ehita telefonikettale juurde kooder, mis loeks vanalt sidevahendilt valimisimpulsside ja saadaks need telefoniliinile kaasaegse kahetoonaalse tiluliluna. Et mõne loenduri ja DTMF-koodri kiviga peaks ju välja tulema.

Mõte jäi kuhugi mälusoppi kummitama ning hakkasin huvi pärast skeemi kritseldama. Selgus, et DTMF-koodri mikroskeem on poeletil täiesti olemas (MT8880), kuid võikalt kallid – üle 200 krooni. Naljakas ärioloogika – mida vähem asja tarbitakse, seda rasvasem hinnasilt talle külge kleebitakse...

Paadunud PICutajana, nagu ma olen, läks mu mõte kohe Microchipi toodangule. DTMF-mulinat annab ju programselt formeerida.

Suhteliselt lihtsa ülesande ehk telefoniketta lugemise ja vastavate DTMF-toonide

väljastamisega saab hakkama juba üsna tagasihoidlik kontrollor, näiteks PIC12F629. Nimelt protsessoril on ka tunduvalt tagasihoidlikum hind: umbes 35 krooni! Ja mis veelgi toredam – skeemidetailide arv kahele «tavaliste» kividega lahendusega võrreldes tunduvalt. Sellega oli asi otsustatud, jäi vaid loota, et kassid pole mu viimast «vaba» PIC-d hiirepüüdismänguks kasutanud.

Skeem

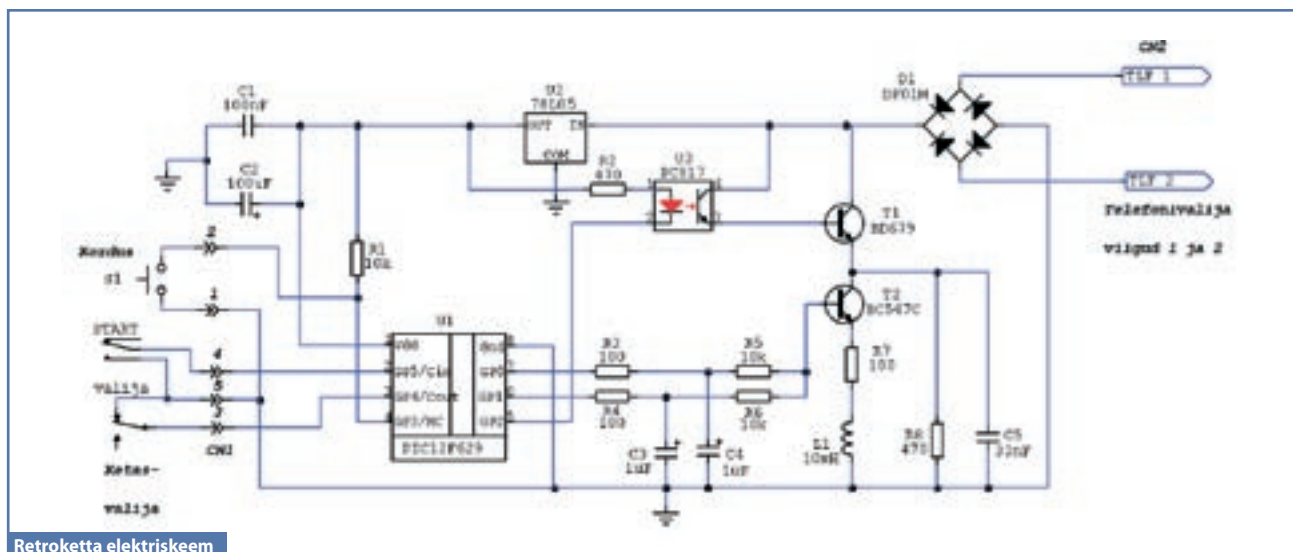
Esialgne, paberinurgale sirgeldatud skeemivariant on näha joonisel. Vaatame, mismoodi see töötada võiks.

Ketasvalijal on kaks paari kontakte. Skeemil on need kujutatud rahuolekus ehk siis, kui kasutaja pole ketast veel närviliselt väänama hakanud.

Kontaktid tähisega «START» avanevad kohe, kui valija «üles keeratakse», ja püsivad nii seni, kuni ketas jälle «koju» tagasi jõuab. Seega on protsessori jaoks selged stardi- ja

lõpusignaaliid hoobilt olemas. Kas tõesti ongi asi nii lihtne? Aga numbrid ise? Nende jaoks on kontaktid tähisega «valija», mis on enamiku ajast suletud. Kui ketas vedru mõjul tagasi vajub, avanevad ja sulguvad kontaktid vastavalt nii mitu korda, kui suure numbriga valisime. Number 7 annab 7 impulssi, 9 annab 9 ja 0 omakorda 10 pulssi! Ülesanne taandub impulsside kokkulugemisele ja sellele vastava DTMF-toonipaari tekitamisele. Polegi teab mis keeruline!





Retroketta elektriskeem

Pulse on protsessoriga mugav lugeda. Lubame igal sarakal kutsuda esile põhiprogrammi katkestuse ja siis analüüsime, milline sisend seisu muutis. Kui see oli «START», tõstame sündmuse meelepidamiseks lipu nimega «turning» (keeleseadusest ei tea me midagi) ja ootame edasi valimisimpulssi. Kuna iga värati seisundi muutus (nii nullist ühte kui ka vastupidine siire) kutsub esile katkestuse, sätime numbril loendamise toimuma vaid siirdel nullist ühte (selle illusaja teaduslik nimi on positiivne front). Seame süsteemile püsti lipu «pulse», et järgmise valija katkestuse peale tulemust mitte topelt loendada. Loendamiseks võtame ühe mälupeša ja liidame iga pulsi puhul ta sisule ühe juurde.

Ja ongi kõik: START-kontaktide avanemine teavitab proset järjekordse numbril valituks saamisest. On aeg see piiksujadana telefonijaama saata.

NB! Osa kettaid annab ühe lisaimpulsi, osa aga mitte, näiteks valides numbril 7, võib

saada 8 impulssi. Koodis taandub parandus lihtsale lahutamistehtele.

Kuidas teada saada, millise ketta me leid-sime/ostsime/rüüstasime? Laadime esmalt prosesse koodi nimega test.hex, ühendame väljundi DTMF0 ja maa vahele ahela, mis koosneb LED-ist ning 470-oomisest takistist. Anname skeemile toite +5 V (otse prose jalgadele) ja valime numbril. Kui LED vilgutab ühe võrra vajalikust rohkem, kustutame koodi (retroketas.asm) alamprogrammist DTMF1 semikooloniga märgitud rea algusest sellesama semikooloni (koodi kommentaarid aitavad õige koha leida). Kes ei viitsi assemblerkoodi süveneda, kasutab prose programmeerimiseks FTP-st leitavat faili «retromiinus.hex» (teised peaksid kasutama faili «retroketas.hex»).

DTMF

Kahetonaalse signaaliga ehk DTMF (*Dual Tone MultiFrequency*) -valimine kasutab kaheksat püsisagedust. Neli neist kuuluvad nn madalsageduslikku rühma ja neli ülejäänut kõrgemasageduslikku rühma. Tegemist on kõrvale täiesti kuuldavate helisagedustega. Tiba musikaalne häkker ehk vilistab need mikrofoniga ise liinile...

Iga numbril puhul lastakse teele üks madala rühma signaal ja kohe talle peale sobiv signaal kõrgete rühmast. Protsessoriga on seda kõige lihtsam teha nn look-up-tabeleid kasutades. Tabel 2 koodis sisaldab madalsageduste väärtusi ja tabel 3 kõrgete rühma numbreid. Väärtusi on kümme – üks iga telefoniketta numbril kohta. Kettalt loetud numbril kasutame tabeli viitena.

Prose laadib tabelist nõutud kahe sageduse koodid mällu ja käivitab koodijupi DTMF1, mis on kogu valija tuum – baidijada

ülesanne on vastavalt etteantud numbriltele genereerida nõutavad sagedused. Asi käib üsna lihtsalt: etteantud numbreid esitatakse loenduritena, mida pidevalt vähendatakse. Iga vähendamisega samas tsükli klõpsitakse väljundeid DTMF0 ja DTMF1. Tooni kestuse 1/10 sekundit määrab kolmas loendur, mis on iga tilulilu-numbril alates alati sama väärtusega.

Saadud signaalid segustatakse filterahelate R3, R5, C4 ja R4, R6, C3 abil. Lülituse mõte on prose kandilisi väljundsignaale pehmedada, sest telefoniliinil eelistatakse näha siinusesarnast võnkumist.

Optronvõti U3 on hargilüliti asemel, sellega saab PIC kordusvalimise puhul toru korraks «ära panna».

Ja lõpuks: dioodsild D1 garanteerib skeemile alati õige polarsusega toitepinge. Ei või ial teada, mismoodi montöör juhtmed ühendas... Ja tavalisi telefone see ei morjendagi, neil on samamoodi sild sees. Teeme siis kah nii, nagu targemad ette näidanud!

Kordusvalimine

See, mis on digitehnika jaoks elementaarne, teeks analoogskeemi pagana keerukaks – juttu siis kordusvalimisest. Kuna meil on digiskeem, otsustasin selle elu lihtsamaks tegeva funktsiooni koodi juurde kirjutada. Iseenesest pole selleks palju vaja – tuleb vaid iga valitav number mällu kirjutada.

Kordusvalimise käivitab nupp SW1. Esiteks imiteerib skeem jaamale toru ärapanekut, lastes opronvõtmega U3 liini vabaks. 0,7 sekundi möödudes võtab PIC uuesti toru, ootab veel veidi, et e-lõvi jõuaks raha kasseerima hakata, ning valib kogu puhvri sisu liinile. Ketta keerutamise käigus peab prose oma kivises mälus numbrilte üle arvet



ja seetõttu teab täpselt, millal vilekontsert lõpetada.

NB! Hargilüliti vajutamine nullib mälu ja kordusvalimist ei toimu!

Montaaž

Montaažiplaadi tegin piisavalt pisikeseks, et teda ketta alla (või suurema telefoni puhul selle sisse) kinnitada saaks. Joonised leiata ajakirja FTP-saidilt (<ftp://ftp.arvutikasutaja.ee/rauakool>). Detailid on «tavalised»: ¼ W takistid ja väiksemat sorti kondensaatorid. Viimaste maksimaalne tööpinge peaks olema vähemalt 16 V. Pingestabilisaator U3 (78L05) tekitab proसेle vajaliku 5-voldise toitepinge.

Drossel L1 silub omakorda pisut nurgelisi valimissignaale. Täpne väärtus on häälestamise küsimus. Kui valimine toimub ilma vigadeta, on õige nominaal leitud. Ise kasutasin maketi jaoks ühe vana taskuraadio väljundtrafot. Sobiva väärtusega drosseli leiab elektroonikakauplusest.

Plaadi väljundid tuleb ühendada telefoni skeemi kontaktidega, kuhu enne modimist lülitus originaalne mehaaniline valija. Valija pulseerivad kontaktid tuleb lühistada, meie skeem ühendub valimise algusest teavitavate kontaktide asemele.

Häälestamine

Timmimist skeemis palju ei ole. Esiteks tasub hargilt tõstetud toru korral mõõta proсе toitepinget – see peab olema umbes-täpselt



Disklaimer

■ DTMF-valimise algoritmi pärineb Talking Electronics'i võrgulehelt, mida on kohandatud PIC12F629 jaoks (alguses skeemis oli PIC12C508, mis on vaid 1 kord progetav). Telefoniketta lugemise kood ning kordusvalimiste lahendus on juba puhtalt omalooming.

■ Avaldatud skeemid ja koodid on kasutamiseks GNU

avaliku litsentsi tingimuste kohaselt. Skeemide/koodide ja/või selle osade oma tarbeks kasutamine on lubatud, lausa soovitatav. Autori mainimine pole kohustuslik. Mingitele kahtlastele kaitsjatele tolli maksma ei pea.

■ Kommertsotstarbeks kasutamisel peab olema autori kirjalik luba (mina tahan ka šampust juua ☺).

5 volti. Nüüd proovime numbreid valida. Kahtluste korral on hea, kui läheduses viibib mõni vanem inimene, kes «aurutelefoni» kasutamise korda tunneb...

Kui ketta tagasikeeramise lõpus midagi ei juhtu, on viga kõige tõenäolisemalt protsessori koodis, õigemini – kood on kuidagi vigaselt kivisse kirjutatud. Aga pole hullu, võime uuesti proovida, sest mikroskeemi annab korduvalt üle kirjutada.

PIC12F629 sisaldab sisemise generaatori sageduse korrigeerimiskoodi, mis on iga konkreetse kivi jaoks erinev ja on tehases kivi sisse salvestatud. Programm laadib esimese asjana pärast käivitumist selle väärtuse ROM-ist ja korrigeerib vastavalt proсе töösagedust. Enamik PIC-kivide programmeerijaid (IS-prog jt) oskab seda numbrit kivist lugeda.

Programmeerimise käigus teatab programm kasutajale, et OSCCAL-i väärtus on xxxx, kuigi püüad kirjutada hoopis väärtust yyyy, kas kasutate kivist loetut? Õige vastus on: JAH, kasuta kivist loetut (tehase CAL-väärtust, mitte koodifailist leitud). Vastasel korral ei saa proсе taktigeneraatorit enam kunagi õigel sagedusel tööle ja ka DTMF-koodid on vale sagedusega. Lugege siis hoolega programmeerijate antavaid teateid...

Konkreetselt ketta puhul tuleb kindlaks teha, kas see annab ühe liigse impulsi või (nagu minul juhtus) väljastab kohe õige arvu särakaid. Kontrollimise protseduurist ja testsofist vestsin juba eespool.

Enne skeemi ehitamist heitke pilk ajakirja FTP-saidile. Panen sinna parandused ja täiustused, mis skeemi käimajooksmise käigus tekkida võivad. Seega võib «õige» skeem ajakirjas avaldatust erineda! Kõik koodid saab tõmmata samalt saidilt. Võimalikest «kaladest» palun teada anda, parandan ära!

Veljo Sinivee
felc@edu.ttu.ee

Lisafunktsioonid

On hulk laiendusvõimalusi, mida mainitud koodis veel pole, kuid mida oleks suhteliselt lihtne lisada. Näiteks:

■ **10 numbril mälu:** PIC-I on sisemine toitest sõltumatu EEPROM-mälu, kuhu saaks salvestada üsna mitu numbrit. Kuna kettal on numbreid kümme, on lihtsam teha ka sama palju kiirvalikukode. Salvestamine võiks käia nii: vajutame kaks korda kordusvalimispupu (*Redial*), valime kettaga abonendi numbrit ja vajutame veel kaks korda sama nuppu. Telefon paneb mõistmise märgiks toru korraks ära. Salvestatud numbrit kiirvalimine: vajutame REDIAL-nupu alla ja valime kettal vajaliku kiirvalikukoodi, laseme nupu lahti – telefon valib vastavasse mälu salvestatud abonendi numbrit.

■ **Lapse telefon:** toru tõstmisel valitakse automaatselt proсе EEPROM-mälu salvestatud number. Ilmselt peab REDIAL-nupuga tegema ümberlülituse «normaalsele» valimisele.

■ **Kõne kestuse näit:** kuna proсе istub pärast valimist nagunii käed rüpes, las teeb midagi mõistlikku. Ajaarvamiseks saab kasutada sisemisi taimerid, näit võiks minna vähese voolutarbega LCD-le (nt Nokia 3310-st rüüstatule). Juhtotstena saab kasutada pärast valimist vabu viike GP5, GP1, GP0. Viimast kahte tuleb plöksutada kiiresti, et keskjaam midagi kahtlustada ei oskaks. LCD juhtsignaalid ongi enamasti lühikesed ja probleeme ei peaks tekkima. NB! Indikaatori taustvalgustuse jaoks liinilt voolu ei jätku!

■ **Kettaga mobiiltelefon:** tööpoolest, telefoni korpusesse saab peita vana, kurvast sahtlis vedeleva «kännu». Toide tuleks tema akult (või laadijalt), numbrit valimine piirdub mobilale vastavate käskude saatmisega. Näiteks kui tahame valida numbrit 1234, peame saatma mobilale jada: ATD 1234. Kõnetoru vastab: «OK». Keerukamas lahenduses kontrollib proсе enne valimist, kas telefon on ikka levis, ja kui ei ole, annab sellest märku. Võrgusolekut saab kontrollida käsuga AT+CREG? Positiivne vastus: AT+CREG 0,1. Enamasti on mobilaga suhtlemisel side parameetrid järgmised: 9600,n,8,1. Kirjeldatud kood on mul juba välja töötatud, ehk saab mõnes järgnevas artiklis seda tutvustada.