

**Forritið GIF10**

**Einar Hrafnkell Haraldsson**

**Greinargerð EHH-94-01**

## Forritið GIF10

Forritið GIF10 er til að lesa gögn inn í PC tölvu úr sérstöku lestrartæki af gerðinni Wild Gif10. Lestrartækið tekur minniskubba úr landmælingartæki. Þessi greinargerð er ekki endanleg lýsing á forritinu heldur skýrir það sig sjálf við notkun.

Lestrartækið GIF10 sendir út gagnaskrár á textaformi. Sendingu er komið af stað ýmist frá takkaborði á því eða frá PC tölvunni. Forritið er notað þannig að fyrst er kveikt á lestrartækinu, það síðan tengt PC tölvunni og skipunin GIF10 er gefin í PC tölvunni. Forritið GIF10 er hægt að ræsa með nokkrum rofum, sem stilla samskiptahraða á milli tölvunnar og lestrartækisins. Eftirtaldir rofar eru notaðir:

/c[1,2] = comport 1 og 2  
/b[110, 300, 1200, 2400, 4800, 9600, 19200] = sendihraði  
/p[0=no, 1=even, 2=odd] = parity  
/s[1,2] = stoppbítar  
/d[7,8] = fjöldi bita í sendingu

Dæmi: GIF10 /c1 /b9600 /p0

Sjálfgefin stilling er: comport1, 9600, no parity, 1 stoppbíti, 8 bitar í sendingu.

Lestrartækið þarf að vera stillt á hliðstæð gildi í COMM valmyndinni:

```
BAUD          9600
PARITY        NONE
PROTOD       XON/XOF
STOPBIT       1 BIT
ENDMARK       CR/LF
CONNECTED    AS DCE
```

Í upphafi spyr forritið um nafn á gagnaskrá, sem mun geyma gögnin frá lestrartækinu. Að því loknu þarf að gefa skipun til lestrartækisins um að hefja sendingu á mæligögnum. Ef PC tölvun hefur ekki undan að taka við gögnum frá lestrartækinu, þá stoppar hún það með XOFF og setur það af stað aftur með XON. Í lok sendingar mæligagna frá lestrartækinu, sendir það frá sér Control-C. Við það lokar forritið GIF10 gagnaskránni í PC tölvunni og hættir keyrslu. Þetta Control-C er ekki geymt í gagnaskránni. Ef forritið GIF10 er keyrt áður en kveikt er lestrartækinu er hætta á að aukastafir komi fremst í gagnaskrána. Línur í gagnaskránni enda á "Newline".

Hægt er að stöðva forritið með því að ýta á Alt-X. Ef gagnaflutningurinn er í gangi, þá er gagnaskránni er lokað um leið. Ef ýtt er á Alt-U á meðan gagnaflutningur er í gangi lokar forritið gagnaskránni en sýnir á skjánum gögnin, sem koma inn. Þá er hægt að ýta á Alt-U aftur og hefja gagnaflutning að nýju.

```

/* Forrit til þess að lesa gögn frá Wild Gif 10.
   Tækið er til að lesa úr minnishylkjum úr landmælingartæki
   Þetta tæki er í notkun hjá Landmælingum Orkustofnunar.
   Stofninn í þessu forriti er term forritið og er smá rusl frá því hér.
   22. mars 1994 Einar Hrafnkell Haraldsson
*/
#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <stdlib.h>
#include <dos.h>
#include "async2.h"
#include "sp2.h"
#define FALSE 0
#define TRUE 1
#define ESC 27

void loka_port(int);
void opna_port(void);
void syna_hjalp(void);
void senda_skra(void);
void upptaka(void);
void nafn_upptoku_skrar(void);
void hallo(void);
void syna_stillingar(void);
char *velja_skra(char *);
long file_size(FILE *stream);
int getopt(int argc, char *argv[], char *options);

FILE *upptokuskra;
char UpptokuNafn[80] = "tmp.hex"; /* Nafn á upptökuskrá */
int er_ad_taka_upp = FALSE;
int hef_opnad_upptoku_skra = FALSE;

int com_port = COM_1;
int baud = 9600;
int parity = 0;
int stop_bits = 1;
int data_bits = 8;
int hand_shake_mode = HND_SOFT;

int err;
int vinna;
int TakaMedCr = FALSE;
char *utgafa = "Forritið GIF-10, útgáfa 1,00 - 22/03/94 - eh";

main(int argc, char **argv)
{
char cc, ck;
char opt;
extern char *optarg;

hallo();
while ( EOF != ( opt = getopt(argc, argv, "c:b:p:s:d:") ) ) {
switch(opt) {
case 'c' : com_port = atoi(optarg) -1; break;
case 'b' : baud = atoi(optarg); break;
case 'p' : parity = atoi(optarg); break;
case 's' : stop_bits = atoi(optarg); break;
case 'd' : data_bits = atoi(optarg); break;
}
}
vinna = TRUE;
opna_port();
fputs("\nBest er að kveikja á GIF10 áður en forritið er keyrt.\n", stdout);
upptaka();
while(vinna) {
checkcom(com_port, &cc, &err);
if(!err) {
if(cc == 03) {
upptaka();
break; // Ef 03 kemur frá Wild GIF10, þá er sendungu lokið.
}
}
}
}

```

```

}
putch(cc);
    if(er_ad_taka_upp) {
        if (cc == '\r') {
            if (TakaMedCr)
                fputc(cc, upptokuskra);
            else
                ;
        }
        else
            fputc(cc, upptokuskra);
    }
}
if(kbhit()) {
    ck = getch();
    if(ck == '\r') {
        sendcom(com_port, '\n', &err);
        sendcom(com_port, ck, &err);
    }
    if(ck == 0) {
        ck = getch();
        switch(ck) {
            case 120: /* Alt-1 senda SND1 til lestrartækis */
                sendcom(com_port, 'S', &err);
                sendcom(com_port, 'N', &err);
                sendcom(com_port, 'D', &err);
                sendcom(com_port, '1', &err);
                sendcom(com_port, '\n', &err);
                sendcom(com_port, '\r', &err);

                break;
            case 22: /* Alt-U Upptaka */
                upptaka();
                break;
            case 30: /* Alt-A Upptökuskrá */
                nafn_upptoku_skrar();
                break;
            case 31: /* Alt-S Senda skrá */
                senda_skra();
                break;
            case 45: /* Alt-X Hætta */
                vinna = FALSE;
                break;
            case 59: /* F1 Hjálp */
                syna_hjalp();
                break;
        }
    }
    else
        sendcom(com_port, ck, &err);
} /* Endir á switch(ck) */
} /* Endir á vinnulykkju */
loka_port(com_port);
if(er_ad_taka_upp)
    printf("Loka upptökuskrá.\n");
fclose(upptokuskra);
return 0;
} /* End main */

void hallo(void)
{
    fputs("\n FORRITID GIF-10\n ", stdout);
    fputs(utgafa, stdout);
    fputs("\n", stdout);
    fputs(" Hætta með Alt-X\n", stdout);
    fputs(" Hjálp : F1\n", stdout);
    fputs(" Sjálfgefin stilling: com1, 9600, no par, 1 stopp, 8 data\n", stdout);
    fputs(" Rofar við keyrslu:\n", stdout);
    fputs(" /c[1,2] = comport 1 og 2\n", stdout);
    fputs(" /b[110-19200] = sendihraði\n", stdout);
    fputs(" /p[0=no, 1=even, 2=odd] = parity\n", stdout);
}

```

```

fputs("    /s[1,2] = stoppbitar\n", stdout);
fputs("    /d[7,8] = fjöldi bita í sendingu\n", stdout);
fputs("    Dæmi: GIF10 /c1 /b2400 /p0\n", stdout);
}

void opna_port(void)
{
    opencom(com_port, baud, parity, stop_bits, data_bits, &err);
    sethandshakemode(com_port, hand_shake_mode);
}

void loka_port(int com_port)
{
    closecom(com_port);
}

void senda_skra(void)
{
    FILE *skra;
    char *nafn, byte, sv[15];
    int err, villa = FALSE;
    if ((nafn = (char*)malloc(15)) == NULL) {
        fputs("Villa: Ónógt minni fyrir nafn.\n", stdout);
        exit (1);
    }

    /* nafn = velja_skra("tmp.hex"); */
    fputs("Hvaða sendiskrá: ", stdout);
    fgets(nafn, 15, stdin);
    if ((skra = fopen(nafn, "rt")) == NULL) {
        fputs("Villa: Get ekki opnað sendiskrá.\n", stdout);
        villa = TRUE;
    }
    if(!villa) {
        fputs("Stærð skráar er : ", stdout);
        itoa(file_size(skra), sv, 15);
        fputs(sv, stdout);
        fputs(" byte\n", stdout);
        while((byte = fgetc(skra)) != EOF) {
            sendcom(com_port, byte, &err);
        }
        fclose(skra);
    }
}

char *velja_skra(char *sjalfgefid_nafn)
{
    static char cp[15];
    fprintf(stdout,
        "Gefðu nafn á skrá. Sjálfgefið nafn er %s :", sjalfgefid_nafn);
    fgets(cp, 14, stdin);
    if (strcmp(cp, "\n") == 0)
        return (sjalfgefid_nafn);
    else {
        cp[strcspn(cp, "\n")] = '\0'; /* Klippa \n aftan af
skráarnafni. */
        return(cp);
    }
}

void syna_hjalp(void)
{
    fputs("\n HJÁLP\n", stdout);
    fputs("    Alt-X : Hætta\n", stdout);
    fputs("    Alt-S : Senda skrá\n", stdout);
    fputs("    Alt-U : Upptaka á skrá\n", stdout);
    fputs("    Alt-A : Nafn á upptökuskrá\n", stdout);

    hallo();
}

```

```

void upptaka(void)
{
    if(!hef_opnad_upptoku_skra)
        nafn_upptoku_skrar();
    if( er_ad_taka_upp == FALSE) {
        er_ad_taka_upp = TRUE;
        fputs(">>>>>>> Upptaka hefst <<<<<<<\n", stdout);
        fputs("Hér eru 2 aðferðir til að segja útlestrartækinu að senda gögnin.\n", stdout);
        fputs("1) Nota SEND aðgerðina útlestrartækinu.\n", stdout);
        fputs("2) Gefa skipanir beint frá lyklaborði PC tölvunnar.\n", stdout);
        fputs(" Það, sem er skrifað sést ekki á skjánum. <ret> er Return takkinn.\n", stdout);
        fputs(" Dæmi: SND1<ret> (skráningartækið sendir skrá 1)\n", stdout);
        fputs(" Dæmi: SND1,1,150<ret> (skrá 1, blokk 1 til 150)\n", stdout);
        fputs(" Ef ýtt er á Alt-1 hér sendir PC tölvun skipunina SND1 til lestrartækisins\n",
        stdout);
        fputs("Hér má einnig gefa aðrar skipanir til GIF10 eins og \n", stdout);
        fputs("ERA<File> CRE<File>,<Size> DIS<File> INI RECMODULE\n", stdout);
        fputs("Hér má hætta í forritinu með því að ýta á Alt-X\n", stdout);
    }
    else {
        er_ad_taka_upp = FALSE;
        fputs(">>>>>>> Upptöku lokið <<<<<<<\n", stdout);
    }
}

/* char *nafn; */

void nafn_upptoku_skrar(void)
{
    struct find t ffbk;
    char sv[15], sv2[15];
    int svar, svar2;
    if( er_ad_taka_upp) {
        fputs("Er að taka upp\n", stdout);
        return;
    }

    strcpy(UpptokuNafn , SkraarNafnSkrifa());
    if ((upptokuskra = fopen(UpptokuNafn, "wb")) == NULL)
    {
        fprintf(stderr, "Villa: Get ekki opnað skrána %s.\n", UpptokuNafn);
        exit(1);
    }
    fprintf(stderr, "\n");

    hef_opnad_upptoku_skra = TRUE;
}

long file_size(FILE *stream)
{
    long curpos, length;

    curpos = ftell(stream);
    fseek(stream, 0L, SEEK_END);
    length = ftell(stream);
    fseek(stream, curpos, SEEK_SET);
    return length;
}

```