VERSATILE MULTI–CHANNEL ANALOG–TO–DIGITAL CONVERTER FOR EXPERIMENTAL STUDY OF THE ATMOSPHERE

A.P. Rostov

Institute of Atmospheric Optics,
Siberian Branch of the Russian Academy of Sciences, Tomsk
Received December 13, 1994

A modification of a 10–bit analog–to–digital converter with the RS–232 interface has been proposed, which is compatible with any personal computer and instruments and installations used in international experiments.

Recently complex field experiments have been more and more often performed in collaboration with foreign colleagues in different countries of different continents. In these cases, it is difficult to interface the equipment delivered by our scientists with computers of different types used in experiment.

Since practically all instruments being developed now feature an analog–to–digital converter (ADC), to solve the problem I have developed a multi–channel ADC equipped with the international standard RS–232 interface.[1] All modern personal computers have such an interface. To operate properly, protocol of this interface calls for intelligent ADC. It is desirable to include a microprocessor. The Intel i80196 or i8051GB single–crystal microcontrollers equipped with a 8–bit ADC and a series port are best suited for these purposes. They are good but very expensive modern microcontrollers. Meanwhile, electronics designers have usually only the first modifications of single–crystal microcontrollers and ADC produced in our country in the 80's. Allowing for the aforesaid, a modification of the cheap device built around such microcircuits is proposed. Figures 1 and 2 illustrate the circuit schematic of the device.

Dynamic range of analog signals to be digitized seldom exceeds 60 dB and a 10–bit ADC is sufficient in most cases. The K1113PV1A single–crystal ADC (see Ref. 2) that is an analog of the Analog Devices Inc. AD571 is well suited for this purpose. This device was produced by the Riga Association "Al'fa" in due time. The conversion time less than 30 μs is more than sufficient since the RS–232 interface restricts the band rate. When the band rate is 9600 bands/s, the real bit rate is about 800 bytes/s.

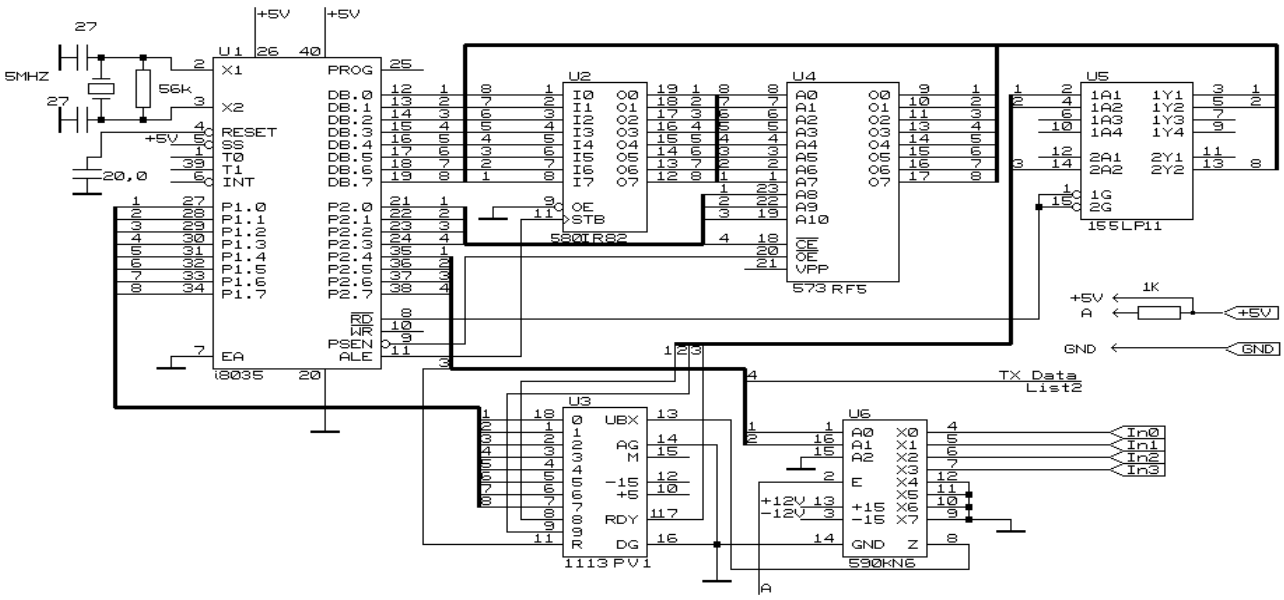

FIG. 1. Circuit schematic of ADC (beginning).

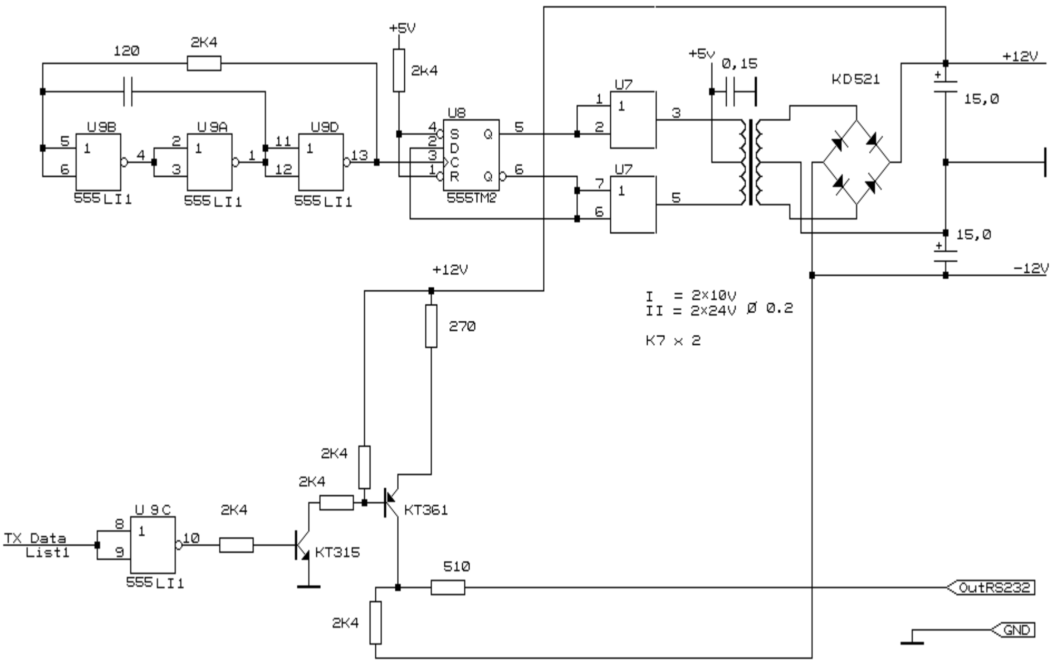

FIG. 2. Circuit schematic of ADC (end).

To control ADC operation and reception and transfer of information as well as its preliminary processing, the KR1816VI35 single–crystal microcontroller[3] is used. It is an analog of the Intel i8035 device. In due time, it was produced in series by the electronics plants of the former USSR. This microcomputer has backing program storage (BPS) that makes it possible to change the ADC operation regime and algorithms for preliminary signal processing by simple replacement of read–only memory (ROM). One version of control program written in assembly language ASM–48 is described below.

```
;=================================
;Control program for ADC–10
;with the KR1816VI35 microcontroller
;-----------------------------------------------------------------------------
-----
;A. Rostov, Tomsk, E-mail andrey@rostov.tomsk.su
;-----------------------------------------------------------------------------
-----
;Output in RS–232(8N1) format
;The marker AA55H is the first, then the three two–byte words containing the values of the three input parameters follow
;-----------------------------------------------------------------------------
-----
;Port 1          input of the ADC low–order byte
;Port 2(4p,5p)   input of the ADC high–order bits
;Port 2(6p)      tag of ADC readiness
;Port 2(7p)      RS–232 output

Volum0    .EQU 32
Volum1    .EQU 34
Volum2    .EQU 36
Volum3    .EQU 38

.ORG 0
JMP Start
.ORG 100h

;subroutine of a 100 µs delay uses R7
Delay1:     mov R7,#10
Loop1:      djnz R7,Loop1
ret

;subroutine of byte output from the accumulator to port P2.7
Out232:

mov         R6,#8
anl         P2,#01111111B ;Start bit
            call Delay1

Rotate:     rrc A
            jc OFF
ON:         anl P2,#01111111B
            jmp Delay
OFF:        orl P2,#10000000B
            jmp Delay
Delay:      call Delay1
            djnz R6, Rotate
            orl P2,#10000000B ;Stop bit
            call Delay1
            ret

Start:      .ORG 200h

;Port relocation

            mov A,#OFFh   ;input
            outl P1,A
            anl P2,#10001111B    ;output(4,5,6,7 bits)

Loop:

            mov R0,#Volum0+1

;Digitization of channel 0
            mov A,#11001111B
            outl P2,A
            anl P2,#10111111B
```

```
Ready0:   ins A,BUS
          jb7 Ready0
          ins A,BUS
          anl A,#3 ;put on a mask because the first and
                second bits carry information
          mov @R0, A
          in A,P1 ;peek the ADC low–order byte
          dec R0
          mov @R0,A
          inc R0
          inc R0
          inc R0 ; shift the pointer to the next reading

;Digitization of channel 1

        orl P2,#01010000B ;digitization of channel 1
        anl P2,#10111111B

Ready1:   ins A,BUS
          jb7 Ready1
          ins A,BUS
          anl A,#3 ;put on a mask because the first and
                second bits carry information
          mov @R0,A
          in A,P1 ;peek the ADC low–order byte
          dec R0
          mov @R0,A
          inc R0
          inc R0
          inc R0 ;shift the pointer to the next reading

;Digitization of channel 2

          mov A,#11101111 ;digitization of channel 2
          outl P2,A
          anl P2,#10111111B

          Ready2: ins A,BUS
          jb7 Ready2
          ins A,BUS
          anl A,#3 ;put on a mask because the first and
                second bits carry information
          mov @R0,A
          in A,P1 ;peek the ADC low–order byte
          dec R0
          mov @R0,A
          inc R0
          inc R0
          inc R0 ;shift the pointer to the
          next reading

;Digitization of channel 3

          orl P2,#01110000 ;digitization of channel 3
          anl P2,#10111111B

Ready3:   ins A,BUS
          jb7 Ready3
          anl A,# ;put on a mask because the first and
              second bits carry information
          mov @R0,A
          in A,P1 ;peek the ADC low–order byte
          dec R0
          mov @R0,A

;serial output of digitized parameters
          mov A,#0AAh ;first output the marker
          call Out232 ;AAh
          mov A,#55h
          call Out232 ;55h

          mov R0,#Volum0
          mov R2,#6 ;number of output bytes
Loop2:    mov A,@R0
          call Out232
          inc R0
          djnz R2,Loop2

          jmp Loop ; do digitization cycle
.END;                     ;program end
```

To commutate analog signals, the KR590KN6 commutator, certainly commutating signals with an amplitude of ±10 V, is used. For operation of the ADC, commutator, receiver, and transmitter of the RS–232, a voltage of ±12 V is supplied. For convenience, a voltage transformer from +5 V to ±12 V was included in the device. So, a voltage of +5 V is supplied to the device. In this case, the current consumed is no more than 500 mA.

To receive information from a device equipped by the ADC, a COPY command is set up on a personal computer (PC) in the simplest case, and program–driver in assembly language is best among that in high level languages. The example of such a program is described below.

```
;================================
;Subroutine AdcComl in micro assembly language MASM.EXE
;For Microsoft programming languages FORTAN and Quick
BASIC
;Receives three 10–bit words from the port COM1 IBM PC
;--------------------------------------------------------
;QB call AdcComl(r3,r2,r1)
;where r1, r2, and r3 are the integer variables
;--------------------------------------------------------
Coml_Dat EQU 3F8h
Com1_St EQU 3Fdh

.MODEL medium
.CODE   ;code segment
PUBLIC ADCCOM1 ;define procedure name
ADCCOM1 PROC

   push bp  ;save the register–pointer BP
   mov bp,sp ;take top of stack pointer

start:    mov dx,Com1_St
mark1:    in al,dx ;check the byte in the receiver
          and a1,1
          jz mark1
          mov dx,Com1_Dat ;yes, take it
          in al,dx
          cmp al,0aah ;beginning of the marker?
          jnz start ;no, go to beginning
          mov dx,Com1_St
mark2:    in al,dx
          and al,1
          jz mark2
          mov dx,Com1_Dat
          in al,dx ;take 2 bytes
          cmp al,55h ;identify them with marker
          jnz start ;no, go to beginning
          mov cx,3 ;the marker has been identified,
          take 6 bytes of information

rws:      mov bx,[bp+6] ;calculate the address
               of the first variable
          mov dx,Com1_St
```

```
m1:     in al,dx
        and al,1 ;wait 1 byte
        jz m1
        mov dx,Com1_Dat
        in al,dx ;take this byte
        mov ah,al;poke this byte in the high–order byte
        of the accumulator
        mov dx,Com1_St
m2:     in al,dx
        and al,1
        jz m2
        mov dx,Com1_Dat
        in al,dx ;take 2 bytes
        xchg al,ah ;exchange the accumulator bytes
        mov [bx],ax ;poke the result in the address of
        the first variable
        inc bp ;increase the pointer
        inc bp ;by 2
        loop rws ;do the cycle of information reception
        ;to 6 bytes
        pop bp ;restore the pointer
        ret 6 ;return to the program call

ADCCOM1 ENDP
END
```

The ADC is a 100×100 mm board in design. The device is in successful operation at the Institute of Atmospheric Optics of the Siberian Branch of the Russian Academy of Sciences as part of an automated marine solar photometer. The device was in successful operation in the Russian–American–Spain expedition aboard Spain *Esperansa del Mar* ship in spring of 1994 and under the integrated Program SATOR–94 in summer of 1994. At the Institute of Strength Physics and Materiology of the Siberian Branch of the Russian Academy of Sciences such ADC is used for digitization and input of signal of electronic microscope to a personal computer.

REFERENCES

1. A.A. Myachev, V.N. Stepanov, and V.K. Sherbo, *Interfaces of Data Processing Systems* (Radio and Svyaz', Moscow, 1989), 416 pp.
2. B.G. Fedorokov and V.A. Telets, *DAC and ADC Microcircuits: Function, Parameters, Application* (Energoatomizdat, Moscow, 1990), 320 pp.
3. V.V. Stashin, A.V. Urusov, and O.F. Mologontseva, *Design of the Digital Devices Built around Single–Crystal Microcontrollers* (Energoatomizdat, Moscow, 1990), 224 pp.