VID2G

Video Identifier & Generator
v 2.01

(English, part I, II & III)

Author: Mijo Kovačevič, S51KQ

Eng. Linguistics: Dave Wheeler, G0GMK

S51KQ, P.O.Box 11, SI-3212 VOJNIK

http: // lea.hamradio.si/~s51kq

1.1. Introduction

Many years ago Radio Amateurs having ATV QSO's used what would be considered a primitive method by todays standards means of sending personal ATV identification.

The simplest way was a sheet of paper and a thick nibbed coloured pencil or felt tip pen. The callsign and other personal data were written on the paper, and then the station ATV video camera focused onto the paper. A few others with more imagination (and probably time) built themselves a fairly complicated mechanical system for identification. An example of this type of mechanical device was made with picture frames and a motor to slowly turn the frames before the lens of a video camera. Similar systems are still in use in some very old cinemas for displaying advertising media onto cinema flax.

With the first appearance of home computers in about 1978 a new era of ATV was also entered. Initially this was only for the wealthier amongst us as the first home computers were very expensive pieces of equipment. Due to this, a number of Radio Amateurs started to build simple Black and White TTL based video identifiers with the low-resolution images being programmed into PROM's or EPROM's.

A couple of disadvantages with this type of device was that there was no facility for mixing of video images (genlocking) and due to the fairly high power consumption of the TTL devices spurious emissions causing interference to either the Video or Audio signals. In 1979 I built such a device which was shown in a CQ-ZRS article in issue number 4/1997.

Many years later Radio Amateurs started to build and publish new video generators with genlocking facilities to superimpose generated text or pictures onto the external incoming video signals. In the early days this was done without the help of microprocessors. Such video generators could be used as video identifiers as the incoming video was also on the output too. This type of device was able to overlay a line of text in one of the video corners. One of these Video identifiers were also published in CQ-ZRS magazine in issue number 4/1994.

This video identifier was recently named VID-01. The device was based on LS and HC integrated circuits. The EPROM used was CMOS, therefore the power consumption is considerably lower than with the older types of video generators/identifiers. And with reduced power consumption also comes the reduction in the unwanted HF emission.

These video generators were superb. The generated picture still had a low resolution, which was limited with both the size and access speed of used the memory circuit (EPROM). The synchronising part of the device used very simple logic with a sync detector and line counters. Therefore, if the incoming video was a bit distorted (with regard to the synchronisation pulses), then the synthetically generated text would be jumping up and down.

Another problem with this device was that it could be very sensitive to changes in ambient temperature, this is mainly caused by value deviations in the used capacitors and resistors. Nowadays all of these, and lots of other problems can easily be overcome with the use of microcontrollers.

Why use Microcontrollers ?. We can, by using one small microprocessor and relevant software build an interesting and easy to operate simple video generator or video identifier. With adequate software such a device can generate a high quality video output, unaffected by changes of the ambient temperature and with much better synchronisation (Genlocking) onto the incoming (input) video signal.

Even with the vast amount of microprocessors available there are still limitations with what can be achieved. It therefore makes sense to use a specialist video microprocessor for sync decoding and for overlaying graphics and text, and to use a microcontroller for interfacing and control. Many companies manufacture Professional video identifier devices for use on TV broadcasting transmitters (TV-logo graphical devices). Their content is similar to ours. But due to much higher specifications and demands these professional devices are very complex to build, and they can cost up to few 10000 $US. In most cases they are high quality graphical stations (computers). They are generally housed in a standard 19" rack type housing, with own keyboard, LAN connection, and with very large hard drives for video storage. Such professional video identifiers (LOGO devices) can offer numerous variations of video output, which would not be possible to reproduce using widely available and ordinary microprocessors.

Anyway, with the correct choice of main microprocessor, graphical microprocessor, and adequate software support, we can home-brew a quality video identifier that in lots of ways is similar to the professional devices. This project, which will be published in few parts, has its origin at the end of 1997. I named this new device: VID2G.

It's uses the latest release of microprocessor based video generator and video identifier in one device, on one single small board. As I mentioned before, The project will be published in few parts. In the first article we will discover the structure of the VID2G device, its main functions, operating system and management facilities. In the following issues, practical instructions for how to build a VID2G at home will be covered, The procedure for testing of the device and possible uses. Finally we will examine the software support for Windows.

1.2. Development of VID2G

The latest version of the VID2G identifier came out in the winter of 1998. This project was started due to the limitations that existed with using my older identifier named VID-01. These limitations were seen mainly with: The limitation to display fixed text strings from EPROM, and also poor synchronisation if the input video quality was anything short of perfect. In order to overcome these limitations either an extremely complex device would need to be designed and built using standard circuits or to use a powerful microprocessor to do the same job.

Using the first option a number of the problems associated the earlier video identifier could be overcome, but this would involve a very complicated device which would be very expensive to build. It is therefore not practical (or cost effective) to take this option. Therefore the logical approach was for the development of new microprocessor based video identifier.

In many of my past projects I have implemented the use of low cost and widely available PIC microprocessors, with FLASH type program area. It is possible to make a video identifier with such microprocessors. But this time the idea was to build something more powerful, to outweigh any other. For this purpose we needed a larger program area, and a much more powerful CPU than the 16F84. One year before this project was started I had cast eyes over the ATMEL-s products, AVR (RISC) microcontrollers.

The primary requisite was that the CPU must be a FLASH type, and must have as large a program area as possible. Despite promises from MICROCHIP Inc (producer of PIC processors) at last years seminar, that this company will bring new controllers onto the market, with FLASH and large program area, nothing was available at that point.

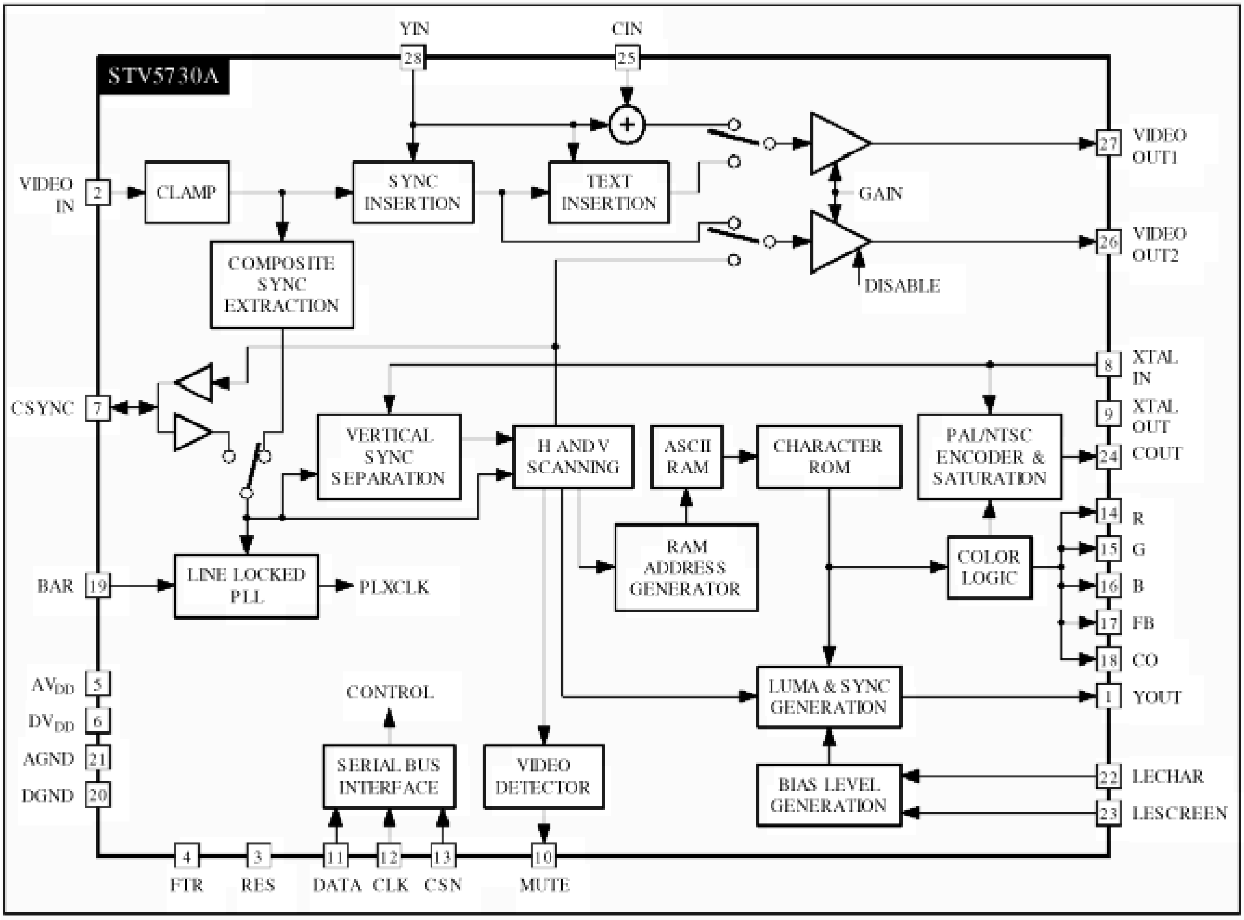
The decision was taken to use latest ATMEL's AVR products. And not only for the VID2G device, but also for some other future complex projects. Yes, there was some doubt about buying and using new products, but the first operational (test) project built with an AVR microprocessor eliminated all of my concerns. The very first test device was built using a AT90s8515 AVR, with great success.

Now the decision had to be made as to what graphical microprocessor should be used. For this job we could use a wide range of available video processors, including those that are available from electronic component suppliers. Unfortunately many of the circuits that are available are fairly old service parts, which probably wouldn't be available in a years time. Producers of these circuits make new types each year. At the same time the older types go out of production.

Because of that fact I tried to select for theVID2G project one of the latest video processors from the company ST (SGS & Thomson). This circuit is not available (on the stock) in ordinary shops. It's intended to be used inside home digital multimedia devices, new era digital video cameras and digital satellite receivers. But there are the worldwide suppliers to the industry and we can buy larger quantities of almost any available circuit from them.

After almost 3 months of waiting on delivery of the ordered graphical circuits, they came. And in the following days the first prototype starts to grow. On the first prototype the AT90s8515 microcontroller was used. The ATmega103 was not available at that time. The 8kB large program area was soon full. The chosen used graphical microprocessor uses many 12-bit long registers, and there was plenty of work to be carried out, before first picture came out of this circuit.

On the final version of the VID2G device I used for controlling, communicating via RS-232 port, and



Picture 1.1. - Block Diagram of STV5730A

Basic features of video processor (picture 1.1.) :

- *ROM with 128 non ASCII characters!*
- *On-chip RAM for one video page*
- *12 dots x 18 lines character composition*
- *11 rows x 28 characters page composition*
- *Line locked PLL*
- *Video timing generator*
- *Input CVBS CLAMP and SYNC extractor*
- *Vertical SYNC separator*
- *Input CVBS SYNC re-insertion*
- *PAL/NTSC chroma encoder*
- *Dedicated pins for LUMA and CHROMA external filtering*

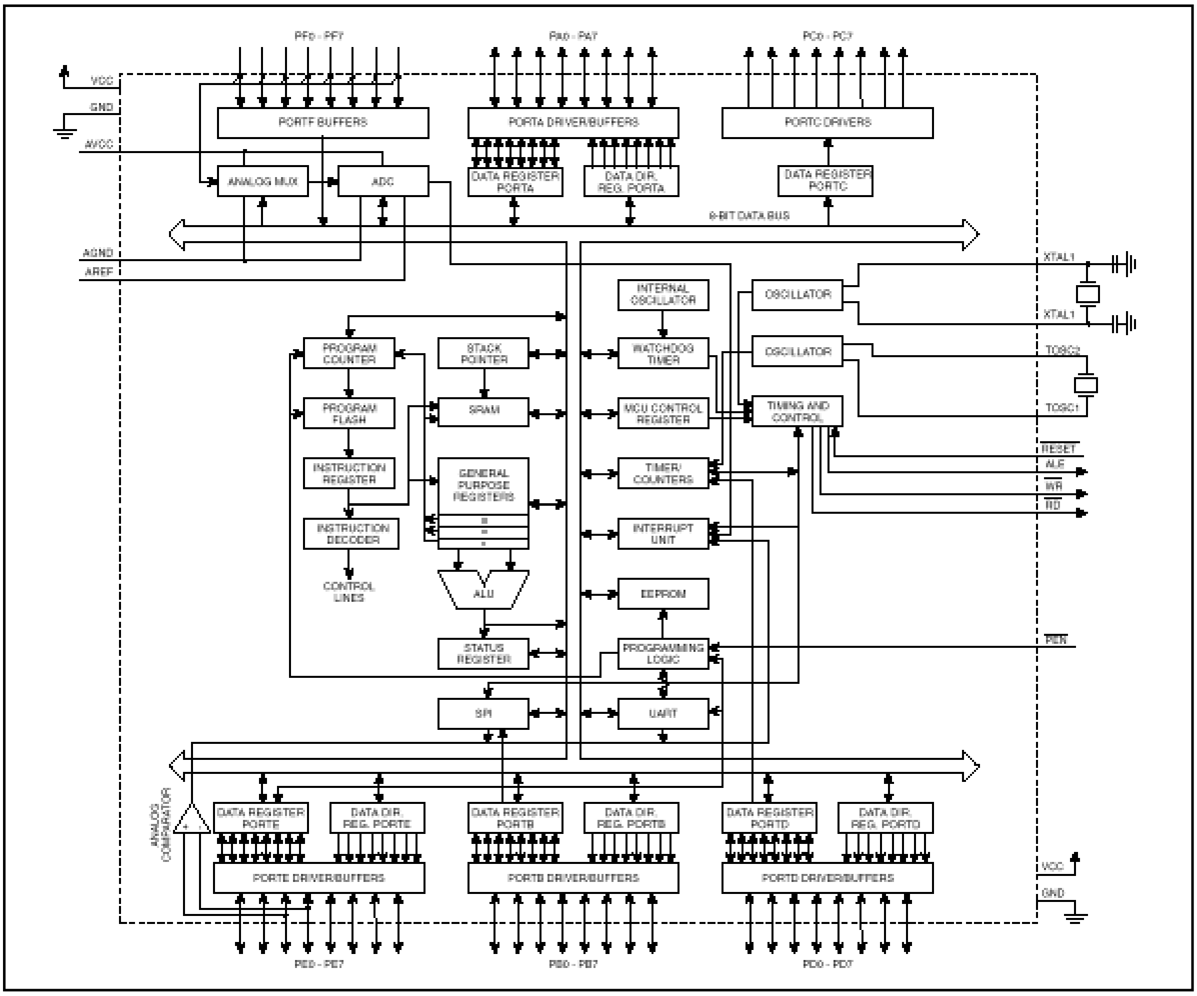
... and other features

for all other jobs, the most powerful and biggest Atmel AVR microprocessor available at that time, the ATmega103 (picture 1.2.). Both microprocessors (graphical and main AVR) are available in a SMD package only. The graphical microprocessor is in 28 pin SO28 package. The ATmega103 is the thin square, with dimensions of 16 x 16mm with 64 pins inside a TQFP64A plastic package. And it's only 1mm thick.

1.3. Description

In the development procedure of VID2G I try to recognise the following requirements. The new video identifier should be available for portable use, with its own keyboard. Users should be able to reprogram their own text scripts many times. We should be able to use for personal data programming any standard PC or VT-100, VT-200 terminal. The power consumption should be as low as possible. And last, but by no means least the price for device should be within everyone's reach.

VID2G device is in terms of its physical dimensions a very small and easy to use device. But in terms of its built-in software, a much more complicated device. At the heart of this project are two microprocessors (ATmega103 and STV). Because of some EEPROM problems on the +5v supply voltage in the ATmega CPU, we chose to use an external EEPROM circuit, a 24c04 with I2C bus (picture 1.3.). EEPROM is used as a storage device for all personal data, including ID's (call signs, QTH, Loc, etc). Inside



Picture 1.2. - The ATmega103 Block Diagram

Basic features of ATmega103 AVR microcontroller :

- Enhanced RISC Architecture
- 121 powerful Instructions - Most Single Cycle Execution
- 128kB of in-system FLASH
- 4kB of internal EEPROM-a (not usable on +5v..)
- 4kB of internal SRAM
- 32 x 8 General Purpose Working + Peripheral registers
- 32 programable I/O lines, 8 output and 8 input lines
- Programable serial UART , plus SPI serial interface
- Vcc supply from +4v to +5.5v
- Up to 6 MIPS troughput at 6Mhz
- RTC with separate oscillator
- Two 8 bit Timer/Counters with separate prescaler and PWM
- One 16 bit Timer/Counters with separate prescaler, compare, capture
- Programmable Watchdog Timer with on-chip oscillator
- 8 channel, 10 bit ADC
- On-chip analog comparator

- Low Power Idle, Power Save and Power Down modes
- Software selctable clock frequency
- Programming lock for software security

of the main microprocessor ATmega103 is written the so-called Operating System. It's located in the FLASH program area of the ATmega103. This operating system contains more than 37000 lines of source code. Time used to write such a huge source was more than two months. Regardless to the large amount of program lines, it still takes only about 50% of available program area inside ATmega103 microprocessor.

The Operating system of VID2G is involved in all processes. It was written as a true multitasking system with three processes. Those processes take care of the low-level communication with the graphical CPU, there is an interpreter compiler built in for user commands (direct entry). Other processes look after the RS-232 port, keyboard reading, managing ADC channels, and setting up states on the output ports of the main microprocessor. Due to the use of an I2C bus on the EEPROM, an additional 2 wire bi-directional communication has to be made (I2C protocol).

1.4. VID2G management

Because of multitasking and to prevent collisions inside the operating system, the main operation is divided on to two main modes: Management from own Keyboard, and on Management via the RS-232 serial port. On the VID2G's own keyboard we have 5 push buttons as follows: DOWN, UP, F1, F2 and CTRL. There are two main and two additional menus available on the keyboard (picture 1.4.).

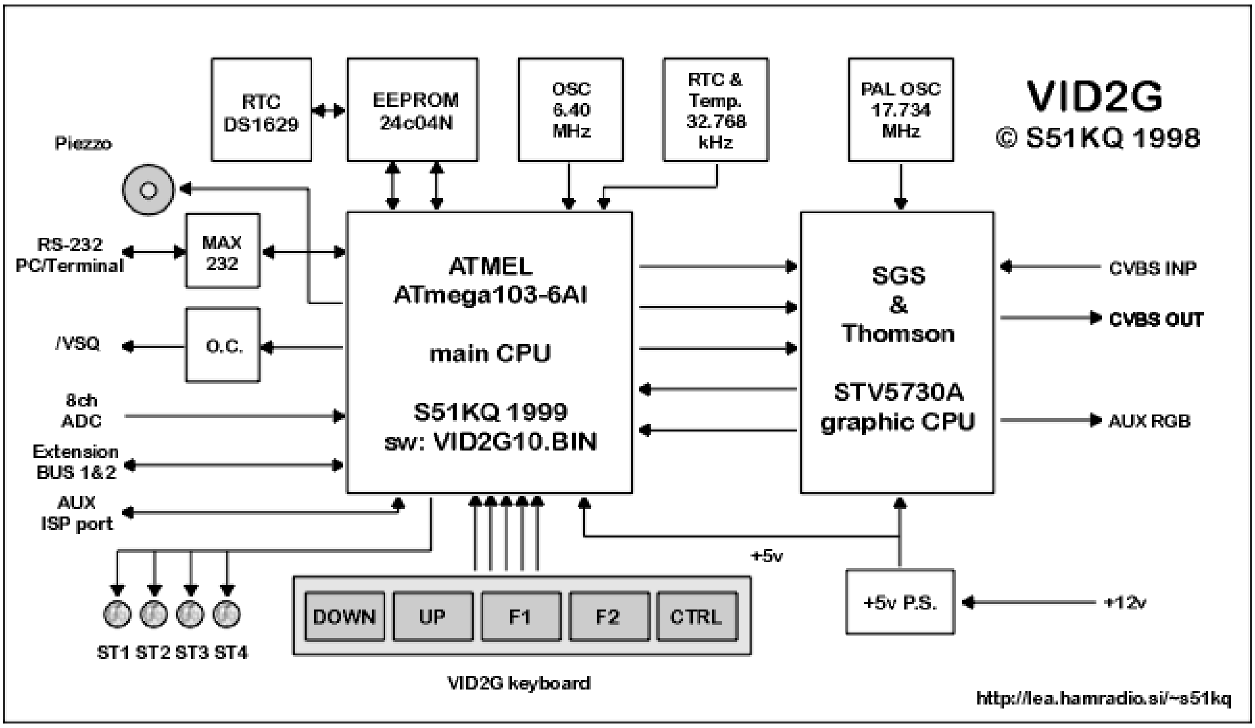
In the first menu (1) - PAGE MODE we can switch page UP/ DOWN, go to the DISPLAY MODE (menu2) with F1, disable or enable printout of generated picture with F2, or start/stop ROLL function on actual page with CTRL button. In the menu2 - DISPLAY MODE, the user can change the displaying mode to NORMAL. The generated picture will be displayed 100% over the incoming video. With next button, change to the MIX mode. The generated picture will be displayed over the incoming video 50% transparent (like on professional TV LOGO devices). And with next button the user can select FULL mode. In this mode, the incoming video will be disabled, and the generated picture displayed in colour (CVBS out). The device will act as stand alone video generator. Colours of the characters used in this mode are weak! Background colours of the

Basic features of VID2G video identifier:

- Stand-alone video identifier and color video generator
- PAL and NTSC standard support
- CVBS input & output 1Vpp + optional RGB output (color mode all the time)
- 3 display modes: NORMAL (100% B/W print over color input video)
MIX (50% transparent B/W print over color input video)
FULL (stand-alone color video generator)
- Separate commands for colors in FULL mode
- Blinking
- 4 ZOOM factors, for X and Y axis, separate for 0., 1..9. and 10. line
- Video output level 0db or +6dB
- Digital Video Squelch output (open collector)
- Digital PLL for synchronization
- Regeneration of synchronization
- Enabling/disabling fo generated picture
- Readout of all 8 ADC inputs
- 12 predefined video pages
- 4 dynamic or one static ID entry + RTC & Temperature printout
- Free possitoning of some video pages, with memory
- Build in macro interpreter for drawing of new video pages via RS-232
- Managing with own keyboard, or via RS-232 serial port (PC, terminal,..)
- All user settings are in EEPROM
- Builded in automatic EEPROM initialization after very first start-up

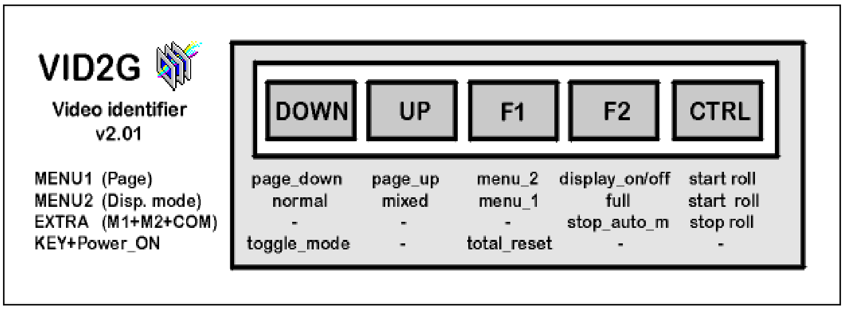
- Software support: Own operating system (v1.00 60kB),
Free VT-100, VT-220 terminal access,
Win95/98 manager (VID2w13.EXE),
Own service software support (OSv1.00).

- Possibility of future expansions for operating-system up to 128kB
- SMD pcb, small dimensions, low power consumption
- Supply voltage: +12v / 80mA



Picture 1.3. - The VID2G Block Diagram

characters are normal. This is not a malfunction. The colour coding is hardware fixed inside of the STV circuit and cannot be changed from the outside world.


VID2G
Video identifier
v2.01

	DOWN	UP	F1	F2	CTRL
MENU1 (Page)	page_down	page_up	menu_2	display_on/off	start roll
MENU2 (Disp. mode)	normal	mixed	menu_1	full	start roll
EXTRA (M1+M2+COM)	-	-	-	stop_auto_m	stop roll
KEY+Power_ON	toggle_mode	-	total_reset	-	-

Picture 1.4. - Functions on VID2G keyboard

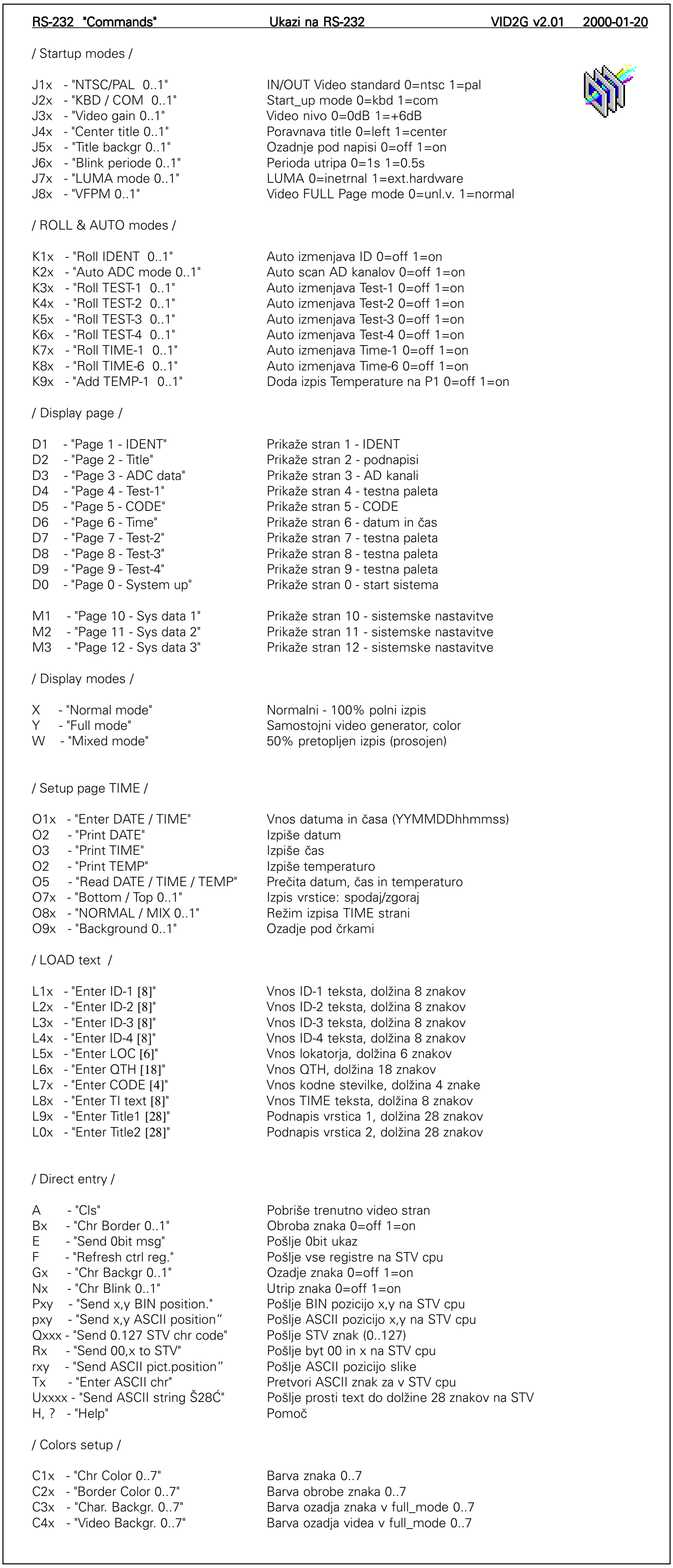
There is also one special menu on the VID2G keyboard. These special functions are hidden and are only available at boot of the VID2G module. Therefore this mode is called POWER_ON. Each function can be executed by switching the device OFF, pressing the selected button, the switching the power ON, waiting for 1second, and then releasing the pushed button.

Under button F1 (F1+Pwr_ON) is hidden a TOTAL RESET. The device will reset itself (all user data) to the factory state. All user-defined data will be lost. After execution of this command there is no Undo function available ! Under button DOWN (DOWN+Pwr_ON) the user can user change the VID2G management mode from RS-232 to KEYBOARD, or vice versa, regardless of the start-up mode written in EEPROM (user function too). After any keyboard command is executed the device will produce one or more short beeps on built-in piezo sounder.

For the maintenance of VID2G we have an RS-232 serial port available. On Operating Systems below version 2.01 the speed on this port is fixed to 9600 bps, 8 bits, no parity, one stop bit, On versions 2.01 and above the baud rate is user programmable.

Regardless of the main mode, VID2G will send all changes and replys every time, via RS-232 to the outside world (PC, terminal, modem, etc). On the other hand, VID2G can execute commands from outside world received via RS-232 port only when in RS-232 mode ! The table 1.5. shows us the available RS-232 commands. Commands are as printed in the table. Only UPPER CASE commands are valid. VID2G won't respond to any lower case command ! Some RS-232 commands need additional numeric or alphabetic data. The maximum length of the additional data is written in brackets in comment near to each command. A few of the critical commands need additional confirmation before execution will take place (using a "Y" character). Lets examine the available RS-232 commands shown on table 1.5.

In section of START-UP modes the user can change few of VID2G's main functions. These functions will remain active after each future power-up (after saving only!). In section of ROLL & AUTO modes the user can enable (or disable) flags for endless loops on various VID2G video pages, examples of Rolling are - changing ID's, adding real time read-out of RTC (Real Time Clock) & Digital Temperature sensor. Or scanning of ADC channels. In the case that user selects one of available flags, VID2G will upon selection of the actual page automatically start the ROLL or SCAN



```
RS-232  "Commands"                 Ukazi na RS-232                  VID2G v2.01    2000-01-20

/ Startup modes /

J1x   - "NTSC/PAL  0..1"           IN/OUT Video standard 0=ntsc 1=pal
J2x   - "KBD / COM  0..1"          Start_up mode 0=kbd 1=com
J3x   - "Video gain 0..1"          Video nivo 0=0dB 1=+6dB
J4x   - "Center title 0..1"        Poravnava title 0=left 1=center
J5x   - "Title backgr 0..1"        Ozadnje pod napisi 0=off 1=on
J6x   - "Blink periode 0..1"       Perioda utripa 0=1s 1=0.5s
J7x   - "LUMA mode 0..1"           LUMA 0=inetrnal 1=ext.hardware
J8x   - "VFPM 0..1"                Video FULL Page mode 0=unl.v. 1=normal

/ ROLL & AUTO modes /

K1x   - "Roll IDENT  0..1"         Auto izmenjava ID 0=off 1=on
K2x   - "Auto ADC mode 0..1"       Auto scan AD kanalov 0=off 1=on
K3x   - "Roll TEST-1  0..1"        Auto izmenjava Test-1 0=off 1=on
K4x   - "Roll TEST-2  0..1"        Auto izmenjava Test-2 0=off 1=on
K5x   - "Roll TEST-3  0..1"        Auto izmenjava Test-3 0=off 1=on
K6x   - "Roll TEST-4  0..1"        Auto izmenjava Test-4 0=off 1=on
K7x   - "Roll TIME-1  0..1"        Auto izmenjava Time-1 0=off 1=on
K8x   - "Roll TIME-6  0..1"        Auto izmenjava Time-6 0=off 1=on
K9x   - "Add TEMP-1  0..1"         Doda izpis Temperature na P1 0=off 1=on

/ Display page /

D1    - "Page 1 - IDENT"           Prikaže stran 1 - IDENT
D2    - "Page 2 - Title"           Prikaže stran 2 - podnapisi
D3    - "Page 3 - ADC data"        Prikaže stran 3 - AD kanali
D4    - "Page 4 - Test-1"          Prikaže stran 4 - testna paleta
D5    - "Page 5 - CODE"            Prikaže stran 5 - CODE
D6    - "Page 6 - Time"            Prikaže stran 6 - datum in čas
D7    - "Page 7 - Test-2"          Prikaže stran 7 - testna paleta
D8    - "Page 8 - Test-3"          Prikaže stran 8 - testna paleta
D9    - "Page 9 - Test-4"          Prikaže stran 9 - testna paleta
D0    - "Page 0 - System up"       Prikaže stran 0 - start sistema

M1    - "Page 10 - Sys data 1"     Prikaže stran 10 - sistemske nastavitve
M2    - "Page 11 - Sys data 2"     Prikaže stran 11 - sistemske nastavitve
M3    - "Page 12 - Sys data 3"     Prikaže stran 12 - sistemske nastavitve

/ Display modes /

X     - "Normal mode"              Normalni - 100% polni izpis
Y     - "Full mode"                Samostojni video generator, color
W     - "Mixed mode"               50% pretopljen izpis (prosojen)


/ Setup page TIME /

O1x   - "Enter DATE / TIME"        Vnos datuma in časa (YYMMDDhhmmss)
O2    - "Print DATE"               Izpiše datum
O3    - "Print TIME"               Izpiše čas
O2    - "Print TEMP"               Izpiše temperaturo
O5    - "Read DATE / TIME / TEMP"  Prečita datum, čas in temperaturo
O7x   - "Bottom / Top 0..1"        Izpis vrstice: spodaj/zgoraj
O8x   - "NORMAL / MIX 0..1"        Režim izpisa TIME strani
O9x   - "Background 0..1"          Ozadje pod črkami

/ LOAD text  /

L1x   - "Enter ID-1 [8]"           Vnos ID-1 teksta, dolžina 8 znakov
L2x   - "Enter ID-2 [8]"           Vnos ID-2 teksta, dolžina 8 znakov
L3x   - "Enter ID-3 [8]"           Vnos ID-3 teksta, dolžina 8 znakov
L4x   - "Enter ID-4 [8]"           Vnos ID-4 teksta, dolžina 8 znakov
L5x   - "Enter LOC [6]"            Vnos lokatorja, dolžina 6 znakov
L6x   - "Enter QTH [18]"           Vnos QTH, dolžina 18 znakov
L7x   - "Enter CODE [4]"           Vnos kodne stevilke, dolžina 4 znake
L8x   - "Enter TI text [8]"        Vnos TIME teksta, dolžina 8 znakov
L9x   - "Enter Title1 [28]"        Podnapis vrstica 1, dolžina 28 znakov
L0x   - "Enter Title2 [28]"        Podnapis vrstica 2, dolžina 28 znakov


/ Direct entry /

A      - "Cls"                     Pobriše trenutno video stran
Bx     - "Chr Border 0..1"         Obroba znaka 0=off 1=on
E      - "Send 0bit msg"           Pošlje 0bit ukaz
F      - "Refresh ctrl reg."       Pošlje vse registre na STV cpu
Gx     - "Chr Backgr 0..1"         Ozadje znaka 0=off 1=on
Nx     - "Chr Blink 0..1"          Utrip znaka 0=off 1=on
Pxy    - "Send x,y BIN position."  Pošlje BIN pozicijo x,y na STV cpu
pxy    - "Send x,y ASCII position” Pošlje ASCII pozicijo x,y na STV cpu
Qxxx - "Send 0.127 STV chr code"   Pošlje STV znak (0..127)
Rx     - "Send 00,x to STV"        Pošlje byt 00 in x na STV cpu
rxy    - "Send ASCII pict.position” Pošlje ASCII pozicijo slike
Tx     - "Enter ASCII chr"         Pretvori ASCII znak za v STV cpu
Uxxxx - "Send ASCII string Š28Ć"   Pošlje prosti text do dolžine 28 znakov na STV
H, ?   - "Help"                    Pomoč

/ Colors setup /

C1x   - "Chr Color 0..7"           Barva znaka 0..7
C2x   - "Border Color 0..7"        Barva obrobe znaka 0..7
C3x   - "Char. Backgr. 0..7"       Barva ozadja znaka v full_mode 0..7
C4x   - "Video Backgr. 0..7"       Barva ozadja videa v full_mode 0..7
```

Picture 1.5a. - Index of RS-232 commands

function. Break-out from an endless loop is only possible by pressing the CTRL button, regardless of selected RS232 or KEYBOARD main control mode.

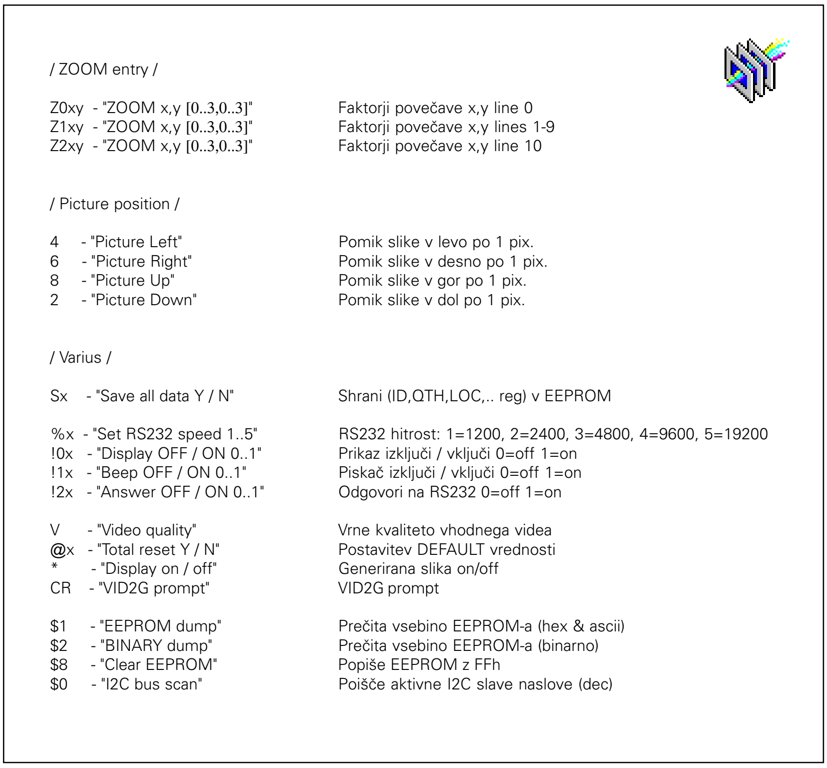
Next section is DISPLAY PAGE. There is 11 available video pages (out of 12) built inside the VID2G device, plus the system-boot page. On the first page (command D1), the device will display in the left upper corner, the first ID (callsign) in 50% transparent mode. If the ROLL flag was selected, the ID's will roll one after the other at 20s intervals. If the Roll-TIME flag was selected (command K71), the printout of time & temperature will be added after fourth ID. Video page 2 (command D2) is made for displaying two subtitle lines. Lines can be justified either in the centre, or left. And can be displayed with background or without. The user can enter free text with up to 28 characters. Both lines can be used for various personal data (telephone numbers, addresses, RTX description, etc).

Texts will be displayed as 100% printout. Page 3 (command D3) will display all 8 on board ADC channels. ADC capturing will be single shot only, unless the SCAN flag is selected (command K21). Page 4 (command D4) will push VID2G in to stand alone video generator, with colour-bar (rainbow) plus ID from first ID bank. If the ROLL flag for this page was selected, ID will roll unless the user stops the endless loop by pressing the CTRL button.

The fifth page (command D5) is dedicated to ATV contests. VID2G will display the first ID, Universal locator and standard 4 digit ATV contest Code number. Characters on this page will be magnified, with 100% printout over the incoming video.

The sixth video page (command D6) can be used for security video recordings or similar purposes. VID2G will display actual date & time, plus an additional 8 character long text. Video pages from 7 to 9 are stand-alone video pages with fixed or dynamic ID's. Pages 10 and 11 are system set-up pages.

There are three display modes available in the DISPLAY MODES section (commands X, Y and W). The next section of RS232



```
/ ZOOM entry /

Z0xy  - "ZOOM x,y [0..3,0..3]"        Faktorji povečave x,y line 0
Z1xy  - "ZOOM x,y [0..3,0..3]"        Faktorji povečave x,y lines 1-9
Z2xy  - "ZOOM x,y [0..3,0..3]"        Faktorji povečave x,y line 10


/ Picture position /

4    - "Picture Left"                 Pomik slike v levo po 1 pix.
6    - "Picture Right"                Pomik slike v desno po 1 pix.
8    - "Picture Up"                   Pomik slike v gor po 1 pix.
2    - "Picture Down"                 Pomik slike v dol po 1 pix.


/ Varius /

Sx   - "Save all data Y / N"          Shrani (ID,QTH,LOC,.. reg) v EEPROM

%x  - "Set RS232 speed 1..5"          RS232 hitrost: 1=1200, 2=2400, 3=4800, 4=9600, 5=19200
!0x  - "Display OFF / ON 0..1"        Prikaz izključi / vključi 0=off 1=on
!1x  - "Beep OFF / ON 0..1"           Piskač izključi / vključi 0=off 1=on
!2x  - "Answer OFF / ON 0..1"         Odgovori na RS232 0=off 1=on

V     - "Video quality"               Vrne kvaliteto vhodnega videa
@x  - "Total reset Y / N"             Postavitev DEFAULT vrednosti
*      - "Display on / off"           Generirana slika on/off
CR   - "VID2G prompt"                 VID2G prompt

$1    - "EEPROM dump"                 Prečita vsebino EEPROM-a (hex & ascii)
$2    - "BINARY dump"                 Prečita vsebino EEPROM-a (binarno)
$8    - "Clear EEPROM"                Popiše EEPROM z FFh
$0    - "I2C bus scan"                Poišče aktivne I2C slave naslove (dec)
```

Picture 1.5b. - Index of RS-232 commands

commands will help the user to set up the real time clock. A brief HELP of available RS-232 commands (index) is available too. With commands in the LOAD TEXT section the user can enter all text data such as: four ID's (callsigns), universal locator, QTH, contest code number, additional text for sixth video page, and both subtitle lines. VID2G internal compiler will accept any ASCII character, but the displayed text is converted to STV format and only those included within the STV character ROM table are displayed properly. Any other characters will be printed out as a blank (space).

The DIRECT ENTRY section is for "direct" access to the STV circuit. In fact these are the commands from internal interpreter compiler, built inside VID2G device. Those commands are for Expert users only! It enables the user to write (draw) self composed VID2G pictures in to devices RAM. Those pictures must be uploaded each time to the VID2G's RAM, either from file or PC keyboard.

How to write (draw) such user made pictures won't be covered here because of the extensive knowledge needed to do this. Users should carefully study the STV PDF file before any attempt to use these direct-entry commands. Controlling of the STV microprocessor is complex! With those commands and a few others, VID2G can generate almost any possible video picture within the limits of graphical microprocessor possibilities.

All new (predefined) user values will be saved to the EEPROM, but only after executing the SAVE command. This command must be confirmed with a "Y". At that point all of the current settings will be valid, and the currently displayed video page becomes the START-UP video page. (The page that will be shown after each Power-On) first. Additionally, positions of a few of the video pages can be changed and they will be automatically saved with their new position (pages: 1, 2, 5 and 6 only).

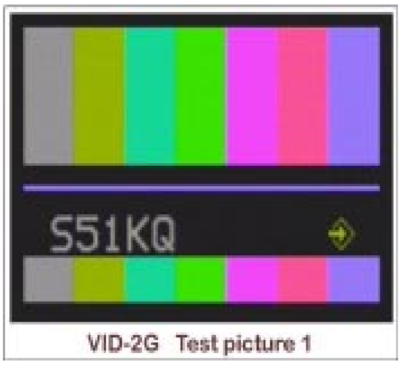
The ZOOM ENTRY enables the user to temporarily change the size in all 3 sectors of the screen. There are separate Zoom factors for X and Y axis in each sector. The numbers must be within the limits from 0 to 3.

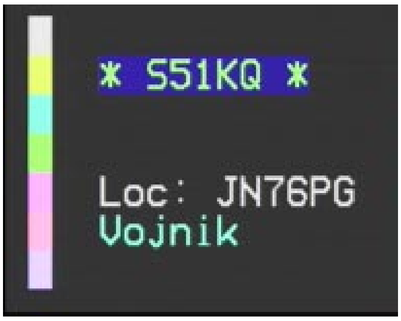
There are few RS-232 commands left. The user can check video quality. VID2G will examine 16 lines of input video for quality. The result can be displayed via the RS-232, and is always present on /VSQ output of VID2G. Putting VID2G in to stand-alone video generator mode will DISABLE this function!

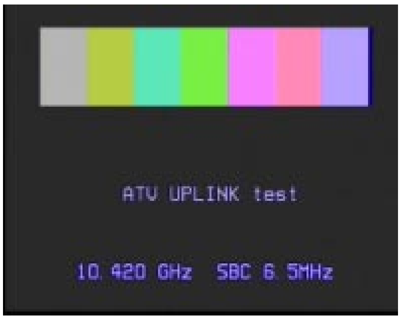
A group of service commands (beginning with a $.) are not all available for user execution.

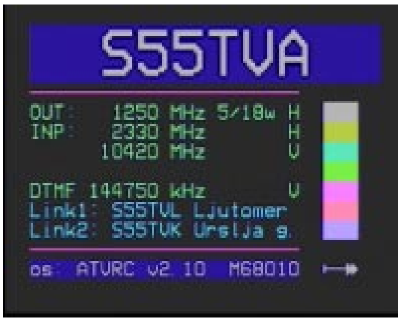
1.5. Conclusion

In next issue how to made a VID2G device at home will be covered, how to made first tests, and how to use VID2G for ATV, etc.









2.1. Introduction

The VID2G was presented, and its main functions were covered in part 1 of this article. This time we will cover the hardware in more detail, cover the assembly of the device at home and cover the testing and setting up of the completed unit.

2.2. Circuit description

The schematic diagram is shown on pictures: 2.2 and 2.3. The VID2G device consists of a few high-density SMD integrated circuits. For the main processor I chose to use ATMEL's ATmega103 AVR (RISC) processor. In order for the processor to run at least one external crystal is required, A 4.000 MHz (Q1) in this application is connected to the main internal oscillator (pins XTAL1 & XTAL2). Crystal Q2 (32.768 kHz) that can be connected to the second internal oscillator is used by the processor for special interrupts (TOSC). In the current application these interrupts are not used, therefore the second crystal is not required and can be omitted.

The VID2G keyboard consists of 5 pushbuttons, These are connected to the MSB bits of PD I/O bus (PD3..PD7). The VID2G's keyboard does not need any external Pull-up resistors, as internal pull-ups are switched on (in software) for these CPU bus inputs. Part of the PC bus (PC0..PC3) is used for the driving of LED's. They are connected directly to the CPU without need for external driving transistors. The CPU I/O pins can deliver enough current to drive small LED's directly (via resistors).

The VID2G project uses 3mm LED's with very low current, connected with 680E current limiting resistors. Part of the PC bus (PC5) is connected to a stand alone low current Piezo oscillator (not single Piezo transducer, as used on UNIPLL project !).

Output PC4 is used with an additional BC547B transistor, configured as an open-collector and is used for the /VSQ (Video Squelch) output.

The MSB bits of PA bus (PA4..PA7) and PC7 are connected to the first spare connector (EXTENSION-A) on board. PA4 and PA5 are also used foe the I2C bus. The 24c04 (serial EEPROM) and Dallas DS1629 are connected onto the I2C bus (picture 2.3). The used main microprocessor has an integrated 4kB internal EEPROM, but accessing of this internal EEPROM was found to be unreliable (on 5v supply). My findings indicate that the internal EEPROM on AVR processors only work properly using lower CPU supply voltages. This was the reason to add an external EEPROM. Another thing to consider was that all EEPROM devices can start to fail after the specified number of read/write cycles has been exceeded, for this reason the DIL version of the EEPROM was chosen which is fitted to a DIL socket and can easily be replaced if failures do start to occur. At the development time of VID2G the chosen Real Time Clock circuit, The DS1629, had only just been released by Dallas (picture 2.3). This circuit needs an external miniature crystal of 32.768 kHz. Over and above the normal RTC functions the DS1629 includes temperature measuring functions (9 bit) and programmable temperature alarms.

The used RTC integrated circuit also contains a Direct Digital Temperature sensor. With this internal sensor the software can measure the approximate temperature inside the device (+/- 2 deg.C). To avoid having to set the RTC each time the VID2G is used we need to provide an external +4.5V supply. VID2G uses 3 standard 1.5v batteries in separate PVC box. The RTC circuit current is about 1mA ! Therefore a miniature NiCad can't be used for this supply.

The LSB bits of PA bus (PA0..PA3) form the multi wire serial bus used for communication between the SGS/Thomsons STV5730A graphical cpu and the 8main AVR ATmega103 cpu. Using a similar method to an I2C bus but on a 3-wire connection, the main cpu, here with the help of the correct software support forms the corresponding communication protocol, well known to the graphical cpu. Using this protocol the cpu can gain access and control of the STV cpu, writing or reading it's internal registers, executing users commands, etc. The STV5730 integrated circuit is intended to be used in VCR, Satellite receiver, PAY-TV systems, digital camcorders, etc.. for CVBS or R/G/B text/graphics insertion. The STV5730A can operate in "mixed mode" or "full page mode". In Mixed mode the device is line locked to the incoming CVBS signal. The text is superimposed over the CVBS input either in B/W or translucent. It may also be superimposed in colour by using the R/G/B outputs. In Full-page mode the device generates a coloured PAL or NTSC CVBS output and simultaneous R/G/B outputs. Both "normal full page" and "video full page" modes are supported. In the "normal full page" mode the screen background is uniformly coloured, while the "video full page" mode permits the display of the unlocked VIDEO IN signal in the screen back-ground area.

The page is composed of 11 rows x 28 characters. Character composition is 12 dots x 18 lines.

Only a few external components are needed for low cost applications. The first one is the crystal. The frequency of this crystal must be selected to correspond to the colour subcarrier frequency (4 x fsc). For PAL standard a 17.734 MHz crystal is needed (4.4335 x 4), and for American NTSC standard a 14.318 MHz crystal (3.5795 x 4) is needed. In addition to this selection the CPU must rewrite selected internal registers of the STV5730A to enable the use of the desired video standard. For proper operation, the STV5730A needs a LUMA filter. A simple LUMA filter can be made using a 20yH inductor, 56p capacitor, and two resistors. This type of LUMA filter is connected between pins 1 and 28 of the graphical processor. Another filter that is required is a so-called CHROMA band pass filter, connected between pins 24 and 25. Those filters are optional, and for low cost applications can be omitted (short circuit).

Composite input (CVBS) is delivered directly via a 75E impedance matching resistor connected to ground, and a serially connected 2.2 yF capacitor. Due the method that the STV uses for CVBS output it is necessary to use both of the available video outputs. Only by using both is it possible to generate video with overlayed text, and stand alone text (without an incoming video source). The output video level is partially dependent on the value of 75E Resistor connected to the BC547B transistor, The value of this resistor can be changed. The Graphical processor STV5730A that is used also has R/G/B outputs. These pins are connected to the external bus (AUX RGB-OUT). With help of an extra RGB modulator circuit connected to those outputs, it should be possible to deliver overlayed text with colour.

The complete PF bus of the main processor (PF0..PF7) is used as eight channel analogue to digital converter input (ADC). The lines are connected to the standard 10 pin on board male connector (ADC INP - P3). This connector can be used with external voltage converters for input level measurements. These are direct inputs onto the main processors (ADC) and allow input voltages between 0 and +5v. The resolution of the internal ADC is 10 bit. As a direct result of measurement on the selected ADC channel VID2G device will display a digital value between 0 and 1023 decimal. There is one jumper JMP2 that enables connection of external precision reference voltage for ADC (A - AREF, pin 62). ADC input channels are multiplexed. All unused ADC inputs must be connected to ground. Otherwise false values can appear on them.

Buses PE (PE6..PE7) and PB (PB4) are connected to the second extension connector (EXTENSION-B, P9). This connector is intended for possible future extensions. From the same buses we also take the wires for the RS-232 communication interface, this interface is made using the standard MAX-232 integrated circuit in DIL package. Through this Serial port we can execute user and sysop commands from either a VT100 Similar terminal, or from a PC with a suitable communications program. The On board Reset button is for Test purposes. The adjacent diodes are needed for establishing of the correct state on the /RES line of the ISP bus. There may be some reset problems at Power on, if either a poor quality power switch is used, or if the device is powered from a large capacity supply (causing the supply voltage rise or drop too slowly). In such cases, an external reset circuit should be used!

Other pins from PE and PB buses are connected to the special service connector called ISP (In System Programming port). This connector enables future upgrading of the on_board system software inside of the soldered ATmega103 processor. Prior to upgrading JMP1 must be in the correct position. The main processor uses a FLASH type of program area to store system software (operating system). Therefore after the previous version of software has been cleared (by the programming device) new software can be uploaded.

This ISP connector is not intended for normal use. Any playing on these pins can be dangerous to the operating system code burned inside processor! And could render the device inoperable. For normal use the position of jumper JMP1 must be between points labelled as A and B (as shown on schematic diagram of VID2G device). Otherwise the RS-232 communication won't work at all.

2.3. Construction

The VID2G generator consists of two small PCB's: one is single sided, and used for the keyboard, with LED's and current protection resistors, the main PCB is double sided with the rest of components. This PCB is dual coated, with its approximate dimensions being. 90 x 65 mm. All resistors and diodes used are in size 1206 SMD packages. All except one are located on the TOP side of the main PCB. All inductors used are standard factory manufactured types (like normal 1/4W resistors with wires). For all of the extension connectors, standard single inline female SIL sockets are used. Wires should not be soldered directly onto these connectors, a male connector should be used with the wires soldered to that. On the main PCB only one circuit socket is used (for EEPROM 24c04). The MAX232 integrated circuit does not need a socket and can be soldered directly onto the PCB. Both of the processors ATmega103 and STV5730A, and the Dallas RTC integrated circuits are only available in packages for surface mounting (TQFP). Therefore they must be soldered directly on to the PCB. All integrated circuits are static sensitive! And care must be taken to avoid damage.

Before any soldering take place, we need to prepare all of the required components (see table of used components), and SMD tools. We will also need some various shapes of tiny tweezers, a vacuum pump for integrated circuit placement. Also two sizes of very high quality solder wire should be prepared: 0.5mm or less and 0.7mm. I prefer "Superspeed" type of solder wire made by Frys Metals Ltd.

When soldering such high-density integrated circuits, low quality solder is NOT an option and will certainly result in an absolute nightmare I personally prefer not to use SMD paste! Instead of that I use: a good quality conformal coating. Spray it all over the PCB and leave to dry. SK10 spray won't cause any unwanted short circuit connections on the PCB after soldering. Specialist soldering tools can be used for SMD, but with care a normal soldering iron, one with the smallest tip available can be used.

Standard soldering and building techniques should be followed. First the smallest parts should be soldered: SMD resistors, capacitors, diodes. All of the parts must be placed in the correct place on the PCB, with values/signatures facing towards the top, This is good practise for all assembly of PCB's and allows the values (where available) to be easily read. Each component must be carefully soldered. First on one connection, using the smallest amount of solder wire (0.5mm or less), verifying it's positioned correctly on the PCB, and only after this soldering onto the second connection. A soldering iron with the correct tip temperature (approx 320 - 340 deg.C) should be used. Special attention should be taken with SMD diodes. On the PCB component print the CATHODE is marked with a DOT for each diode placement.

After all of the SMD resistors, caps and diodes are soldered, the hard work is on the way - soldering of SMD integrated circuits. Soldering should start with the smallest one - The DS1629, and end with largest one - The ATmega103.

Integrated circuits (as well as diodes) MUST be correctly placed onto the PCB before soldering. Normally, pin 1 of each SMD circuit is indicated with a DOT. The ATmega103 does have a dot, but it's not clearly visible. But, there is also a large triangle showing the same circuit corner and pin 1. If processor is placed so that the signature of the cpu is readable and up the right way, then pin 1 is in the left-upper corner. And this one must be placed near JMP1 and JMP2 on PCB.

Soldering of ATmega103 processor will take a bit more effort. The CPU's pins are of a higher density, and the integrated circuit itself must be precisely placed on PCB (carefully check all four sides a few times!). Before soldering the CPU go and have a cup of coffee (or similar) then check the orientation again.

Next, all of the other electronic components are soldered, and finally the 10 pin connectors, crystals and 7805 regulator. Pay attention to ensure the correct polarity of the capacitors and Piezo oscillator. Not all Piezo oscillators will work in this device correctly. The finished - soldered PCB should be cleaned with a brush and PCB cleaning liquid (Electrolube FLU or similar). Do not use Acetylene based liquids. Once dry the PCB can be protected using the same SK10 type spray, but please cover all of the connectors with adhesive tape prior to spraying.

The keyboard PCB is single sided. The buttons used are standard ones (the same as used in UNIPLL project). The buttons and LED's are placed on the front. Buttons directly onto the PCB, and the LED's should be approximately 10mm above PCB (top of LED).

Four 680E SMD resistors and two SIL single line female sockets are placed on the track side of the PCB. 10 wire flat cable is later soldered directly to these two SIL sockets. The other end of the cable is fitted with a 10 pin IDC female connector for connecting the keyboard to the main PCB. Wires are kept in the same orientation: wire nr.1 on the keyboard PCB is connected to wire nr.1 on the main PCB. Therefore the flat cable will not be twisted. This 10 wire flat cable should have length of approximately 12cm.

The soldered and cleaned VID2G PCB is then mounted into a factory made PVC box. A "VELLEMAN" type G407 (made in Germany) as shown in picture 8 is ideal for use and was used by me. The dimensions of the box are 120 x 60 x 30mm. Inside there is sufficient free space for both of the PCB's and for single line PVC battery box. The front cover is cut and drilled with the help of X/Y table, one large hole 12.5 x 62 mm for the keyboard, plus holes of a suitable size for the LED's. On the back cover, holes are cut and drilled for all of the connectors: DB9F for RS-232, above him a DB9M for ADC port, two RCA type or BNC connectors for Video IN and OUT, A 3.5mm DIN female connector for / VSQ output, A hole for a small power switch, plus a hole for the power supply cable.(A standard 2.1mm power socket could also be used). Although I have specified the Velleman enclosure, any home made project box can be used, providing it is large enough to carry the PCB's and battery box. The battery box should be positioned at the lowest point within box ! Video input and output must be connected from PCB to box connectors using 75E coaxial AV cable.

2.4. Device test

Before the first connection of VID2G device to the power supply, we must check both PCB's again. Check that all parts are in the correct positions and that not a single short circuit exists on either PCB. For the initial power_up the VID2G must be connected to the +12v power supply through an Ammeter. The maximum supply current that is permitted is below 100mA. If all is well the normal current will be around 65mA on +12v. Now if all is OK, we can proceed by connecting video input to the external video source (camera, SAT receiver, VCR, etc). And VID2G's output to the CVBS monitor, or to TV set via the Scart/Chinch input. A few seconds after power_on you should see the main system video page on the monitors screen showing the software version, release date, followed by single beep or Morse ident from the on board Piezo oscillator. A few seconds later the VID2G will display the first selected video page. At this time the VID2G device is ready for use.

Normally a number of CPU processes (tasks) run at same time. Upon CPU start-up the first one will read the contents of the EEPROM. If the EEPROM is new one (first run) it will perform an automatic initialisation. This function is only called when a new EEPROM is found, otherwise the EEPROM would be rewritten with default data every time the VID2G is powered. The same initialisation procedure is carried out if the user executes a power_on command called TOTAL RESET. Next the low level initialisation of the STV5730A graphic processor is carried out, followed by the data forming the default system page (devices name, software version, release date,etc). The second process (task) will take care of what to indicate on the LED's. It will actually uses some internal registers from main task and displays the actual state of the device on LED panel.

Concurrent with these procedures one almost invisible task takes place - the flow of ASCII data through the RS-232 interface to the outside world. This data can easily be monitored on any VT100 compatible terminal, or PC equipped with terminal emulation software. Or even better: with the VID2win.EXE software package (downloadable from www server). If only a terminal emulator is available, the communication parameters must be initially set to 9600 8N1 - 9600 bps, 8 bits, no parity, 1 stop bit. However, the Windows manager package "VID2win" does not need any additional set up. Only one free COM port and PC mouse for control. After power_on some system messages will be sent to the receiving window of the PC or terminal.

These are the first signs that the main processor is alive and running. If nothing appears to happen, the RxD and TxD of RS-232 cable (pin 2 and 3) are probably reverse connected. These early system messages will be displayed every time the system is powered_on (or after a CPU reset), and can't be omitted. Also some internal error messages can be seen if anything is wrong with the hardware itself. Also all of these messages will be seen regardless of the VID2G mode (keyboard, or RS-232 mode).

After successful initialisation and displaying of selected power_on picture, the user can start using the device. If the keyboard is active the first LED will be lit and the user can execute available commands from the keyboard. If RS-232 mode is active the first LED will blink, and keyboard commands will not work. The device can be accessed in this mode only via the RS-232 port. Refer to the table of available RS-232 commands in the first VID2G article. If LED's are blinking one by one from left to right

ACCEL Bill of Materials **VID2G.PCB**

Qty	Pattern	Value	Designators
1	100	32.768 KHz	Q4
1	200	4.000 MHz	Q1
1	200	17.734 M / 14.318 M	Q3
1	TRIMC 7.5mm	4-25pF	C26
1	CC1206	12p	C17
2	CC1206	22pF	C12 C13
2	CC1206	39pF	C10 C9
1	5mm NPO	56pF	C20
1	CC1206	220pF	C23
1	CC1206	470pF	C18
1	CC1206	68nF	C22
10	CC1206	100nF	C11 C14 C15 C19 C24 C25 C27 C6 C7 C8
1	2.5mm	2y2/16v	C21
4	2.5mm	10yF/16v	C1 C2 C3 C4
2	2.5mm	100yF/16v	C16 C5
1	CC1812	1N4007	D1
4	CC1206	1N4148	D2 D3 D4 D5
2	CC1206	68E	R12 R2
2	CC1206	75E	R10 R15
4	CC1206	680E	R1K R2K R3K R3K
2	CC1206	1K	R11 R7
1	CC1206	1K2	R18
1	CC1206	2K7	R13
1	CC1206	3K9	R8
2	CC1206	4K7	R3 R5
1	CC1206	5K6	R16
1	CC1206	6K8	R9
1	CC1206	8K2	R14
1	CC1206	12K	R4
2	CC1206	22K	R1 R17
1	INDUCTOR400	20yH	N3
1		27yH	N4
2		47yH	N1 N2
1	JMP2	JMP2	JP2
1	JMP3	JMP3	JP1
2	IDC10M	ADC, KEYBOARD	P2 P3
4	SIP2	DIL	P8 P11 Pn Pps
1	SIP3	DIL rs232	P1
1	SIP4	DIL video in/out	P4-5
3	SIP5	DIL kbd, etx-b	P9 P1k P2k
2	SIP6	DIL isp, etx-a	P2 P7
1	SIP7	DIL aux rgb	P6
1	DB9F	RS-232	P1x
1	DB9M	ADC INP	P3x
2	CHINCH	CVBS IN, OUT	P4 P5
1	3.5mm DIN MONO	/VSQ OUT	P11x
1	DIL8	DIL SOCKET	U3s
1	SO8	DS1629	U6
1	TQFP64	ATmega103	U1
1	SOL28	STV5730A	U2
1	DIP8	24c04 DIL	U3
1	DIP16	MAX-232 DIL	U4
1		7805	U5
2	NPN	BC547B	T1 T2
1	RED 3mm	LED	LD1
1	YELLOW 3mm	LED	LD2
2	GREEN 3mm	LED	LD3 LD4
1	CAP300RP	+5..12v Piezzo OSC.	SP1
1	RESET	RES Button MINI	B0
5	BUTTON	keyboard	B1 B2 B3 B4 B5
1	3.5mm switch	POWER SW	SW1
1		VID2G.PCB	PCB1
1		VID2KBD.PCB	PCB2
1	Project Box	Velleman type:G421	Box1

(running light), then the device is at this point executing a TOTAL RESET - rewriting the EEPROM with default data.

The user can now easily set up callsign(s) - ID1..ID4, QTH, contest Code, title text, Real Time Clock, roll bits, etc. Any data is set to VID2G's RAM area following execution of the SEND button in the Windows program. If a SAVE command is issued all of the data in the VID2G RAM area will be stored onto the EEPROM, A confirmation of the SAVE request must be given when prompted.

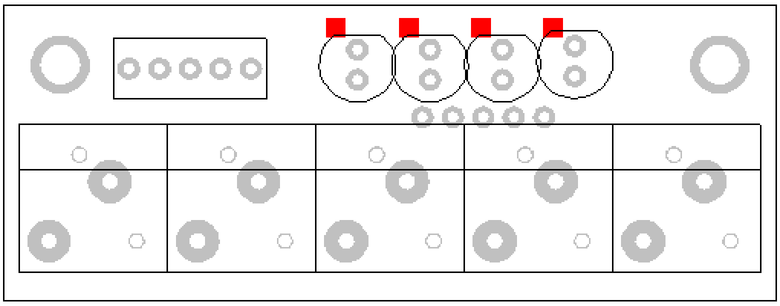
At this point the currently selected video page becomes the starting video page, and any selected roll markers become active. An In-depth description on how to use VID2win Windows package will follow in third part of this article.

Back to hardware. The only adjustments that are needed to the VID2G device is the setting of the 4-20pF trimmer capacitor. With this, the correct clock speed of graphic processor can be adjusted. This calibration can be carried out without the use of an special test equipment by setting the VID2G to FULL mode (stand alone video generator), showing any internal colourful picture. The trimmer should be adjusted so that the colours appear normal and are visible and properly decoded on the video monitor. For best results an accurate frequency counter must be used, measuring the crystal oscillator frequency of the STV5730A. There is one more tuning that can be required if Real Time Clock does not run as accurately as it should. Generally there are two types of 32.768 kHz crystals available. The main difference lies in the crystals internal capacity. One type will run very accurate and no further calibration will be necessary. For the second type, one extra capacitor connected in parallel with the RTC crystal is needed. Therefore, extra pad's labelled "Cx" are available on the PCB. The actual capacity should be determined with time measuring tests (over long time periods) via the RS-232 port. If required the capacitor will normally have a value between 5pFand 82pF, and must be temperature independent. Now for a few final words about the availability of components. Generally the components used in this project are standard, This means that almost all parts are widely available from any specialised electronic component supplier. There are however a few "strange ones". The Dallas DS1629 RTC circuit is new one and available directly from Dallas representatives. The STV5730A and reprogrammed ATmega103 processors are available from the author and developer of theVID2G project: Mijo Kovacevic, S51KQ. High quality factory made PCB's with stop mask and component screen-printed overlay are also available from author in a single package along with complete with the CPU's. There is no full KIT (all parts together in one package) available at this time. The Windows software is available for free download from web server: http://lea.hamradio.si/čs51kq

2.5. Conclusion

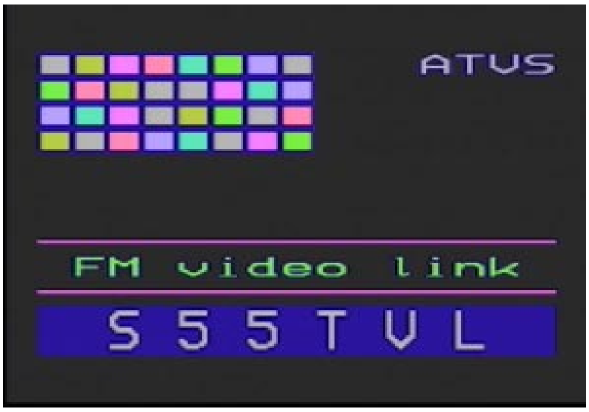
A novice constructor would not normally attempt the building and soldering of theVID2G device at home (due to the fine soldering requirement). By comparison to similar devices manufactured for professional TV broadcasting use, the device is actually very simple. However it is surprising what all constructors with enough care can achieve. And those who say in advance that they cannot solder a few SMD components onto a factory made PCB, probably don't know themselves very well...

The current operating system of the VID2G device is huge, but still only fills about half of the program area available in the main processor. There has been a lot of time spent in developing this simple device, and there will certainly be more. Plans for the future include the addition of further functions, obviously fix any bugs that are found, and addition of sensible requests !!. In the third and final part of the VID2G article we will examine the use of the latest version of the Windows manager package, and cover the requirements for an external Safe_Reset_Device should it be found necessary for your application.



Picture 2.5. - Component placement on Keyboard PCB

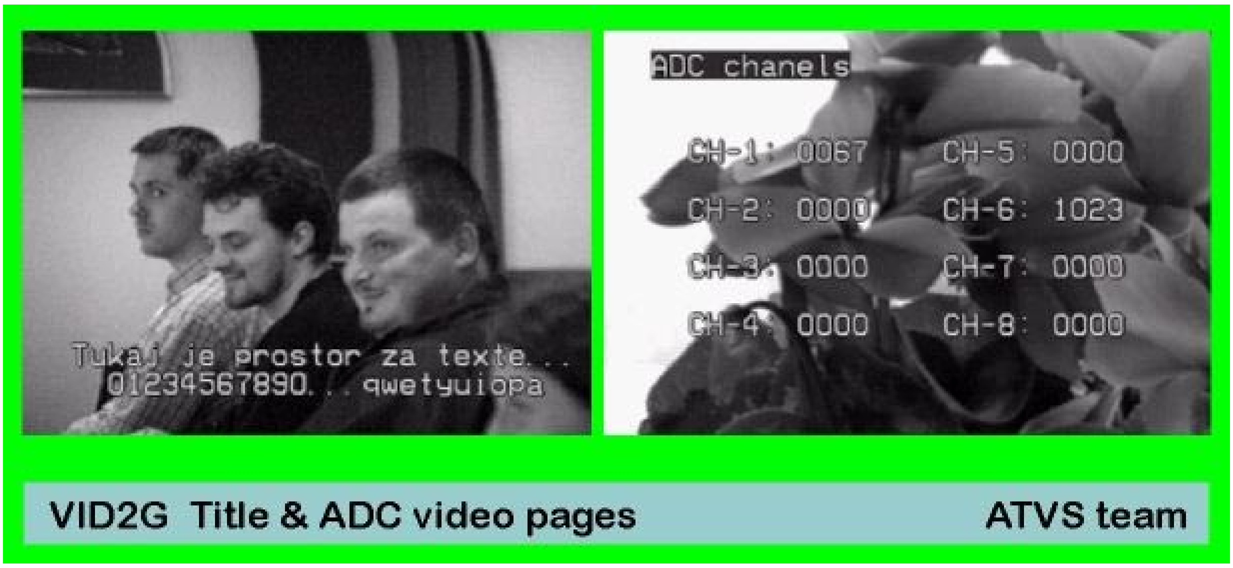
VID2G Video identifier & generator
http://lea.hamradio.si/~s51kq



ATVS

FM video link

S 5 5 T V L

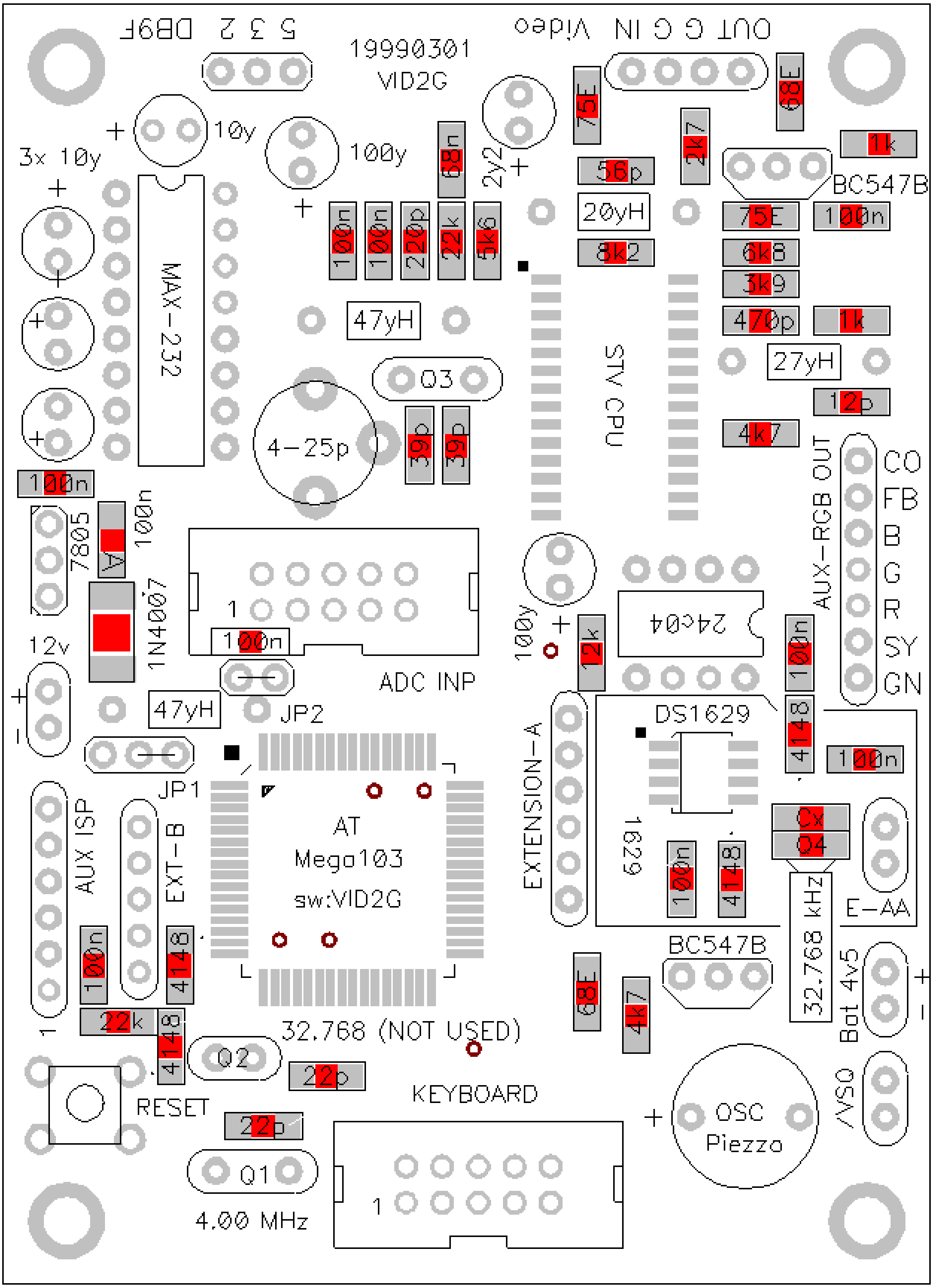


Tukaj je prostor za texte...
01234567890...qwetyuiopa

ADC chanels

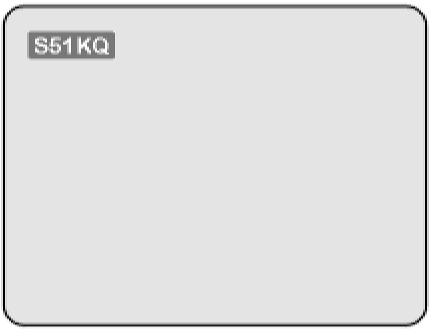
CH-1: 0067 CH-5: 0000
CH-2: 0000 CH-6: 1023
CH-3: 0000 CH-7: 0000
CH-4: 0000 CH-8: 0000

VID2G Title & ADC video pages ATVS team

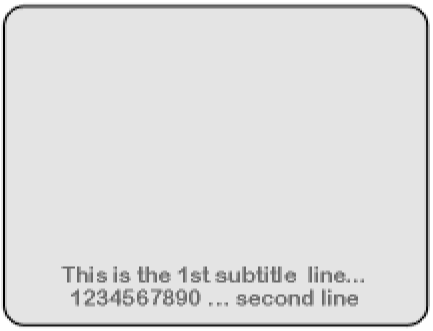


Picture 2.6. - Component placement on VID2G main PCB

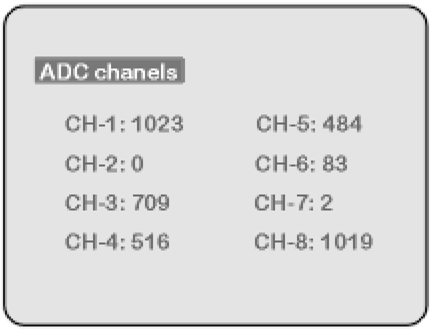
1999-02-19


S51KQ

#1: IDENT fixed ID1, dynamic ID1..ID4
Default mode: MIX (50%)

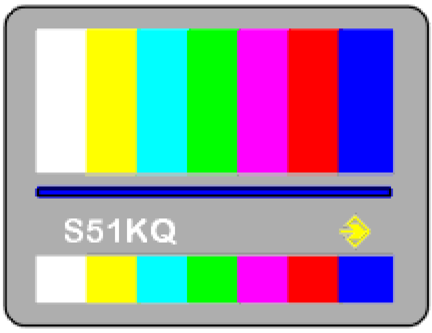

This is the 1st subtitle line...
1234567890 ... second line

#2: Title (1 or 2 lines)
Default mode: NORMAL

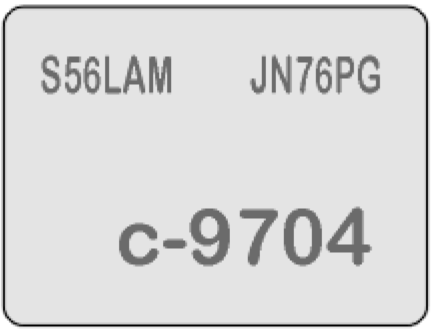

ADC chanels

CH-1: 1023	CH-5: 484
CH-2: 0	CH-6: 83
CH-3: 709	CH-7: 2
CH-4: 516	CH-8: 1019

#3 : ADC chanels (one time or Scan)
Default mode: NORMAL

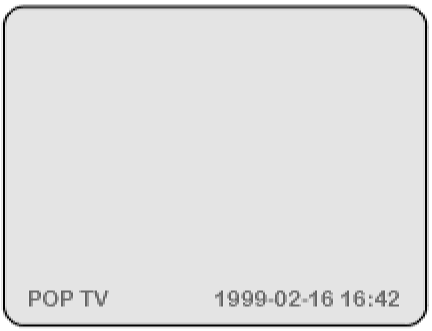

S51KQ

#4 : Test-1
Default mode: FULL - video gen.

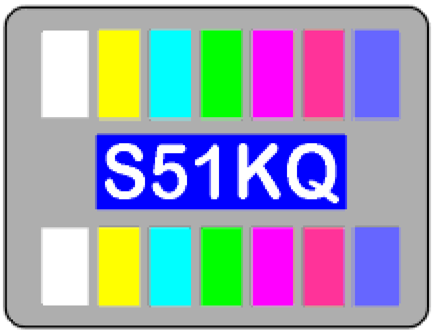

S56LAM   JN76PG

c-9704

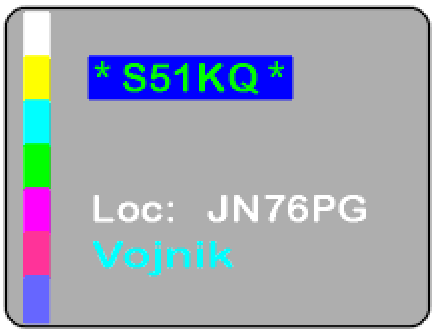
#5 : Contest page
Default mode: NORMAL


POP TV   1999-02-16 16:42

#6 : Time page, normal or mix mode
Default mode: NORMAL

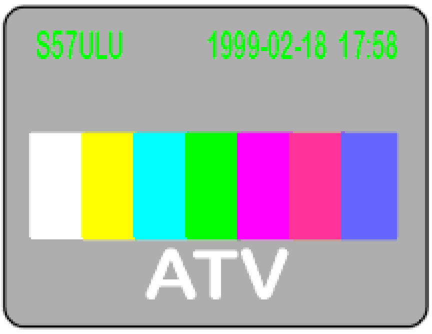

S51KQ

#7 : Test-2
Default mode: FULL - video gen.


* S51KQ *

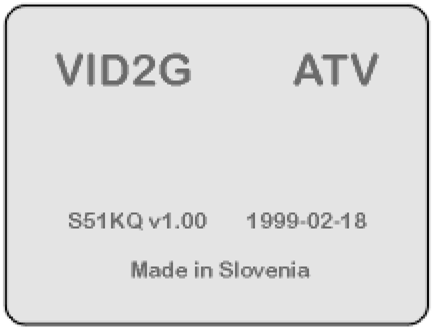
Loc: JN76PG
Vojnik

#8 : Test-3
Default mode: FULL - video gen.


S57ULU   1999-02-18 17:58

ATV

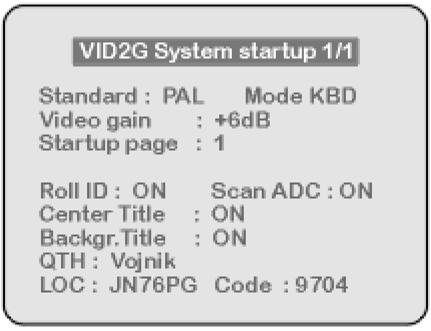
#9 : Test-4
Default mode: FULL - video gen.


VID2G   ATV

S51KQ v1.00   1999-02-18

Made in Slovenia

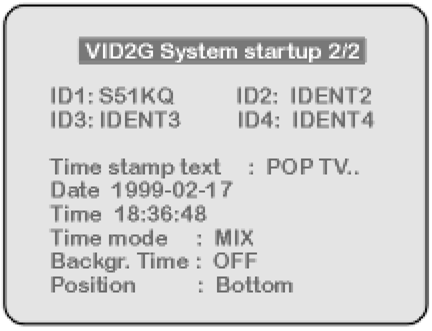
#0 : System start
Default mode: NORMAL


VID2G System startup 1/1

Standard : PAL   Mode KBD
Video gain   : +6dB
Startup page   : 1

Roll ID : ON   Scan ADC : ON
Center Title   : ON
Backgr.Title   : ON
QTH : Vojnik
LOC : JN76PG Code : 9704

#10 : System data 1/1
Default mode: NORMAL


VID2G System startup 2/2

ID1: S51KQ   ID2: IDENT2
ID3: IDENT3   ID4: IDENT4

Time stamp text   : POP TV..
Date 1999-02-17
Time 18:36:48
Time mode   : MIX
Backgr. Time : OFF
Position   : Bottom

#11 : System data 2/2
Default mode: NORMAL

VID2G v1.00 - Predefined video pictures available in NORMAL, MIX or FULL mode

S51KQ 1998/1999   http://lea.hamradio.si/~s51kq

VID2w - Windows software support for VID2G

3.1. Introduction

The VID2G stand alone video generator and identifier has been previewed in both CQ-ZRS (Slovenia) in the issues 1,2 and 3/1999 and also in REPEATER (Netherlands) in the 3/1999, 4/1999 and 1/2000 issues.

Although for normal use and displaying of generated pages it is not necessary for the VID2G to be connected to any other control equipment (PC or VT100 Terminal), the setting up of VID2G and the configuration of user pages will require the use of either a PC or VT100 Terminal.

The RS232 settings for the PC or terminal should be set to 9600,8,N,1. This will allow the setting up of things like the RTC, Video standard, startup page and rolling modes etc.

Version 1.00 of the VID2G operating system is now more than a year old. During this time the system has been extended to include many new features. One of the first things that were done was to add a number of RS232 commands. In the previous release (v1.30) only the 2/3 of the commands used pure Text parameters, while the remaining 1/3 of the commands needed Binary parameters. The result of such a structure was that in order to retain simple control, the use of special PC software was needed.

This problem reminds me some users from the USA, where the VID2G is used in NTSC mode on ATV repeaters as system video display. Therefore in the most recent issue of the operating system version 2.01, we have included some additional Bin parameters, also full support for all commands with Text parameters. In last release some in-depth system changes have taken place. Because of the insecure internal reset of the main CPU - when a low quality power switch is used on VID2G. Or a Power Supply with large amount of capacitors causing slow voltage dropout and raise, the system software automation of self-initialisation of the EEPROM was removed from v2.01. From now on, the user must execute a manual command for EEPROM initialising, when the VID2G is powered up for the first time. Also the additional safe hardware Reset circuit was designed and built (see last figure 3.6.).

There were also a number of other changes made which are not visible to the user. All modes on the keyboard and operation remained the same, but now, after power on the VID2G device will wake up with a "VID" message in CW. The execution speed of commands was raised, and a third system page has been added.

VID2G maintenance can be performed with the help of any computer capable of RS232 communication with 9600bps 8N1. But for very simple use, and without the need to know any of the devices commands, the special program needed to be written. Many of our home computers are PC compatibles with Windows operating system & mouse, or track ball. Therefore any complicated maintenance software can be written for easy - user friendly use.

3.2. VID2w201

Last release of VID2G's maintenance PC software v2.01 was released on the19th February 2000, and was written for use with Win95/98. It now has support for all of the possible commands implemented in the VID2G device. All of the required files, including samples and help files are compressed in a single ZIP package. There is no special installation procedure built in. Such automation takes up a lot of free disk space. And in the case of VID2w software there is no need for any special actions. Therefore the complete package can be manually unzipped into a new folder (C:ĐVID2w201, or similar).

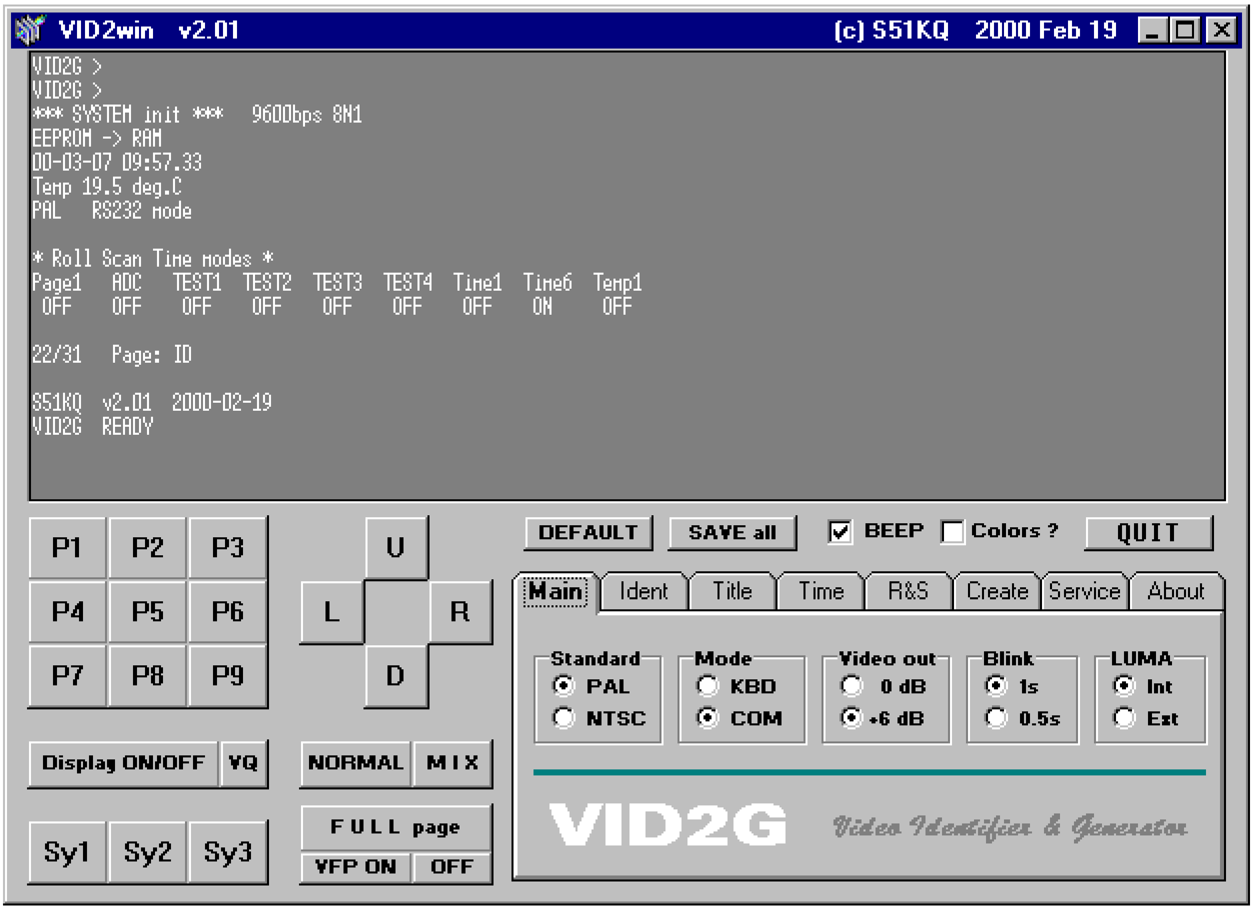
Users are free to manually add a shortcut on to their desktop (with right mouse button click on to exe) if required . If there is an INI file included in package, it should to be copied into the Windows folder (C:ĐWINDOWS). All other files (samples, readme, etc) should remain in the same folder as the main executable. Now the VID2G device needs to be connected with an RS232 cable onto PC's COM port.

With a double click on VID2w's icon the program will run. After initialisation of data, a screen similar to that shown on figure 3.1.will appear in the centre of your computers monitor. The layout of the screen is split horizontally into two sections. The upper half is a big RS-232 monitor. VID2G device will send important data on to its RS232 output at all times, regardless of the selected mode (Keyboard, or RS232). All of this data is captured and displayed by the Windows program in this particular area. This display area is purely a monitor area and therefore commands can not be written to it. . For writing commands the user should use the lower half of the command console and use the buttons and boxes for entering data.

This particular software, VID2w is written a bit differently from my other Windows programs. You won't find any ordinary drop down menus along the top line, also there is no ordinary extended Help file built in. However, each button or entry field has its own Hint - short help built in. This Hint will be displayed when the mouse cursor stays on the selected area for more then one second.

In the main window (figure 3.1., left below) are 9 buttons which are used for displaying any one of the 9 predefined video pages. The buttons are numbered as P1 to P9 (for page 1 to page 9). The currently displayed (generated) picture can be switched OFF and ON very fast, at the press of the "DISPLAY ON/OFF" button. This function is only operational when there is no Roll or Scan functions running. VID2G device will return the input Video quality report at the press of the "VQ" button. When the user has selected FULL mode (for the output) the device will report back "video quality bad..." This is not an error, and such a report should be taken as normal. If the user presses on the Sy1 to Sy3 buttons, the system data page will be displayed. On each of these some of the important set-up information will be shown.

Buttons with letters U, D, L, R are used to move the displayed



```
VID2G >
VID2G >
*** SYSTEM init ***   9600bps 8N1
EEPROM -> RAM
00-03-07 09:57.33
Temp 19.5 deg.C
PAL   RS232 mode

* Roll Scan Time modes *
Page1   ADC   TEST1  TEST2  TEST3  TEST4  Time1  Time6  Temp1
 OFF    OFF    OFF    OFF    OFF    OFF    OFF    ON     OFF

22/31   Page: ID

S51KQ  v2.01  2000-02-19
VID2G  READY
```

☑ BEEP □ Colors ?

Standard: ● PAL ○ NTSC

Mode: ○ KBD ● COM

Video out: ○ 0 dB ● +6 dB

Blink: ● 1s ○ 0.5s

LUMA: ● Int ○ Ext

Figure 3.1. Main window of VID2Wxxx software

picture in all directions. Some of the pictures have a 'Save' option built in, for the saving of the predefined picture position.

Pressing on buttons "NORMAL", "MIX" and "FULL" will execute display mode changes in the VID2G device. Normal mode will display 100% B/W generated text over coloured incoming video. Mode MIX will display 50% transparent B/W generated text over coloured video. The Full mode is in this latest software release extended onto VFPM sub mode. Normally the text was generated in full mode, as a stand-alone video generator, without the displaying of the input video source. In version 2.01 the active VFPM mode (Video Full Page Mode) will allow user to add (in main full mode) the unsynchronised input video in the background of coloured generated text. First impressions this new function may be that it has no value at all, but if the device is used on an ATV repeater, or ATV beacon, the incoming video source can be switched onto the background at any time by using a video squelch device or external processor. This is for monitoring of video presence on input. And all of this can be done very fast, without need to switch into Mix or Normal mode.

Pressing on button named "DEFAULT" will cause a total reset - initialisation of the VID2G device with so called "factory" set-up, as if it was the first run, or on delivery of VID2G device. All user data will be erased, and the EEPROM be written with default data. This command is dangerous (loss of personal data in device), therefore the mouse cursor will change when over this area to a "no park" symbol.

There is no UNDO function available after this button is pressed! Similar cursor indications are used to indicate any unusual area. When new personal & other data is set up and sent to the VID2G device it is only stored in the devices RAM area! Prior to turning the power off, the user needs to execute a "SAVE ALL" command. With a press of this button, the VID2G device will copy all registers from RAM into the EEPROM area.

Before The "SAVE ALL" command is issued there are a number of things to be considered: The currently displayed VID2G picture will be saved as the default start-up video page. If there is a Roll function needed on the current page, the function should be enabled, but after that the page should not be restarted. Otherwise the roll function will start, and the save option will be disabled.

There is no confirmation needed to execute the "SAVE" command. If the user is using a Terminal or another communications programme, the "SAVE" command must be confirmed by pressing "Y" when prompted in the same way as any other critical command.

There are some further functions to be mentioned. The first one is called "BEEP". It's new one built into the latest operating system. If checked, The VID2G device will confirm certain actions with beeps from the onboard Piezo beeper. If left unchecked the beeper will generally remain silent but will still sound for important reports (I2C failure etc.). The second one is called "Colour".

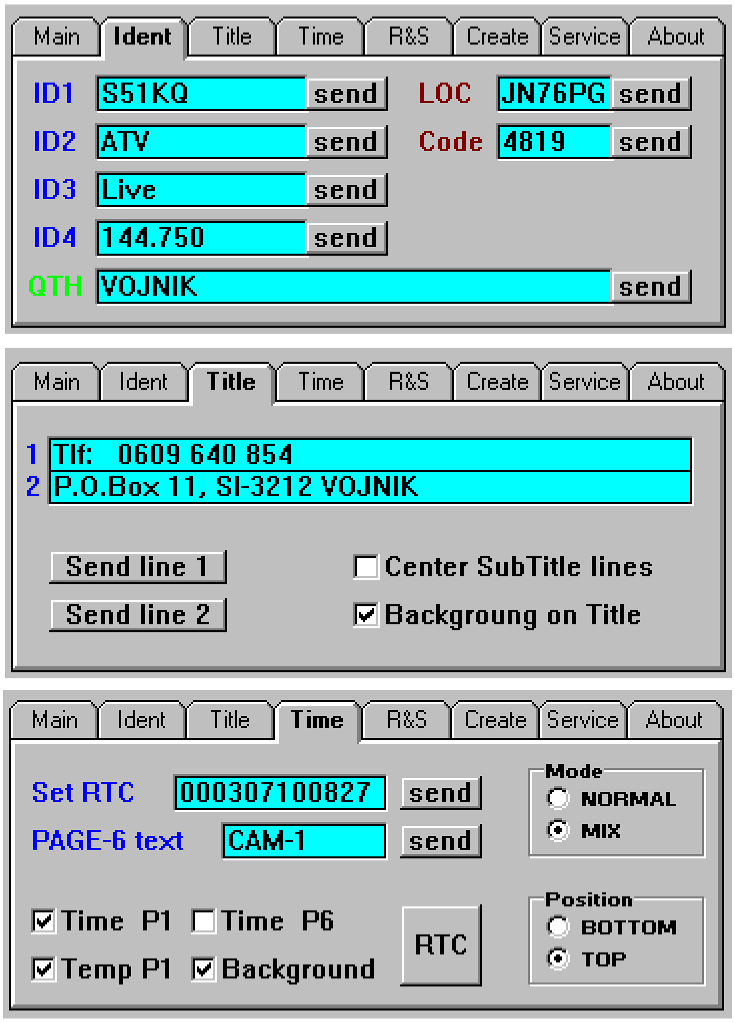
After checking, the colour coding Help table will be displayed. And finally pressing on the button "QUIT" will terminate this Windows software.

Until now all of the discussed command buttons are positioned on the front layer of the main window, because those buttons are needed most of the time. All of the other buttons and entry areas are 'hidden' on eight sub menus (figure 3.1.,).

The first sub menu is called MAIN. There is not a single button on this sub menu only a few groups of check areas. Each area, when selected is checked with a dot. In the first group the PAL or NTSC video standards can be selected. The VID2G device as factory default uses PAL as primary standard. The second group is used to select commanding mode: via COM, or via five-button keyboard. The third area is used to change video output gain. Factory predefined gain is +6dB. The Blink period will change timings for blinking duration of characters (if used). The last one is called LUMA control, which is new in V2.01.

The Graphical Processor used in the VID2G does not have full internal LUMA controls. Therefore the displayed picture when displayed on a monochrome monitor would not have any levels of grey. With the help of this software switch the user is able to connect an external LUMA control circuit based on three trimmer resistors onto the processors inputs, and manually adjust LUMA levels. Both of the inputs on the Graphic Processor can be connected through this resistor network onto the RGB output from the device to generate LUMA information that can be used for greyscale information for use with a monochrome monitor. However this can in turn cause problems with the colour output when using 'FULL' mode. This software option has been included to limit the effects.

The next sub menu window called IDENT is shown on figure 3.2.,. There are 7 entry fields with buttons for sending each entry into the VID2G devices RAM. There are four fields for Identification (callsign, etc). The primary one is ID1. Other three will be displayed if the Roll function is activated on that particular page. QTH and LOC fields are used to send City name and Universal Locator to the device. The CODE field is used to send a 4 digit ATV contest code into devices RAM. Note that any data that is sent by pressing the "SEND" button adjacent to each field is stored only in the devices RAM area. After sending all of the required data, it should be Saved into the EEPROM using the "SAVE" command!



ID1 S51KQ send
LOC JN76PG send
ID2 ATV send
Code 4819 send
ID3 Live send
ID4 144.750 send
QTH VOJNIK send

1 Tlf: 0609 640 854
2 P.O.Box 11, SI-3212 VOJNIK
Send line 1
Send line 2
□ Center SubTitle lines
☑ Backgroung on Title

Set RTC 000307100827 send
PAGE-6 text CAM-1 send
Mode
○ NORMAL
● MIX
☑ Time P1 □ Time P6
☑ Temp P1 ☑ Background
RTC
Position
○ BOTTOM
● TOP

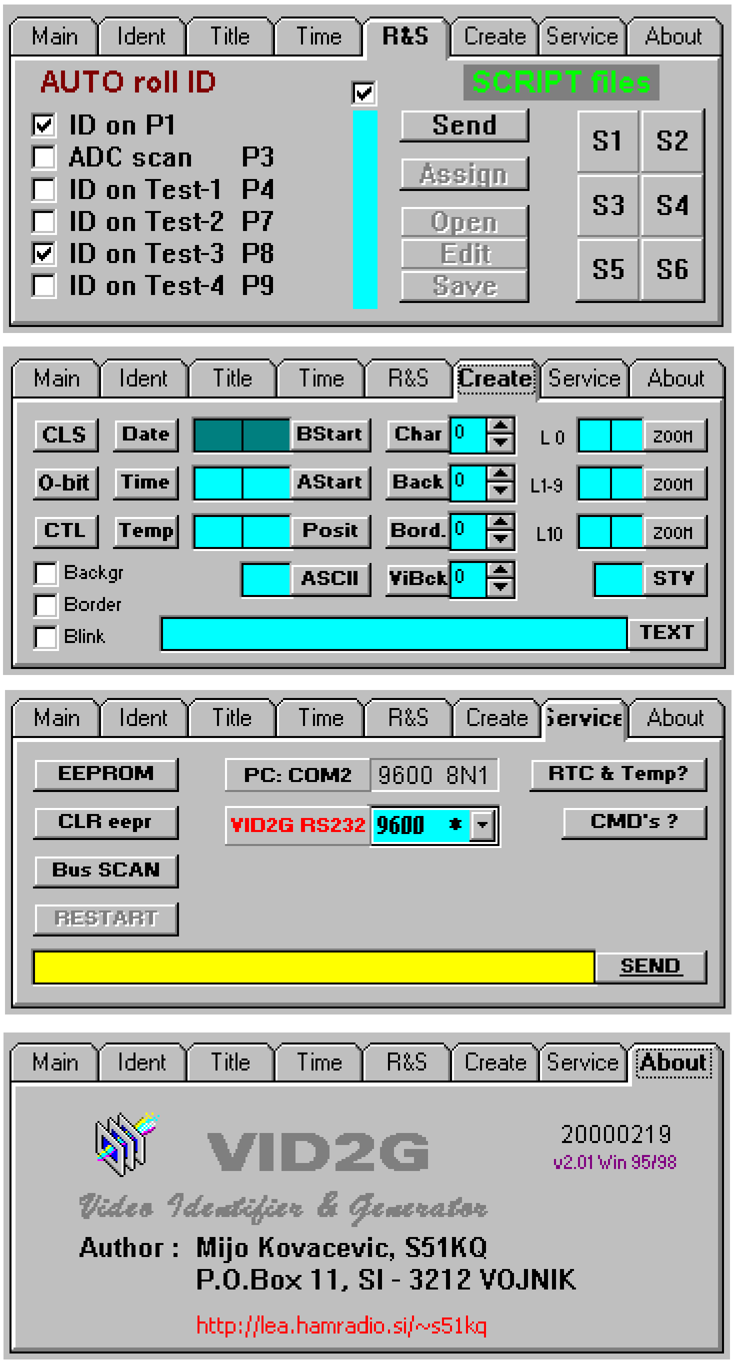
Figure 3.2. Sub menus: IDENT, TITLE and TIME

The third sub menu window is called TITLE (figure 3.2.). All entries on this sub menu are used by the Title video page only. Each of two entry lines can hold up to 28 characters, and the data is sent to devices RAM after pressing on the "SEND LINE" buttons. Title lines can be automatically centred if the "Center SubTitle Lines" field is checked. They may also have a background block displayed too.

The forth sub menu window is used to control and set the Real Time Clock and its data. The RTC can be simply set-up by entering two digit numbers into the "Set RTC" field. Example: 000308142947 is the correct entry for year 2000, month 03, day 08, hour 14, minutes 29, and seconds 47. The used RTC circuit is one of the latest designed by Dallas semiconductors, and would not (did not) have any problems with Year 2000. After sending the RTC data, the clock starts to run using these details. This function does not need the "SAVE" command to be issued as it is sent directly to the RTC circuit which is battery powered.

Changing of any other data on this sub menu will need the "SAVE" command to be issued after all actions. The user is able to add time and/or temperature printout on some of the pages, by checking the relevant fields. Note: The display of the clock and temperature details can only be used with certain pages and will only update if the "Roll" option is activated.

The fifth sub menu window (figure 3.3. on top) is called ROLL & SCRIPT. On the left half of window are located check fields for enabling



☑ ID on P1
☐ ADC scan P3
☐ ID on Test-1 P4
☐ ID on Test-2 P7
☑ ID on Test-3 P8
☐ ID on Test-4 P9

PC: COM2 9600 8N1
VID2G RS232 9600

20000219
v2.01 Win 95/98

Author : Mijo Kovacevic, S51KQ
P.O.Box 11, SI - 3212 VOJNIK

http://lea.hamradio.si/~s51kq

Figure 3.3. Sub menus: R&S, Create, Service and About

Roll Functions on pages. The right half is used to manipulate so called SCRIPT files. These files are a brand new property of the latest operating system. Script files are in fact ordinary text files. Each file can contain all of the needed RS232 commands to create your own VID2G video page. The main property of these files is that they can be simply created and edited in any text based editor. After the Script file is created, it can be simply uploaded into the VID2G using the "SEND" command.

Also if the VID2G device is used as system display on an ATV repeater, the sysop defined video pages can be simply uploaded into the graphic CPU RAM with the help of the main repeater computer, or microcontroller. All RS-232 commands are allowed to be used in each Script file. Automation built into the device will take care of interpretation of most entries.

Naturally execution of any Script line must be checked before final use. Only after a successful pass may the "COM answer OFF" command may be added in the front of first script command. See the samples in package! Transfering data thru RS232 cable at low speed is a very slow process. Plus the built in interpreter is slow, therefore inhibiting the device from sending answers to the outside world will improve the uploading speed a lot! Also, the command sequence in Script files is very important to maintain reasonably upload speeds. The proper way is to write commands in a single line, close to each other, with one space character as a delimiter, with the exception of a few variable length entries which must be either at the end of a line or on their own line. Comments are allowed, but should be avoided as they are still sent to the CPU and then discarded, dramatically slowing down upload times.

All users should read through the included help and sample files to gain maximum efficiency of VID2G device. All Script files use an extension of *.IMG.

The number of Script files is not limited, but at present only one picture can be displayed. This is a limitation of the STV graphic CPU. It has sufficient RAM area for one single picture only. After the VID2G device is switched OFF, contents of the last shown Script file in the device are lost. It will need to be uploaded after power ON again. VID2w package V2.01 includes 6 final Script files, plus two DEMO Script files fully commented (English and Slovene).

The sixth sub menu window is called CREATE. This one is for advanced users, and is used for experimenting with STV registers and commands. All needed register data can be found in the STV5730A PDF file available from the manufacturer. Three ZOOM fields on the right-hand side of the window can be used to Zoom IN or OUT current video output in both directions (X/Y).

SERVICE is the seventh sub menu window. There are some special functions for performing system tests located here. Pressing on the EEPROM button will read out the EEPROM contents in HEX and ASCII. Button CLR EPR will erase the EEPROM circuit - FFh will be written in all locations. Button BUS SCAN will check and display all available slave addresses on the I2C bus.

RTC&TEMP will readout date/time and the temperature. CMD's? will display short help (located inside VID2G device).

The lower entry field and adjacent SEND button, can be used to send any command to the device manually, like on VT100 terminal. There are two fields left on this sub window. The upper one determinates the Windows COM software speed, while lower one determinates VID2G's RS232 speed. Both numbers must be the same. It is recommended to use the highest possible (operational) speed.

Finally the eighth sub window will display some software information (version, release date, etc), plus the authors address.

3.3. O.S. upgrading

The new operating system does not need any upgrades, but if you have an older operating system than v2.01 it can be upgraded, like flash BIOS in modems, ISDN phones, SAT receivers , etc. The VID2G uses a 'flash' type of main processor, therefore with internal support and relevant system upgrading software this is an easy task. There are no hardware changes needed to the VID2G device.

Upgrading of operating system can be done in two ways: if user brings their device to our lab, to be upgraded. Or you can upgrade the device by yourself at home.

Yes, it is an easy job, but it's not without risk! Incorrect connection between the PC and VID2G can cause permanent damage to either or both devices. Anyway all that is needed is a simple 5 wire cable (as shown on figure 3.5.), a PC computer with Win95/98 operating system, The VID2gUPG package, free LPT-1 port, plus the latest VID2G.SUT file. This upgrade file can be obtained from the author. All other files are available free on WWW.

Once the cable is prepared and connections checked, JMP-1 on VID2G board must be set to the service position "B-C". The VID2G device is then powered and connected to the PC's LPT-1. The upgrading software should now be run, and SUT file is manually loaded.

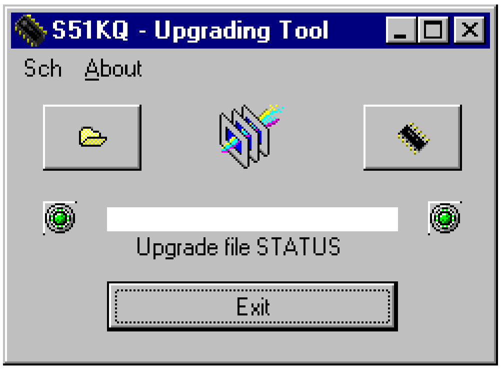


Figure 3.4. VID2G Operating System upgrading tool

After pressing on right button the operating system inside VID2G device is upgraded in less than a minute. A colour bar will show the upgrading progress. While the upgrad is being carried out both the PC and VID2G should NOT be interrupted, or powered down !!!

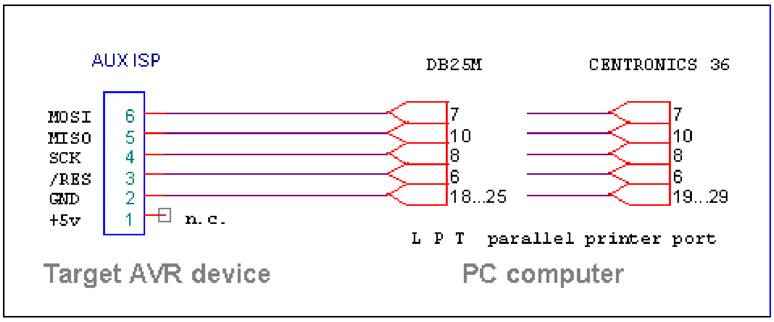


Figure 3.5. Operating system upgrading cable

There are no known limitations on PC side. Any computer from PC386, to 400MHz Pentium should work OK. The upgrading software will first identify the target device and installed version, then the upgrading process will start. This software is not an AVR programmer. It would not program a blank AVR processor with your data or any software! For actions like this you need to use an AVR programming software package with compiled source code. After upgrading is completed, JMP-1 must be switched back on to normal position "A-B", allowing RS232 communication with the device.

3.4. Conclusion

In this third and final article of the VID2G device we described the Windows manager package. With the help of this package the life of a VID2G owner is simplified. But the story of this project does not end here. There are already some ideas for new functions being considered, and the future will bring new ones too. The selected main processor is the biggest and most powerful one, out of the AVR family. Although version 2.01 of the operating system for the VID2G is already very comprehensive it still only uses 70% of the program area.

What about full colour pictures to be displayed over incoming colour and synchronised video? There is an aux RGB output available on board. An external dual input RGB modulator circuit would need to be added. A simple one can be built using an old Motorola MC1378 circuit. But, a better approach would be to choose one of the latest, low power RGB modulator circuits designed for primary usage in video camcorders.

Well, with this last article, the VID2G series are concluded. This project has also shown in which direction the development of new ATV devices is going. Not only miniaturisation, but also lower and lower power consumption, with more powerful functions, and without leaving good picture quality behind.

- - -

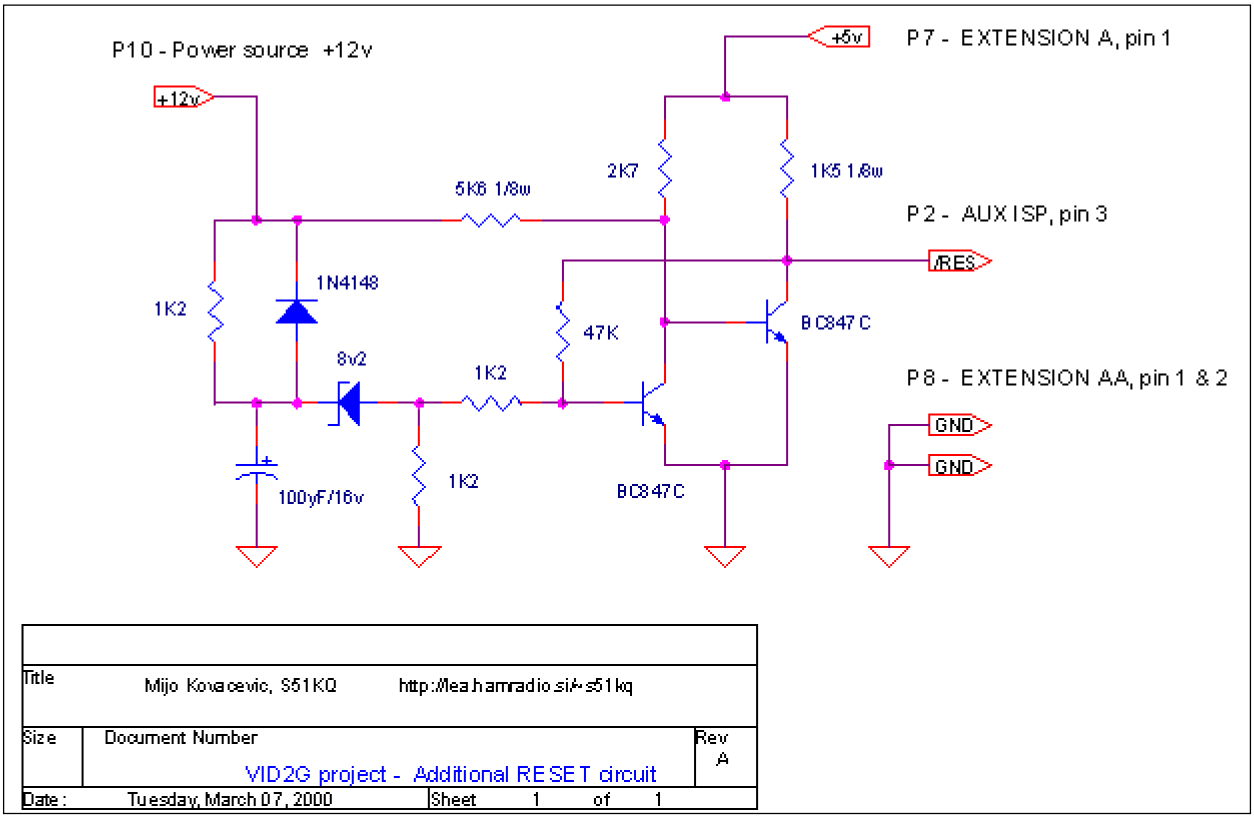
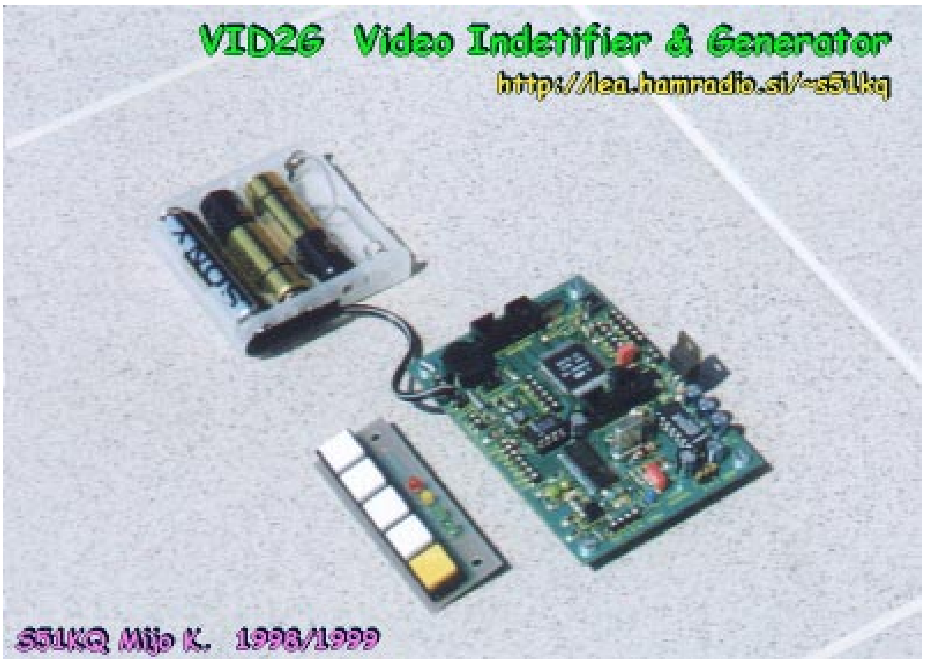


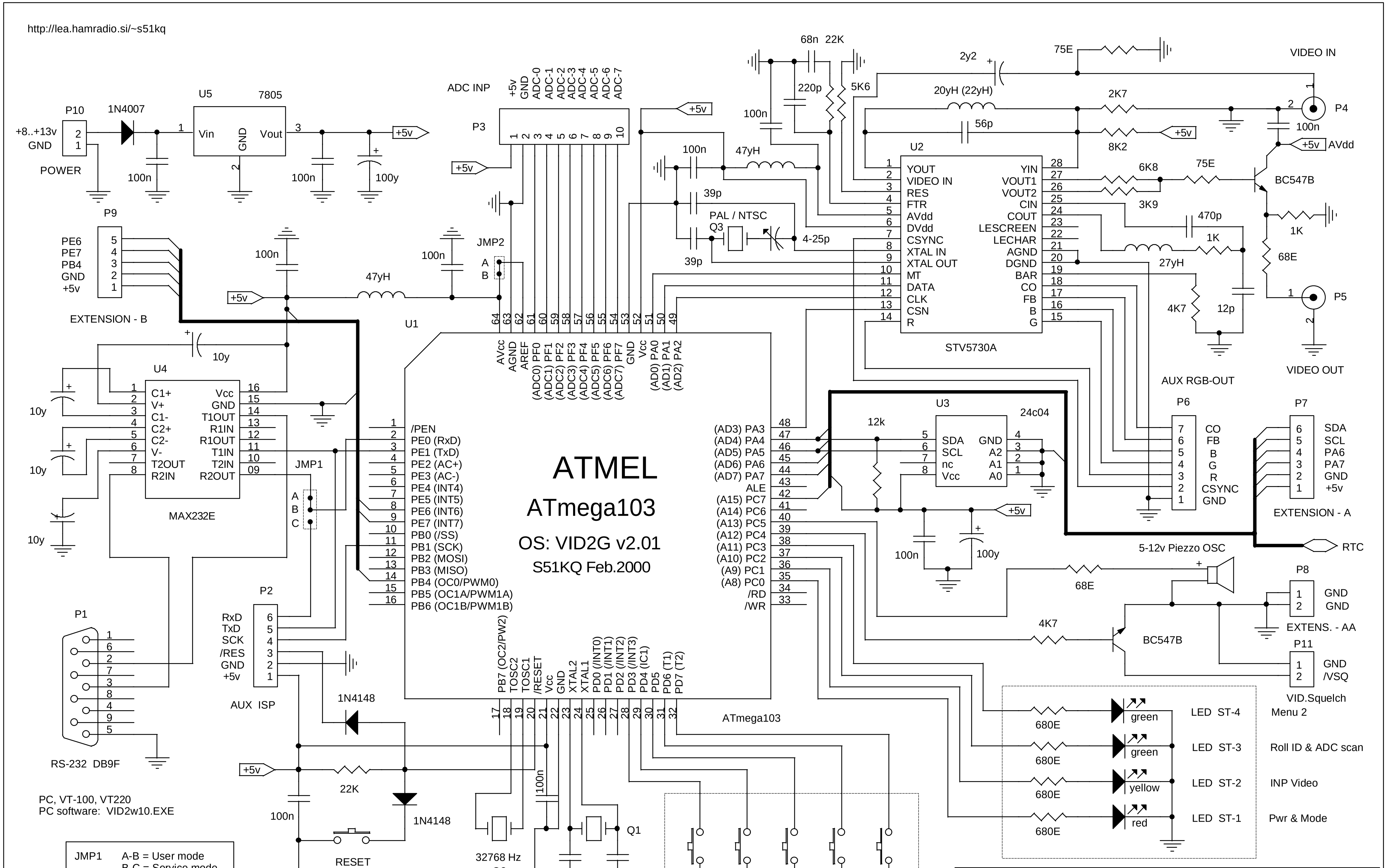
Figure 3.6. Additonal hardware Reset circuit


VID2G Video Indetifier & Generator
http://lea.hamradio.si/~s51kq
S51KQ Mijo K. 1998/1999

Reference

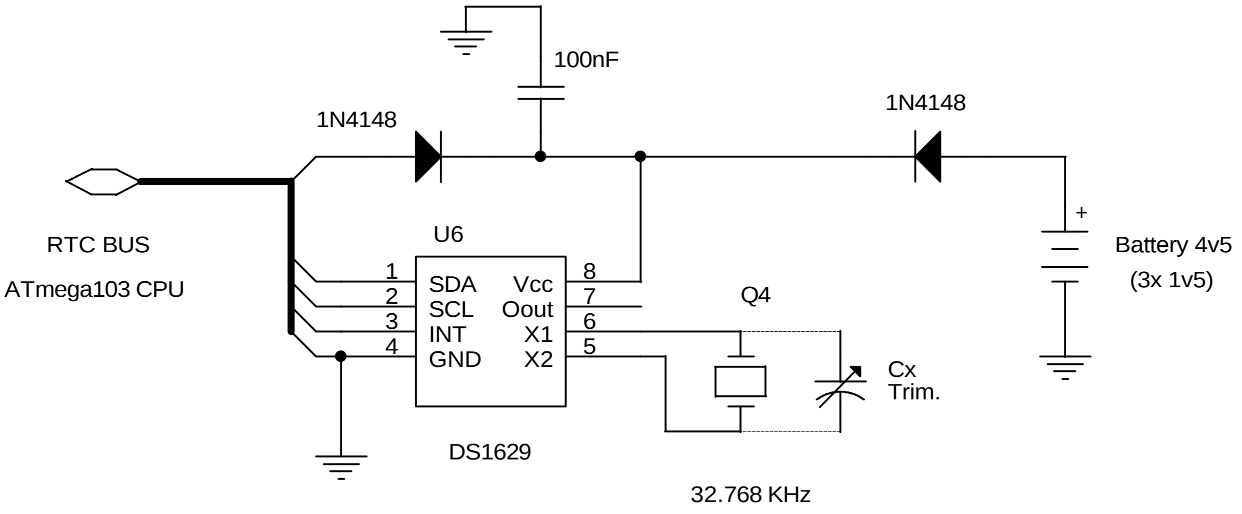
SGS-THOMSON home page	***ENG***	***STV5730A.PDF***
ATMEL CD, October 1998	***ENG***	***ATmega103.PDF***
CQ-ZRS 1, 2, 3 / 1999	***SI***	***VID2G generator***
REPEATER 3, 4 / 1999, 1 / 2000	***NL***	***VID2G Video Callgever***
CQ-ZRS 4 / 1994 pages 29-34	***SI***	***VID-01 ATV video identifikator***
S51KQ CD, PDF book	***ENG***	***IAT.PDF An Introduct. to ATV*** ***ISBN 0-9513779-2-2***

http://lea.hamradio.si/~s51kq



Title		
VID2G - Video identifier & generator S51KQ (c) 1998 / 1999		
Size A4	Document Number Mijo Kovacevic, S51KQ, P.O.Box 11, SI-3212 VOJNIK	Rev 1.1
Date: Tuesday, March 07, 2000	Sheet 1 of 2	

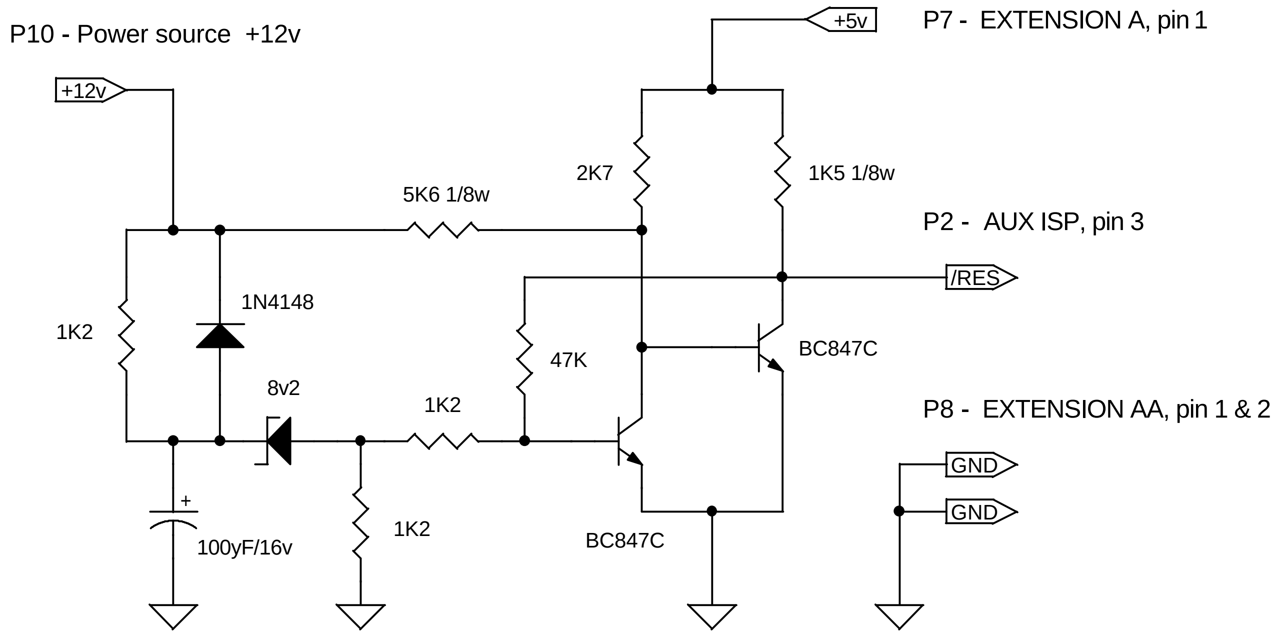
Schematic of VID2G module, main board



VID2G Real Time Clock & Digital Thermometer

All rights reserved.

Title	VID2G - Video identifier & generator (RTC) S51KQ (c) 1998 / 1999	
Size	Document Number Mijo Kovacevic, S51KQ, P.O.Box 11, SI-3212 VOJNIK	Rev 1.1
Date:	Tuesday, March 07, 2000	Sheet 2 of 2



Title	Mijo Kovacevic, S51KQ http://lea.hamradio.si/~s51kq		
Size	Document Number VID2G project - Additional RESET circuit		Rev A
Date:	Tuesday, March 07, 2000	Sheet 1 of 1	