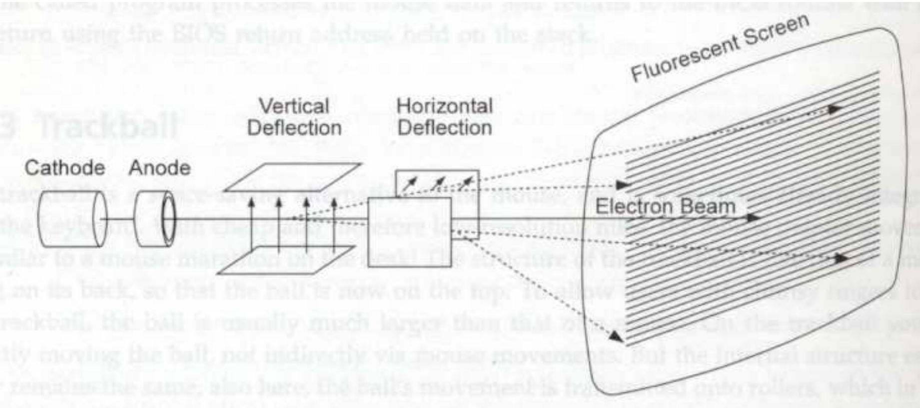
Monitors and Graphic Adapters

- To the process of displaying the information a **graphic adapter** and **monitor** are involved.
- Graphic adapter: an element between a processor (and its I/O bus) and a monitor. They are interconnected by a **cable**.

Monitors

- To display an image the method similar to conventional TV is used (in CRT monitors – Cathode Ray Tube).



- Displaying images on a monitor with a **cathode ray tube (CRT)**.
  - The screen's content is divided into many horizontal rows – every row is formed by **pixels** or **pixel elements**.
  - The cathode emits electrons, they are accelerated by the electric field of the anode and hit the screen.
  - The screen is coated with a **fluorescent material** which illuminates when fast electrons emitted by cathode hit it.
  - The negatively charged electrons are **deflected** by electrical fields generated by electrical voltages at the **deflection plates**, in this way scan lines are formed.

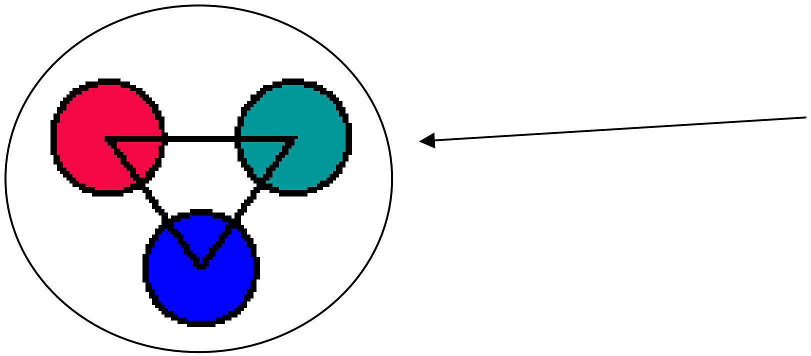
- **Horizontal retrace**: the electron beam has reached the right hand end of a row – then it must return to the beginning of the next scanline.
- **Vertical retrace**: the electron beam has reached the lower right hand corner of a row – then it must return to the upper left corner.
- Actual display of an image consisting of many pixels – electron beam must be modulated accordingly: **bright pixel – strong beam hits the screen, dark pixel – no beam (or less intensive)**.
- Displaying coloured images: 3 electron beams are required - R (red), G (green), B (blue) - **additive colour mixing**.
- White point – all three colours are illuminated at the same intensity.
- Absolute beam intensity: it determines the brightness.
- Relative beam intensity: it determines the colour.

One pixel consists of three small dots – R (red), G (green), B (blue). The coloured dots are very close each other – eyes are not able to distinguish them – in our eye the three dots are seen as a mixture of three colours (additive composition of basic colours)

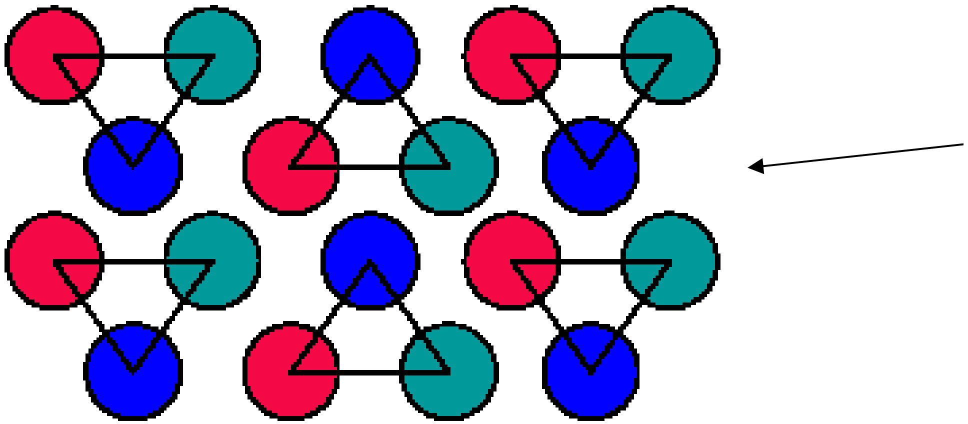
Coloured CRT monitor works with 3 electron beams (one per colour), each of them hits its colour dot on the screen.

If the information about colour is digital – the particular colour component participates/does not participate on the resulting colour (the table lower).

If the information about colour is analog – then the particular colour component has certain level of brightness – as a result a great number of colours can be developed as based on the number of bits representing the colour.



One pixel consisting of three colour dots R, G, B



Pixels on the screen

The composition of colours (digitally controlled monitor):

Basic colours: R, G, B	**Resulting colour**
no	black
R	R
G	G
B	B
R, B	magent
R,G	yellow
B,G	grey
R,G,B	white

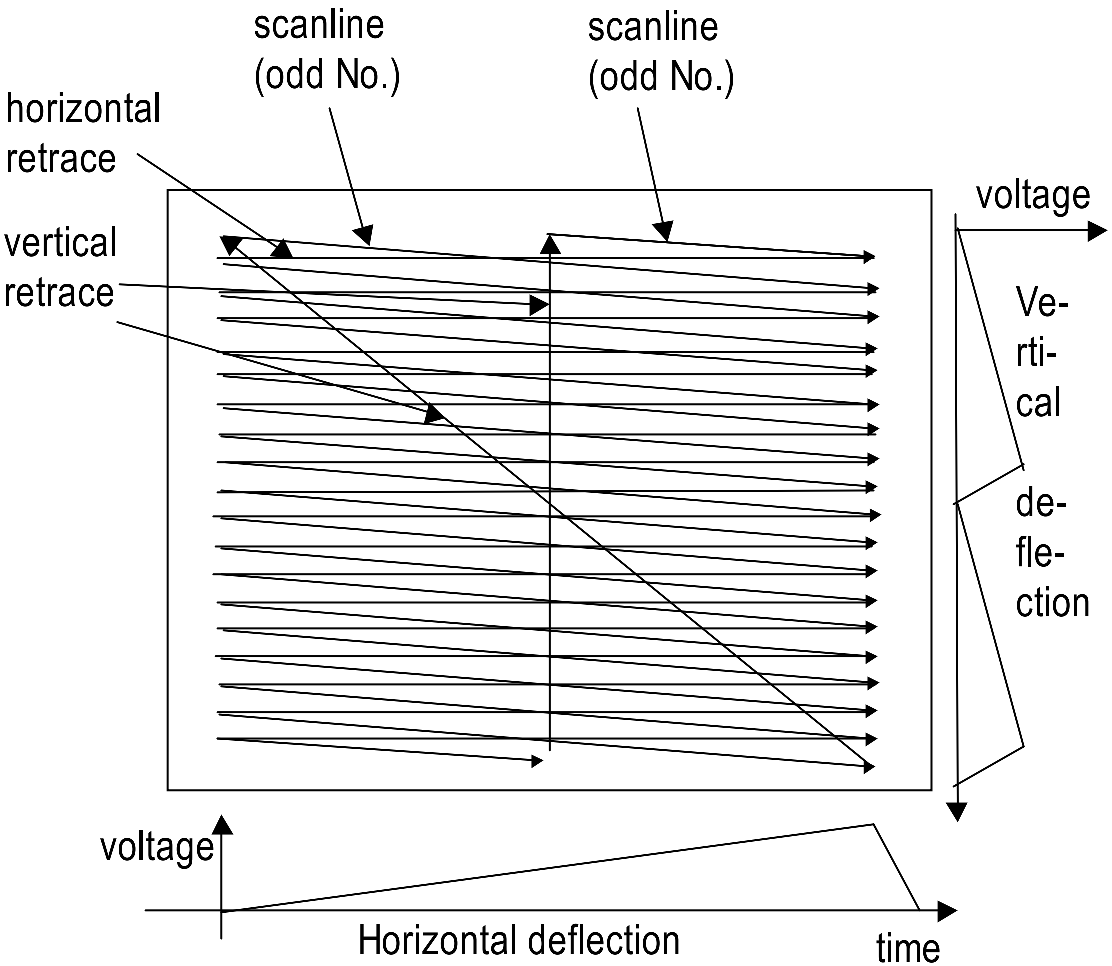
- The electron beam as well as the horizontal and vertical retraces must be synchronised.

- The **graphics adapter has to provide the video signals required for the individual pixels** (i. e. the intensity and colour signals), as well as **the synchronisation signals for the horizontal and vertical retraces** – all these signals can be identified in the cable between graphics adapter and the monitor.

- **Example**: VGA adapter with a resolution of 640 * 480 pixels =>
  - every line has 640 pixels,
  - the image is generated by 480 scanlines,
  - after 480 horizontal retraces a vertical retrace occurs,
  - the image is built 60 times per second, that is, a vertical retrace occurs every 16,7 ms,
  - video bandwidth is 25 175 MHz (the rate at which the pixels are written onto the screen) => every second more than 25 million pixels have to be transferred between adapter and monitor and displayed.

- How the bandwidth is calculated:
  the number of pixels on the screen x the number of screens/s
  - VGA adapter (higher type): 1024 x 768 pixels = 786432 pixels/one screen
  - To calculate the bandwidth, the number of screens displayed per one second must be taken into account.
  - Usual values now: the bandwidth of 100 MHz.

- **Interlaced mode of operation** – a simple and cheap way out.



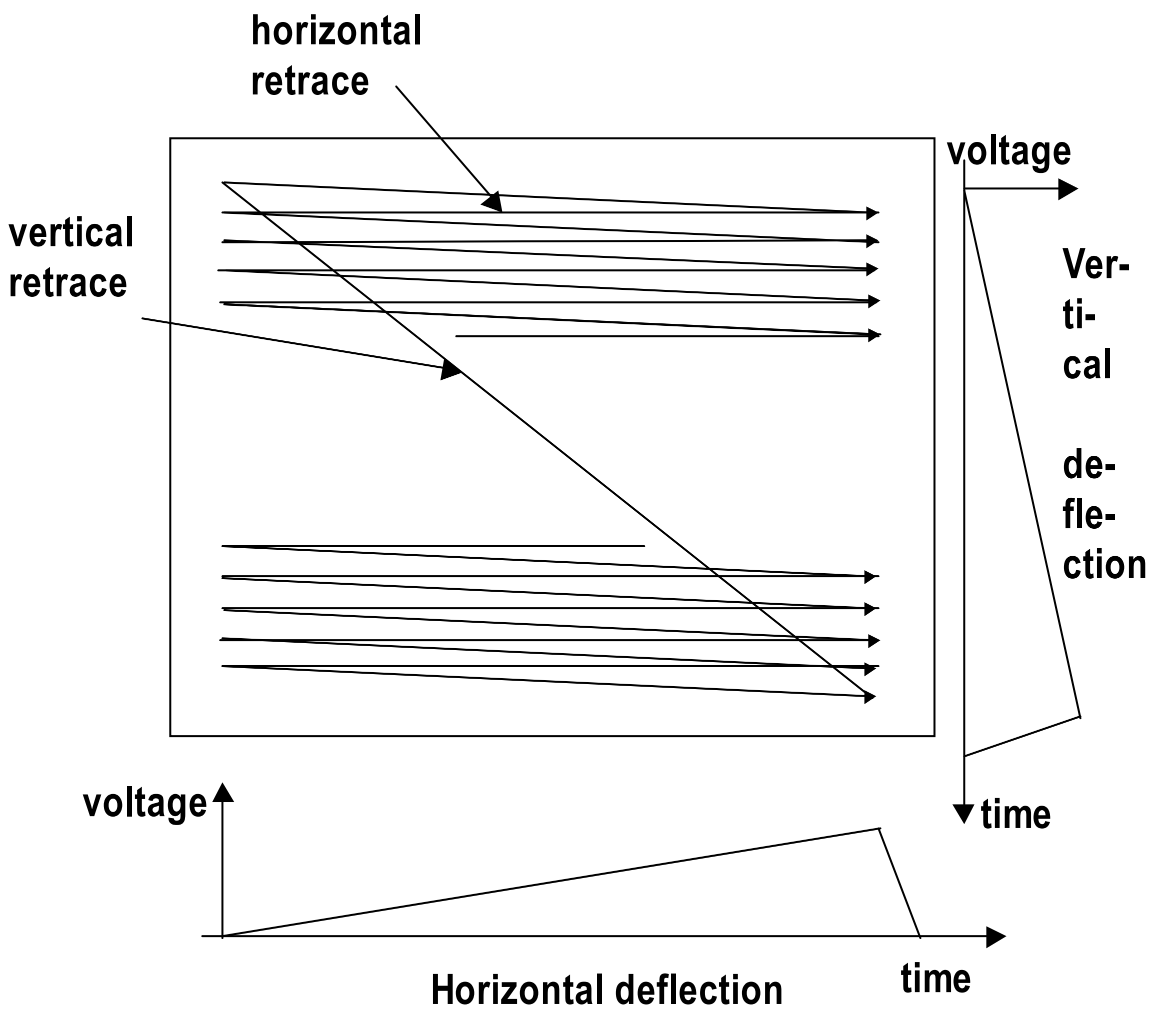
Interlaced mode of operation

- The contents of the screen is written in two passes:
  1st pass: scanlines with odd number are displayed.
  2nd pass: scanlines with even number are displayed.
- **The consequences**:
  The line frequency and thus the video bandwidth is halved, but the picture frequency (the number of vertical retraces) remains the same.
  The beam runs through the odd lines during the 1st run (one half of scanlines is run through), during the 2nd run it goes through the lines which are adjacent to those from the first run - **our eye is not able to recognise it**.

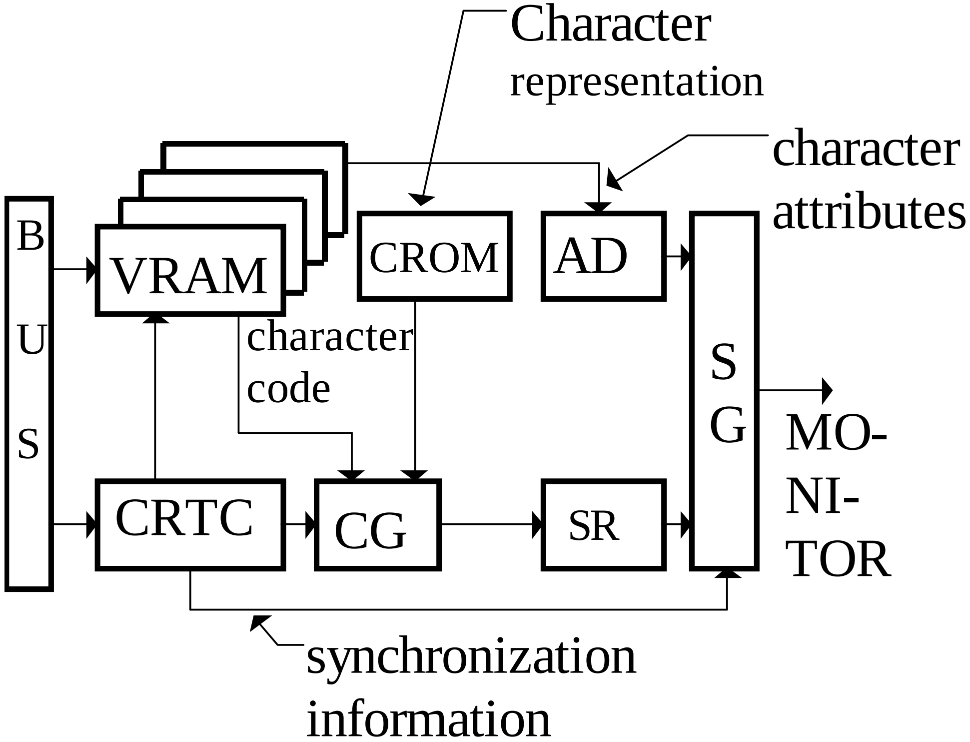
- Interlacing is not a new method – it has been used since the start of the TV era (to prevent the bandwidth of TV channels from going beyond all possible bounds).
- Europe:

- TV image consists of 625 lines separated into two partial images of 312,5 lines.
- The partial images are transmitted 50 times per second, so that the TV effectively displays 25 complete images per second.

- **Non-interlaced mode of operation**
  The lines are written in succession (line number 0, 1, 2, 3, ……).



Non-interlaced mode of operation



CROM - character ROM, AD - attribute decoder, SR - shift register, CG - character generator, CRTC - cathode ray tube controller, SG - signal generator

Block diagram of a graphics adapter

- The modes of operation of a graphic adapter:
  - text mode of operation
  - graphics mode of operation

Text mode of operation

- Characters are defined by their **ASCII codes**, they are assigned **attributes** => in text mode every character on screen is assigned a word of two bytes in **video RAM** (**VRAM**).
- **CROM (Character ROM)** – holds a pixel pattern for the corresponding character (for every ASCII code).
- The ASCII codes of characters together with their attributes are stored in video RAM.
- The character code is the index into the character ROM (where the first scan line is stored).

- A character is displayed on the screen in the way which depends on the value of the attribute: **blinking**, with **high intensity**, **inversely** or with a **certain colour**.

15							8	7							0
BLNK	$BACK_2$	$BACK_1$	$BACK_0$	INT	FOR_2	FOR_1	FOR_0	CHR_7	CHR_6	CHR_5	CHR_4	CHR_3	CHR_2	CHR_1	CHR_0

BLNK	Blink 1 = on 0 = off
$BACK_2$ - $BACK_0$	background colour
INT	Intensity 1 = high 0 = normal
FOR_2 - FOR_0	Foreground colour
CHR_7 - CHR_0	Character code

The structure of two bytes reserved for character code and character attributes in video RAM

- The meaning of bits:
  BLNK is set: the character is displayed blinking.
  BAK_2 - BAK_0: background colour of the character (eight different colours are possible):
  The actual colour depends on:
  - the value of the bits BAK_2 - BAK_0
  - the graphics adapter (the selected graphics palette, whether a monochrome or colour monitor)

  FOR_2 - FOR_0: the character foreground colour (in the same way as BAK_2 - BAK_0)
  CHR_7 - CHR_0: **the code of the displayed character and serve as an index into the character table in character ROM**.
- Text mode: every text row is created from a certain number of scanlines (in our example 14 scanlines).
- Pixel pattern of "O" in text mode in character ROM/RAM (14 scanlines for text row).

- An example: the representation of "0" character is stored in character ROM in the following way:

	1							**8**	
1	0	0	0	0	0	0	0	0	00
	0	0	1	1	1	0	0	0	38
	0	1	1	0	1	1	0	0	6C
	1	1	0	0	0	1	1	0	C6
5	1	1	0	0	0	1	1	0	C6
	1	1	0	0	0	1	1	0	C6
	1	1	0	0	0	1	1	0	C6
	1	1	0	0	0	1	1	0	C6
	0	1	1	0	1	1	0	0	6C
10	0	0	1	1	1	0	0	0	38
	0	0	0	0	0	0	0	0	00
	0	0	0	0	0	0	0	0	00
	0	0	0	0	0	0	0	0	00
14	0	0	0	0	0	0	0	0	00

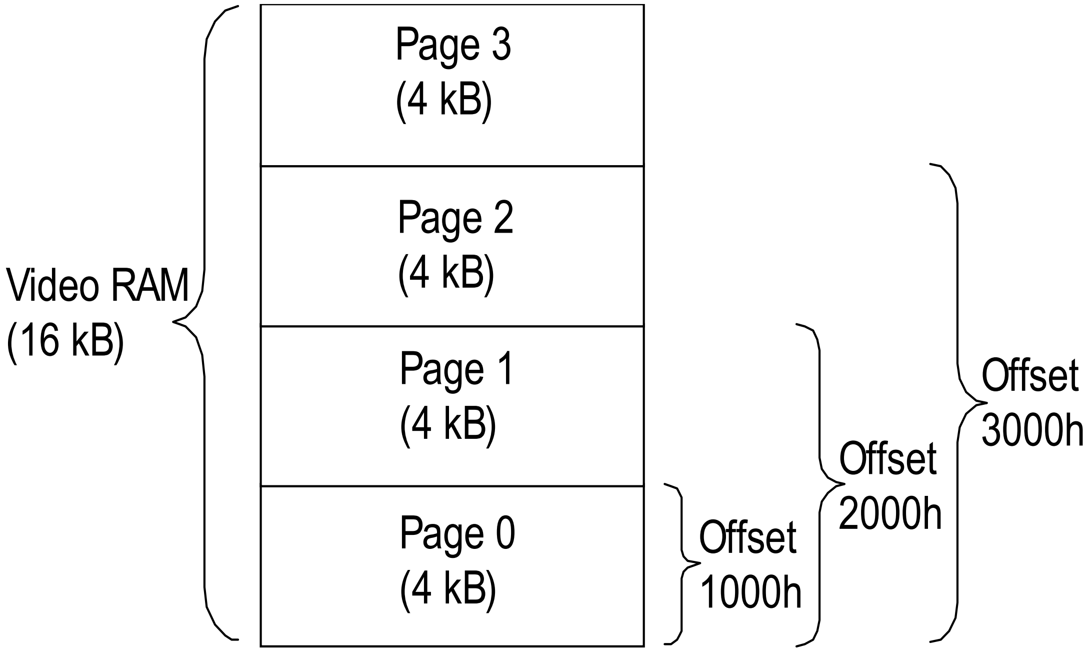
- Every alphanumerical character is displayed as a pixel pattern held in the character ROM or RAM (00 38 6C C6 ….00).
- A "1" at particular location: a pixel with the foreground colour is displayed.
- A "0" at particular location: a pixel with the background colour is displayed.
- On the basis of character code and its pixel pattern in ROM, sequence of bits is generated and transferred into a shift register.
- In more sophisticated adapters RAM was used instead of ROM (starting from EGA adapter) to store the patterns of characters => user defined fonts could be used.
- The inputs to the signal generator:
  - the bit stream from the shift register,
  - the attribute information from RAM ,
  - the synchronisation signals from the CRTC.
- The monitor processes the video signals and displays the symbolic information on the screen of the monitor.

The conclusion: the character information in the video RAM thus modulates the electron beam of the monitor through the intermediate stages of **character ROM**, **character generator**, **shift register** and **signals** from adapter to monitor.

- RAM organisation in text mode:
  - RAM is regarded as a linear array.
  - The first word (two bytes) is assigned to the character in the upper left corner, i. e. the character in row 1, column 1.
  - The second word then describes in row 1, column 2, etc.
  - Depending upon the text resolution (number of characters on the screen), a varying number of words is necessary to accommodate the whole screen's contents.

- An example:
  The standard resolution of 25 rows with 80 character each => 2000 display memory words with 2 bytes each => a total of 4 kB of. **High resolution SuperVGA** adapters with 60 rows of 132 characters each need 15840 bytes (60 x 132 x 2).

- The video RAM installed on the adapter is always much larger than a single screen requires.
- (S)VGA - in text mode the video RAM is divided into **pages** - every page can accommodate a whole screen page.



- The calculation of offset pages:
  $1000h = 0.\ 16^0 + 0.16^1 + 0.16^2 + 1.16^3 = 4096$ (4 k)
  $2000h = 0.\ 16^0 + 0.16^1 + 0.16^2 + 2.16^3 = 8192$ (8 k)

Graphics mode of operation

- Graphics mode: the bytes directly determine the intensity and colour of the corresponding pixel, the information in video RAM is directly used for generating the picture on the screen.
- The data in the video RAM is directly transferred to the shift register and the signal generator - **character ROM and hardware character generator do not play role - they are disabled**.
- **Example**: how the BASIC command PRINT "A" is interpreted in:
  Text mode: **ASCII code of the character "A"** is written into the corresponding location in the video RAM.
  Graphics mode: **the pixel matrix of the character "A"** is copied to the corresponding location in the video RAM.
- The power of the graphics mode: not displaying text but the **capability of drawing free graphics and lines**.
- Depending upon the number of displayable colours, one pixel on the monitor is assigned a number of bits in the video RAM required to distinguish between colours.

- Monochrome graphics - only a single bit corresponding to bright (bit = 1) or dark (bit = 0) per pixel.
- Multicolour graphics - several bits per pixel:

  VGA with 256 different colours - 8 bits per pixel are required => the storage capacity of the must increase rapidly with:

  1. increasing resolution
  2. the rising number of colours in the pictures.
- The consequences:
  1. When (S)VGA appeared on the market, it was equipped with 1 MB of video RAM.
  2. High resolution graphics adapters could accommodate up to 8 MB of video RAM.

The situation with modern graphic adapters and monitors:

Each pixel is represented by 3 colours: R, G, B

For each colour, 8 bits are used to represent the “value of the colour” in the video memory, altogether 24 bits for one pixel (R – 8 bits, G – 8 bits, B – 8 bits)

The capacity of video memory needed to store the information displayed on the monitor in graphics mode is much greater than in text mode.

The function of CRTC on video adapter

- It controls the graphics adapter as a whole.
- One important function - control, to cover it, the CRTC must be able:
  - **To address the video memory**, where the codes of characters are stored (the character in the upper left corner corresponds to the lowest address provided immediately after a vertical retrace).
  - **To address the character ROM/RAM** where the scan lines are stored.

    => the address generated by the CRTC consists of two parts:

  An example: in the text mode video RAM has the capacity of 16 kB, divided into 4 pages, each with the capacity of 4 kB.
  - To address 16 kB , 14 bits (MA0 - MA13) are needed (generated by CRTC).
  - To address 14 scanlines within each character, additional 4 bits (RA0 - RA3) are needed.

- The electron beam reaches the end of the scanline => **the CRTC activates the output HS** to issue a horizontal retrace (horizontal synchronisation).
- The whole text row is displayed after 14 scanlines => the line address RA0 - RA3 returns to the value 0, the CRTC provides new addresses MA0 - MA13.
- The end of the last scanline of the last text row => **the CRTC activates the output VS** to issue a vertical retrace (vertical synchronisation) => exactly one picture has thus been formed on screen.
- The screen image cannot persist (and as the CPU might overwrite the video RAM with new values in the meantime => the screen must be refreshed within a short time interval (typically 50 - 100 Hz).
- **Important**: (S)VGA holds the pixel patterns in a character RAM into which you can load user defined characters and thus define your own characters.

  - If more than 4 bits are available to address => characters with a maximum height of 32 scanlines are possible.
- Graphics mode: MA0 - MA13 and RA0 - RA3 are combined into one 18 bit address => a video RAM of up to 256 kB can be addressed.