```
'*************************************************************************
**
'                    Channelscan example to record audio
'
'     The recorded audio can be played in any WAV compatible player
'
'              VisualRadio Basic Keywords are Color Coded
'                       Copyright© 2002 Liedtke GmbH
'*************************************************************************
**

      'all variables to be declared
      Option Explicit

      'state of the server's scanner
      Const SERVER_SCANNER_STARTED        =          1
      Const SERVER_SCANNER_PAUSED         =          2
      Const SERVER_SCANNER_CONT           =          3
      Const SERVER_SCANNER_STOPPED        =          4

      'PCR-1000 Constants
      Const Mode_LSB                      =          0
      Const Mode_USB                      =          1
      Const Mode_AM                       =          2
      Const Mode_CW                       =          3
      Const Mode_FM                       =          4
      Const Mode_FMW                      =          6

      Const WIDTH_3kHz                    =          0
      Const WIDTH_6kHz                    =          1
      Const WIDTH_15kHz                   =          2
      Const WIDTH_50kHz                   =          3
      Const WIDTH_230kHz                  =          4

      'folder to hold wave data
      Const WAV_FOLDER                    =          "C:\WAVDATA\"

      'we delete small records
      Const DELETE_SMALL_RECORDS          =          True

      'records should last at least 3 seconds
      Const MIN_RECORDING_TIME            =          3

      'index into the 'Type' box of the server
      Const SCANTYPE_CONT                 =          0
      Const SCANTYPE_HALT                 =          1
      Const SCANTYPE_PAUSE                =          2
      Const SCANTYPE_STOP                 =          3

      'allow server to start up
      Const SERVER_START_DELAY            =          20

      Const SERVER_IP                     =          "192.168.0.8"
      Const SERVER_PORT                   =          3200

      dim timersave
      dim mytemp
      dim fs, f, s
```

```
dim GotAudioDriver
dim i, j
dim RecordingAudioFlag
dim SaveBandScope

'function StartServer launches the VisualRadio server -
'returns TRUE if succesfully started
if StartServer("VRADIO /CCS") then

      'display message
      StatusMessage = "Launching Server...."

      'allow server to startup
      Wait SERVER_START_DELAY

      ' built in properties
      IP = SERVER_IP
      Port = SERVER_PORT

      'display status
      StatusMessage = "Connecting to Server...."

      'OpenSocket returns TRUE if successfull, built in function
      if OpenSocket then

            'built in properties

            'sample the audio with 8 kHz/sec
            AudioSampleRate = 8000

            'use mono
            Channels = 1

            '8 bits might be enough for shortwave
            BitsPerSample = 8

            'save bandscope status into variable SaveBandScope
            SaveBandScope = BandScopeStatus

            'disable bandscope
            SetBandScope(False)

            'macro selects VisualRadio's type of scanning
            ScanType = SCANTYPE_HALT

            StartTheChannelScanner

            'Loop until we 'Break' the loop
            do
                  'Strg-C pressed or 'Run -> Break' selected
                  'or 'Break' button pressed ?
                  'built in property Break
                  if Break then
                        'Break received
                        exit do
                  end if
```

```
                        'someone fiddled with the server buttons
                        if ScannerStatus <> SERVER_SCANNER_STARTED then
                              StartTheChannelScanner
                        end if

                        if CommunicationError then
                              StatusMessage = "Communication Error"
                              'reset errorflag
                              CommunicationError = False
                        end if

                        'ScannerBusy = True -> scanner is scanning
                        'ScannerBusy = False -> scanner is halted
                        'so if scanner is halted -> record audio
                        if Not ScannerBusy then
                              if Not RecordingAudioFlag then
                                    'prepare for recording
                                    GotAudioDriver = AudioOpen()

                                    'AudioOpen returns TRUE if driver found
                                    if GotAudioDriver then
                                          RecordingAudioFlag = true

                                          ' built in property AudioFileName
gets assigned
                                          'by helper function FileName
                                          mytemp = Signal
                                          AudioFileName = FileName(Frequency)
                                    else
                                          exit do
                                    end if

                                    'start recording
                                    AudioRecord
                                    'we need this timer to delete short
records later on
                                    timersave = timer
                              end if
                              StatusMessage = "Recording " & AudioFileName
                        else
                              'scanner is busy, i.e. scanning
                              if GotAudioDriver then
                                    AudioClose
                                    GotAudioDriver = false

                                    'delete records < MIN_RECORDING_TIME
seconds
                                    'if flag DELETE_SMALL_RECORDS is set to
true

                                    if DELETE_SMALL_RECORDS then
                                          if timer < timersave +
MIN_RECORDING_TIME then
                                                set fs =
CreateObject("Scripting.FileSystemObject")
                                                set f =
fs.GetFile(AudioFileName)
                                                s = f.delete
```

```
                                                set f = Nothing
                                                set fs = Nothing
                                          end if
                                    end if
                                    'reset flag
                                    RecordingAudioFlag = false
                              end if
                              'display current activity
                              StatusMessage = "Scanning..."
                        end if
                  loop

                  if GotAudioDriver then
                        AudioClose
                  end if

                  'macro stops VisualRadio's scanner
                  ChannelScannerStop

                  'enable bandscope
                  if SaveBandScope = -1 then
                        SetBandScope(True)
                  end if

                  'unconditional wait for operation complete of macro
SetBandScope()
                  WaitOPC 5

                  'we close the connection to the radio server
                  CloseSocket
            else
                  'we connected to an IP currently not available
                  MsgBox "IP not available"
            end if
      end if

'-------------------------------------------------------------------------------
----------------
      'helper function to assign a filename
      ' format: [drive[folder]] [frequency in kHz] [date as ddmmyy] [time as
hhmmss]

      function Filename(TagFrequency)
      dim temp

      temp = WAV_FOLDER
      temp = temp & TagFrequency * 1000000

      temp = temp & "_" & lenN(4, mytemp)

      'date
      temp =  temp  &  "_" & lenN(2, day(now)) & lenN(2, month(now)) & lenN(2,
year(now))

      'time
      temp =  temp & "_" & lenN(2, hour(now)) & lenN(2, minute(now))  &
lenN(2, second(now))
```

```
      temp = temp & ".wav"

      FileName = temp

      end function

'-------------------------------------------------------------------------------
-------------------
      'helper function to format date and time
      ' 1 -> 01, 2  - > 02 ... 9  -> 09
      '2001 -> 01

      function LenN(N, temp1)

      if len(cstr(temp1)) > N then
            temp1 = right(temp1, N)
      end if
      if len(cstr(temp1)) < N then
            temp1 = string(N - len(cstr(temp1)), "0") & temp1
      end if
      lenN = temp1

      end function
'-------------------------------------------------------------------------------
-------------------
      sub StartTheChannelScanner
      dim TimerSave1

      'macro starts VisualRadio's scanner
      ChannelScannerStart
      TimerSave = Timer
      do
            wait .1
            if Timer > TimerSave1 + 3 then
                  exit do
            end if
      loop until ScannerStatus = SERVER_SCANNER_STARTED
      end sub
```