

Steuerbefehle VisualRadio Basic

Version 3.1

19.06.2019

BASIS-BEFEHLE

Name	Beschreibung	Typ	Beispiel
Record (Flag As Boolean)	Startet/Stoppt die Audioaufzeichnung	Methode Doppelt die Taste 'RECORD'	ar.Record True ar.Record False
BandWidth()	Setzen/Ausgabe der Bandbreite	Eigenschaft	ar.bandwidth = "200 k"
Frequency()	Setzen/Ausgabe der Frequenz [MHz]	Eigenschaft	ar.frequency = 12.663
Attenuator()	Setzen/Ausgabe des Abschwächers	Eigenschaft	ar.attenuator="ON"
Squelch()	Setzen/Ausgabe des Squelch	Eigenschaft	ar.squelch = 250
Mode()	Setzen/Ausgabe der Betriebsart	Eigenschaft	ar.mode = "AM"
Antenna()	Setzen/Ausgabe der Antenne	Eigenschaft	ar.antenna = 1
DebugPrint()	Gibt Text in das Debugfenster aus	Methode	DebugPrint ar.mode
Sound()	Entspricht dem BEEP	Methode	sound
Direct (LAN Command)	Sendet einen Textdtring direkt an VR Flash	Method	Direct ("SYSTEM:RECORD:ON")

FLASH-BEFEHLE

Span()	Setzen/Ausgabe des Span [MHz]	Eigenschaft	ar.span = 5
SetFreerunOn()	Verwendet einen Windows Timer als Tigger für die Datenerfassung	Methode Doppelt die Taste 'FREE' (aktive)	ar.SetFreerunOn
SetFreerunOff()	Verwendet VR Basic als Trigger für die Datenerfassung, i.e. TakeSweep	Methode Doppelt die Taste 'FREE' (inaktive)	ar.SetFreerunOff
GetFreerunStatu	Gibt den Freerun	Funktion	ar.GetFreerunStat

s()	Status zurück, i.e. TRUE/FALSE		us
SetTriggerLevel(Level)	Setzt den Triggerlevel in dBm	Methode	ar.SetTriggerlevel -97
GetTriggerLevel()	Gibt den Triggerlevel in dBm zurück	Funktion	Debugprint ar.GetTriggerLevel
SetTriggerOn()	Zeigt die Triggerline, i.e TRIGGER = EIN	Methode Doppelt die Taste 'TRIGGER' (aktiv)	ar.SettriggerOn
SetTriggerOff()	Verbirgt die Triggerline , i.e TRIGGER = AUS	Methode Doppelt Taste 'TRIGGER' (inaktive)	ar.SetTriggerOff
SetScaleMinimum(Level)	Setzt den unteren Level des angezeigten Spektrums in dBm	Methode Doppelt die Taste 'Min. Level'	ar.SetScaleMinimum -100
GetScaleMinimum()	Gibt den unteren Level des angezeigten Spektrums in dBm aus	Funktion	Debugprint ar.GetScaleMinimum
SetScaleMaximum(Level)	Setzt den oberen Level des angezeigten Spektrums in dBm	Methode Doppelt Taste 'Max. Level'	ar.SetScaleMaximum -20
GetScaleMaximum()	Gibt den oberen Level des angezeigten Spektrums in dBm aus	Funktion	Debugprint ar.GetScaleMinimum
SetMarker1(Frequency)	Setzt Marker1 auf die gewünschte Frequenz in MHz	Methode	SetMarker1 85
GetMarker1()	Gibt die Frequenz des Marker1 in Mhz zurück	Funktion	Debugprint ar.GetMarker1
SetMarker2(Frequency)	Setzt Marker2 auf die gewünschte Frequenz in MHz	Methode	SetMarker2 93.5
GetMarker2()	Gibt die Frequenz des Marker2 in	Funktion	Debugprint ar.GetMarker2

	Mhz zurück		
SetMarkersOn()	Zeigt beide Marker	Methode	ar.SetMarkersOn
SetMarkersOff()	Verbirgt beide Marker	Methode	ar.SetMarkersOff
GetMarkerStatus()	Gibt den Status beider Marker zurück (ON / OFF)	Funktion	Debugprint ar.GetMarkerStatus
SetMarkerToLimits()	Setzt die Marker auf Anfang und Ende des Spektrums	Methode	ar.SetMarkerToLimits
SetSampleCount (Samples)	Definiert die Probengröße für Histogramm und Mittelwert	Methode Doppelt Combo 'Samples'	ar.SetSampleCount 12
SetHistogramOn()	Zeigt das Histogramm mit dem Spektrum	Methode Doppelt Taste 'HIST.' (aktive)	ar.SetHistogramOn
SetHistogramOff()	Verbirgt das Histogramm	Methode Doppelt Taste 'HIST.' (inaktive)	ar.SetHistogramOff
GetHistogramStatus()	Gibt den Status Histogramms aus (ON / OFF)	Funktion	Debugprint ar.GetHistogramStatus
GetSampleCount()	Gibt die Probengröße für Histogramm and Mittelwert zurück	Funktion	z = ar.GetSampleCount
GetTriggerStatus()	Gibt die Sichtbarkeit der Triggerlinie zurück (TRUE / FALSE)	Funktion	Debugprint ar.GetTriggerstatus
TakeSweep()	Löst einen Trigger aus	Funktion. Gibt geordnete Paare von Frequenz und Level über 'TRIGGERLEVEL' aus	ar.takesweep

