The Annex32 Book of Programs

Volume 1

ESP8266 Programs

FOREWORD by Starraker

This book contains a list of programs posted in Annex32 Forum since 2021. The programming is context and version sensitive and users will need to be able to adapt the program as necessary.

Some of the earlier programs as in this volume were for the ESP8266 which has now been superceded by the ESP32 range of MCUs. So members wishing to use the old code with the new MCU should note the slight and major differences in the functioning of the MCU and also the firmware version that they will be using.

Members who are using the very new ESP32-S3 and ESP32-P4 should be aware of the enhanced capabilities of their MCU with each having more memory and inbuilt support than the ESP8266 and modify their program where necessary to take advantage of this increased capability.

Screen based modules add a further dimension to programming and those modules with TOUCH an ability for instant interaction and control and the use of LVGL, Lovyan GFX and graphics. This can also be simulated with VGA and Mouse/Keyboard control (later firmware versions)

CONTENTS

Program 1 : SLIDER/ TEXTBOX INTERACTION by Electroguard

Preamble

Seeing a question about how to have a responsive slider / textbox interaction, I created a little snipped that show how this can be done relying more on the browser functionalities.
This snippet is a good example on how it is possible to integrate html / javascript with annex.
The instructions are directly inside the code below.

```
'simple snippet showing how manage a slider with
'its value shown in a textbox sending the updated value
'clicking on an "Update' button

'The variable is MyVar and is defined in the basic and also in the html code
'if 'value="20" is removed from the HTML, it will be automatically filled issuing the refresh command

'run the program and look for the variable value in the console.
'move the slider and see the effect in the textbox.
'clicking on the Update button will update the variable.
'Then, in the immediate window, type myVar = 25 and then type refresh
'both the slider and the textbox will be updated
cls
myVar = 20

a$ =||
a$ = a$ + |<input type="range" data-var="myVar" oninput="_$('myTextBox').value = this.value;" value="20" min="16" max="30" step="1">|
a$ = a$ + |<br>|
a$ = a$ + |<input type="text" data-var="myVar" id="myTextBox" value="20" style="width:140">|
a$ = a$ + |<br>|
a$ = a$ + |<button onclick="connection.send('cmd:immediatx' + 'myVar=' + _$('myTextBox').value);">Update</button>|
html a$
refresh

for z=1 to 100000
print myVar
next z
```

https://cicciocb.com/forum/viewtopic.php?t=13

Program 2 : ASSIGNING TITLES TO TEXTBOX AND BUTTON OBJECTS by Electroguard

Preamble

The css can become easily complicate and very hard to understand.

A "fast and dirty" solution is to use the attribute "title" that enable the tooltip text on the objects.

This snippet shows how it can be applied to textbox and button objects

```
'CODE: tooltip-easy.bas
cls
b$="ciao"

t$ = textbox$(b$)
html replace$(t$, "<input ", "<input title='this is a textbox'")

t$ = button$("Press Me", gohere)
html replace$(t$, "<button ", "<button title='this is a button'")

gohere:
```

https://cicciocb.com/forum/viewtopic.php?t=14

Program 3: DRIVEWAY MONITOR by ZIM

Preamble

This set of two programs did not come with any explanation except what is shown. However, the programs and diagrams (in the references) are self explanatory.

https://cicciocb.com/forum/viewtopic.php?p=40&hilit=annex+code#p40

CODE: Outside unit

```
cls
PIN.MODE 5, output 'alive indicator led
PIN(5) = 0
i = 0
trg = 0
dist = 0
vlt = ADC
batt = 0
msg$ = "0"
MAC1$ = "86:F3:EB:10:0B:2C" 'house
espnow.begin
espnow.add_peer MAC1$
'espnow.add_peer MAC2$
timer0 700, update
wait

update:
PIN(5) = 0
dist = DISTANCE(14,4)
pause 200
'if dist < 300 then trg = trg + 1 else trg = 0        ' uncheck if needed
'if trg >= 2 gosub alert                              ' uncheck if needed
if dist < 300 gosub alert
return

alert:
vlt = ADC
PIN(5) = 1
batt = CONVERT.MAP(vlt, 708, 1006, 12, 17)* 100 + 6        '+6 is for calibration
msg$ = str$(batt)
EspNow.Write(msg$)
pause 300
PIN(5) = 0
trg = 0
return
```

CODE: Inside unit

```
cls
pin.Mode 4, output                    ' blue led low battery alert
pin.Mode 5, output                    ' red led
pin.mode 12, input, pullup
INTERRUPT 12, display
pin.Mode 14, output                   ' bell or buzzer
PIN(4) = 1
PIN(5) = 0                            ' alive indicator on boot
PIN(14) = 1                           ' alive indicator on boot
pause 500
PIN(14) = 0
PIN(5) = 1
TM1637.SETUP 13, 16
batt$ = "0000"
trigs = 0
TM1637.PRINT str$(trigs), 4, 0
espnow.begin
onEspNowMsg message
autorefresh 1000
wlog "listening..."
wait

message:
trigs = trigs + 1
TM1637.PRINT str$(trigs), 4, 0
batt$ = espnow.read$
if val(batt$) > 120 PIN(5) = 0 else  PIN(4) = 0
PIN(14) = 1
pause 1000
PIN(14) = 0
pause 10000
PIN(4) = 1
PIN(5) = 1
return

display:
TM1637.PRINT batt$, 4, 255
if pin(12) = 0 return
TM1637.PRINT str$(trigs), 4, 0
refresh
return
```

https://cicciocb.com/forum/viewtopic.php?p=40&hilit=annex+code#p40

Program 4: ANNEX KEYLESS ENTRY by ZIM

Preamble

After locking my keys in the truck, I always wanted a keyless entry. I hate the idea of drilling holes for a keypad or having any kind of switch out in the elements.

Along came the XKC-Y25-PNP fluid level switch from china- Easy to position inside a vehicle behind glass. I have since relocated to the side rear window

You will have to dd your own homework on how to attach to your door locks. Connect across the outside key switch is the easiest, as it will also disable the alarm if it has one. I am lucky to have a vehicle schematic, so I'm going to the body Control Module. The esp8266 parasitic drain is too high, so deep sleep is applied when unlocked, or 30 seconds after the last code attempt

The 30 seconds also allows you to log in to AP mode as well, if you want to change the code; The added electronics allows the code entry pin to also issue the "wake up" call, so no need for 2 sensors;

Watch the bottom Video for operation.

```
GOSUB CREATE_PAGE
CREATE_PAGE:
CLS
TIMER0 1000, CHECK
TIMER1 30000, SLEEEP        'puts it to sleep after 30 sec of inactivity
CNT = 0
CNT2 = 0
UNL$ = ""
D1=5:D2=4:D6=12:D7=13     'D1 mini format to io
PIN.MODE D6, OUTPUT       ' keeps reset high, preventing reset while awake
PIN(D6) = 1
PIN.MODE D2, OUTPUT       'D2 is an output to relay
PIN.MODE D1, INPUT        'D1 is input pin for sensor. It also wakes from sleep
PIN(D2) = 0
AUTOREFRESH 1000
WAIT

CHECK:
IF PIN(D1) = 1 THEN TIMER1 30000, SLEEEP 'sleep 30 sec. after last key press
IF UNL$ = "AB" THEN LET CNT2 = (CNT2 + 1) ELSE CNT2 = 0
IF CNT2 = 6 THEN LET UNL$ = ""
IF PIN(D1) = 0 THEN LET CNT = 0
IF PIN(D1) = 0 RETURN
PIN.MODE D7, OUTPUT       ' led flash
PIN(D7) = 1
PAUSE 100                 ' for indicator visibility
PIN(D7) = 0
PAUSE 800                 'critical for finger off response time
CNT = CNT + 1
```

```
IF PIN(D1) = 0 AND (CNT = 2) THEN LET UNL$ = "AB"                        ' unlock
IF PIN(D1) = 0 AND (CNT = 1) AND (UNL$ = "AB") THEN LET UNL$ = "ABCD"      ' code
IF PIN(D1) = 0 AND (CNT = 3) AND (UNL$ = "ABCD") GOSUB UNLOCK ELSE RETURN  ' 213
IF PIN(D1) = 0 RETURN
RETURN

UNLOCK:
PIN.TONE D7, 8, 1500     ' flashes led to confirm unlock
PIN(D2) = 1
PAUSE 900                ' pulse relay
PIN(D2) = 0
UNL$ = ""
PAUSE 500
SLEEP 0
RETURN

SLEEEP:
PIN(D2) = 0
PAUSE 10
SLEEP 0
```

https://cicciocb.com/forum/download/file.php?id=14

Program 5: KINETIC CLOCK by PeterN

Preamble

A hardware and software project for a friend who loves clocks in all variants.
It has two concentric NeoPixel rings and some moving parts to form a kinetic object that can even display time in an old-fashioned "analog" way 😀. Hours are displayed on the inner ring, minutes and seconds on the outer ring.

The electronics are located in the base of the object.

```
'#########################################################################################
##############################
VERSION$ = "1.1"
' Uhr mit zwei  NeoPixel-Ringen an D4(!!!)
' und PWM-Signal  an D7 für  einen gesteuerten Motor

' Jan 2020: Erweiterung von 84 auf 85 LEDs zur
'          Mittenbeleuchtung der zusätzlichen Mechanik der Uhr

cls

'für drehzahlgesteuerten Motor mit PWM-Signal an D7
D7 = 13 ' D7 = GPIO13
VAL_PWM = 0
VAL_PWM_alt = 1
OPTION.PWMFREQ 10000   '100Hz ist in 1.37beta die kleinste mögliche Frequenz für PWM. De-
fault ist 1000Hz
pwm(D7)=VAL_PWM

x = 0: y = 0: z = 0
mm = 0:  hh = 0: ss = 0
R = 5: G = 25 : B = 0
R_alt = R + 1
G_alt = G
B_alt = B
Mark_R = 0 : Mark_G = 1 : Mark_B = 1
Sec_R = 1 : Sec_G = 1 : Sec_B = 1
t$ = "TIME"

'gosub SHOW_IP
Gosub READ_RGB_DATA
gosub STARTUP
OnHtmlReload HTML_PAGE
gosub HTML_PAGE
timer0 1000, SHOW_TIME

wait

' ###################################
' ###################################
```

```
' ###################################

Mode_CLOCK:
timer0 0
mm = 0
hh = 0
gosub STARTUP
timer0 1000, SHOW_TIME
wait
'

' ###################################

SHOW_TIME:
mm_alt = mm
hh_alt = hh
ss_alt = ss
t$ = time$
'dw = mid(bla,1,3) 'dow
'mh = mid(bla,5,3) 'month
'dt = mid(bla,9,2) 'date
hh = val(mid$(t$,1,2)) 'hour
mm = val(mid$(t$,4,2)) 'min
ss = val(mid$(t$,7,2)) 'sec

'Motor in den letzten 5 Sekunden einer Minute  kurz anwerfen VAL_PWM = 0
if ss  >54 then
  VAL_PWM = 800
else
  ' Motor zu Sicherheit ausschalten;
  ' Dazwischen geht auch manuelle Bedienung mit dem Slider
  if ss = 1 then VAL_PWM= 0
end if

if hh > 11 then hh = hh -12
hh = (hh * 2) + 60

if mm > 31 then hh = hh + 1

'neo.strip 0,84,1,1,1,1
neo.strip 0,84,0,0,0,1

'
"neo.pixel ss_alt,0,0,0,1
"neo.pixel mm_alt,0,0,0,1
"neo.pixel hh_alt,0,0,0,1

'Stundenmarkierungen  aussen
for i = 0 to 58 step 5
  neo.pixel i,Mark_R,Mark_G,Mark_B,1
next i
'Stundenmarkierungen  innen
```

```
'for i = 60 to 84 step 6
'  neo.pixel i,Mark_R,Mark_G,Mark_B,1
'next i

'die Zeiger setzten
neo.pixel ss,Sec_R,Sec_G,Sec_B,1
neo.pixel mm,R,G,B,1
neo.pixel hh,R,G,B,1

'Mittelbeleuchtung
neo.pixel 85,R*3 mod 100,G*3 mod 100,B*3 mod 100,1

'erst jetzt den Neopixel-puffer senden
neo.pixel 84,R,G,B,0
'
'PWM an D7 aendern, wenn neuer Wert
if VAL_PWM <> VAL_PWM_alt then
  pwm(D7) = VAL_PWM mod 1024
  VAL_PWM_alt = VAL_PWM
end if

refresh
return

' ######################################
LIGHTSHOW1:
timer0 0
t$="LIGHTSHOW1"
neo.strip 0,84,0,0,0
'do
for xx = 1 to 5
  for p = 0 to 59
    neo.pixel p,55,0,0
    neo.pixel 59 - p,55,0,0
    neo.pixel p,0,0,0
    neo.pixel 59-p,0,0,0
  next p
  for p = 0 to 59
    neo.pixel p,0,55,0
    neo.pixel 59-p,0,55,0
    neo.pixel p,0,0,0
    neo.pixel 59-p,0,0,0
  next p

  for p = 0 to 59
    neo.pixel p,0,0,55
    neo.pixel 59-p,0,0,55
    neo.pixel p,0,0,0
    neo.pixel 59-p,0,0,0
  next p
next xx
```

```
timer0 1000, SHOW_TIME
return

' ###################################
LIGHTSHOW2:
'pacman
timer0 0
t$="PACMAN"
neo.strip 60,84,0,10,10
schritt_x = 1
schritt_y = 1.8
schritt_z = 1.2

do
  x_alt = x
  x = x +  schritt_x
  if (x > 59) or (x < 1) then
    schritt_x = schritt_x * -1
  end if

  y_alt = y
  y = y +  schritt_y
  if (y > 59) or (y < 1) then
    schritt_y = schritt_y * -1
  end if

  z_alt = z
  z = z + schritt_z
  if (z > 59) or (z < 1) then
    schritt_z = schritt_z * -1
  end if

  neo.pixel x,84,0,0,1
  neo.pixel x_alt,0,0,0,1

  neo.pixel y,0,60,0,1
  'neo.pixel y_alt,0,0,0,1

  neo.pixel z,0,0,60,1
  neo.pixel z_alt,0,0,0,0
  pause 40

loop until 0

wait

' ###################################
LIGHTSHOW3:
timer0 0
t$="LIGHTSHOW 3"
neo.strip 0,93,0,0,0
neo.strip 0,84,0,0,0
```

```
for i = 0 to 59

  if i < 29 then
    neo.pixel 29-i,5,5,0,0
    neo.pixel 29+i,5,5,0,0
    neo.pixel (59 + 12 - i / 2),5,5,0,0
    neo.pixel (59 + 12 + i / 2),5,5,0,0

  end if
  neo.pixel i,1,1,5,0
  neo.pixel (60 -i),1,5,1,0
  neo.pixel (60 + 12 - i / 2),1,1,5,0
  neo.pixel (60 + 12 + i / 2),1,5,1,0
  pause 20
next i

neo.strip 0,84,0,0,0
for i = 1 to 30000
  ii = i MOD 60
  iii= (ii/5)*2
  'if ii = 1 then
  gg= i mod 50
  rr= i+15 mod 50
  bb= i+30 mod 50
  'end if
  neo.strip 0,59+24+1,0,0,0,1

  neo.pixel ii,rr,gg,bb,1
  neo.pixel (ii+20)mod 60,gg,bb,rr,1
  neo.pixel (ii+40)mod 60,bb,rr,gg,1

  ' iii=iii/1.5
  ' neo.pixel 83-iii,rr,gg,bb,1
  ' neo.pixel 83 -abs((iii-8)mod 23),gg,bb,rr,1
  ' neo.pixel 83 -abs((iii-16)mod 23),bb,rr,gg,1


  neo.pixel 60+iii,rr,gg,bb,1
  neo.pixel 60+abs((iii-8)mod 23),gg,bb,rr,1
  neo.pixel 60+abs((iii-16)mod 23),bb,rr,gg,1

  neo.pixel 84,bb,rr,gg,0

  pause 25
next i

return

' ######################################
STARTUP:
neo.setup  86
neo.strip 0,86, 0,0,0
```

```
for i = 0 to 29
  neo.pixel (30-i),30-i,2+i,0,1
  neo.pixel (30+i),30-i,2+i,0,1
  neo.pixel (72+((i/5)*2)),1,i,0,1
  neo.pixel (72-((i/5)*2)),1,i,0,1
  neo.pixel (84),1,i,0,0
  pause 120-i*3
next i

neo.strip 0,86,11,211,11
pause 10
neo.strip 0,86,0,0,0
return

' ####################################


' ####################################
HTML_PAGE:
cls
autorefresh 750

a$ = ""
a$ = a$ & "
S T A R G A T E -V“& VERSION$ & „
"
a$ = a$  & textbox$(t$)
a$ = a$ & "
"
a$ = a$ &  "Colour of the hour and minute hands:"
a$ = a$ & "
"
a$ = a$ &  SLIDER$(R,0,100)
a$ = a$ &  "R:"
a$ = a$ &  textbox$(R)
a$ = a$ &  "
"

a$ = a$ &  slider$(G,0,100)
a$ = a$ &  "G:"
a$ = a$ &  textbox$(G)
a$ = a$ &  "
"

a$ = a$ &  slider$(B,0,100)
a$ = a$ &  "B:"
a$ = a$ &  textbox$(B)

a$ = a$ &  "
"
a$ = a$ &  button$("Save RGB", SAVE_RGB_DATA)
```

```
a$ = a$ & "

"
a$ = a$ &  "Colour of the second hand:"
a$ = a$ & "
"
a$ = a$ &  SLIDER$(Sec_R,0,100)
a$ = a$ &  "R:"
a$ = a$ &  textbox$(Sec_R)
a$ = a$ &  "
"

a$ = a$ &  slider$(Sec_G,0,100)
a$ = a$ &  "G:"
a$ = a$ &  textbox$(Sec_G)
a$ = a$ &  "
"

a$ = a$ &  slider$(Sec_B,0,100)
a$ = a$ &  "B:"
a$ = a$ &  textbox$(Sec_B)

a$ = a$ &  "
"
a$ = a$ &  button$("Save RGB", SAVE_RGB_DATA)

a$ = a$ & "

"
a$ = a$ &  "Colour of the hour and 5-minute markers:"
a$ = a$ & "
"
a$ = a$ &  SLIDER$(Mark_R,0,10)
a$ = a$ &  "R:"
a$ = a$ &  textbox$(Mark_R)
a$ = a$ &  "
"

a$ = a$ &  slider$(Mark_G,0,10)
a$ = a$ &  "G:"
a$ = a$ &  textbox$(Mark_G)
a$ = a$ &  "
"

a$ = a$ &  slider$(Mark_B,0,10)
a$ = a$ &  "B:"
a$ = a$ &  textbox$(Mark_B)
a$ = a$ &  "
"
a$ = a$ &  button$("Save RGB", SAVE_RGB_DATA)
a$ = a$ &  "
```

```
"

a$ = a$ &  "PWM-Signal for motor at D7:"
a$ = a$ &  "
"
a$ = a$ &  slider$(VAL_PWM,0,1023)
a$ = a$ &  "
VAL_PWM:"
a$ = a$ &  textbox$(VAL_PWM)
a$ = a$ &  "

MODUS: "
a$ = a$ &  button$("LIGHTSHOW", LIGHTSHOW1)
a$ = a$ &  button$("PacMan LIGHTSHOW2", LIGHTSHOW2)
a$ = a$ &  button$("LIGHTSHOW3", LIGHTSHOW3)
a$ = a$ &  button$("CLOCK", Mode_CLOCK)

html a$
a$=""
return
' ####################################

Exit:
end

' ####################################
SHOW_IP:
IPADR$= word$(word$(IP$,1),4,".")
return


' ####################################
TestExit:
timer0 0
neo.strip 0,84,0,8,0
pause 500
neo.strip 0,84,0,5,0
pause 500
neo.strip 0,84,0,2,0
pause 500
neo.strip 0,84,0,1,0
pause 500
for i = 0 to 29
  neo.pixel i,0,0,0,1
  neo.pixel 60-i,0,0,0,1
  neo.pixel i/2+60,0,0,0,1
  neo.pixel 84-i/2,0,0,0,0
  pause 50
next i
neo.pixel 30,0,10,0,1
neo.pixel 72,0,10,0,0
```

```
'######
end
'######

' ################################################################
SAVE_RGB_DATA:
' schreibt die Einstellungen in einzelne Dateien im Flash
t$ = "DATA ..."
xxx$ = "RGB-DATA"
file.save "/clock/settings",xxx$
file.save "/clock/R", str$(R)
file.save "/clock/G", str$(G)
file.save "/clock/B", str$(B)
file.save "/clock/Sec_R", str$(Sec_R)
file.save "/clock/Sec_G", str$(Sec_G)
file.save "/clock/Sec_B", str$(Sec_B)
file.save "/clock/Mark_R", str$(Mark_R)
file.save "/clock/Mark_G", str$(Mark_G)
file.save "/clock/Mark_B", str$(Mark_B)
t$ = "DATA SAVED"
return

' ################################################################
READ_RGB_DATA:
if file.exists("/clock/settings") then SETTINGS$ = file.read$("/clock/settings")
if SETTINGS$ <> "" then
  R       = val(file.read$("/clock/R"))
  G       = val(file.read$("/clock/G"))
  B       = val(file.read$("/clock/B"))
  Sec_R   = val(file.read$("/clock/Sec_R"))
  Sec_G   = val(file.read$("/clock/Sec_G"))
  Sec_B   = val(file.read$("/clock/Sec_B"))
  Mark_R  = val(file.read$("/clock/Mark_R"))
  Mark_G  = val(file.read$("/clock/Mark_G"))
  Mark_B  = val(file.read$("/clock/Mark_B"))
endif
return

' ################################################################
```

https://cicciocb.com/forum/viewtopic.php?t=25

Program 6 : PHONE TEXT / EMAIL ALERT by ZIM

Preamble

Mice in your parked RV can be a problem, especially when they go undetected when you are not around. No, this is not a joke. Sending text messages and emails was not easy until Annex was born.

With just a few lines of code and a gmail account, ESP8266 can send you text and email warnings so you can remove the decaying mouse before smell....and find the hole! Can also be retrofitted to any trap - No 3rd party required!!!! (you must turn on less secure apps in your gmail account, so create a new one just for this)

If you are sending a text, you will need to know the receivers carrier so you can use the appropriate domain name. find it here:

https://computer.howstuffworks.com/e-mail-messaging/how-to-send-text-messages-computer.htm

```
'Annex Code

EMAIL.SETUP "smtp.gmail.com", 465, "whoever@gmail.com", "password"
PRINT EMAIL ("whoever@gmail.com", "4037672441@msg.telus.com", "Mice", "Mouse trap has been tripped")                ' this sends a text
PRINT EMAIL ("whoever@gmail.com", "John@Yahoo.com", "Mice", "Mouse trap has been tripped")                ' this sends an email
PAUSE 10000   'stays awake for 10 seconds, so you can catch it before sleep to edit it, if needed...
SLEEP 10                ' goes into deep sleep until reset, to save battery
```

https://cicciocb.com/forum/viewtopic.php?t=28

Program 7 : PET HOUSE HEATER by ZIM

Preamble

I took a 40 watt incandescent light bulb, put it in a appropriate socket and mounted it inside a piece of heavy walled pipe. I capped both end of the pipe and poked several small holes to let the heat escape. I made sure there was more than one inch clearance between the bulb and pipe inner wall. Using the appropriate wiring, I connected up a robotdyne module to the esp. If the temperature gets too hot or too cold, the esp will send me a text warning on the phone.

The beauty of the dimmer function is that the bulb always has a "soft start" and will have a very long life. The code also provides more or less heat with more or less temperature spread. Its not just on/off. Its progressive.

See the Project Text/email alert for text setups

```
'Annex Cat House Thermostat/heater v.4   40W lightbulb
dimmer.stop
pause 10
onHtmlReload create_page
onhtmlchange update
'file.save "/tmpset.txt", str$(5)   'XXXXXXXXXXXXXXXXXXX   uncheck this for the first run so it creates a txt file
PWR = 0                                  ' variable for dimmer intensity
gosub create_page
timer0 2000, update
wait

create_page:
cls
autorefresh 2000
D4 = 2    ' sensor input
setp = val(file.read$("/tmpset.txt"))
curr = val(TEMPR$(D4,1))*(1.05)        ' 1.05 is calibration if needed
curr1 = 99                              'displayed sensor reading if sensor issue
PWR = 0
DIMMER.SETUP 4, 5 , 0, 1    'ZC signal io4 ,  Output pulse io5 , short io13 to io4
DIMMER.LIMITS 1300, 8300   'actual min/max delay in triac firing
pause 10
stat$ = "Heat is off"
ltmp = 0                                ' low temp warning variable
htmp = 0                                ' high temp warning variable
phoneN = 12345678910 'phone number for alert text XXXXXXXXXXXXXXXXXXX
qsbutt$ = "background-color:white;text-align:center;dis-
play:block;width:130px;height:50px;font-size:15;font-weight:bold;box-shadow: 3px 3px black;border-radius:5px;line-height: 1"
txtbstat$ = "background-color:white;text-align:center;dis-
play:block;width:160px;height:26px;font-size:15;font-weight:bold"
```

```
txtbsetp$ = "background-color:white;text-align:center;dis-
play:block;width:54px;height:42px;font-size:30;font-weight:bold"
txtbcurr1$ = "background-color:white;text-align:center;dis-
play:block;width:54px;height:42px;font-size:30;font-weight:bold"

A$ = ""
A$ = A$ + |<!DOCTYPE html><html><body><style>|
A$ = A$ + |body {background-color: black;}|
A$ = A$ + |p {font-size:24px;}|
A$ = A$ + |</style>|
A$ = A$ + |<table align='center' width='300' height='100' bgcolor='lightgreen' border='1'
cellpadding='8'>|
A$ = A$ + |<th>|
A$ = A$ + |<p>Cat House&#10;Temperature</p></th></center>|
A$ = A$ + |<th><center>|
A$ = A$ + textbox$(curr1, "txt1")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("txt1", txtbcurr1$)
A$ = A$ + |</table>|
A$ = A$ + |<table align='center' width='300' height='100' bgcolor='red' border='1' cellpad-
ding='8'>"|
A$ = A$ + |<center><th>|
A$ = A$ + |<p>Thermostat<br>is Set At</p>|
A$ = A$ + |</th></center>|
A$ = A$ + |<th><center>|
A$ = A$ + textbox$(setp, "txt2")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("txt2", txtbsetp$)
A$ = A$ + |</table>|
A$ = A$ + |<center>|
A$ = A$ + |<p><font color = 'white', font size = '4'>Type Desired Temp. in Above Win-
dow<br>or select a Quick Button below</font></p>|
A$ = A$ + |</center>|
A$ = A$ + |<table align='center' width='300' bgcolor='red' border='1' cellpadding='5'>|
A$ = A$ + |<center><th>|
A$ = A$ + button$("<b><big>Quick set to&#10;5 deg.</big><b/>", Qset5, "but1")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("but1", qsbutt$)
A$ = A$ + |<center><th>|
A$ = A$ + button$("<b><big>Quick set to&#10;10 deg.</big><b/>", Qset10, "but2")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("but2", qsbutt$)
A$ = A$ + |</table>|
A$ = A$ + |<table align='center' width='300' bgcolor='lightblue' border='1' cellpad-
ding='8'>|
A$ = A$ + |<td><center>|
A$ = A$ + |<b><big>Heater Status</big></b>|
A$ = A$ + |</center><center>|
A$ = A$ + textbox$(stat$, "txt3")
A$ = A$ + cssid$("txt3", txtbstat$)
A$ = A$ + |</center></td></table>|
```

```
A$ = A$ + |<br><center>|
A$ = A$ + button$("<b><big>Refresh Page if Needed</big><b/>", freshweb)
A$ = A$ + button$("<b><big>Turn OFF</big><b/>", offf)
A$ = A$ + |</center></table></body></html>|
HTML A$
A$ = ""
return

hheat:
'wlog "high heat"
stat$ = "Heat is On High"
if curr < -10 gosub alert_low                  'if temp low, go to alert
return

off2:
'wlog "Off"
stat$ = "Heat is now OFF"
DIMMER.BRIGHTNESS PWR

if curr > 25 gosub alert_high                  ' if temp high, go to high_alert
return

Qset10:
setp = 10
return

Qset5:
setp = 5
return

update:
curr = val(TEMPR$(D4,1))*(1.05)                ' 1.05 is calibration if needed
'wlog "actual temp -> " ;curr
'wlog "set temp " ;setp
if setp <= -1 or setp > 30 then let setp = val(file.read$("/tmpset.txt"))              'if temp
has been set out of normal range, it defaults to last setting
if (setp <> val(file.read$("/tmpset.txt"))) and (setp < 30) and (setp > -1) then file.save
"/tmpset.txt", str$(setp)              'writes last acceptable setting to memory
if curr < 40 and curr > -40 then curr1 = int(curr)  'int gives no decimal xxxxxxxxxxxxxxx
use this line to calibrate eq. int(curr) - 5
setp = val(file.read$("/tmpset.txt"))

PWR = (setp - curr) * 100 / 3  ' linear increase in power with linear decrease in temp
drop from set point (3 deg span)
If PWR < 0 then let PWR = 0
If PWR > 100 then let PWR = 100

DIMMER.BRIGHTNESS int(PWR)
pause 100
if PWR = 100 then gosub hheat
if PWR < 100 and PWR > 0 then stat$ = "Heat is On"
```

```
if PWR = 0 then gosub off2
return

freshweb:
dimmer.stop
gosub create_page
return

alert_low:
ltmp = ltmp + 1
if ltmp < 450 return
ltmp = 0
EMAIL.SETUP "smtp.gmail.com", 465, "joeblow@gmail.com", "mypassss" 'replace with
your login / password
print EMAIL ("joeblow@gmail.com", "phoneN@msg.telus.com", "Title : Thermostat", "Cat
house is below -5")
pause 100
'wlog "low message sent"
return

alert_high:
htmp = htmp + 1
if htmp < 450 return
htmp = 0
EMAIL.SETUP "smtp.gmail.com", 465, "joeblow@gmail.com", "mypassss" 'replace with
your login / password
print EMAIL ("joeblow@gmail.com", "phoneN@msg.telus.com", "Title : Thermostat", "Cat
house is above 25")
pause 100
'wlog "high message sent"
return

offff:
dimmer.stop
timer1 0
cls
A$ = ""
A$ = A$ + |<!DOCTYPE html><html><body><style>|
A$ = A$ + |body {background-color: black;}|
A$ = A$ + |p {font-size:36px;}|
A$ = A$ + |</style>|
A$ = A$ + |<br><center>|
A$ = A$ + button$("<b><big>Boot Up Thermostat</big><b/>", stopme)
A$ = A$ + |</center></table></body></html>|
HTML A$

onHtmlReload offff
refresh
return

stopme:
```

dimmer.stop
cls
reboot

https://cicciocb.com/forum/viewtopic.php?t=29

Program 8: ESP8266 THERMOSTAT by ZIM

Preamble

There was no text preamble to this project. Before using check with initial posting at:

https://cicciocb.com/forum/viewtopic.php?t=30

```
'Zim's Annex Thermostat Celcius v.7

onHtmlReload create_page
onhtmlchange update
gosub create_page
timer0 2000, update
wait

create_page:
cls
autorefresh 1000
D1 = 5    ' output to relay
D2 = 4    ' sensor
PIN.MODE D1, OUTPUT 'relay for furnace D2
PIN(D1)= 1
'file.save "/tmpset.txt", str$(5)   'XXXXXXXXXXXXXXXXXXXX   uncheck this for the first run so it creates a txt file  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
setp = val(file.read$("/tmpset.txt"))
DHT.SETUP 4, 11
curr = DHT.TEMP
curr1 = 99              'displayed sensor reading if sensor issue
stat$ = "Furnace is now OFF"
otw = 0  'On Time Warning variable
ltmp = 0   'low temp warning variable
htmp = 0   'high temp warning variable
qsbutt$ = "background-color:white;text-align:center;display:block;width:130px;height:50px;font-size:15;font-weight:bold;box-shadow: 3px 3px black;border-radius:5px;line-height: 1"
txtbstat$ = "background-color:white;text-align:center;display:block;width:160px;height:26px;font-size:15;font-weight:bold"
txtbsetp$ = "background-color:white;text-align:center;display:block;width:54px;height:42px;font-size:30;font-weight:bold"
txtbcurr1$ = "background-color:white;text-align:center;display:block;width:54px;height:42px;font-size:30;font-weight:bold"

A$ = ""
A$ = A$ + |<!DOCTYPE html><html><body><style>|
A$ = A$ + |body {background-color: black;}|
A$ = A$ + |p {font-size:24px;}|
A$ = A$ + |</style>|
A$ = A$ + |<table align='center' width='300' height='100' bgcolor='lightgreen' border='1' cellpadding='8'>|
A$ = A$ + |<th>|
```

```
A$ = A$ + |<p>Room<br>Temperature</p></th></center>|
A$ = A$ + |<th><center>|
A$ = A$ + textbox$(curr1, "txt1")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("txt1", txtbcurr1$)
A$ = A$ + |</table>|
A$ = A$ + |<table align='center' width='300' height='100' bgcolor='red' border='1' cellpad-
ding='8'>"|
A$ = A$ + |<center><th>|
A$ = A$ + |<p>Thermostat<br>is Set At</p>|
A$ = A$ + |</th></center>|
A$ = A$ + |<th><center>|
A$ = A$ + textbox$(setp, "txt2")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("txt2", txtbsetp$)
A$ = A$ + |</table>|
A$ = A$ + |<center>|
A$ = A$ + |<p><font color = 'white', font size = '4'>Type Desired Temp. in Above Window<br>or
select a Quick Button below</font></p>|
A$ = A$ + |</center>|
A$ = A$ + |<table align='center' width='300' bgcolor='red' border='1' cellpadding='5'>|
A$ = A$ + |<center><th>|
A$ = A$ + button$("<b><big>Quick set to&#10;5 deg.</big><b/>", Qset5, "but1")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("but1", qsbutt$)
A$ = A$ + |<center><th>|
A$ = A$ + button$("<b><big>Quick set to&#10;15 deg.</big><b/>", Qset15, "but2")
A$ = A$ + |</th></center>|
A$ = A$ + cssid$("but2", qsbutt$)
A$ = A$ + |</table>|
A$ = A$ + |<table align='center' width='300' bgcolor='lightblue' border='1' cellpadding='8'>|
A$ = A$ + |<td><center>|
A$ = A$ + |<b><big>Furnace Status</big></b>|
A$ = A$ + |</center><center>|
A$ = A$ + textbox$(stat$, "txt3")
A$ = A$ + cssid$("txt3", txtbstat$)
A$ = A$ + |</center></td></table>|
A$ = A$ + |<br><center>|
A$ = A$ + button$("<b><big>Refresh Page if Needed</big><b/>", freshweb)
A$ = A$ + button$("<b><big>Turn OFF</big><b/>", offf)
A$ = A$ + |</center></table></body></html>|
HTML A$
return

on2:
otw = otw + 1
if otw > 900 gosub alert_time  'if furnace on over 30 min. sends alert
'if otw > 30 gosub alert_time  'test
PIN(D1)= 0
stat$ = "Furnace is now ON"
return
```

```
off2:
otw = 0  'resets alarm timer
PIN(D1)= 1
stat$ = "Furnace is now OFF"
return

Qset15:
setp = 15
return

Qset5:
setp = 5
return

update:
curr = DHT.TEMP
if setp <= -1 or setp > 30 then let setp = val(file.read$("/tmpset.txt"))             'if temp has been set
out of normal range, it defaults to last setting
if (setp <> val(file.read$("/tmpset.txt"))) and (setp < 30) and (setp > -1) then file.save "/tmpset.txt",
str$(setp)            'writes last acceptable setting to memory
if curr < 40 and curr > -40 then curr1 = int(curr)  'int gives no decimal xxxxxxxxxxxxxxx use this
line to calibrate eq. int(curr) - 5
if curr1 < (setp - 1) then gosub on2 'swing adjustment
if curr1 > (setp + 1) then gosub off2 'swing adjustment
if curr > 32 gosub alert_high ' if temp high go to alert
if curr < 0 gosub alert_low  'if temp low go to alert
'wlog " On time alert  "; otw     'shows the count up to alert after 30 minutes of on time
'wlog " Low temp alert  "; ltmp   'shows the count up to alert after 15 minutes of low temp
'wlog " High temp alert  "; htmp  'shows the count up to alert after 15 minutes of high temp
return

freshweb:
gosub create_page
return

alert_time:
otw = 0  'resets alarm timer
EMAIL.SETUP "smtp.gmail.com", 465, "Zim@gmail.com", "passw" 'replace with your login /
password
print EMAIL ("Zim@gmail.com", "780xxxxxxx@msg.telus.com", "Title : Thermostat", "Furnace
running for over 30 minutes")
pause 100
'wlog "on-time alert sent"
return

alert_low:
ltmp = ltmp + 1
if ltmp < 450 return
ltmp = 0
EMAIL.SETUP "smtp.gmail.com", 465, "Zim@gmail.com", "passw" 'replace with your login /
password
print EMAIL ("Zim@gmail.com", "780xxxxxxx@msg.telus.com", "Title : Thermostat", "Garage
```

```
temperature is low")
pause 100
'wlog "low temp message sent"
return

alert_high:
htmp = htmp + 1
if htmp < 450 return
htmp = 0
EMAIL.SETUP "smtp.gmail.com", 465, "Zim@gmail.com", "passw" 'replace with your login / password
print EMAIL ("Zim@gmail.com", "780xxxxxxx@msg.telus.com", "Title : Thermostat", "Garage temperature is high")
pause 100
'wlog "high temp message sent"
return

offf:
PIN(D1)= 1
timer0 0
cls
A$ = ""
A$ = A$ + |<!DOCTYPE html><html><body><style>|
A$ = A$ + |body {background-color: black;}|
A$ = A$ + |p {font-size:36px;}|
A$ = A$ + |</style>|
A$ = A$ + |<br><center>|
A$ = A$ + button$("<b><big>Turn on Thermostat</big><b/>", gogo)
A$ = A$ + |</center></table></body></html>|
HTML A$
PIN(D1)= 1
'wlog " pin D1 is  "; PIN(D1)
onHtmlReload offf
return

gogo:
cls
reboot
```

https://cicciocb.com/forum/viewtopic.php?t=30

Program 9 : PETROL INJECTOR FLOW TEST / CLEANER by ZIM

Preamble

As per ZIMs postings there was a graphical introduction which should be viewed before constructing at:

https://cicciocb.com/forum/viewtopic.php?t=31

```
' Zim's Gasoline Injector Flow Tester/Cleaner

onhtmlchange refresh_vars
onHtmlReload menu
goto menu

menu:
cls
PIN.MODE 4, output
PWM(4) = 0
durt$ = "?"
hrts$ = "?"
mms$ = "?"
hrts2 = 0
dur = 0
bar = 0
pw = 0
i = 0
Ct1 = 10

startb$ = "background-color:lightblue;text-align:center;width:55px;height:40px;font-size:14;font-weight:bold"
butt$ = "background-color:lightblue;text-align:center;width:120px;height:65px;font-size:14;font-weight:bold"
dropd$ = "background-color:white;text-align:center;height:40px;font-size:20;font-weight:bold"
tbox$ = "background-color:lightblue;text-align:center;width:60px;height:40px;font-weight:bold"
abortb$ = "text-align:center;width:310px;height:50px;font-size:20;font-weight:bold"
barb$ = "width:310px;height:20px"

A$ = ""
A$ = A$ + |<!DOCTYPE html><html><body><style>|
A$ = A$ + |body {background-color: black;}|
A$ = A$ + |</style><center>|
A$ = A$ + |<p><font color = 'white', font size = '6'>Gasoline Injector Tester</font></p>|
A$ = A$ + |</center>|
A$ = A$ + |<table align='center' width='310' height='100' bgcolor='lightgreen' border='1' cellpadding='8'>|
A$ = A$ + |<center><th>|
A$ = A$ + BUTTON$("Automatic<br>Clean Cycle<br>250 seconds", clean, "but11")
A$ = A$ + cssid$("but11" , butt$)
A$ = A$ + |</center></th><center><th>|
```

```
A$ = A$ + BUTTON$("Automatic<br>Flow Test<br>10 seconds", flow, "but12")
A$ = A$ + cssid$("but12" , butt$)
A$ = A$ + |</center></th></center></table><center>|
A$ = A$ + meter$(bar, 0, 100, "bar1")
A$ = A$ + cssid$("bar1" , barb$)
A$ = A$ + |</center><center>|
A$ = A$ + |<table align='center' width='310' height='100' bgcolor='lightgreen' border='1' cellpad-
ding='4'>|
A$ = A$ + |<tr><th>Manual<br>Mode</th><th>Fre-
quency<br>Hertz</th><th>Pulse<br>mSec</th><th>Timer<br>Seconds</th></tr><td>|
A$ = A$ + BUTTON$("Start", manual, "but13")
A$ = A$ + cssid$("but13" , startb$)
A$ = A$ + |</td><td>|
A$ = A$ + listbox$(hrts$,"10, 20, 30", "lbox5")
A$ = A$ + cssid$("lbox5" , dropd$)
A$ = A$ + |</td><td>|
A$ = A$ + listbox$(mms$,"5, 10, 20, 30", "lbox6")
A$ = A$ + cssid$("lbox6" , dropd$)
A$ = A$ + |</td><td>|
A$ = A$ + listbox$(durt$,"5, 10, 60, On", "lbox7")
A$ = A$ + cssid$("lbox7" , dropd$)
A$ = A$ + |</td></table></center><center>|
A$ = A$ + BUTTON$("Stop Timer", abort, "but14")
A$ = A$ + cssid$("but14" , "background-color:red;" & abortb$)
A$ = A$ + |<br>|
A$ = A$ + BUTTON$("Exit", EXITs, "but15")
A$ = A$ + cssid$("but15" , "background-color:lightblue;" & abortb$)
A$ = A$ + |</center>|
A$ = replace$(A$, "Choose here", "?")  ' change all the "Choose here" instances with ""?"
HTML A$
refresh
wait

Flow:  'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
pause 1000
if bar > 10 then goto abort
i = 0
j = 1
Ct1 = 10
refresh

do
bar = Ct1
Ct1 = j*10
refresh
bar = Ct1
PWM(4) = 256  '25% duty cycle
OPTION.PWMFREQ 10
pause 1000
i = i + 1    ' for the loop
j = j + 1    ' for the bar
LOOP UNTIL i > 9
```

```
PWM(4) = 0
bar = 0
refresh
return

manual:  'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
i=0 ' for the loop
j=0 ' for the bar
dur = val(durt$)
if durt$ = "On" then dur = 10000
if dur * val(mms$) * val(hrts$)=0 then goto abort  'aborts if no entrys
Ct1 = 10
if durt$ = "On" then let Ct1 = 50
do
if val(hrts$)=10 then pw = val(mms$) * 1024 / 100  'calculates pw to milliseconds
if val(hrts$)=20 then pw = val(mms$) * 1024 / 50
if val(hrts$)=30 then pw = val(mms$) * 1024 / 33.3
refresh
  if dur = 5 then let Ct1 = j*20
  if dur = 10 then let Ct1 = j*10
  if dur = 60 then let Ct1 = j*1.6
  if dur = 10000 then let Ct1 = 50
  bar = Ct1
  PWM(4) = pw
  OPTION.PWMFREQ val(hrts$)
  pause 1000
i = i + 1
j = j + 1    ' for the bar
refresh
LOOP UNTIL i > dur
PWM(4) = 0
dur = 0
bar = 0
refresh
return

Clean:  'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
f=10
i=0
j=10
bar = j
PWM(4) = 0
do
refresh
bar = j
PWM(4) = 512
OPTION.PWMFREQ f
pause 1000
f = f + 0.08    ' for the frequency
i = i + 1       ' for the loop
j = j + 0.36    ' for the bar
bar = j
```

```
refresh
LOOP UNTIL i > 249
PWM(4) = 0
bar = 0
refresh
return

refresh_vars: 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
refresh
return

abort: 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
i=300    'stops the do loop
refresh
PWM(4) = 0
hrts2 = 0
dur = 0
bar = 0
HZ = 0
refresh
return

abort2: 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
i=300    'stops the do loop
refresh
PWM(4) = 0
hrts2 = 0
dur = 0
bar = 0
HZ = 0
Stat$ = "OFF"
refresh
return

EXITs:  'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
PWM(4) = 0
refresh
cls
```

Program 10 : HVAC DUCT CONTROL by ZIM

Preamble

My home has a heated crawl space with furnace ducts. These ducts need to be opened in winter and closed in summer for A/C. I'm getting too old to belly crawl 100' twice a year, so I decided to install motorized ducts until I seen the price! $180 each in Canada!

Then you need a control of sorts. I always look for reasons to play with Annex, so I set out to make my own. I cut six pieces of pipe, threaded 6 pieces of 1/8" welding rod and bought 6 damper plates. I found 6 servo motors MG996R on ebay for cheap.

The output from the ESP is adequate for the servo motor's PWM sensor input, but the servo likes its own separate power source 4.5 - 6v.

The two power supplies must have a common ground. The old IDE hard drive external power supplies work good for this as they have 2 outputs. One is 5v for the servos, and the other is 12v which is easily regulated down to 3.3/5v for the ESP.

All my six motors are connected in parallel. The signal strength is to help find an ideal location for the ESP. I like this so much I will use it on all my projects.

https://cicciocb.com/forum/viewtopic.php?t=32

```
'Zims Duct control

'You need to create a file and upload it to ESP called "mode.txt" for position memory

onHtmlReload create_page
onhtmlchange create_page
gosub create_page
wait

create_page:
cls
pin.mode 5, output
servo.setup 1, 5
out = 0
in = 0
calc = 0
butt$ = "background-color:lightblue;text-align:center;display:block;width:310px;height:50px;font-size:24;font-weight:bold;box-shadow: 3px 3px black;border-radius:5px;line-height:1"
butt2$ = "background-color:lightblue;text-align:center;display:block;width:200px;height:80px;font-size:24;font-weight:bold;box-shadow: 3px 3px black;border-radius:5px;line-height:1"
tbox$ = "background-color:white;text-align:center;display:block;width:310px;height:50px;font-size:24;font-weight:bold"
tbox2$ = "background-color:white;text-align:center;display:block;width:60px;height:60px;font-size:24;font-weight:bold"
```

```
'file.save "/mode.txt", str$(0)   ' you could uncomment this line for your first run, then recomment it. That would create the memory file
setmode = val(file.read$("/mode.txt"))

B$ = ""
B$ = B$ + |<!DOCTYPE html><html><body><style>|
B$ = B$ + |body {background-color: black;}|
B$ = B$ + |</style><center>|
B$ = B$ + |<center>|
B$ = B$ + |<p><font color = 'white', font size = '6'>Basement Duct Control</font></p>|
B$ = B$ + |</center>|
B$ = B$ + BUTTON$("Fully Closed", mode0, "but0")
B$ = B$ + cssid$("but0", butt$)+ | <br>|
B$ = B$ + BUTTON$("1/4 Open", mode1, "but1")
B$ = B$ + cssid$("but1", butt$) + |<br>|
B$ = B$ + BUTTON$("1/2 Open", mode2, "but2")
B$ = B$ + cssid$("but2", butt$) + |<br>|
B$ = B$ + BUTTON$("3/4 Open", mode3, "but3")
B$ = B$ + cssid$("but3", butt$) + |<br>|
B$ = B$ + BUTTON$("Fully Open", mode4, "but4")
B$ = B$ + cssid$("but4", butt$) + |<br>|
B$ = B$ + |<table align='center' width='310' height='70' bgcolor='lightgreen' border='1' cellpad-
ding='8'>|
B$ = B$ + |<center><th>|
B$ = B$ + BUTTON$("Check Signal Strength <span>&#37;</span>", sig, "but5")
B$ = B$ + cssid$("but5" , butt2$)+ |</center></th>|
B$ = B$ + |<center><th>|
B$ = B$ + TEXTBOX$(out ,"txtb2")  'signal strength
B$ = B$ + cssid$("txtb2" , tbox2$)+ |</center></th>|
B$ = B$ + |</table></center>|
B$ = B$ + |</center></body></html>|
HTML B$

if setmode = 0 gosub mode0
if setmode = 1 gosub mode1
if setmode = 2 gosub mode2
if setmode = 3 gosub mode3
if setmode = 4 gosub mode4
refresh
return

sig: ' This is optional to check your ESP AP signal strength. Very helpful to find a good location for ESP
mySSID$ = "ESP blabla" ' define your ESP8266 SSID here
wifi.scan
while wifi.networks(a$) = -1
wend
NL$ = chr$(10)   ' this means New Line
for z=1 to word.count(a$, NL$)  'browse all the lines
  z$ = word$(a$, z,NL$) 'extract line by line
  s$ = word$(z$, 1, ",") ' extract the SSID (the 1st argument)
  if (s$ = mySSID$) then ' if the SSID is the one we search
```

```
    RSSI$ = word$(z$, 3, ",") ' extract the SSID (the 3rd argument)
   'print s$, RSSI$
  end if
 next z
in = val(RSSI$)
calc = (in + 100) / 100 * 110 'formula to go from db to percent (with a little 10% tweak)
out = int(calc)
refresh
return


mode0:
file.save "/mode.txt", str$(0)
servo 1, 5   'closed
html cssid$("but0", "box-shadow: 12px 0px red;")
html cssid$("but1", "box-shadow: 3px 3px black;")
html cssid$("but2", "box-shadow: 3px 3px black;")
html cssid$("but3", "box-shadow: 3px 3px black;")
html cssid$("but4", "box-shadow: 3px 3px black;")
refresh
return


mode1:
file.save "/mode.txt", str$(1)
servo 1, 25   '1/4 open
html cssid$("but0", "box-shadow: 3px 3px black;")
html cssid$("but1", "box-shadow: 12px 0px red;")
html cssid$("but2", "box-shadow: 3px 3px black;")
html cssid$("but3", "box-shadow: 3px 3px black;")
html cssid$("but4", "box-shadow: 3px 3px black;")
refresh
return


mode2:
file.save "/mode.txt", str$(2)
servo 1, 50   '1/2 open
html cssid$("but0", "box-shadow: 3px 3px black;")
html cssid$("but1", "box-shadow: 3px 3px black;")
html cssid$("but2", "box-shadow: 12px 0px red;")
html cssid$("but3", "box-shadow: 3px 3px black;")
html cssid$("but4", "box-shadow: 3px 3px black;")
refresh
return


mode3:
file.save "/mode.txt", str$(3)
servo 1, 75   '3/4 open
html cssid$("but0", "box-shadow: 3px 3px black;")
html cssid$("but1", "box-shadow: 3px 3px black;")
```

```
html cssid$("but2", "box-shadow: 3px 3px black;")
html cssid$("but3", "box-shadow: 12px 0px red;")
html cssid$("but4", "box-shadow: 3px 3px black;")
refresh
return

mode4:
file.save "/mode.txt", str$(4)
servo 1, 100   'fully open
html cssid$("but0", "box-shadow: 3px 3px black;")
html cssid$("but1", "box-shadow: 3px 3px black;")
html cssid$("but2", "box-shadow: 3px 3px black;")
html cssid$("but3", "box-shadow: 3px 3px black;")
html cssid$("but4", "box-shadow: 12px 0px red;")
refresh
return
```

https://cicciocb.com/forum/viewtopic.php?t=32

Program 11 : CAPACITIVE LEVEL METER by ELECTROGUARD and MCGUINN

Preamble (by mcguinn)
I was looking for a way to use Annex to measure waterlevels, and that with as little extra hardware as possible. I ended up using pwm and analogue voltage measurement. If a probe is used with enough capacitance swing, it is accurate enough for home-use

Details and schematic are in the code.

https://cicciocb.com/forum/viewtopic.php?t=40

Constructing a capacitive probe doesn't have to be too complicated: I used 2 copperfoil strips in a laminating pouch.
When you have to monitor levels of more than the size of such a pouch, you will have to be creative with, for instance, metal strips and heat-shrinkable tube.

With the dimensioned 27k resistor the capacitance (when fully immersed) must be about 150 pf or over. Much bigger is much better.
Expect a jitter of about 4 % (less with higher capacitance). Twisted pair leads, not too long, can be used to connect the probe(s)

After starting the program begin with adding a new probe. Then open the setup for that probe and fill in the particulars.
If no valid pwm pinnumber is supplied (ie <0), then the program works in demo-mode.
When calibrating a probe (learning what is 100% and what is 0%) the hintbox shows the steps to take.

I think it can also be used to monitor the moisturelevel in a flowerpot, but I'm not sure about how water is distributed in the soil.
Or use it to detect waterspillage on the floor.

On the pictures:
One shows the laminated probes, the others show a rough test-setup.
The bottle was previously calibrated on 100% bottle is 100% probe. Then it is filled up at 25%, 50%, 75% and 100%.

Program too lengthy with graphics to show here. It can be found at:

https://cicciocb.com/forum/viewtopic.php?p=83&hilit=annex+code#p83:~:text=maxprobes%3D5,cancelprobe%3A%0A%20%20%20%20dofinally%0Areturn

Program 12 : MULTI ALARM CLOCK by ZIM

Preamble:

```
' Zims annex 12 hour alarm clock with DST v9

ONHTMLRELOAD CREATE_PAGE
ONHTMLCHANGE UPDATE
GOSUB CREATE_PAGE
TIMER0 1000, UPDATE
WAIT

CREATE_PAGE:
CLS
AUTOREFRESH 1000
I2C.SETUP 4, 5
GOSUB RTC2ESP                                   ' sets ESP time and date from RTC at startup
INMEM$ = FILE.READ$("/alarms.txt")          '  reads file and puts in ram
DSTCHK$ = RTC.DATE$(2) + " " + RTC.TIME$         'Year/Month/Day Hours:Minutes:Seconds

ALR1H$ = LEFT$(WORD.GETPARAM$(INMEM$, "alarm1"), 2)
ALR1M$ = MID$(WORD.GETPARAM$(INMEM$, "alarm1"), 4, 2)
ALR1AP$ = RIGHT$(WORD.GETPARAM$(INMEM$, "alarm1"), 2)
ALR2H$ = LEFT$(WORD.GETPARAM$(INMEM$, "alarm2"), 2)
ALR2M$ = MID$(WORD.GETPARAM$(INMEM$, "alarm2"), 4, 2)
ALR2AP$ = RIGHT$(WORD.GETPARAM$(INMEM$, "alarm2"), 2)
ALR3H$ = LEFT$(WORD.GETPARAM$(INMEM$, "alarm3"), 2)
ALR3M$ = MID$(WORD.GETPARAM$(INMEM$, "alarm3"), 4, 2)
ALR3AP$ = RIGHT$(WORD.GETPARAM$(INMEM$, "alarm3"), 2)
ALR4H$ = LEFT$(WORD.GETPARAM$(INMEM$, "alarm4"), 2)
ALR4M$ = MID$(WORD.GETPARAM$(INMEM$, "alarm4"), 4, 2)
ALR4AP$ = RIGHT$(WORD.GETPARAM$(INMEM$, "alarm4"), 2)
ALR5H$ = LEFT$(WORD.GETPARAM$(INMEM$, "alarm5"), 2)
ALR5M$ = MID$(WORD.GETPARAM$(INMEM$, "alarm5"), 4, 2)
ALR5AP$ = RIGHT$(WORD.GETPARAM$(INMEM$, "alarm5"), 2)
ALR1MODE$ = LEFT$(WORD.GETPARAM$(INMEM$, "mode1"), 9)
ALR2MODE$ = LEFT$(WORD.GETPARAM$(INMEM$, "mode2"), 9)
ALR3MODE$ = LEFT$(WORD.GETPARAM$(INMEM$, "mode3"), 9)
ALR4MODE$ = LEFT$(WORD.GETPARAM$(INMEM$, "mode4"), 9)
ALR5MODE$ = LEFT$(WORD.GETPARAM$(INMEM$, "mode5"), 9)
D5=14:D6=12
PIN.MODE D5, OUTPUT    'is feed to buzzer
PIN.MODE D6, OUTPUT     'is feed to blue LED
PIN(D5) = 0             'buzzer off
PIN(D6) = 0              'blue LED off
HTMLEVENTVAR$ = ""      ' needed for auto invoke alarm selections
MODE$ = "Alarm Off,Alarm On ,Bell Only,Lite Only"
HLIST$ = "01,02,03,04,05,06,07,08,09,10,11,12"
MLIST$ =
"00,01,02,03,04,05,06,07,08,09,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31
,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59"
```

```
APLIST$ = "AM,PM"
HOUR$ = "0"
MIN$ = "0"
HOUR = VAL(HOUR$)
MIN = VAL(MIN$)
T24$ = RTC.TIME$   '24hr time
T12$ = ""          '12hr time
CNT = 0     'timer for led
CNT2 = 0    'timer for buzzer
CLCKAP$ = ""        'AM/PM for clock setting
DROPDM$ = "background-color:white;text-align:center;width:100px;height:40px;font-size:15;font-weight:bold"
DROPD1$ = "background-color:white;text-align:center;width:60px;height:40px;font-size:20;font-weight:bold"
DROPD$ = "background-color:white;text-align:center;width:55px;height:40px;font-size:20;font-weight:bold"
CSET$ = "background-color:red;text-align:center;width:55px;height:40px;font-size:14;font-weight:bold"
CLKDISP$ = "background-color:white;text-align:center;display:block;width:200px;height:42px;font-size:30;font-weight:bold"

A$ = ""  '                                  CLOCK DISPLAY
A$ = A$ + |<!DOCTYPE html><html><body><style>|
A$ = A$ + |body {background-color: black;}|
A$ = A$ + |p {font-size:20px;}|
A$ = A$ + |</style>|
A$ = A$ + |<table align='center' width='300' height='60' bgcolor='lightgreen' border='1'>|
A$ = A$ + |<th>|
A$ = A$ + |<p>Time</p></th></center>|
A$ = A$ + |<th><center>|
A$ = A$ + TEXTBOX$(T12$, "txt1")
A$ = A$ + |</th></center>|
A$ = A$ + CSSID$("txt1", CLKDISP$)
A$ = A$ + |</table>|
HTML A$

A$ = "" '
A$ = A$ + |<table align='center' width='300' height='100' bgcolor='lightgreen' border='1' cellpadding='4'>|
A$ = A$ + |<tr><th>Apply</th><th>Hours</th><th>Minutes</th><th>AM/PM</th></tr>|
A$ = A$ + |<td>|
A$ = A$ + BUTTON$("Set Clock", SETCLOCK, "but0")
A$ = A$ + CSSID$("but0" , CSET$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(HOUR$, HLIST$, "lboxa")
A$ = A$ + CSSID$("lboxa" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(MIN$, MLIST$, "lboxb")
A$ = A$ + CSSID$("lboxb" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(CLCKAP$, "AM,PM", "lboxc")
A$ = A$ + CSSID$("lboxc" , DROPD1$)
```

```
A$ = A$ + |</td>|
A$ = A$ + |</center>|
A$ = REPLACE$(A$, "Choose here", "")  ' change all the "Choose here" instances with ""
HTML A$

A$ = "" '                                ALARM 1
A$ = A$ + |<table align='center' width='300' height='50' bgcolor='lightgreen' border='1' cellpad-
ding='1'>|
A$ = A$ + |<tr><th>Mode</th><th>Hours</th><th>Minutes</th><th>AM/PM</th></tr>|
A$ = A$ + |<td>|
A$ = A$ + LISTBOX$(ALR1MODE$, MODE$, "mde")
A$ = A$ + CSSID$("mde" , DROPDM$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR1H$, HLIST$, "lbox1")
A$ = A$ + CSSID$("lbox1" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR1M$, MLIST$, "lbox2")
A$ = A$ + CSSID$("lbox2" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR1AP$, APLIST$, "lbox3")
A$ = A$ + CSSID$("lbox3" , DROPD1$)
A$ = A$ + |</td>|
A$ = A$ + |</center>|
A$ = REPLACE$(A$, "Choose here", "")  ' change all the "Choose here" instances with ""?"
HTML A$

A$ = "" '                                ALARM 2
A$ = A$ + |<table align='center' width='300' height='50' bgcolor='lightgreen' border='1' cellpad-
ding='1'>|
A$ = A$ + |<td>|
A$ = A$ + LISTBOX$(ALR2MODE$, MODE$, "mde")
A$ = A$ + CSSID$("mde" , DROPDM$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR2H$, HLIST$, "lbox4")
A$ = A$ + CSSID$("lbox4" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR2M$, MLIST$, "lbox5")
A$ = A$ + CSSID$("lbox5" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR2AP$, APLIST$, "lbox6")
A$ = A$ + CSSID$("lbox6" , DROPD1$)
A$ = A$ + |</td>|
A$ = A$ + |</center>|
A$ = REPLACE$(A$, "Choose here", "")  ' change all the "Choose here" instances with ""?"
HTML A$

A$ = "" '                                ALARM 3
A$ = A$ + |<table align='center' width='300' height='50' bgcolor='lightgreen' border='1' cellpad-
ding='1'>|
A$ = A$ + |<td>|
A$ = A$ + LISTBOX$(ALR3MODE$, MODE$, "mde")
A$ = A$ + CSSID$("mde" , DROPDM$)
```

```
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR3H$, HLIST$, "lbox7")
A$ = A$ + CSSID$("lbox7" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR3M$, MLIST$, "lbox8")
A$ = A$ + CSSID$("lbox8" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR3AP$, APLIST$, "lbox9")
A$ = A$ + CSSID$("lbox9" , DROPD1$)
A$ = A$ + |</td>|
A$ = A$ + |</center>|
A$ = REPLACE$(A$, "Choose here", "")  ' change all the "Choose here" instances with ""?"
HTML A$

A$ = "" '                                ALARM 4
A$ = A$ + |<table align='center' width='300' height='50' bgcolor='lightgreen' border='1' cellpad-
ding='1'>|
A$ = A$ + |<td>|
A$ = A$ + LISTBOX$(ALR4MODE$, MODE$, "mde")
A$ = A$ + CSSID$("mde" , DROPDM$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR4H$, HLIST$, "lbox10")
A$ = A$ + CSSID$("lbox10" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR4M$, MLIST$, "lbox11")
A$ = A$ + CSSID$("lbox11" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR4AP$, APLIST$, "lbox12")
A$ = A$ + CSSID$("lbox12" , DROPD1$)
A$ = A$ + |</td>|
A$ = A$ + |</center>|
A$ = REPLACE$(A$, "Choose here", "")  ' change all the "Choose here" instances with ""?"
HTML A$

A$ = "" '                                ALARM 5
A$ = A$ + |<table align='center' width='300' height='50' bgcolor='lightgreen' border='1' cellpad-
ding='1'>|
A$ = A$ + |<td>|
A$ = A$ + LISTBOX$(ALR5MODE$, MODE$, "mde")
A$ = A$ + CSSID$("mde" , DROPDM$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR5H$, HLIST$, "lbox7")
A$ = A$ + CSSID$("lbox7" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR5M$, MLIST$, "lbox8")
A$ = A$ + CSSID$("lbox8" , DROPD$)
A$ = A$ + |</td><td>|
A$ = A$ + LISTBOX$(ALR5AP$, APLIST$, "lbox9")
A$ = A$ + CSSID$("lbox9" , DROPD1$)
A$ = A$ + |</td>|
A$ = A$ + |</center>|
A$ = REPLACE$(A$, "Choose here", "")  ' change all the "Choose here" instances with ""?"
```

```
HTML A$
A$ = ""
RETURN

SETCLOCK:
IF CLCKAP$ = "PM" THEN HOUR = VAL(HOUR$) + 12 ELSE HOUR = VAL(HOUR$)
IF HOUR = 24 THEN LET HOUR = 00
IF HOUR = 12 AND CLCKAP$ = "AM" THEN LET HOUR = 00
MIN = VAL(MIN$)
T$ = RTC.TIME$
D$ = RTC.DATE$(2)           'Year, Month, Day,
D = VAL(WORD$(D$,3,"/"))
M = VAL(WORD$(D$,2,"/"))
Y = VAL(WORD$(D$,1,"/"))
S = VAL(WORD$(T$,3,":"))
RTC.SETTIME Y, M, D, HOUR, MIN, S  ' Year, Month, Day, Hours, Minutes, Seconds
SETTIME Y, M, D, HOUR, MIN, S      ' Year, Month, Day, Hours, Minutes, Seconds
RETURN

UPDATE:                          '<<<<<<<UPDATE>>>>>>>
T$ = RTC.TIME$
D$ = RTC.DATE$(2)
DSTCHK$ = RTC.DATE$(2) +" "+ RTC.TIME$
IF (DSTCHK$ = "21/03/14 02:00:00") OR (DSTCHK$ = "22/03/13 02:00:00") GOSUB AD-
DHOUR
IF (DSTCHK$ = "20/11/1 02:00:00") OR (DSTCHK$ = "21/11/7 02:00:00") OR (DSTCHK$ =
"22/11/6 02:00:00") GOSUB CUTHOUR

IF CNT > 0 THEN CNT = CNT - 1 ELSE CNT = 0
IF CNT2 > 0 THEN CNT2 = CNT2 - 1 ELSE CNT2 = 0
IF CNT = 1 THEN PIN.TONE D6, 0
IF (CNT2 = 8) OR (CNT2 = 5) OR (CNT2 = 2) THEN PIN(D5) = 1
IF (CNT2 = 7) OR (CNT2 = 4) OR (CNT2 = 1) THEN PIN(D5) = 0

T24$ = RTC.TIME$
IF VAL(T24$) > 12 THEN T12 = VAL(T24$)-12 ELSE T12 = VAL(T24$)
IF VAL(T24$) = 0 THEN T12 = 12
IF VAL(T24$) >= 12 THEN PORAC$ = " PM" ELSE PORAC$ = " AM"
T12$ = STR$(T12) + RIGHT$(T24$, 6) + PORAC$
IF T12 > 0 AND T12 < 10 THEN LET T12$ = "0" + T12$
IF T12$ = WORD.GETPARAM$(INMEM$, "alarm1") AND WORD.GETPARAM$(INMEM$,
"mode1") <> "Alarm Off" THEN
    IF WORD.GETPARAM$(INMEM$, "mode1") = "Alarm On" GOSUB ALARMON
    IF WORD.GETPARAM$(INMEM$, "mode1") = "Bell Only" GOSUB BELLONLY
    IF WORD.GETPARAM$(INMEM$, "mode1") = "Lite Only" GOSUB LITEONLY
ENDIF
IF T12$ = WORD.GETPARAM$(INMEM$, "alarm2") AND WORD.GETPARAM$(INMEM$,
"mode2") <> "Alarm Off" THEN
    IF WORD.GETPARAM$(INMEM$, "mode2") = "Alarm On" GOSUB ALARMON
    IF WORD.GETPARAM$(INMEM$, "mode2") = "Bell Only" GOSUB BELLONLY
    IF WORD.GETPARAM$(INMEM$, "mode2") = "Lite Only" GOSUB LITEONLY
ENDIF
```

```
IF T12$ = WORD.GETPARAM$(INMEM$, "alarm3") AND WORD.GETPARAM$(INMEM$,
"mode3") <> "Alarm Off" THEN
    IF WORD.GETPARAM$(INMEM$, "mode3") = "Alarm On" GOSUB ALARMON
    IF WORD.GETPARAM$(INMEM$, "mode3") = "Bell Only" GOSUB BELLONLY
    IF WORD.GETPARAM$(INMEM$, "mode3") = "Lite Only" GOSUB LITEONLY
ENDIF
IF T12$ = WORD.GETPARAM$(INMEM$, "alarm4") AND WORD.GETPARAM$(INMEM$,
"mode4") <> "Alarm Off" THEN
    IF WORD.GETPARAM$(INMEM$, "mode4") = "Alarm On" GOSUB ALARMON
    IF WORD.GETPARAM$(INMEM$, "mode4") = "Bell Only" GOSUB BELLONLY
    IF WORD.GETPARAM$(INMEM$, "mode4") = "Lite Only" GOSUB LITEONLY
ENDIF
IF T12$ = WORD.GETPARAM$(INMEM$, "alarm5") AND WORD.GETPARAM$(INMEM$,
"mode5") <> "Alarm Off" THEN
    IF WORD.GETPARAM$(INMEM$, "mode5") = "Alarm On" GOSUB ALARMON
    IF WORD.GETPARAM$(INMEM$, "mode5") = "Bell Only" GOSUB BELLONLY
    IF WORD.GETPARAM$(INMEM$, "mode5") = "Lite Only" GOSUB LITEONLY
ENDIF

IF HTMLEVENTVAR$ = "" RETURN
GOSUB MEM2FILE
RETURN

RTC2ESP:                ' Synch RTC to ESP
T$ = RTC.TIME$          ' Hours, Minutes, Seconds
D$ = RTC.DATE$(2)           'Year, Month, Day,
D = VAL(WORD$(D$,3,"/"))
M = VAL(WORD$(D$,2,"/"))
Y = VAL(WORD$(D$,1,"/"))
H = VAL(WORD$(T$,1,":"))
MN = VAL(WORD$(T$,2,":"))
S = VAL(WORD$(T$,3,":"))
SETTIME Y, M, D, H, MN, S     'Year, Month, Day, Hours, Minutes, Seconds
RETURN

MEM2FILE:
WORD.SETPARAM INMEM$, "alarm1", ALR1H$ + ":" + ALR1M$ + ":" + "00 " + ALR1AP$
WORD.SETPARAM INMEM$, "alarm2", ALR2H$ + ":" + ALR2M$ + ":" + "00 " + ALR2AP$
WORD.SETPARAM INMEM$, "alarm3", ALR3H$ + ":" + ALR3M$ + ":" + "00 " + ALR3AP$
WORD.SETPARAM INMEM$, "alarm4", ALR4H$ + ":" + ALR4M$ + ":" + "00 " + ALR4AP$
WORD.SETPARAM INMEM$, "alarm5", ALR5H$ + ":" + ALR5M$ + ":" + "00 " + ALR5AP$

WORD.SETPARAM INMEM$, "mode1", ALR1MODE$
WORD.SETPARAM INMEM$, "mode2", ALR2MODE$
WORD.SETPARAM INMEM$, "mode3", ALR3MODE$
WORD.SETPARAM INMEM$, "mode4", ALR4MODE$
WORD.SETPARAM INMEM$, "mode5", ALR5MODE$


FILE.SAVE "/alarms.txt", INMEM$
HTMLEVENTVAR$ = ""
RETURN
```

```
ALARMON:
CNT = 30
CNT2 = 9
PIN.TONE D6,1
GOSUB RTC2ESP
RETURN

BELLONLY:
CNT2 = 9
GOSUB RTC2ESP
RETURN

LITEONLY:
CNT = 30
PIN.TONE D6,1
GOSUB RTC2ESP
RETURN

CUTHOUR:   '            DST set back 1 hour
T$ = RTC.TIME$
D$ = RTC.DATE$(2)          'Year, Month, Day,
D = VAL(WORD$(D$,3,"/"))
M = VAL(WORD$(D$,2,"/"))
Y = VAL(WORD$(D$,1,"/"))
H = VAL(WORD$(T$,1,":"))
MN = VAL(WORD$(T$,2,":"))
S = VAL(WORD$(T$,3,":"))
RTC.SETTIME Y, M, D, H-1, MN, S  ' Year, Month, Day, Hours, Minutes, Seconds
SETTIME Y, M, D, H-1, MN, S      ' Year, Month, Day, Hours, Minutes, Seconds
RETURN

ADDHOUR:     '              DST set ahead 1 hour
T$ = RTC.TIME$
D$ = RTC.DATE$(2)          'Year, Month, Day,
D = VAL(WORD$(D$,3,"/"))
M = VAL(WORD$(D$,2,"/"))
Y = VAL(WORD$(D$,1,"/"))
H = VAL(WORD$(T$,1,":"))
MN = VAL(WORD$(T$,2,":"))
S = VAL(WORD$(T$,3,":"))
RTC.SETTIME Y, M, D, H+1, MN, S  ' Year, Month, Day, Hours, Minutes, Seconds
SETTIME Y, M, D, H+1, MN, S      ' Year, Month, Day, Hours, Minutes, Seconds
RETURN
```

https://cicciocb.com/forum/viewtopic.php?t=47

Program 13 : FORUM EXTENSIONS and I2C ADDRESS SCANNER by CICCIOCB

Preamble:

I did some background work on the new forum, in particular implementing the following extensions :

- **Clipboard Image**

Allows to paste image from clipboard to be added as an attachment.
Any image copied in the clipboard can be pasted directly into the message (this avoid the annoying action to paste into an image file and then upload it)

- **Google Translator**

Adds Google Translator to the Forum Header.
This is self explanatory; actually implemented English, Spanish, German, Italian, French (other can be implemented on request)

- **Media Embed PlugIn**

An official phpBB extension that allows users to embed content from allowed sites using {media] BBCode, or from simply posting a supported URL in plain text.
This allows to include images / videos and other elements from other hosting sites (i.e. youtube) simply copying the corresponding url

- **phpBB Studio Prism JS syntax highlighter**

This is probably the most useful on the list as it enable to show the code with the correct syntax highlight.
I customised this plugin to show the code following the colours defined in the Annex internal editor.

To use it, simply use the code button

https://cicciocb.com/forum/viewtopic.php?t=48

```
'I2C Address Scanner
'print in the console the address of the devices found
I2C.SETUP 21, 22  ' set I2C port on pins 4 and 5

for i = 0 to 120
  i2c.begin i
  if i2c.end = 0 then
    wlog"found "; i , hex$(i)
    pause 10
  end if
next i

end
```

Program 14 : SIMPLE STYLISH CLOCK by ELECTROGUARD

Preamble:

This example brings HTML and CSS and Javascript all together to create a simple but stylish digital clock.

Most of the style is initially commented out, so uncomment a particular line to see how it affects the style.

Note that although style values are usually embedded in the HTML, the script shows how Annex can specify style values in variables if wished, so don't forget to uncomment the appropriate variables as well.

Obviously the specified font file will only display if it is available on the Annex device - you can find some font files and more detailed instructions in the Skin Clock project here...
https://sites.google.com/site/annexwifi ... skin-clock

And here is a handy W3Schools page for selecting different colours...
https://www.w3schools.com/colors/colors_groups.asp

In the same way that some of the style lines are commented out, feel free to insert some of your own between lines 7 and 11 to add more style.

Note that the syntax to specify a style parameter and its value is... style-parameter : value ;

Notice that white space is optional, but a colon must follow the parameter, and a semi-colon must follow the value, else the parameter will be ignored.

Don't forget to tick the Editors 'Stop Log' checkbox to prevent it clogging things up.
The simple example shows how a 'live' clock display might easily be incorporated into your own projects, with or without styling.

```
'Simple Stylish Clock - uncomment lines below to add styles, and tailor style values to your own liking
'clocksize$ = "5"
'clockcol$ = "darkblue"
'clockbak$ = "lightyellow"
'fontfile$ = "/dig7monoitalic.ttf"       'filename.ext of optional font file
'a$ = a$ + |<style> @font-face { font-family: myfont; src: url('| + fontfile$ + |');} </style>|
a$ = a$ + |<div id='clock' style=' |
'$ = a$ + |font-family:myfont; background:|+clockbak$+|; color:|+clockcol$+|; font-size:|+clock-size$+|em; |
'a$ = a$ + |border:1px solid gray; border-radius:.3em; text-align:center;padding:.18em; |
'a$ = a$ + |display: table; margin-right:auto;margin-left: auto;padding-left:.4em; padding-right:.4em; |
a$ = a$ + |;'>| + time$ + |</div><br>|
```

```
html a$
timer0 1000,ticktock
wait

ticktock:
jscall |_$('clock').innerHTML = "| + time$ + |"|
return

'------- End ------
```

https://cicciocb.com/forum/viewtopic.php?t=63

Program 15 : SANDCLOCK by PETERN

Preamble:

I created this project some time ago as an example for newbies to ANNEX.
A clock - both analogue and digital - consisting of
- ESP8266-Wemos-D1-Mini,
- TM1637 4digit 7-segment display
- SG90 mini-servo and some cheap peripheral components like
- a LDR
- 18B20 temperature sensor
- 4 NeoPixel LEDs
- mini hour glass (10 minutes)

To add to the confusion, it will not only display the atomic internet time but turn the emptied hour-glass, show the temperature and the brightness of the display will respond to the ambient light, while the background of the hourglass is changed in color.
The IP address of the web interface is cut into four parts and displayed at startup.

Circuit and Description can be found at :

https://cicciocb.com/forum/viewtopic.php?t=115

```
' ################### S A N D C L O C K ############################
' 4 digits digital watch + 10 minutes hourglass
  VERSION$  = "v3.3e"
' 11/2019  Peter.Neufeld@gmx.de   min:ANNEX_1.39b6
' - ESP8266-Module:    WEMOS-D1-Mini
' - Display:           4digit 7segment TM1637
' - Servo:             analog servo (e.g. SG90)
' - Backlight:         4*NeoPixel-LED
' - Temperatur sensor: DS18B20
' - Light sensor:      LDR at analog-input
'GPIO-Mapping for WEMOS- und NodeMCU-boards
D0=16:D1= 5:D2= 4:D3= 0:D4=2
D5=14:D6=12:D7=13:D8=15:D9=3:D10=1
SERVO_PIN   = D5  'Servo at GPIO14=D5
TM_DATA     = D2  'TM1637 Data  at GPIO4=D2
TM_CLOCK    = D1  'TM1637 Clock at GPIO5=D1
TM_BRIGHT   = 7   'TM1637 brightnes
TURN_TIME   = 10  'turn glass each xx minutes
SERVO_LEFT  = 180
SERVO_RIGHT = 0
SERVO_POS   = 0
BLINK       = 0   '255= red colon at TM1637
ADC_DARK    = 450 'max ADC-value,while LDR dark
'           depends on  LDR and  Pullup-resistor
LED1_STATUS = 0 'Background-LEDs: 0=automatik mode, 1=manual
STATUS$     = "Mode: AUTOMATIC"
ADC_IN = 0: t$="":TEMP$="":HH=0:MM=0:SS=0 ' working variables
R=0:G=0:B=0
```

```
'----Setup sensors and actuators ------------------
SERVO.SETUP 1, SERVO_PIN
TM1637.SETUP TM_DATA, TM_CLOCK
'!!! NeoPixel-data-output at D4=GPIO2 only!!!
NEO.setup 4   '4 cascaded NeoPixel-LEDs

gosub        web_page
onHtmlReload  web_page
onhtmlchange  web_page
gosub TOGGLE_SERVO 'turn sandclock 180deg
gosub SHOW_IP      'show assigned IP-adress

'Start main routines with timer once per second
TIMER0 1000, MAIN_ROUTINE
wait
end
'####################################################################
'SUBROUTINES#########################################################
'####################################################################
MAIN_ROUTINE:
HH=val(left$(time$,2))     'double-digit hours
MM=val(word$(time$,2,":")) 'double-digit minutes
SS=val(right$(time$,2))    'double-digit seconds
gosub LIGHT_SENSOR         'Get strength of the ambient light
if SS > 24 and SS mod 25 < 6 then
  gosub SHOW_TEMP          'Show temperature every 25s for 5s
else
  gosub SHOW_TIME 'TM1637-refresh (time and :)
endif
gosub BACKGROUND_LEDS
'Turn hourglass every TURN_TIME minutes after full hour
if SS = 0 and MM mod TURN_TIME = 0 then gosub TOGGLE_SERVO
t$=time$
refresh                    'refresh webpage
return
'####################################################################
LIGHT_SENSOR:
'Light sensor for adjusting the brightness of the LED display
' bright = low value on the ADC (~ 30)
' dark = high values at the ADC (> 500)
ADC_IN = ADC
if ADC_IN > ADC_DARK then   ADC_IN=ADC_DARK
TM_BRIGHT = int(8-(ADC_IN/(ADC_DARK/7)))
return
'####################################################################
SHOW_TEMP:
'Get only two digits from temperature (quick and dirty:-)
TEMP$= right$("  " + word$(TEMPR$(D6,1),1,".")+ " C",4)
'Display temperature on TM1637 display every 25 s
if SS mod 24 = 1 then
  TM1637.PRINT "   C", TM_BRIGHT, 0
```

```
else
  TM1637.PRINT TEMP$, TM_BRIGHT, 0
endif
return
'###################################################################
SHOW_TIME:

BLINK = 255 - BLINK  'toggle colon in TM1637
'Show time and toggled colon again
TM1637.PRINT left$(time$,2)+mid$(time$,4,2),TM_BRIGHT,BLINK
return
'###################################################################
TOGGLE_SERVO:
if SERVO_POS = SERVO_LEFT then
  for SERVO_POS = SERVO_LEFT to SERVO_RIGHT Step -4
    servo 1, SERVO_POS
    pause 100
  next SERVO_POS
  SERVO_POS = SERVO_RIGHT
else
  FOR SERVO_POS = SERVO_RIGHT to SERVO_LEFT step 4
    servo 1, SERVO_POS
    pause 100
  NEXT SERVO_POS
  SERVO_POS = SERVO_LEFT
endif
return
'###################################################################
BACKGROUND_LEDS:
'Set background LEDs
if R+G+B = 0 then
  'take generated RGB values, if sliders not in use
  NEO.strip 0,3,SS mod 20,30-(SS mod 30),20-(SS mod 20)
else
  'manual adjusted slider or textbox  values for R, G, B
  NEO.strip 0,3,R,G,B
endif
return
'###################################################################
SHOW_IP:
'shows the 4 parts of the IP address on the TM1637 display
TM1637.PRINT  "IP  ", 7, 255
pause 2000
for i = 1 to 4
  TM1637.PRINT  word$(REPLACE$(IP$, " ", "."),i,"."), 7, 0
  pause 1500
next i
pause 4000
return
'###################################################################
WEB_PAGE:
cls
```

```
a$ = "<center><h2> - S A N D C L O C K - "+ VERSION$ +" - </h2>"
a$ = a$ + "<br>" + textbox$(t$,"cssTB")
a$ = a$ + METER$(SS,0,60,"cssMET")
a$ = a$ + textbox$(TEMP$,"cssTB")+"<br><br>"
a$ = a$ + "<br>R: "+ slider$(R, 0,255)+ textbox$(R,"cssTB")
a$ = a$ + "<br>G: "+ slider$(G, 0,255)+ textbox$(G,"cssTB")
a$ = a$ + "<br>B: "+ slider$(B, 0,255)+ textbox$(B,"cssTB")
a$ = a$ + "<br>"+ LED$(LED1_STATUS)
a$ = a$ + "<br>"+ textbox$(STATUS$)
a$ = a$ + "<br>"+ BUTTON$("automatic / manual",MAN_AUTO,"cssBT")
' simple adaptation of the appearance
a$ = a$ + cssid$("cssTB"," width:70px;text-align:center")
a$ = a$ + cssid$("cssMET"," transform:rotate(-90deg);")
a$ = a$ + cssid$("cssBT","font-size:1.8em;border-radius:1.4em;")

html a$
a$ = ""
return
'#########################################################################
MAN_AUTO:
'Called from the MANUAL / AUTO button from the web interface
'Switches the backlight mode.
LED1_STATUS= 1 - LED1_STATUS 'Switch Webpage LED1 red / green
IF LED1_STATUS = 0 then
  R=0: G=0: B=0
  STATUS$="Mode: AUTOMATIC"
else
  R=20: G=100: B=20
  STATUS$="Mode: MANUAL   "
endif
refresh
return
```

https://cicciocb.com/forum/viewtopic.php?t=115