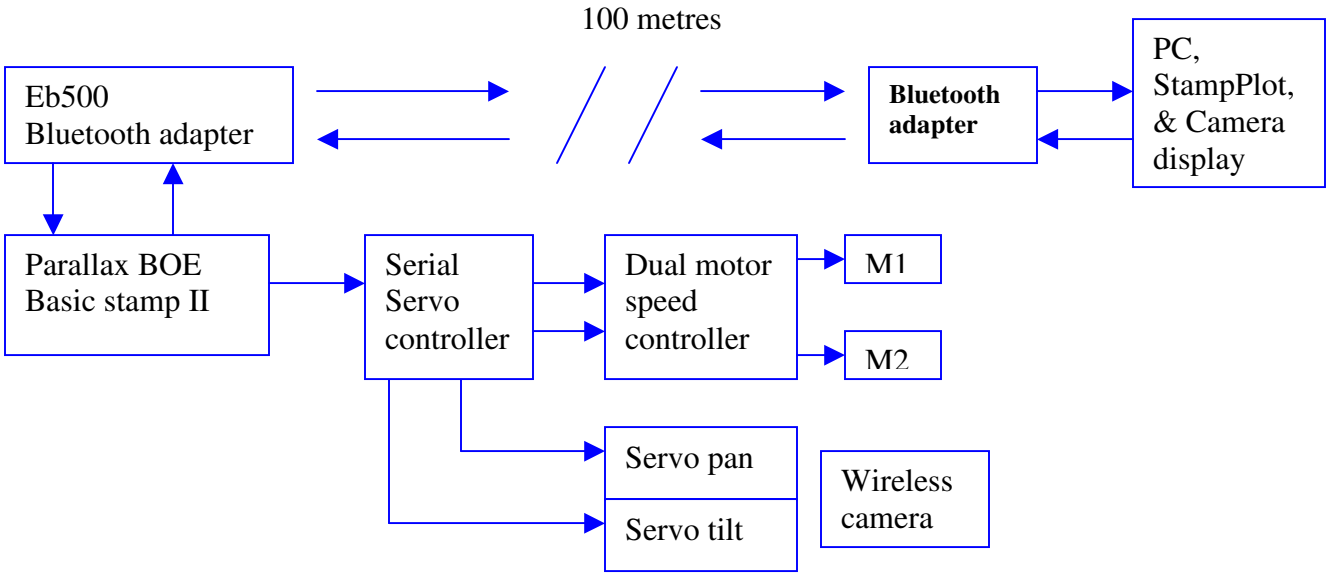
StampPlot GUI control of vehicle via class I Bluetooth

Stuart Weightman Jan 2006



This project was inspired by the robotic vehicle recently sent to Mars, although in this case data is only sent < 100 metres to and fro from the control station (PC StampPlot) to the tracked vehicle (BOE). A model tank was utilized for the platform as the project was control of and not the building of a vehicle. The original tank motors only lasted a couple of hours and consequently were replaced with more durable car windscreen motors.

The programme code is downloaded to the BS2 with the macro residing in StampPlotPro on the PC.
Initial startup is as follows.

1. Apply power to BS2 and associated connected equipment, except the motor control board.
2. Start StampPlot then open the GUI macro, when connect is clicked on the StampPlot toolbar a Bluetooth connection is achieved. (via com port 5 in this case)
3. The camera display is set to “always on top” so is visible on the space allocated on the GUI with the pan / tilt sliders beside.
4. Apply power to motor control board.

Bi directional serial data can now be sent between the PC and the remote controlled vehicle.
Initially the GUI had one slider for control of each vehicle track but this was found to be less than ideal. Buttons were found to give better control with some having a timed function. i.e. - vehicle is traveling forward and one click spins the vehicle 45 deg then continues forward.

BS2 code

```
'{$STAMP BS2}    Bluetooth GUI control of tracked vehicle     Stuart Weightman 2005
' **** StampPlot control/interface with Bluetooth EB500 and Basic Stamp2****
' **** Pins 10 through to 19 are available for I/O, sounder on P11, SSC on P10 ****


  INPUT 5                       'EB500 I/O Line 5 provides the connection status
  PAUSE 1000                      'Wait for the eb500 radio to be ready
WaitForConnection:
  SEROUT 10,$4054,[255,0,127]         'Set Left servo0 to stop, mid position
  SEROUT 10,$4054,[255,2,127]         'Set Right servo1 to stop, mid position
  IF IN5 = 0 THEN WaitForConnection
  GOSUB Beep1                     'Beep when connected and continue
```

```
PAN VAR Byte
TILT VAR Byte
EStop VAR Bit
FF VAR Bit
FS VAR Bit
BS VAR Bit
BF VAR Bit
FR VAR Bit
FL VAR Bit
BR VAR Bit
BL VAR Bit
SR VAR Bit
SL VAR Bit
L VAR Byte
R VAR Byte
XL VAR Byte
XR VAR Byte

Main:
  IF IN5 = 0 THEN WaitForConnection     'If Bluetooth connection is lost then wait to re-establish

  SEROUT 1,84,["!READ (PAN)",CR]        'Read StampPlot PAN slider
  SERIN 0,84,1000,Main,[SDEC PAN]       'Accept returning data
  SEROUT 10,$4054,[255,4,PAN]           'Scaled output to SSCI servo4 (needs baud $4054 not 84)

  SEROUT 1,84,["!READ (TILT)",CR]       'Read StampPlot TILT slider
  SERIN 0,84,1000,Main,[SDEC TILT]      'Accept returning data
  SEROUT 10,$4054,[255,6,TILT]          'Scaled output to SSCI servo6 (needs baud $4054 not 84)

  SEROUT 1,84,["!READ (EStop)",CR]      'Request status (1/0) of STOP checkbox
  SERIN 0,84,1000,Main,[DEC EStop]      'Store returning data
  IF EStop = 1 THEN EmgStop

  SEROUT 1,84,["!READ (FF)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC FF]         'Store returning data
  IF FF = 1 THEN Forwfast

  SEROUT 1,84,["!READ (FS)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC FS]         'Store returning data
  IF FS = 1 THEN ForwSlow

  SEROUT 1,84,["!READ (BS)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC BS]         'Store returning data
  IF BS = 1 THEN BackSlow

  SEROUT 1,84,["!READ (BF)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC BF]         'Store returning data
  IF BF = 1 THEN BackFast

  SEROUT 1,84,["!READ (FR)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC FR]         'Store returning data
  IF FR = 1 THEN ForwRight

  SEROUT 1,84,["!READ (FL)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC FL]         'Store returning data
  IF FL = 1 THEN ForwLeft

  SEROUT 1,84,["!READ (BR)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC BR]         'Store returning data
  IF BR = 1 THEN BackRight

  SEROUT 1,84,["!READ (BL)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC BL]         'Store returning data
  IF BL = 1 THEN BackLeft

  SEROUT 1,84,["!READ (SR)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC SR]         'Store returning data
  IF SR = 1 THEN SpinRight

  SEROUT 1,84,["!READ (SL)",CR]         'Request status (1/0) of checkbox
  SERIN 0,84,1000,Main,[DEC SL]         'Store returning data
  IF SL = 1 THEN SpinLeft
```

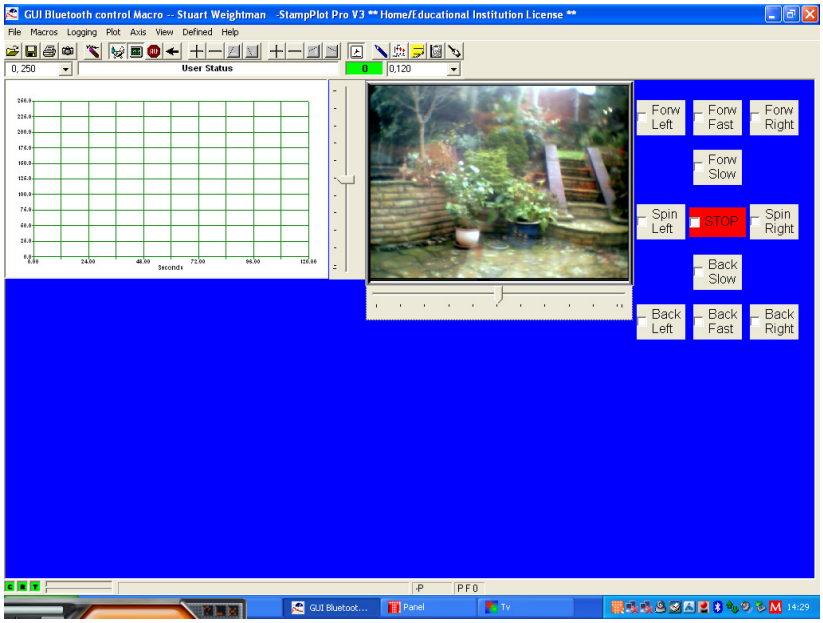
```
  GOTO Main

EmgStop:
  SEROUT 10,$4054,[255,0,127]          'Set Left SSCI Motor0 to stop mid position
  SEROUT 10,$4054,[255,2,127]          'Set Right SSCI Motor2 to stop mid position
  PAUSE 100                            'Pause to view check tick
  SEROUT 1,84,["!O EStop=0",CR]        'Clear the check box
  GOSUB ClearChecks : GOSUB Beep2      'Clear all checks and Two beeps to acknowledge Emg stop initiated
  L=127 : R=127                        'L & R variables to mid stop value
  GOTO Main

ForwFast:
  L=255 : R=255 : GOSUB ClearChecks : GOTO Tracks

ForwSlow:
  L=190 : R=190 : GOSUB ClearChecks : GOTO Tracks

BackSlow:
  L=64 : R=64 : GOSUB ClearChecks : GOTO Tracks

BackFast:
  L=0 : R=0 : GOSUB ClearChecks : GOTO Tracks

ForwLeft:
  XL=L : XR=R                          'Store in use values of L & R
  L=127 : R=255
  SEROUT 1,84,["!O FL=0",CR]           'Clear the check box
  GOSUB TrackTimer
  L=XL : R=XR                          'Restore in use values of L & R
  GOTO Tracks

ForwRight:
  XL=L : XR=R                          'Store in use values of L & R
  L=255 : R=127
  SEROUT 1,84,["!O FR=0",CR]           'Clear the check box
  GOSUB TrackTimer
  L=XL : R=XR                          'Restore in use values of L & R
  GOTO Tracks

BackRight:
  XL=L : XR=R                          'Store in use values of L & R
  L=0 : R=127
  SEROUT 1,84,["!O BR=0",CR]           'Clear the check box
  GOSUB TrackTimer
  L=XL : R=XR                          'Restore in use values of L & R
  GOTO Tracks

BackLeft:
  XL=L : XR=R                          'Store in use values of L & R
  L=127 : R=0
  SEROUT 1,84,["!O BL=0",CR]           'Clear the check box
  GOSUB TrackTimer
  L=XL : R=XR                          'Restore in use values of L & R
  GOTO Tracks

SpinRight:
  XL=L : XR=R                          'Store in use values of L & R
  L=255 : R=0
  SEROUT 1,84,["!O SR=0",CR]           'Clear the check box
  GOSUB TrackTimer
  L=XL : R=XR                          'Restore in use values of L & R
  GOTO Tracks

SpinLeft:
  XL=L : XR=R                          'Store in use values of L & R
  L=0 : R=255
  SEROUT 1,84,["!O SL=0",CR]           'Clear the check box
  GOSUB TrackTimer
  L=XL : R=XR                          'Restore in use values of L & R
  GOTO Tracks
```

```
Tracks:
  SEROUT 10,$4054,[255,0,R]          'Output VAR R to Motor0
  SEROUT 10,$4054,[255,2,L]          'Output VAR L to Motor2
  GOTO Main

TrackTimer:
  SEROUT 10,$4054,[255,0,R]          'Output VAR R to SSCI Motor0
  SEROUT 10,$4054,[255,2,L]          'Output VAR L to SSCI Motor2
  PAUSE 1000
  RETURN

ClearChecks:
  SEROUT 1,84,["!O FF=0",CR] : SEROUT 1,84,["!O FS=0",CR] 'Remove checks
  SEROUT 1,84,["!O BF=0",CR] : SEROUT 1,84,["!O BS=0",CR] 'Remove checks
  RETURN

Beep1:
  FREQOUT 11,750,2500
  RETURN
Beep2:
  FREQOUT 11,200,2500
  FREQOUT 11,200,2800
  RETURN
```

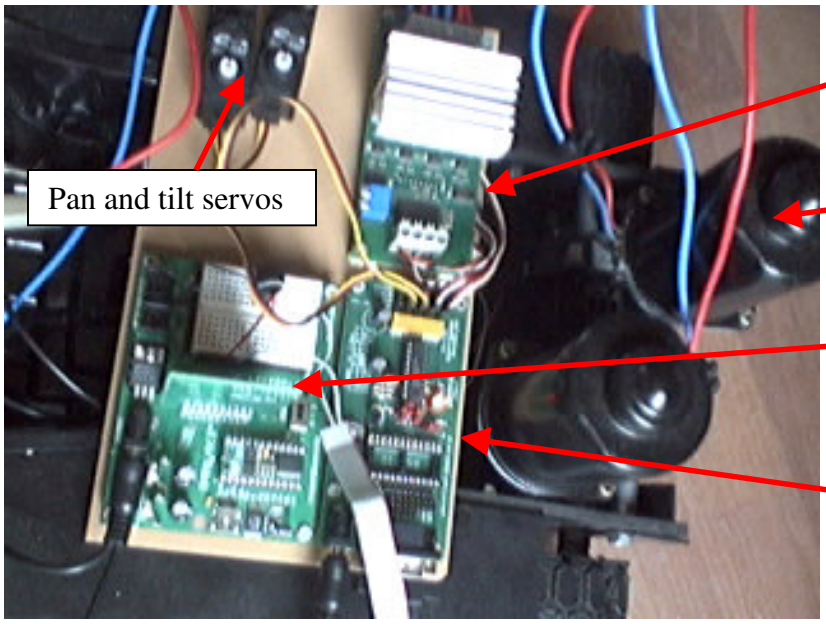
MACRO (Running on StampPlotPro on PC)

```
01-11-2005
Vehicle GUI via Bluetooth
Stuart Weightman

'          ****    StampPlot GUI construction   ****

INIT:
!RSET
!NPSU OFF
' Set as default macro
!DEFS (ME)
!MACR .OBJECTS
' Set Queue size
!QSIZ 5185
' Set analog span
!SPAN  0, 250
' Set max time
!TMAX 120
' Set number of samples
!PNTS 2000
' Title plot
!TITL GUI Bluetooth control Macro -- Stuart Weightman
' Flush old data
!FLSH ON
!DISP D9 G2 A0 S0 B15 R
!ECHO OFF
!SETD 1,1
' Reset the plot
!RSET
' Enable plotting
!PLOT ON
' Connect port on startup
!CONN ON

ENDMAC

OBJECTS:
' Clear all objects
!POBJ Clear
' Set plot percentage
!PPER 40,40
' Background = Blue
!O oBack=(Blue)
```

```
' Create a H slider object called PAN
!O oHSlider.PAN=44.5,59,33,30,0,255,127

' Create a V slider object called TILT
!O oVSlider.TILT=40,100,10,40,0,255,127

' Create a red stop button with check box
!O oCheck.EStop=84.5,74.5,7,6,STOP,0,12,0,10

' Create check object
!O oCheck.FF=85,96,6,7,Forw Fast,8
' Tip
!O FF.tip=Both tracks forward fast

' Create check object
!O oCheck.FS=85,86,6,7,Forw Slow,8
' Tip
!O FS.tip=Both tracks forward slow

' Create check object
!O oCheck.BS=85,65,6,7,Back Slow,8
' Tip
!O BS.tip=Both tracks Backward slow

' Create check object
!O oCheck.BF=85,55,6,7,Back Fast,8
' Tip
!O BF.tip=Both tracks Backward fast

' Create check object
!O oCheck.FR=92,96,6,7,Forw Right,8
' Tip
!O FR.tip=Both tracks Forward right

' Create check object
!O oCheck.FL=78,96,6,7,Forw Left,8
' Tip
!O FL.tip=Both tracks forward left

' Create check object
!O oCheck.BL=78,55,6,7,Back Left,8
' Tip
!O BL.tip=Both tracks Backward left

' Create check object
!O oCheck.BR=92,55,6,7,Back Right,8
' Tip
!O BR.tip=Both tracks Backward right

' Create check object
!O oCheck.SR=92,75,6,7,Spin Right,8
' Tip
!O SR.tip=Both tracks Spin right

' Create check object
!O oCheck.SL=78,75,6,7,Spin Left,8
' Tip
!O SL.tip=Both tracks Spin left

ENDMAC
```



Screen shot showing vehicle controls and view from wireless camera with pan / tilt sliders alongside. Plot is available for displaying information transmitted from vehicle.

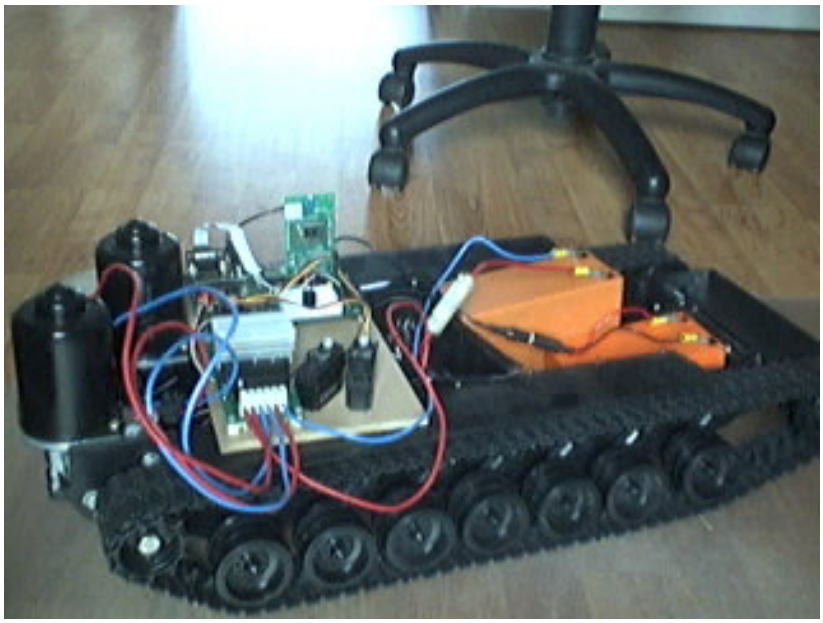


Dual motor speed controller board

Windscreen wiper motor

Layout showing BS2 board of education with EB500 inserted.

Serial servo controller board.



Not cosmetically tidied up yet….. but, all control components work OK.
Pan and tilt servos still to be fitted to camera mount mechanism.