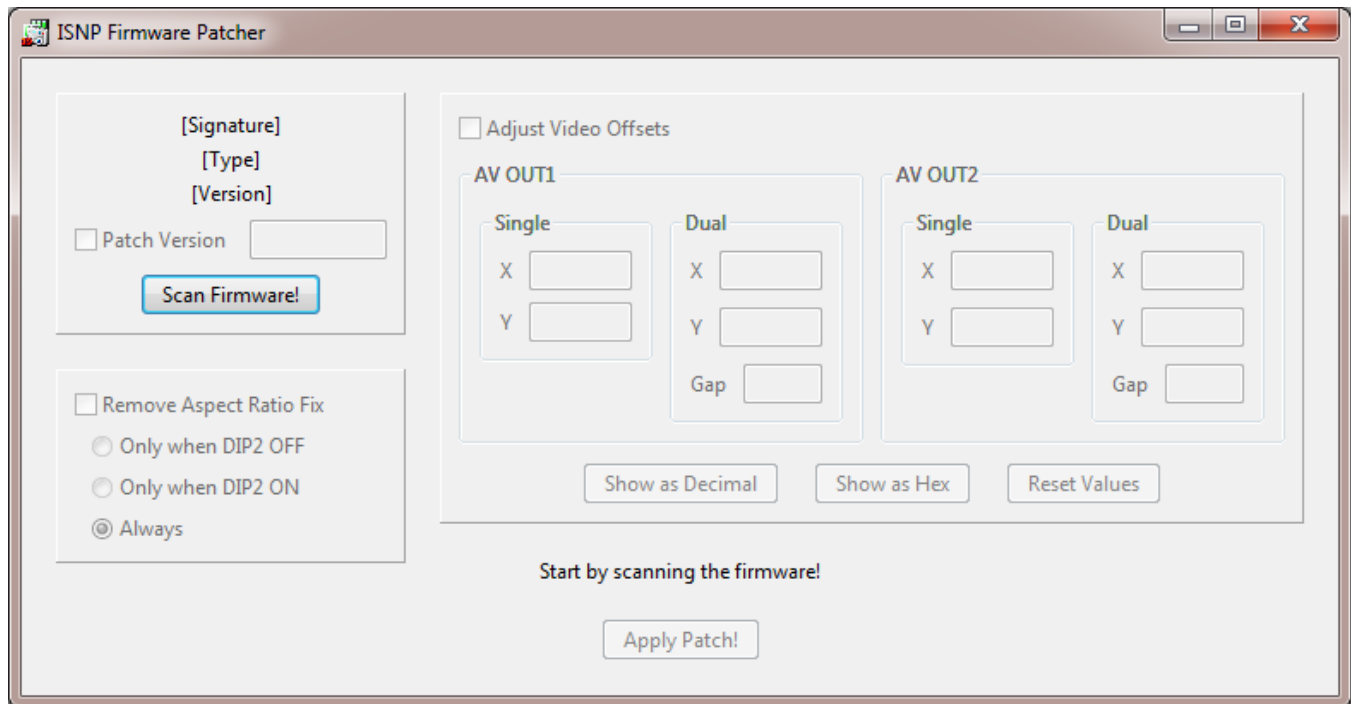


IS-NITRO-CAPTURE Video Adjustment Utility



[@famiac](#) and I wanted to remove the aspect ratio "fix". The "fix" would apply uneven scaling which isn't pixel perfect...

Thus we created this utility to remove that "feature". This utility will patch the firmware, enabling the patch to persist and work without having a PC connected.

As always, lots of thanks to [@Gericom](#) for documenting the IS-NITRO-EMULATOR and the IS-NITRO-CAPTURE hardware registers.

Note this utility is only for the CAPTURE. We have made patches for the EMULATOR in another post.

Introduction

This utility features removing the aspect ratio fix and adjusting the video offsets.

(Removing the aspect ratio fix will cause the image to be uncentered. Plus, we want fine adjustments for a CRT setup.)

For convenience, the utility auto-increments the firmware version so we don't have to reset the firmware everytime we wanted to adjust the video offsets. (ISNP.exe checks if the firmware version is higher than the version installed onto the machine.)

This utility relies on Cheat Engine. Our process patches the firmware stored inside RAM when ISNP.exe runs.

Prerequisites

- Use Windows 7 or earlier
- Install Cheat Engine 7.1 or greater
- Install IS-NITRO-CAPTURE_v2_15_07102300.exe
 - Under Win7, use XP SP3 compatibility mode to install

Usage

1. Open ISNP.exe
2. Open ISNP-Patcher.CT
3. Press **[Scan Firmware!]** The signature, type, and version should be updated.
4. Select any of the options you want to patch.
 - **Patch Version**
 - The firmware version should be 0x1DE.
 - Please enable it; The version number must be higher than the current version.
 - **Remove Aspect Ratio Fix**
 - There are 3 options for DIP2 configuration. When DIP2 is OFF, ON, or Always (Ignore DIP2).
 - (DIP2 allows user settings separate from the default.)
 - **Adjust Video Offsets**
 - You can adjust the settings for each AV out.
 - X and Y offsets for single screen mode.
 - X and Y offsets for dual screen mode.
5. Finally, press **[Apply Patch!]**
6. In ISNP.exe, proceed with a firmware update as usual.

Method for Centering the Image

When the aspect ratio fix is removed, the bottom left corner of the image stays the same using the same offset values.

The image horizontally shrinks by 64 pixels (on a 640*480px grid).

To center the image, Famiac added 32 to the X offset. The Y offset is centered by eye according to the specific monitor.

Famiac's Values:

- Single Screen X & Y: 0xE4, 0x2A
- Dual Screen X & Y: 0x60, 0x26
- Default Single: 0xC4, 0x2E (for reference)

Single Screen Ranges:

- X: 0x87 (left), 0xF6 (right)
- Y: 0x1C (top), 0x3E (bottom)

The ranges represent when the image touches the left, right, top, or bottom edges.

Comments:

Based on the ranges, these units roughly correspond to a 640*480 resolution.

X: $0xF6 - 0x87 = 0x6F = 111px$; $256 * 2 = 512px$ (NDS X Res * 2); $512 + 111 = 623px$

Y: $0x3E - 0x1C = 0x22 = 34px$; $192 * 2 = 384px$ (NDS Y Res * 2); $384 + 34 = 418px$

Technical Details

The firmware is stored inside ISNP.exe and not in some DLL file.

We believe ISNP.exe is packed by ASProtect (ASProtect 2.0x as reported by PEiD). We did manage to unpack the program but we couldn't figure out a way to make it run after patching it... Thus, we gave up on that route... (If you want to give it a shot, I have tried using ImpREC 1.7e and Skylla to try "repacking" the program...)

Regardless, we figure using Cheat Engine might work... and indeed it does! We searched for the firmware header in memory and was able to patch it using Cheat Engine. Thus after a few lines of Lua and making a GUI inside the GUI editor for Cheat Engine, we had a working patcher.

The firmware is "encrypted" simply by decrementing/incrementing each byte... (Just like the firmware for EMULATOR.) Try searching for "HMSDKKHFDMS". (This is what our LUA script searches for to get the reference point for the firmware.)

Look at "script.lua" and "ISNP.CT" if you want to take a look at the code or adjust it. A majority of the code is GUI related while the rest references locations inside the firmware. (Various byte changes here and there...) GUI "code" is done via Cheat Engine's form design tool. Since Cheat Engine can edit memory, executables produced by Cheat Engine can be flagged as viruses... Which is why we opted to use the *.CT format. (Since it's just text afterall.)

```
<?xml version="1.0" encoding="utf-8"?>
<CheatTable CheatEngineTableVersion="31">
  <Forms>
    <Patcher Class="TCEForm" Encoding="Ascii85">*mvX_,r{BEL}Re=$N%Ca3xXo-nk[D{8S?Rv^/HxmJ9h2Vi#stpqQ%8P%e2LsCKZv2=E(3{A.3[q#w0_4Rv;#zx3?
e5$W!_2ry+{EKYPZuVnDhT@d?=vf+=w7-5q*SCmu.Lwwfxqn7,I3=*( {jyYIe[I]?BzVal[U-^8vQ60)FVjhwpYE5i)(4_l7]?;XbN_av:I.
+AFVRy=TSxgbUCB4QnEH@lHbxBn,]:=81XZcB?v9;NwMBW.pXIz#/MYqNC:I:y0H)gbgk/*?Mbp011VSWM(!:(K9^)kh6XKJwK0IVB@Iy.ctvmSq7,Y*0I+T/lgn0+WbmBY+[]12lq-
pd?e],Wc9myJh^9@0R}vhl$Jm{G})}{%}ajvmGB(ok!=:8R$R%mf)Twq*NbhYaJ=sTXLyB(QEBMmw(mG0?M.moMwnMwmr_stJ)tnHhsQX(6V)ILtA[xPAM!E(s^ipTT1gIk!
B53h7fTrXEQDl0}6VrJGHlJw;V!rma7!B14)z[{Eu52gdy@!{#yVxb5!UySUS$2M[R1qzCGY3Th_v?4Wqv?wMZ{ZTX=U/=)z-p2)7N?7[H.4k{0v$zS:,6[0%HCpp^=wV@RxpPU^wmm!
f8506rAaRxySeb8JdPMk31AKf;8?OT!;U%!46%Y6Fj72}tdBwPNEZ!=%rqqv)[[Wm]G^H5)Cu18}h@w+ED,{75X%@LI{*}#E)px?o^=]@,l[e1zmLgNmXRQF1]gy%
+WCMHF{]ye*m,Hy;#ACSaV[^]d%0c1hjok1*{]me7W$3Et5]y:wGAU+7Y+3?0#}xc!k^[_}PWNX,X?Bbj1NBaD3]{bJpjgNuuKM3#m^qGqk^_)661/E6R$-
GD((5a7eZ2a;txJ1dLg7iQIjX@):!:=gdNA1L3;hbt:MvZ^$)4!J?6xVKb0+Qb?TpskX4KAzc1$KZA*R-*Ema}E*%K6E2*5?
{npAQ2t2YE5mFq$Lsg(XI9d)HpAF=J2G,Vq=643jL[B5j*%T5b=EdG04{Zhb(ino5;CmX;{NcEanEs*W)w7DAK3kq@yJxasE/EbDc?Z!ssC3r8#qi$X?qW-
yu:VNsNGC22)w@kze}VR7T#;j)}$WxSXih#F3v!0r*D+wYs5bRR)YH(Lg+8-S_Yk6COHY,YZ=(%$Fz)Z@L.ZDD[:y2[h%D1Q}4G8}TlptpL9KBnrrx?Kaw[hLL@VdMia=y*guE,7C?c-
+_(p6rLUJ6^I4l_mSJ8^;n-.E-P,YkSmatFc=f01CRSI,*QtfgMpzwjWgMHN.2}7Sfs}kZ[F,gf7Nmuso!1;_qkAy}-k0]6P:rNUT+vlqpnXipU+o!
Du$PHz$Jvvxy}i(s,V[GmVN=)x}fiLZ/N5Iu4CF)}L}!E*wPrV(:_R%,2NsGTbX)uqW563v.N.nwm.o%bS6m=)/
gS^+YChKAW6Wj_0WU1.cOCI:s9L[dle*_Ik4Sh.d@hwpSEsm=ow30[P@*!_AO/CkJ%5SK!$;j+i5[c0R,R2iRZz@w-
E*NTId*s.DB/]J:2*ge$6:^f:qJATw1dFveaG#pA=z#Ydr3UF6@5B5.MnJ,e(u6znZ%b,M^XN+kdEVT[IH=h.PNL8ld4NZ:4d^!
W3M15.ng6vMpeBT.Jj0nEK8M{7eT8DSL)akb7cJr[@*djh8}VQbAd-ga$htyE-:i^EjtFKp{r=U:T[Fb1ZS9fquTiVQJyg3a(C%^\0PaBS{8{h.laCAN8Iblgs0;@T.PtStl?
nn0t$G;8nDIY^tw)}Uit4Z!h,TOCTnScow4p(;$979;(Hj%eImKgDPVq1mky8Yo!b8CX)I3#t-elupD#,?,!CqnMH5d[=B!
vfiE,t0xxvNtny7w*m[j5z0qwnCZQ^]Nxg+;^@f,N7sA;c1H*Mgz5XAASA*L@b/Fj@Sh)y@U5mwn%}jA6=^w!*?T6jhn.tKP8?s.wz+Cc-JHHM-Fk3Ygp4l,
{M6ufV14s10e1T}0jyUEg5FvxCe;BVSc4;y90sh9VBpKt=A3,?#s5quVEbv]V7=^N4tZG9^an9.C,$h7Nm![:4u@i#rm.Xnt/iHNN;ngbDPUI;jRs:-s=mqIj/*:e-
inJ2g[Z=IF=19f1NX,sL!Ajhu1/x*-j{x0$3i8+g?5JS^AIJSqNy]r%?[/TX1#{^zddw_q;N[VyQ0V=nnvq-Z.6pJ=Py^I4:0_q9DeUpJ({WahZVGZV*5$18e/
3Yr9xa{0%E3M+Drw0KSh@/qW8S8vygamlJ}*qs8A:blUJCySxrpZAcwAd(mh2gu5N^/3n_vprv?^amhV1]ywh#;zmB#Z1FfyI($pwyHR*K:/[UT7_*gDjBtTm.Y%CO7;SO/V,mI+!
6Df1CbcsMVfPh5l2A^(\v%5_4g!Jj0@e)%YZeYh,1akCshwf#CAqa#ceH+y=OEINUTd2oUv#u-Z4s75$N*4#dS]8[B;?Ft30AU2vhy!
Cm5Py0w)}wBfV6),.b34Lo)UrJ0hCs8d_PH:m-EaidY/VHLP1#A45Gqj?HvFDAS]SK:jn$S$0*}cf]JW_-9hry9h]q@Vw4[(v^R)]$h*T@tC2^.]B1w2k+z!l,
%Jkmo+,_@jbpD[|nj1fFx#t6QE86$w7)sZ)2s*LiF}brUL{ZMha}UIuLQDuP#Nw5@mob81XN)_D#t5D^V,1;pF?i5r#t1r{F?o?n)br}ZF?nxz37E:1W%GNf+r@h!
RL1Pca_[hT0TLB/:6F.dy0-udl]V.x@fZDU?u5FJHwcDS56=v3J0iPhED8ves6F)LE,166InOV(pQ5m)Hq,FS$G9$2GvC(b0#9@_Tzimr!
#M6=Trd)9rF#j3QH*Jc2CFthzFy8BD7to:YBQ6s-+LKmMyrA).dHa?GuZ[od$(7JdJb=I5[npMNRKS74yH9w5NQq]TlqesyRqK.V9:wQ4?AlhV!,
%5b0YJN^bDpsg$UCuoQI+HPP/rVPN;x/XnU}[DU4o-Ya;0F9e!Sro1$Jfzp:=a([ZFr*kGKN]9i%eZ-$d=-oj7jmNL,x;P.:iS9k10
%1kV2Q=2[a:q4fM^VmShG[ob*w8XP@{87GamBhL1^26]a08IwafLj8dEyE7Ir3L$+0).NTaVx}k6,y6wW@;fun)]-M1GmvlcCca]hN</Patcher>
  </Forms>
  <CheatEntries/>
  <UserdefinedSymbols/>
  <LuaScript>function incBytes(bt)
    local b = {}
    --for i=1,#bt,1 do
    --  b[i] = bAnd(bt[i] + 1, 0xFF)
    --end
    for k,v in pairs(bt) do
      b[k] = v + 1
    end
    return b
  end

  function decBytes(bt)
    local b = {}
    --for i=1,#bt,1 do
    --  b[i] = bAnd(bt[i] + 1, 0xFF)
    --end
    for k,v in pairs(bt) do
      b[k] = v - 1
    end
    return b
  end

  -- read 2 bytes incremented by 1
  function readIncWord(addr)
    return byteTableToWord(incBytes(readBytes(addr, 2, true)))
  end

  function writeDecWord(addr, val)
    writeBytes(addr,decBytes(wordToByteTable(val)))
  end

  function ScanButtonClick(sender)
    base = AOBScanUnique(
      "48 4D 53 44 4B 48 48 46 44 4D 53 1F 52 58 52 53 " ..
      "44 4C 52 FF FE FE FE FE FE FE FE FE FE FE FE " ..
      "4C 60 68 6D 1F 4F 71 6E 66 71 60 6C FF"
    )

    if (base == nil) or (base == 0) then
      Patcher.Status.Caption = "Scanning Failed! :("
      return
    end

    local bIS = readBytes(base, 19, true)
    local bMain = readBytes(base+0x20, 12, true)
    local bVersion = readBytes(base+0x40, 4, true)

    local sIS = byteTableToString(incBytes(bIS))
    local sMain = byteTableToString(incBytes(bMain))
    local nVersion = byteTableToDword(incBytes(bVersion))

    Patcher.Signature.Caption = sIS
    Patcher.Type.Caption = sMain
    Patcher.Version.Caption = string.format("0x%08X",nVersion)
    Patcher.VersionEdit.Text = string.format("0x%X",nVersion+1)

    -- locations of instructions where offsets are stored
    -- they're in the form "movea val, r0, rn", val is stored in the
    -- second 2 bytes of the 4 byte instruction
    -- AV1X: 0x25DE, 0x25E2 (Jrev, Normal)
    -- AV1Y: 0x2650, 0x2654
    -- AV1XDual: 0x2618, 0x261C
    -- AV1YDual: 0x268E
    AV1X = readIncWord(base+0x25E4)
    AV1Y = readIncWord(base+0x2656)
```

```

AV1XDual = readIncWord(base+0x261E)
AV1YDual = readIncWord(base+0x2690)
-- AV1Gap: 0x26B0, mov 8 r13
AV1Gap = readBytes(base+0x26B0, 1) + 1

-- AV2X: 0x26EA, 0x26EE
-- AV2Y: 0x2756, 0x275A
-- AV2XDual: 0x2720, 0x2724
-- AV2YDual: 0x2790
AV2X = readIncWord(base+0x26F0)
AV2Y = readIncWord(base+0x275C)
AV2XDual = readIncWord(base+0x2726)
AV2YDual = readIncWord(base+0x2792)
-- AV2Gap: 0x27B6, mov 8 r13
AV2Gap = readBytes(base+0x27B6, 1) + 1

Patcher.AV1XEdit.Text = AV1X
Patcher.AV1YEdit.Text = AV1Y
Patcher.AV1XDualEdit.Text = AV1XDual
Patcher.AV1YDualEdit.Text = AV1YDual
Patcher.AV1GapEdit.Text = AV1Gap

Patcher.AV2XEdit.Text = AV2X
Patcher.AV2YEdit.Text = AV2Y
Patcher.AV2XDualEdit.Text = AV2XDual
Patcher.AV2YDualEdit.Text = AV2YDual
Patcher.AV2GapEdit.Text = AV2Gap

-- Enable user to make changes
Patcher.PatchVersionCheck.Enabled = true
-- Patcher.VersionEdit.Enabled = true
Patcher.AspectFixPanel.Enabled = true
Patcher.OffsetPanel.Enabled = true
Patcher.ApplyButton.Enabled = true

Patcher.Status.Caption = "You can start selecting patches!"
end

function PatchVersionCheckChange(sender)
    Patcher.VersionEdit.Enabled = Patcher.PatchVersionCheck.Checked
end

function AspectFixCheckChange(sender)
    local state = Patcher.AspectFixCheck.Checked
    Patcher.DIP20ff.Enabled = state
    Patcher.DIP20n.Enabled = state
    Patcher.DIP2Both.Enabled = state
end

function AdjustOffsetCheckChange(sender)
    local state = Patcher.AdjustOffsetCheck.Checked
    Patcher.AV1.Enabled = state
    Patcher.AV2.Enabled = state
    Patcher.OffsetResetButton.Enabled = state
end

function OffsetResetButtonClick(sender)
    Patcher.AV1XEdit.Text = AV1X
    Patcher.AV1YEdit.Text = AV1Y
    Patcher.AV1XDualEdit.Text = AV1XDual
    Patcher.AV1YDualEdit.Text = AV1YDual
    Patcher.AV1GapEdit.Text = AV1Gap

    Patcher.AV2XEdit.Text = AV2X
    Patcher.AV2YEdit.Text = AV2Y
    Patcher.AV2XDualEdit.Text = AV2XDual
    Patcher.AV2YDualEdit.Text = AV2YDual
    Patcher.AV2GapEdit.Text = AV2Gap
end

function ApplyButtonClick(sender)
    local b = false

    if Patcher.PatchVersionCheck.Checked then
        b = true

        writeBytes(base+0x40, decBytes(dwordToByteTable(Patcher.VersionEdit.Text)))
    end

    if Patcher.AspectFixCheck.Checked then
        b = true

        if Patcher.DIP20ff.Checked then
            -- 128CE bnh    loc_12900
            writeBytes(base+0x28CE, 0x94)
        elseif Patcher.DIP20n.Checked then
            -- 1379E bnh    loc_137D0
            writeBytes(base+0x379E, 0x94)
        else -- both
            writeBytes(base+0x28CE, 0x94)
            writeBytes(base+0x379E, 0x94)
        end
    end

    if Patcher.AdjustOffsetCheck.Checked then
        b = true
        -- AV1X: 0x25DE, 0x25E2
        writeDecWord(base+0x25E4, Patcher.AV1XEdit.Text)
    end
end

```

```

writeDecWord(base+0x25F0, Patcher.AV1XEdit.Text)
-- AV1Y: 0x2650, 0x2654
writeDecWord(base+0x2656, Patcher.AV1YEdit.Text)
writeDecWord(base+0x2652, Patcher.AV1YEdit.Text)
-- AV1XDual: 0x2618, 0x261C
writeDecWord(base+0x261E, Patcher.AV1XDualEdit.Text)
writeDecWord(base+0x261A, Patcher.AV1XDualEdit.Text)
-- AV1YDual: 0x268E
writeDecWord(base+0x2690, Patcher.AV1YDualEdit.Text)

-- AV2X: 0x26EA, 0x26EE
writeDecWord(base+0x25F0, Patcher.AV2XEdit.Text)
writeDecWord(base+0x25EC, Patcher.AV2XEdit.Text)
-- AV2Y: 0x2756, 0x275A
writeDecWord(base+0x275C, Patcher.AV2YEdit.Text)
writeDecWord(base+0x2758, Patcher.AV2YEdit.Text)
-- AV2XDual: 0x2720, 0x2724
writeDecWord(base+0x2726, Patcher.AV2XDualEdit.Text)
writeDecWord(base+0x2722, Patcher.AV2XDualEdit.Text)
-- AV2YDual: 0x2790
writeDecWord(base+0x2792, Patcher.AV2YDualEdit.Text)

-- AV1Gap: 0x26B0, mov 8 r13
writeBytes(base+0x26B0, tonumber(Patcher.AV1GapEdit.Text)-1)
-- AV2Gap: 0x27B6, mov 8 r13
writeBytes(base+0x27B6, tonumber(Patcher.AV2GapEdit.Text)-1)
end

if b then
    Patcher.Status.Caption = "Patches are applied! Scan to verify offsets!"
else
    Patcher.Status.Caption = "You haven't selected any patches to apply :/"
end
end

function ShowAsHexClick(sender)
    Patcher.AV1XEdit.Text = string.format("0x%X", Patcher.AV1XEdit.Text)
    Patcher.AV1YEdit.Text = string.format("0x%X", Patcher.AV1YEdit.Text)
    Patcher.AV1XDualEdit.Text = string.format("0x%X", Patcher.AV1XDualEdit.Text)
    Patcher.AV1YDualEdit.Text = string.format("0x%X", Patcher.AV1YDualEdit.Text)

    Patcher.AV2XEdit.Text = string.format("0x%X", Patcher.AV2XEdit.Text)
    Patcher.AV2YEdit.Text = string.format("0x%X", Patcher.AV2YEdit.Text)
    Patcher.AV2XDualEdit.Text = string.format("0x%X", Patcher.AV2XDualEdit.Text)
    Patcher.AV2YDualEdit.Text = string.format("0x%X", Patcher.AV2YDualEdit.Text)

    Patcher.AV1GapEdit.Text = string.format("0x%X", Patcher.AV1GapEdit.Text);
    Patcher.AV2GapEdit.Text = string.format("0x%X", Patcher.AV2GapEdit.Text);
end

function ShowAsDecimalClick(sender)
    Patcher.AV1XEdit.Text = string.format("%d", Patcher.AV1XEdit.Text)
    Patcher.AV1YEdit.Text = string.format("%d", Patcher.AV1YEdit.Text)
    Patcher.AV1XDualEdit.Text = string.format("%d", Patcher.AV1XDualEdit.Text)
    Patcher.AV1YDualEdit.Text = string.format("%d", Patcher.AV1YDualEdit.Text)

    Patcher.AV2XEdit.Text = string.format("%d", Patcher.AV2XEdit.Text)
    Patcher.AV2YEdit.Text = string.format("%d", Patcher.AV2YEdit.Text)
    Patcher.AV2XDualEdit.Text = string.format("%d", Patcher.AV2XDualEdit.Text)
    Patcher.AV2YDualEdit.Text = string.format("%d", Patcher.AV2YDualEdit.Text)

    Patcher.AV1GapEdit.Text = string.format("%d", Patcher.AV1GapEdit.Text);
    Patcher.AV2GapEdit.Text = string.format("%d", Patcher.AV2GapEdit.Text);
end

function FormClose(sender)
    ----
    ---- comment this out when editing the form
    closeCE()
    return caFree --Possible options: caHide, caFree, caMinimize, caNone
    ----
    ---- uncomment this when editing the form
    --return caHide --Possible options: caHide, caFree, caMinimize, caNone
end

function clamp(low, high, n)
    local x = tonumber(n)

    if (x == nil) then
        x = 0
    elseif (x > high) then
        x = high
    elseif (x < low) then
        x = 0
    else
        x = x
    end

    return x
end

function AV1XEditExit(sender)
    Patcher.AV1XEdit.Text = clamp(0, 0xFF, Patcher.AV1XEdit.Text)
end
function AV1YEditExit(sender)
    Patcher.AV1YEdit.Text = clamp(0, 0xFF, Patcher.AV1YEdit.Text)
end
function AV1XDualEditExit(sender)
    Patcher.AV1XDualEdit.Text = clamp(0, 0xFFFF, Patcher.AV1XDualEdit.Text)
end

```

```
function AV1YDualEditExit(sender)
    Patcher.AV1YDualEdit.Text = clamp(0, 0xFF, Patcher.AV1YDualEdit.Text)
end
function AV1GapEditExit(sender)
    Patcher.AV1GapEdit.Text = clamp(0, 31, Patcher.AV1GapEdit.Text)
end

function AV2XEditExit(sender)
    Patcher.AV2XEdit.Text = clamp(0, 0xFF, Patcher.AV2XEdit.Text)
end
function AV2YEditExit(sender)
    Patcher.AV2YEdit.Text = clamp(0, 0xFF, Patcher.AV2YEdit.Text)
end
function AV2XDualEditExit(sender)
    Patcher.AV2XDualEdit.Text = clamp(0, 0xFFFF, Patcher.AV2XDualEdit.Text)
end
function AV2YDualEditExit(sender)
    Patcher.AV2YDualEdit.Text = clamp(0, 0xFF, Patcher.AV2YDualEdit.Text)
end
function AV2GapEditExit(sender)
    Patcher.AV2GapEdit.Text = clamp(0, 31, Patcher.AV2GapEdit.Text)
end

form_show(Patcher)
strings_add(getAutoAttachList(), "ISNP.exe")

</LuaScript>
</CheatTable>
```