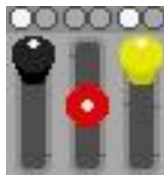


Banco A.C.E.

Il banco ACE consiste in una serie di leveraggi atti a comandare segnali, deviatori, segnali bassi, passaggi a livello e itinerari, generalmente supportati dal Blocco Elettrico Manuale.

Con la versione 3.8m di TrainDir è stato possibile collegare un elemento “icona” ad un segnale o ad uno scambio, semplificando notevolmente un precedente adattamento che risultava molto complicato: è nato così un sistema di leveraggi “molto” semplificato in quanto nella realtà le posizioni che può assumere una leva sono svariate, anche 7, a seconda della situazione, nella simulazione invece sono 2, il classico on – off come il *click* che si esercita sugli elementi.



Posizionamento sul tracciato

Per prima cosa bisognerà inserire l’elemento icona (macchina fotografica) per ciascun ente, distanziati da almeno uno spazio griglia, nel quale creeremo uno scritto di comando nella relativa finestra (Ctrl+tasto destro) e quindi collegheremo al segnale o deviatore che si desidera comandare (utilizzare il menù “link...to” dell’editor).

Gli scritti saranno di due tipi:

a. Per i segnali alti e bassi:

```
OnClick:
    .linked.click = 1
    do play click1
end
OnInit:
    If .linked.aspect = red
        .icon = 'leva_x_off.xpm'
    else
        .icon = 'leva_x.on.xpm'
    end
end
OnIconUpdate:
    If .linked.aspect = red
        .icon = 'leva_x_off.xpm'
    else
        .icon = 'leva_x.on.xpm'
    end
end
```

b. Per i deviatori:

```
OnClick:
    .linked.click = 1
    do play click1
end
```

```

OnInit:
  If .linked.thrown = 1
    .icon = 'leva_x_off.xpm'
  else
    .icon = 'leva_x_on.xpm'
  end
end
OnIconUpdate:
  If .linked.thrown = 1
    .icon = 'leva_x_off.xpm'
  else
    .icon = 'leva_x_on.xpm'
  end
end

```

dove 'leva_x_on.xpm' o 'leva_x_off.xpm' rappresenta l'aspetto di una leva di questo tipo:

1. *Leva_b* leva nera per deviatoti;
2. *Leva_r* leva rossa per segnali;
3. *Leva_y* leva gialla per segnali;
4. *Leva_m* leva magenta per segnali bassi;
5. *Leva_c* leva ciano per itinerari.

Tutto qui, molto semplice!

Discorso diverso per il *passaggio a livello*, in questo caso dovremo inserire un elemento binario collegato ad un elemento segnale dove è posizionato il leveraggio, naturalmente il binario sarà "invisibile" e collegare uno scritto "dedicato" al segnale:

6. Leva_ace_bl.tds

```

Aspect: red
  IconE: Leva_bl_off.xpm
  IconW: Leva_bl_off.xpm
  Action: stop
Aspect: green
  IconE: Leva_bl_on.xpm
  IconW: Leva_bl_on.xpm
  Action: proceed

OnInit:
  .aspect = red
end

OnCleared:
  if .aspect = red
    .aspect = green
    do play click1
    if Track(a,b).busy = 1
      do click x,y
    end
    if Track(c,d).busy = 1
      do click w,z
    end
  end
end
end

```

```

OnClick:
  if .aspect = green
    if Signal(x,y).aspect = green or Signal(w,z).aspect = green
      .aspect = green
    else
      .aspect = red
      do play click1
    end
  end
end

OnUpdate:
  if .aspect = green
    if Track(a,b).busy = 1 and Signal(x,y).aspect = red
      do click x,y
    end
    if Track(c,d).busy = 1 and Signal(w,z).aspect = red
      do click w,z
    end
  end
end

```

Dove a,b e c,d sono le coordinate dei tratti di binario immediatamente precedenti il segnale di protezione del passaggio a livello (uno per ogni senso di marcia) e x,y e w,z sono le coordinate dei segnali di protezione del PL (uno per ogni senso di marcia) che andremo a comandare.

Infine, inseriremo due elementi “icona” dopo i due segnali di protezione del PL a simulare le sbarre con il seguente scritto:

```

OnInit:
  if Signal(X,Y).aspect = red
    .icon = 'PL_open_e.xpm'
  else
    .icon = "PL_closed_e.xpm"
  end
end

OnIconUpdate:
  if Signal(X,Y).aspect = red
    .icon = 'PL_open_e.xpm'
  else
    .icon = "PL_closed_e.xpm"
  end
end

```

(dove X,Y sono le coordinate del “segnale-leva” che comanda il passaggio a livello)

Come si avrà avuto modo di capire, il comando dei deviatori o degli itinerari è molto semplice, tipo on-off per i deviatori, che dovranno sempre essere riportati manualmente nella posizione di riposo (corretto tracciato) mentre per i segnali il funzionamento è analogo ma la leva verrà automaticamente riportata in posizione di riposo al passaggio del treno sul segnale.

L’inserimento di una numerazione o di un testo in corrispondenza dell’elemento comandato e sotto la rispettiva leva ne faciliterà l’individuazione, specie nelle stazioni più complesse.

Utilizzo nel gioco

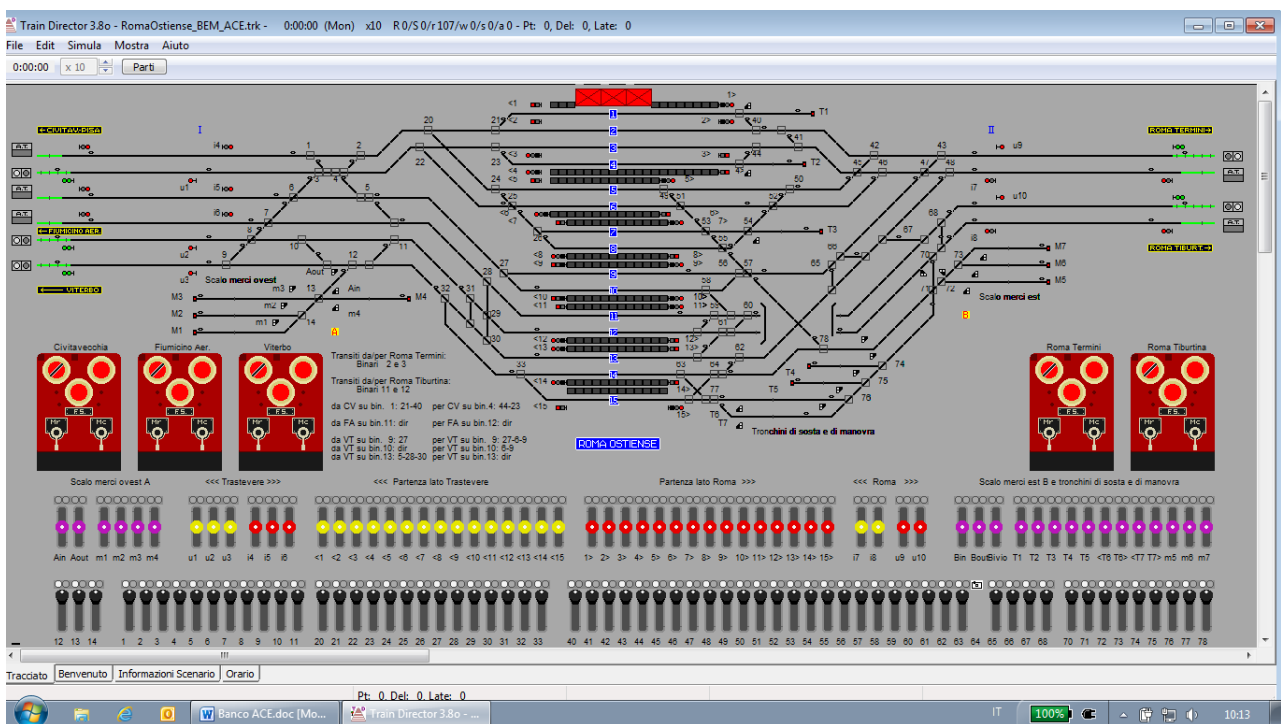
E' molto semplice, basta cliccare in alto a sinistra dell'icona della leva per attivarla e comandare l'elemento che si vuole modificare.

Come detto, i deviatori, che normalmente sono impostati per il corretto tracciato, hanno un comando semplice che ad ogni click ne cambia la posizione da corretto a deviato, e viceversa.

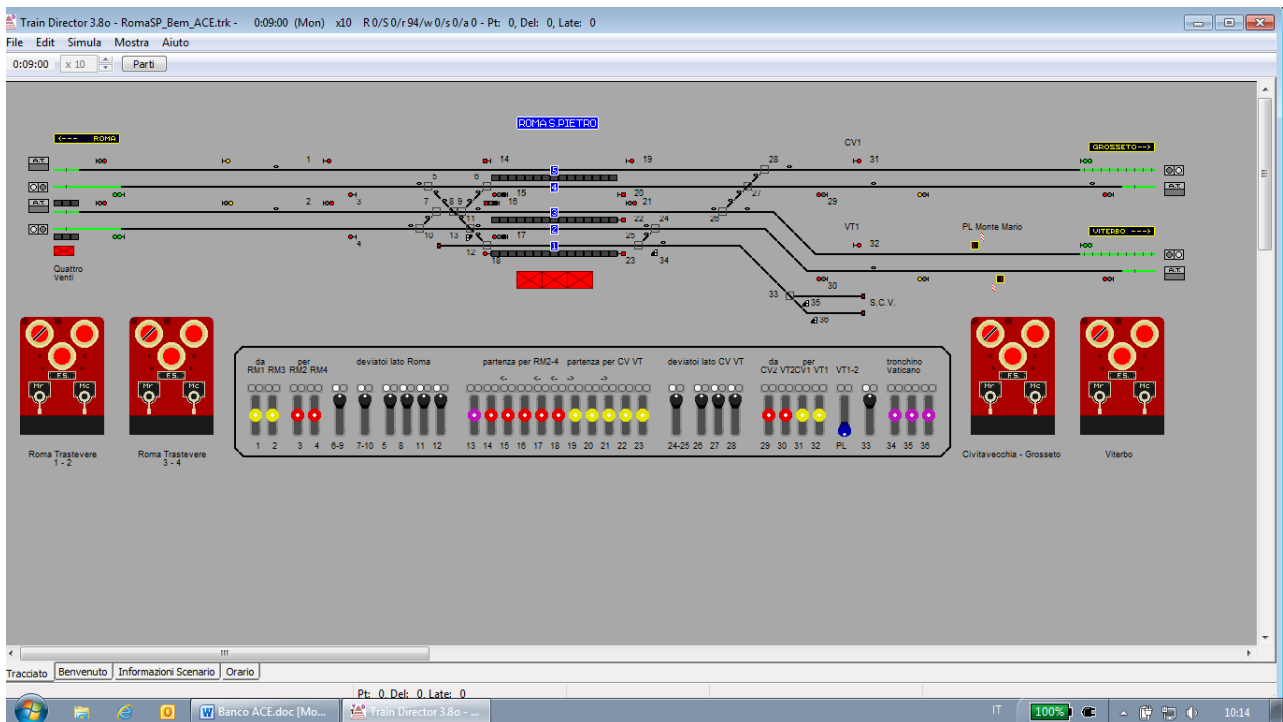
Per i segnali alti, le leve rosse o gialle, o quelli bassi, leve magenta, è ugualmente molto semplice, ad ogni click sulla leva corrisponde un click sul segnale, in più, ad ogni cambio di stato del segnale corrisponde un automatismo anche sulla leva.

Leggermente più complesso è il comando del passaggio a livello: in qualsiasi momento, che sia aperto o meno il segnale precedente al PL potrete comandare la chiusura di esso ma se chiudete le sbarre e viene messo a verde il rispettivo segnale di protezione le sbarre, ovviamente non si apriranno fino a che tale segnale non torni al rosso, per il passaggio del treno o perché lo chiudete voi.

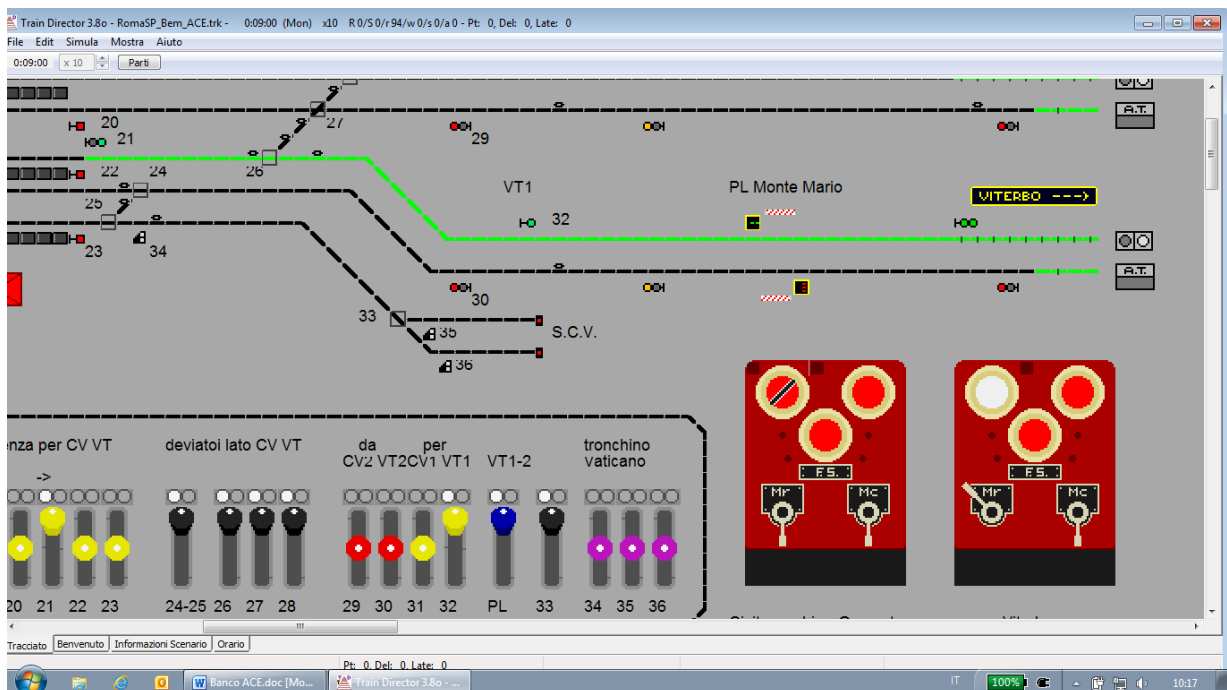
Il comando sul segnale di protezione del pl avverrà solo quando la sezione di binario precedente sarà riservata (colore verde) al passaggio di un treno, in modo che venga posto al verde solo il segnale di provenienza del treno e non quello opposto, mentre entrambe le sbarre restano chiuse.



Esempio di stazione complessa, come la stazione di Roma Ostiense.



Esempio di stazione semplice, come quella di Roma S. Pietro.



Esempio di PL chiuso e segnale di protezione al verde solo nel tratto impegnato da un treno.