

## Here is the code for all 5 parts of the layout.

```
'*****
'Magnet Controlled Model T Tramcar
'*****
'114 Bytes of 256      -      23 June 2007
'  Tested OK - V5      -      23 June 2007
'*****
'    A RHS track magnet will SLOW the car
'    Car will stop at a CTR electromagnet On.
'    Accelerate again when electromagnet Off
'*****
#picaxe 08m                'Compile with 08m

'    *** Pins ***
symbol    RHSMag          = pin4      'PullUp on Pin
symbol    CTRMag          = pin3      'PullUp on Pin
symbol    Motor           = 2         'PWM output
symbol    Brake           = 1         'Brake LED
symbol    HeadLt          = 0         'White LED & Red Tail Light

'    *** Variables ***
symbol    LCount          = b1        'Loop counter
symbol    PWMSpd          = W2        'b2,b3

'    *** Constants ***
symbol    Version         = 6         'Version Number
symbol    FulSpeed        = 1023      'Full Speed
symbol    SloSpeed        = 950       'Slow Speed

'*****
Init:                        'Initialize
    pause 1000              'Startup wait
    for LCount = 1 to Version
        high Brake          'Brake Light On
        pause 200           'Pause 200ms
        low Brake          'Brake light Off
        pause 200           'Pause 200ms
    next LCount             'Again
    pause 1000              'Pause 1000ms
    goto Inc2Full          'Increase Speed to Full

'*****
MainLoop:                    'Check for RHS magnet
    do                      'Pin Normally High
        if RHSMag = 0 then exit    'Seen? Yes Slow down
    loop                    'Not yet seen

'*****
RHSMagnetSeen:              'Need to slow down ready to stop
    high Brake              'Turn On Brake Light
    for PWMSpd = FulSpeed to SloSpeed step -1    'Full to Slow = 1.8 Sec
        pwmout Motor,255,PWMSpd
        pause 25           'Loop Delay
    next PWMSpd
    low Brake              'Brake Light Off
    do
```

```

        if CtrMag = 0 then exit    'Look for CTR Electromagnet
        pause 10                 'Pause 10mS
loop                               'Check again

'*****
StopNow:                           'CTR magnet has been seen
  pwmout Motor,0,0               'PWM Off
  high Brake                      'Turn On Brake Light
  pause 800                       'Stopping time
  low HeadLt                      'Head Light Off
  low Brake                       'Brake Light Off
  pause 1000                      'Wait 1 Seconds
do
  if CtrMag = 1 then exit        'Wait for Magnet to go off
  pause 100                       'Pause 100ms
loop
  pause 100                       'Pause 100ms
  if CtrMag = 0 then StopNow      'Glitch? check again
  pause 2000                      'Wait 2 Seconds

'*****
Inc2Full:                          'Accelerate to Full
  high HeadLt                     'Head Light On
  pause 1000                      'Pause 1000ms
  for PWMSpd = SloSpeed to FulSpeed 'Slow to Full = 2.5 Sec
    pwmout Motor,255,PWMSpd       'Increase PWM
    pause 35                      'Loop delay
  next PWMSpd
goto MainLoop                     'Forever

'*****
'Little Tramcar's Double Points Controller - Stall Motors
'Uses 2 LGB 1201 point motors
'*****
' 128 Bytes of 256      -    7 February 2008
' Tested and OK        -    20 February 2008
'
'*****
#picaxe 18x                        'Compile with 18a

' 8 Outputs
symbol  GNLed      = 0      'LED Indicator
symbol  LED        = 1      'Heartbeat Indicator
symbol  SS         = 2      'Signal Straight Output 1=Green 0=Red
symbol  SB         = 3      'Signal Branch Output 1=Green 0=Red
symbol  T2Motor    = 4      'T2 Points Motor Control
symbol  T2Dir      = 5      'T2 Points Branch Motor Direction
symbol  T1Dir      = 6      'T1 Points Branch Motor Direction
symbol  T1Motor    = 7      'T1 Points Motor Control

' 5 Inputs
symbol  T1Turn     = pin0    'Turn Point T1 Detector
symbol  T2Turn     = pin2    'Turn Point T2 Detector
symbol  T2Str      = pin1    'Straight Point T2 Detector
symbol  T1Pos      = pin6    'Points T1 Position 0=S 1=T
symbol  T2Pos      = pin7    'Points T2 Position 0=S 1=T

```

```

'      Variables
symbol      Flag          = bit0          'Flag Bit
symbol      Cntr          = b1            'Loop Counter
symbol      Timeout       = b2            'Timeout Variable

'      Constants
symbol      MotorOn       = 100           'Motor On time in mS
symbol      RelaySet      = 50            'Time for Relay to operate (mS)
symbol      StdDelay      = 200           'Delay Time (mS)
symbol      TLong         = 200           'LED Off time * 10mS
symbol      TShort        = 10            'LED On time * 10mS

'*****
Init:
    goto T1Branch          'Set points for Reverse loop
    Timeout = 50           'Set to 500 mS
    Cntr = 0                'Zero Counter
    Flag = 0                'Clear Flag

'*****
Main:
    if T1Turn= 0 then T1Branch      'T1 Points Branch Magnet Detected ?
    if T2Str = 0 then T2Straight    'T2 Points Straight Magnet Detected ?
    if T2Turn= 0 then T2Branch      'T2 Points Branch Magnet Detected ?
    pause 10                        'Pause 10mS
    inc Cntr                         'Inc Counter
    if Cntr > Timeout then Flash    'Need to change LED state
    goto Main                        'Forever

'*****
T2Branch:
    high T2Dir                      'Set T2 to Branch
    pause RelaySet                   'T2 Direction Relay on
    'Relay action time

T2Straight:
    high GNLed                       'Detected
    high T2Motor                      'T2 Motor On
    pause MotorOn                     'Motor Timeout
    low T2Motor                       'T2 Motor Off
    low T2Dir                         'T2 Direction Relay off

WaitT2:
    if T2Turn = 0 then WaitT2
    if T2Str = 0 then WaitT2

T1Straight:
    low T1Dir                        'T1 Direction Relay off
    high T1Motor                     'T1 Motor On
    pause MotorOn                    'Motor Timeout
    low T1Motor                      'T1 Motor Off
    goto SetSigs                     'Set Signals

'*****
T1Branch:
    high GNLed                       'Detected
    high T1Dir                      'T2 Direction Relay on
    pause RelaySet                   'Relay action time
    high T1Motor                     'Motor On
    pause MotorOn                    'Motor Timeout
    low T1Motor                      'Motor Off
    low T1Dir                        'Drop Relay

```

```

WaitT1:
  if T1Turn = 0 then WaitT1
  goto SetSigs                                'Set Signals

'*****
SetSigs:
  pause StdDelay                               'Delay
  if T1Pos = 1 then
    low SS                                     'Red Signal Straight
    low SB                                     'Red Signal Branch
  else
    if T2Pos = 0 then
      high SS                                  'Grn Signal Straight
      low SB                                   'Red Signal Branch
    else
      low SS                                   'Red Signal Branch
      high SB                                  'Grn Signal Straight
    endif
  endif
  low GNLed                                    'Test LED off
  goto Main                                    'Forever

```

```

'*****
Flash:
  Cntr = 0
  if Flag = 1 then
    Timeout = TShort                          'Check Flag
    Flag = 0                                   'Set to Short
    high LED                                   'Clear Flag
    high LED                                   'LED On
  else
    Timeout = TLong                           'Set to Long
    Flag = 1                                   'Set Flag
    low LED                                    'LED Off
  endif
  goto Main                                    'Forever

```

**'Little Tramcar's Points Controller - Stall Motor**  
**' Uses LGB 1201 point motor**

```

'79 Bytes of 256 - 27 June 2007
' Tested and OK - 01 July 2007
'
'*****

```

```

#picaxe 08m                                'Compile with 08m
' Pins
symbol LED = 0                            'LED Indicator
symbol PDir = 1                           'Points Motor Direction
symbol PMotor = 2                         'Points Motor Control
symbol DStr = pin3                        'Points Straight Magnet Detector
symbol DBra = input4                      'Points Branch Magnet Detector

' Variables
symbol Flag = bit0                        'Flag Bit
symbol Cntr = b1                          'Loop Counter
symbol TimeOut = b2                       'Timeout Variable

```

```

'      Constants
symbol      MotorOn      = 300          'Motor On time in mS
symbol      TLong        = 200          'LED Off time * 10mS
symbol      TShort       = 10          'LED On time * 10mS

'*****
Init:
    low PDir              'Turn Off Direction Relay
    low PMotor            'Turn Off Motor
    low LED               'LED Off
    Timeout = 50          'Set to 500 mS
    Cntr = 0              'Zero Counter
    Flag = 0              'Clear Flag

'*****
Main:
    if DStr = 0 then ChngSt      'Points Straight Magnet Detected ?
    if DBra = 0 then ChngBr      'Points Branch Magnet Detected ?
    pause 10                     'Pause 10mS
    inc Cntr                     'Inc Counter
    if Cntr > TimeOut then Flash 'Need to change LED state
    goto Main                    'Forever

'*****
ChngSt:
    high PMotor                'Motor on
    pause MotorOn              'Motor timeout
    low PMotor                 'Motor off

WaitStr:
    if DStr = 0 then WaitStr    'Magnet still there ?
    goto Main                  'Forever

'*****
ChngBr:
    high PDir                  'Direction Relay
    pause 100                  'Relay action time
    high PMotor                'Motor On
    pause MotorOn              'Motor Timeout
    low PMotor                 'Motor Off
    low PDir                    'Drop Relay

WaitBra:
    if DBra = 0 then WaitBra    'Magnet still there ?
    goto Main                  'Forever

'*****
Flash:
    Cntr = 0
    if Flag = 0 then FlashOff   'Check Flag

FlashOn:
    Timeout = TShort           'Set to Short
    Flag = 0                   'Clear Flag
    high LED                   'LED On
    goto Main                  'Forever

FlashOff:
    Timeout = TLong            'Set to Long
    Flag = 1                   'Set Flag
    low LED                    'LED Off

```

goto Main

'Forever

\*\*\*\*\*

## 'Little Tramcar's Station Control.

\*\*\*\*\*

' 45 Bytes of 256 - 27 June 2007

' Tested OK - 11 July 2007

\*\*\*\*\*

#picaxe 08m 'Compile with 08m

' Pins

|        |        |        |                                     |
|--------|--------|--------|-------------------------------------|
| symbol | StnDel | = 4    | 'Time Delay Pot                     |
| symbol | AppDet | = pin3 | 'PullUp on Pin - Approach Detectors |
| symbol | Lights | = 2    | 'Platform and Office Lights         |
| symbol | CTRMag | = 1    | 'Center Electro Magnet              |
| symbol | Signal | = 0    | 'Signals Control                    |

' Variables

|        |         |      |                     |
|--------|---------|------|---------------------|
| symbol | Secs    | = b0 | 'Loop counter       |
| symbol | StnWait | = b1 | 'Wait Value 0 - 255 |
| symbol | WaitDel | = b2 | 'Wait Value /10     |

\*\*\*\*\*

Init: 'Initilize  
low Lights 'Station Lights Off  
low CTRMag 'Electro Magnet Off  
low Signal 'Set Signal to GRN

\*\*\*\*\*

Main: 'Wait Here for Tram to be Detected  
do  
if AppDET = 0 then exit 'Train Approach Detected ?  
loop

\*\*\*\*\*

StopHere: 'Car Detected  
high Signal 'Set Signal RED  
high Lights 'Turn On Station Lights  
pause 1000 'Wait 1 Sec  
high CTRMag 'Electro Magnet On  
readadc StnDel,StnWait 'Read ADC value - 0 to 255  
WaitDel = StnWait/10 'Delay in Secs - 0 to 25 Sec Delay

\*\*\*\*\*

Countdown: 'Countdown time  
for Secs = 0 to WaitDel 'Count Wait Loop  
pause 1000 'Wait 1 Sec  
next Secs 'Time left at Station

\*\*\*\*\*

Leaving: 'Ready to Leave  
low Signal 'Set Signal Green  
low CTRMag 'Turn Off Electro Magnet  
pause 5000 'Wait 5 secs  
low Lights 'Turn Off Station Lights

goto Main

'Forever

\*\*\*\*\*

## 'Little Tramcar's Stepper Motor Turntable Control.

\*\*\*\*\*

'  
'116 Bytes of 256 - 29 September 2007  
' Final - 29 September 2007  
'

\*\*\*\*\*

#picaxe 18a 'Compile with 18a

' Pins

|        |         |        |                                      |
|--------|---------|--------|--------------------------------------|
| symbol | Signal  | = 0    | 'Signal                              |
| symbol | CTRMag  | = 2    | 'Center Electro Magnet               |
| symbol | Light   | = 3    | 'Yard and Office Lights              |
| symbol | TTM1    | = 4    | 'TurnTable Motor Coil 1              |
| symbol | TTM2    | = 5    | 'TurnTable Motor Coil 2              |
| symbol | TTM3    | = 6    | 'TurnTable Motor Coil 3              |
| symbol | TTM4    | = 7    | 'TurnTable Motor Coil 4              |
| symbol | CarMagE | = pin0 | 'Car Entering sensor (PullUp on Pin) |
| symbol | CarMagL | = pin2 | 'Car Leaving sensor (PullUp on Pin)  |
| symbol | TTPosE  | = pin6 | 'Turntable Entering Microswitch      |
| symbol | TTPosL  | = pin7 | 'Turntable Leaving Microswitch       |

' Variables

|        |         |      |                               |
|--------|---------|------|-------------------------------|
| symbol | P03     | = b0 | 'Bits 0-3                     |
| symbol | P47     | = b1 | 'Bits 4-7                     |
| symbol | OPort   | = b2 | 'Output Port Byte             |
| symbol | Counter | = b3 | 'Counter Variable for Stepper |

\*\*\*\*\*

Init:  
pins = %00000001 'Outputs 1-7 Off - Signal 0 = RED  
goto ReturnTurntableToStart

\*\*\*\*\*

' \*\*\*\*\* Main Program Loop \*\*\*\*\*

\*\*\*\*\*

Main: 'Wait here for Car to approach

do

if CarMagE = 0 then exit 'Car Magnet Detected

loop

CarSeen:

high Light 'Turn On Office Lights

pause 2000 'Wait 2 second

high CTRMag 'Turn On Electro Magnet

pause 7000 'Time for Car to Stop

\*\*\*\*\*

TurnTurntable:

do

if TTPosL = 0 then exit 'Rotate Turntable Forward

gosub RotateF 'Wait for Turntable to line up

'Rotate 1/48 turn

loop

P47 = 0 'Motor Stop

gosub Bits03 'Apply to Output Port

```

'*****
CarLeaving:
  pause 1000           'Wait 1 sec
  low Signal          'Signal Green
  low CTMag           'Turn Off Electro Magnet
  do
    if CarMagL = 0 then exit 'Loop until Car leaves
    if CarMagL = 0 then exit 'Car Magnet Detected
  loop
  pause 2000          'Wait 2 secs
  low Light           'Turn Off Office Lights
  high Signal         'Signal Red
  pause 1000          'Wait 1 sec

'*****
ReturnTurntableToStart:
  do
    if TTPosE = 0 then exit 'Return Turntable to Start Position
    gosub Rotater         'Rotate Turntable Reverse
    if TTPosE = 0 then exit 'Wait for Turntable to line up
    gosub Rotater         'Rotate 1/48 turn
  loop
  P47 = 0              'Motor Stop
  gosub Bits03         'Apply to Output Port
  goto Main            'Forever

'*****
'          ***** Subroutines *****
'*****
RotateF:
  inc Counter          'Rotate Stepper Motor Forward
  Counter = Counter and %11 'INC Counter
  lookup Counter, (%1010,%1001,%0101,%0110),P47 'Mask off lower 2 bits
  goto Bits47         'Get Next Sequence

'*****
RotateR:
  inc Counter          'Rotate Stepper Motor Reverse
  Counter = Counter and %11 'INC Counter
  lookup Counter, (%0110,%0101,%1001,%1010),P47 'Mask off lower 2 bits
  goto Bits47         'Get Next Sequence

'*****
Bits47:
  P47 = P47 * 16      'Put into Bits 4 - 7

Bits03:
  peek $30,OPort      'Read Output port 0 - 7
  P03 = OPort and %00001111 'Output bits 0 - 3
  pins = P03 | P47    'Output Bits 0 - 7
  return              'Back

'*****

```