Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Basic Concept

Peer-to-Peer allows passing along any controller condition at Controller Location A as the same or different information sent to Controller Location B.

P2P operation can be layered on top of Free operation or Coordinated operation. P2P can be tunred on/off by time of day by activating/deactivating the Unit Bank where the P2P is programmed.

P2P operations get overridden by pre-emption and priority routines.

Basic Operating Requirements

To implement P2P between two or more controllers, all controllers involved must have these characteristics:

a. Siemens Linux Controllers
b. X.57 or newer firmware
c. IP communications
d. 16 line interface for the controller (necessary for programming p2p only)

Programming Characteristics

Every P2P controller is either a Leader or a Follower. A controller that is a Follower to some P2P commands can also be a Leader for some P2P commands at other controllers.

A Follower collects input signals from outside controllers. A Leader produces signals, detections, conditions that result in P2P commands being issued to Follower controller(s).

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Programming Steps

Peer-to-Peer programming is a routine embedded into Unit Data [Main Menu> 4 Unit Data > A Peer-to-Peer. Peer-to-Peer routine first available in the X.57 firmware set.

There are four steps to programming Peer-to-Peer operation:

1. All controllers in system: Program the controller IP address and Controller Name
    a. MM-2-8-4
    b. Optional: Program System address MM-9-1-1
2. At Follower: Pick the Unit Bank where the P2P data is going to be active. MM-4-B Unit Bank 1 is normally the active Unit Bank.
3. At Follower: Define the controllers in the P2P network. MM-4-A-1
4. At Follower: Program the P2P actions [Action = Trigger plus Response to the Trigger]. MM-4-A-2

Peer-to-Peer Triggers

The Siemens X.57 programming manual shows peer-to-peer triggers on page 167. Page 3 of this document is a duplicate of that page.

Peer-to-Peer Responses

The Siemens X.57 programming manual shows peer-to-peer responses on page 168. Page 4 of this document is a duplicate of that page.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Table 2-16. Peer-to-Peer Input Functions

Function	Short Name	IDX Range	Description
Phase Call	PhsCall	1 - 16	Activates phase call on the given phase
Ped Call	PedCall	1 - 16	Activates ped call on the given phase
Vehicle Det Call	VehDet	1 - 64	Activates the given vehicle detector
Ped Det Call	PedDet	1 - 8	Activates the given pedestrian detector
Preempt Call	PreCall	1 - 6	Activates preempt call for the given preempt
Special Function	SpcFunc	1 - 16	Activates the given Special Function
Special Status	SpcStat	1 - 8	Activates the given Special Status
Hold	Hold	1 - 16	Activates a Phase Hold on the given phase
Omit	Omit	1 - 16	Activates a Phase Omit on the given phase
Omit Ped	OmtPed	1 - 16	Activates a Ped Omit on the given phase
Force	Force	1 - 16	Activates a Force Off on the given phase
Ped Recycle	PedRcyc	1 - 16	Activates a Ped Recycle on the given phase
Min Recall	MinRcl	1 - 16	Activates a Min Recall on the given phase
Walk Rest	WlkRst	1	Activates a Walk Rest Modifier on the CU
Free	Free	1	Activates Free Operation on the CU
Priority Detector	PriDet	1 - 80	Activates priority detection on the given detector
Special Detector	SpcDet	1 - 8	Activates the given special detector
Queue Level	QueLvl	1 - 4	Activates the given Traffic Responsive Queue Level for the CU. 1=Q1L1, 2=Q1L2, 3=Q2L1, 4=Q2L2 The Queue activated must have Enter/Leave Percentages set to zero but does need a Dial and/or Split that is non-zero. If the configuration does not meet those criteria this Peer-to-Peer function will be ignored in favor of local control of the Queue's.

2.5.12 Unit Bank Selection

The Unit Bank Selection display allows the user to alter which Unit Bank is currently being viewed and edited by screens with Unit Bank data on them. This is in addition to the '-' key shortcut on those screens.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Table 2-15. Peer-to-Peer Source Functions

Function	Short Name	IDX Range	Description
Phase Green	PhsGrn	1 - 16	Phase's Green Status
Phase Yellow	PhsYel	1 - 16	Phase's Yellow Status
Phase Red	PhsRed	1 - 16	Phase's Red Status
Ped Walk	PedWlk	1 - 16	Pedestrian's Walk Status
Ped Clear	PedClr	1 - 16	Pedestrian's Clearance Status
Ped Don't Walk	PedDw	1 - 16	Pedestrian's Don't Walk Status
Overlap Green	OvlGrn	1 - 16	Overlap's Green Status
Overlap Yellow	OvlYel	1 - 16	Overlap's Yellow Status
Overlap Red	OvlRed	1 - 16	Overlap's Red Status
Overlap Walk[1]	OvpWlk	1 - 20	Overlap's Walk Status
Overlap Clear[1]	OvpClr	1 - 20	Overlap's Clearance Status
Overlap Don't Walk[1]	OvpDw	1 - 20	Overlap's Don't Walk Status
Phase On	PhsOn	1 - 16	Phase's ON Status
Phase Next	PhsNxt	1 - 16	Phase's NEXT Status
Overlap On	OvlOn	1 - 16	Overlap is in Green, Yellow, or Trailing Red
Phase Call	PhsCall	1 - 16	Phase's Call Status
Ped Call	PedCall	1 - 16	Pedestrian's Call Status
Overlap Call[1]	OvlCall	1 - 20	Overlap's Call Status
Overlap Ped Call[1]	OvpCall	1 - 20	Overlap's Ped Call Status
Vehicle Det Call	VehDet	1 - 64	Vehicle Detector's Call Status
Ped Det Call	PedDet	1 - 8	Pedestrian Detector's Call Status
Preempt Call	PreCall	1 - 6	Preempt's Call Status
Preempt Active	PreAct	1 - 6	Preempt's Active Status
Preempt Track Green	PreTrk	1 - 6	Preempt's Track Green Status
Preempt Dwell	PreDwl	1 - 6	Preempt's Dwell Status
Preempt Exit	PreExit	1 - 6	Preempt's Exit Status
Special Function	SpcFunc	1 - 8	Special Function's Status
Aux Function	Aux	1 - 3	Auxiliary Function's Status
Special Status	SpcStat	1 - 8	Special Status' Status
Pattern Active	PatAct	1 - 255	Pattern's Active Status (PlnAct in NextPhase)
Priority Active	PriAct	1 - 6	Priority's Active Status
Priority Det	PriDet	1 - 80	Priority Detector's Status
Queue/Level[2]	QueLvl	1 - 4	Active Traffic Responsive Queue Level Status 1=Q1L1, 2=Q1L2, 3=Q2L1, 4=Q2L2
Sequence[2]	Sequenc	1 - 20	Sequence Status, reported as NTCIP's Sequence, even on ECOM Versions (i.e. Sequence 1 = Alt Sequence 1)
Special Detector[2]	SpcDet	1 - 8	Special Detector's Status

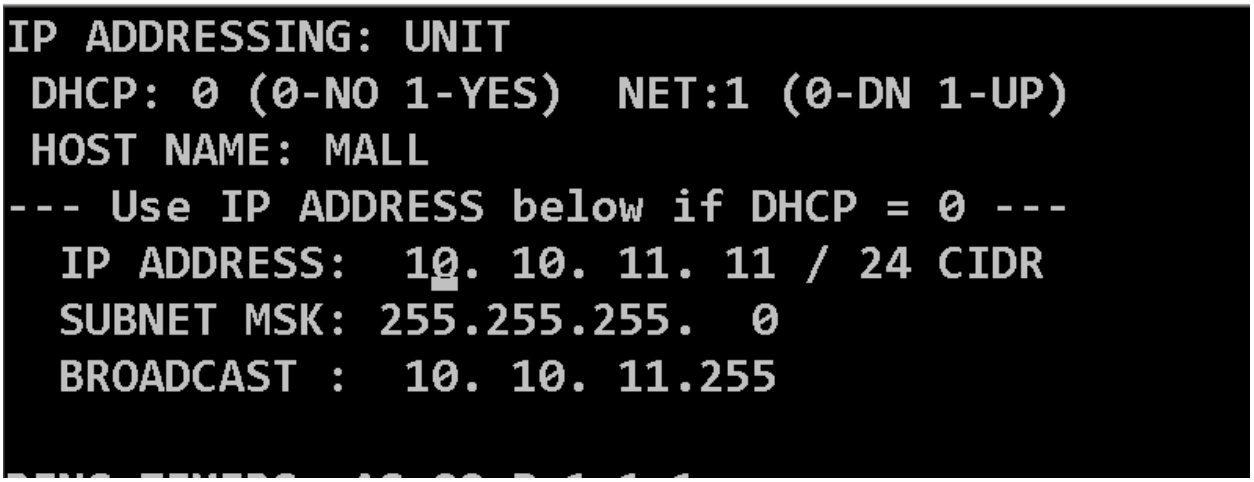
[1]Only supported by NextPhase Peer, [2]Only supported by SEPAC Peer
NOTE: For ranges when connecting to a NextPhase CU see the NextPhase Manual. SEPAC allows for input of the greater of SEPAC and NextPhase ranges.

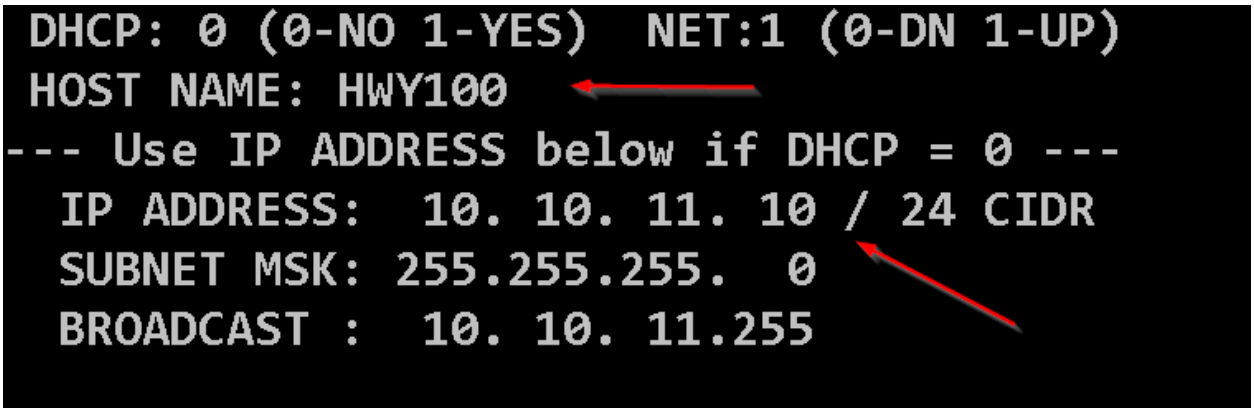
Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Program the IP address and Name





Host Name: The default value for this entry is ATC-2070. Press the number 0 to activate the keyboard to enter a controller NAME. 0 is the Backspace Key. Limited to 14 characters (I think) – no space key.

Desirable to fill in the host name as it gives us a 'real' label to reference the controller.

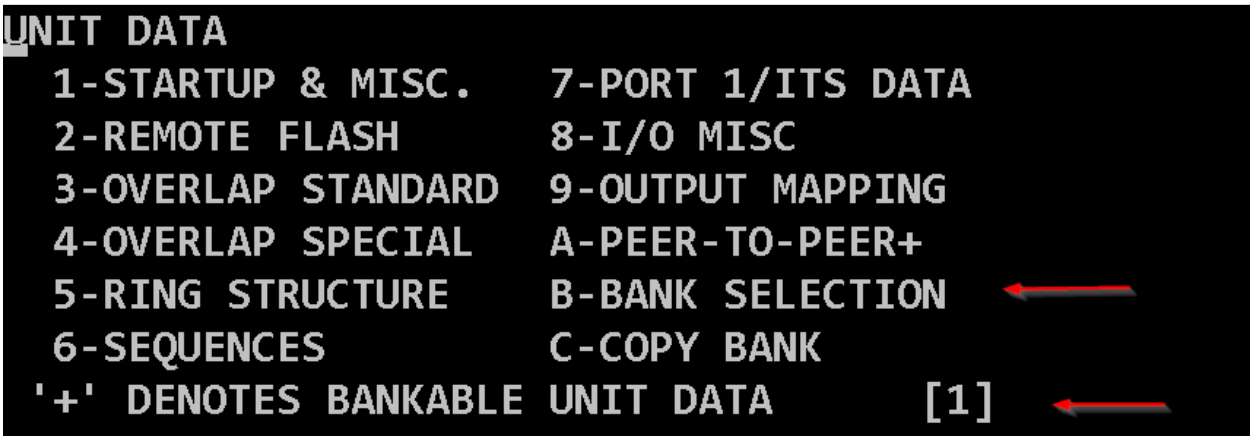
IP Address: Same as always.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Pick the Unit Bank where P2P is Programmed



The selection B allows you to select which Unit Bata bank you are editing.

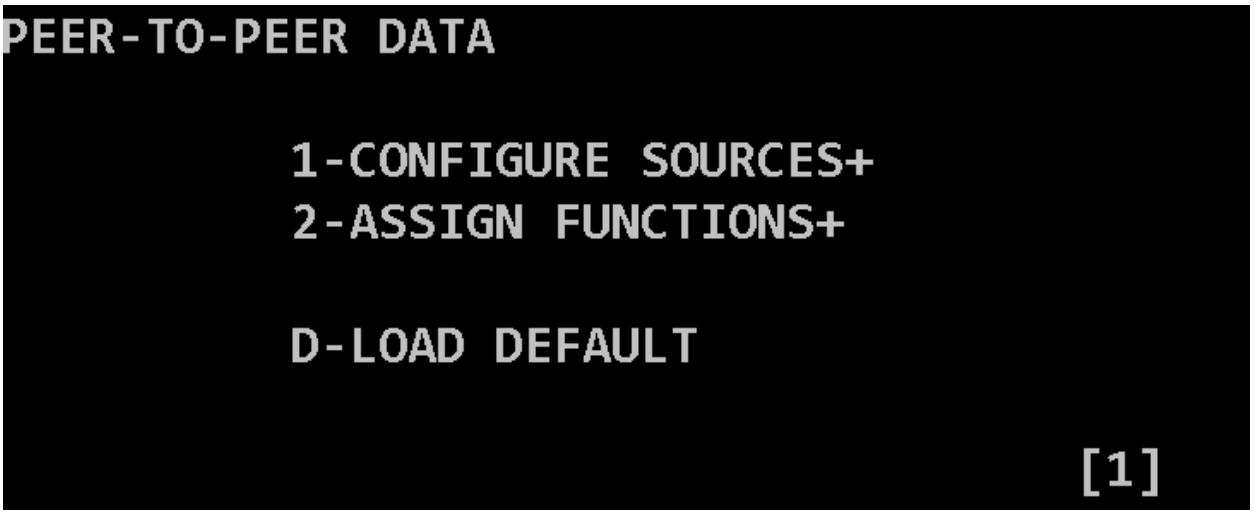
[1] indicates that the active unit data bank for editing is Unit Bank 1.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Define the P2P Players

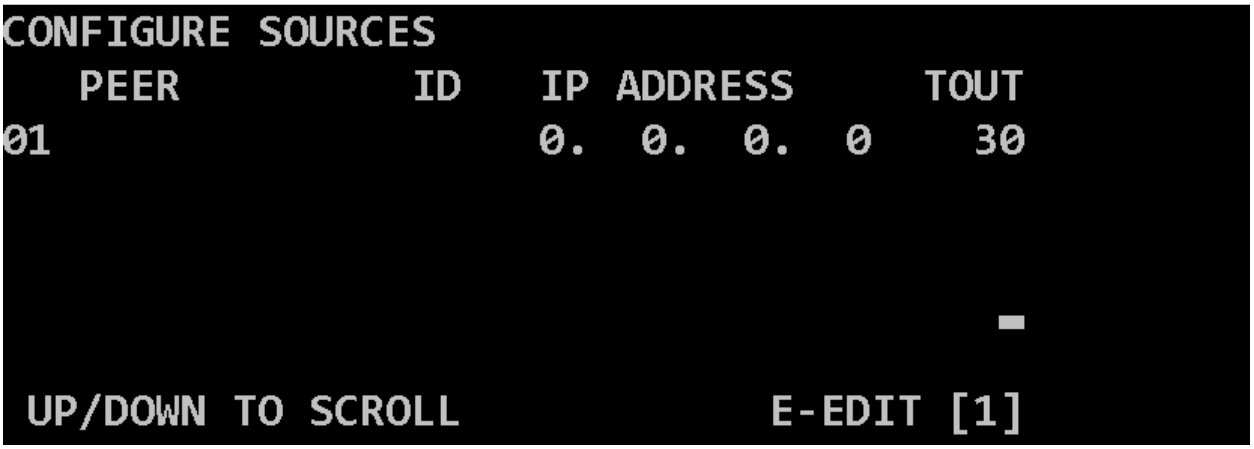


This programming is done at the Follower controller only. Choose number 1 from the P2P Menu

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

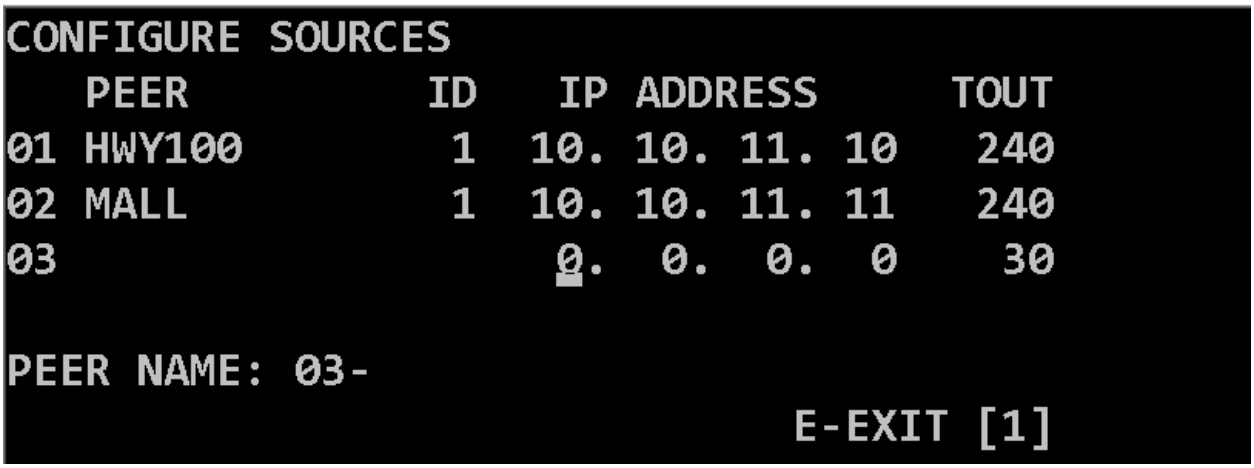


This is the empty screen. The only items you are being asked to define are the IP Address and the Failure Timeout time allowed for the controller.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required



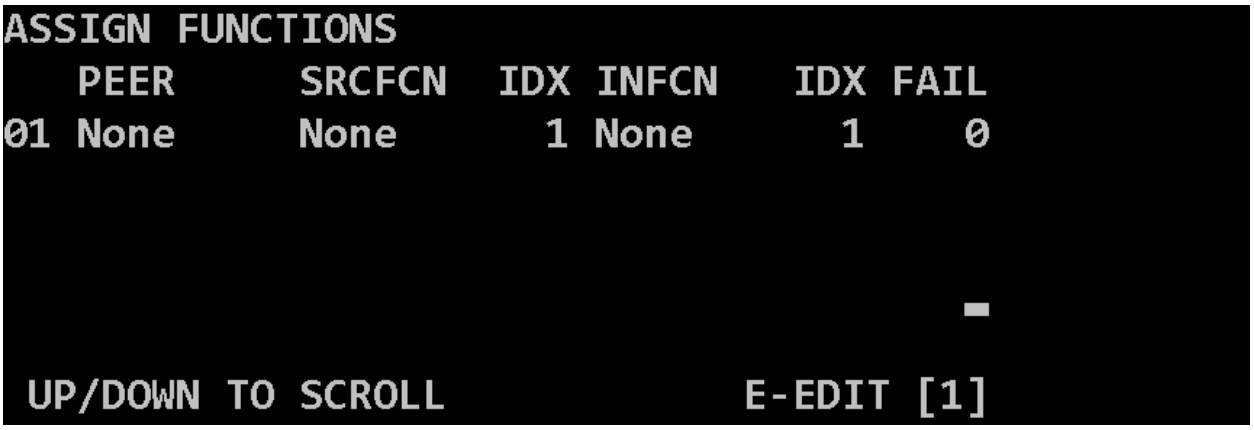
Enter the IP Addresses and the TOUT for the devices that provide P2P inputs on the network. TOUT – The follower controller will send out a 'heartbeat' (I think once per minute, because the default value of 30 seconds does not keep the controller P2P links alive.) I recommend 240 seconds for the Fail Timeout timers. If the P2P source is 'alive' the name of the device populates. If the P2P source is 'failed', the name will populate Fail01, Fail02, etc.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Program the Actions

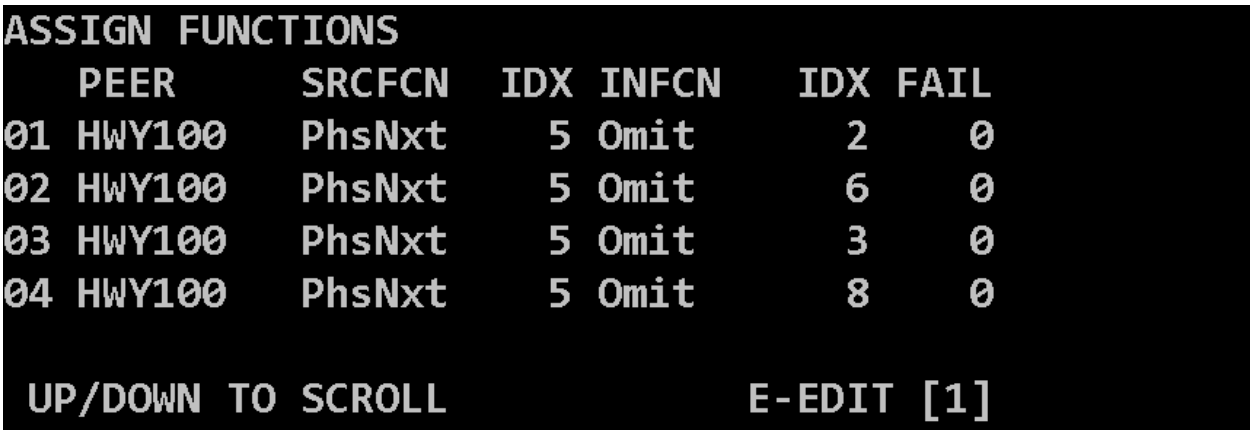


This programming is done at the Follower controller only. This is the empty screen for P2P choice 2. All the programming for P2P actions is done at the Follower signal.

Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required



Press E to begin editing the first line. The columns PEER, SRCFCN, INFCN are programmed with the PREV and NEXT buttons of the M60 screen. Arrow buttons of M60 screen move cursor around screen.

The first column, Peer, represents the Trigger controller; the column SRCFCN represents the trigger condition; the column INFCN represents the response at the follower.

The Fail Value represent what the trigger is supposed to do in case the 'heartbeat' fails. 0 – fail off; 1 – fail on.

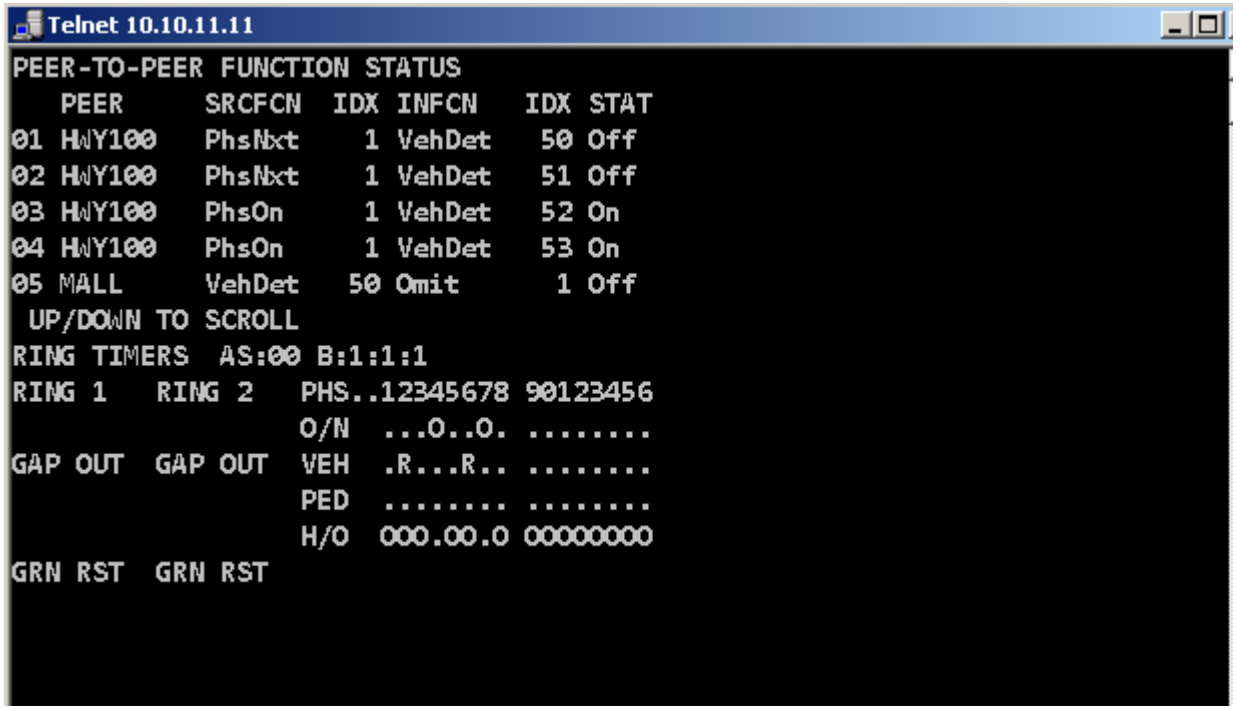
Peer-to-Peer Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Peer-to-Peer Active Status

Active Status Peer-to-Peer Function Status [MM-1 Active Status 5 – Communications 3 - Peer-to-Peer 2 – Function Status] allows user to see the active status condition of each of the P2P Logic Links.



Limitation: The screen can only show you five lines at a time.

Peer-to-Peer Sample Programming

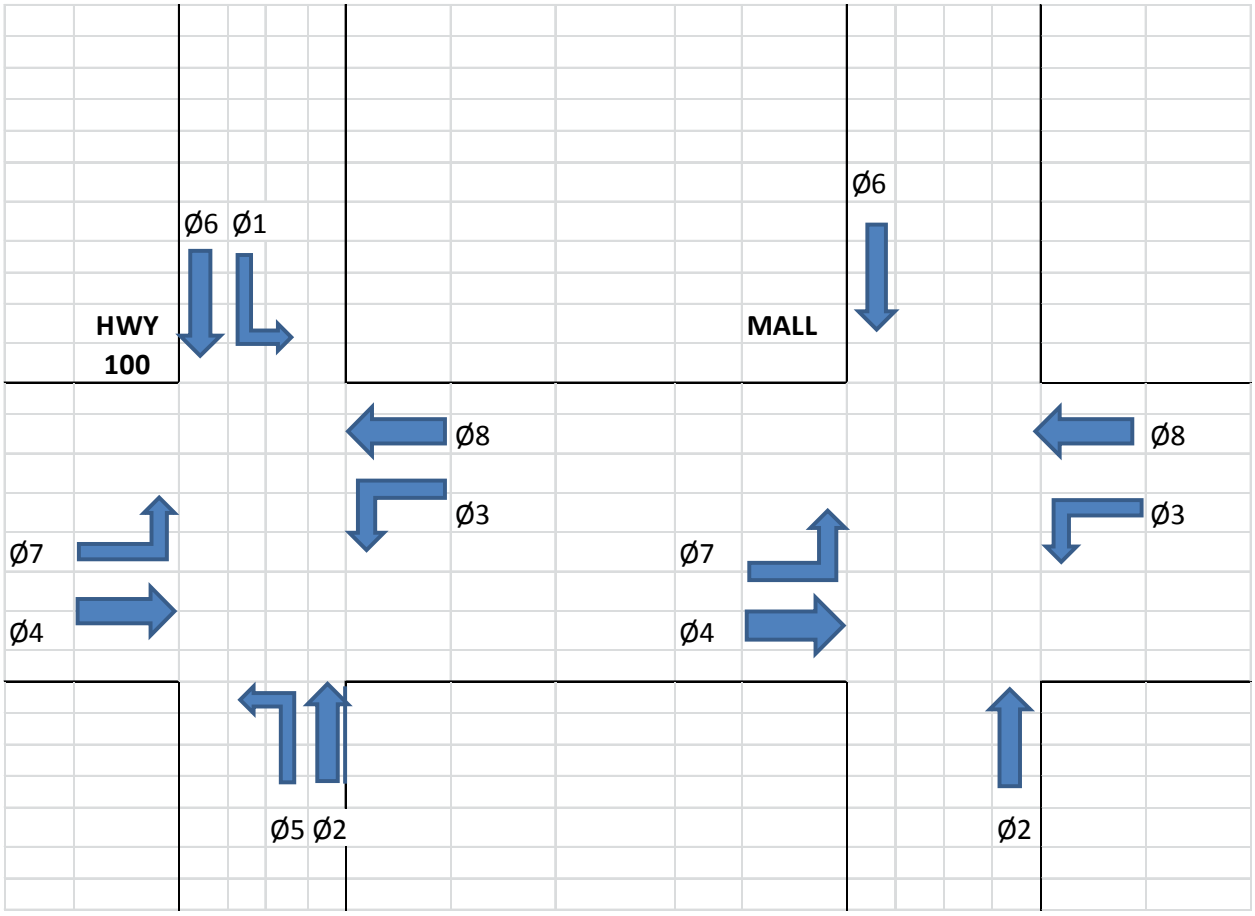
By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

This document shows a peer-to-peer program installation.

Example Problem

Two signalized intersections, HWY 100 and Mall. HWY 100 is part of an arterial corridor running a large coordination dial. Mall is a parallel road that would operate better in free mode.



Objective: Create a green path at MALL to accept traffic during HWY 100 phase 1

Peer-to-Peer Sample Programming

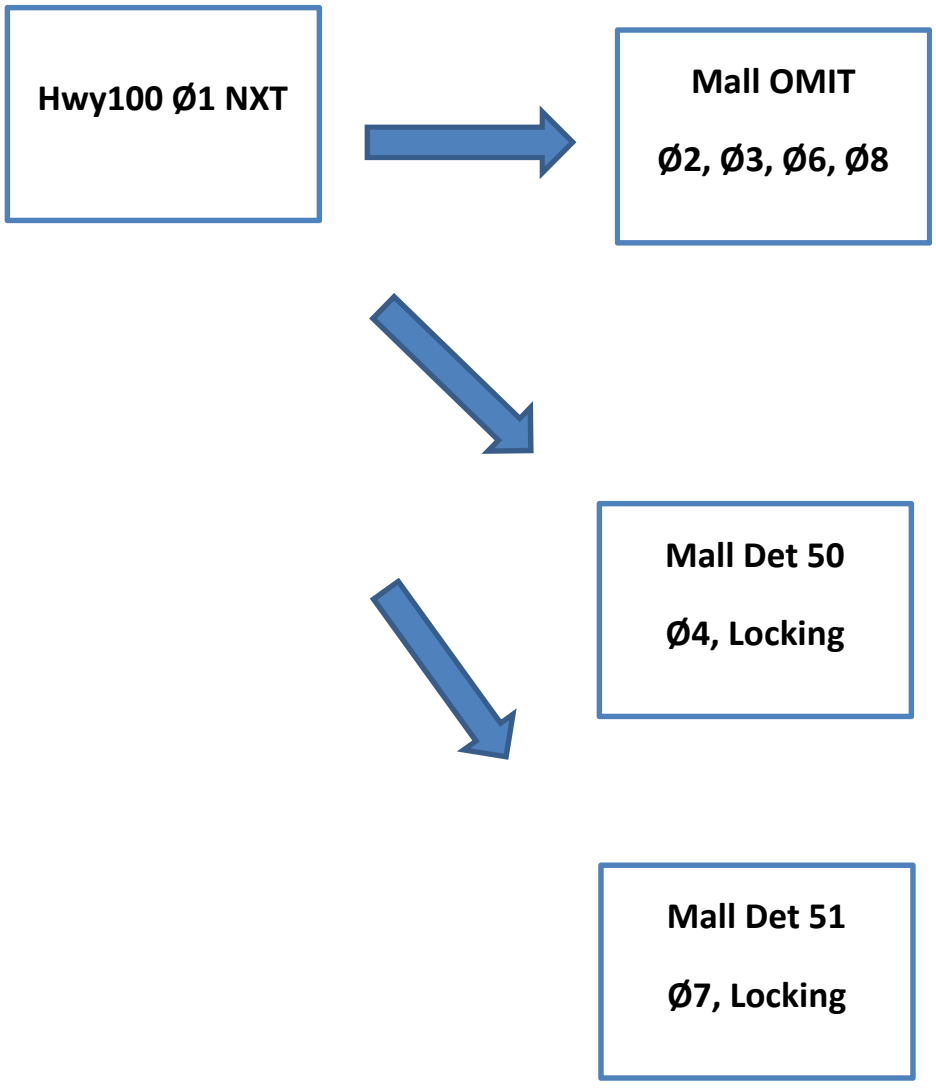
By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Designing P2P Commands

Following our Objective, first step is to design the command set:

Logic Step 1

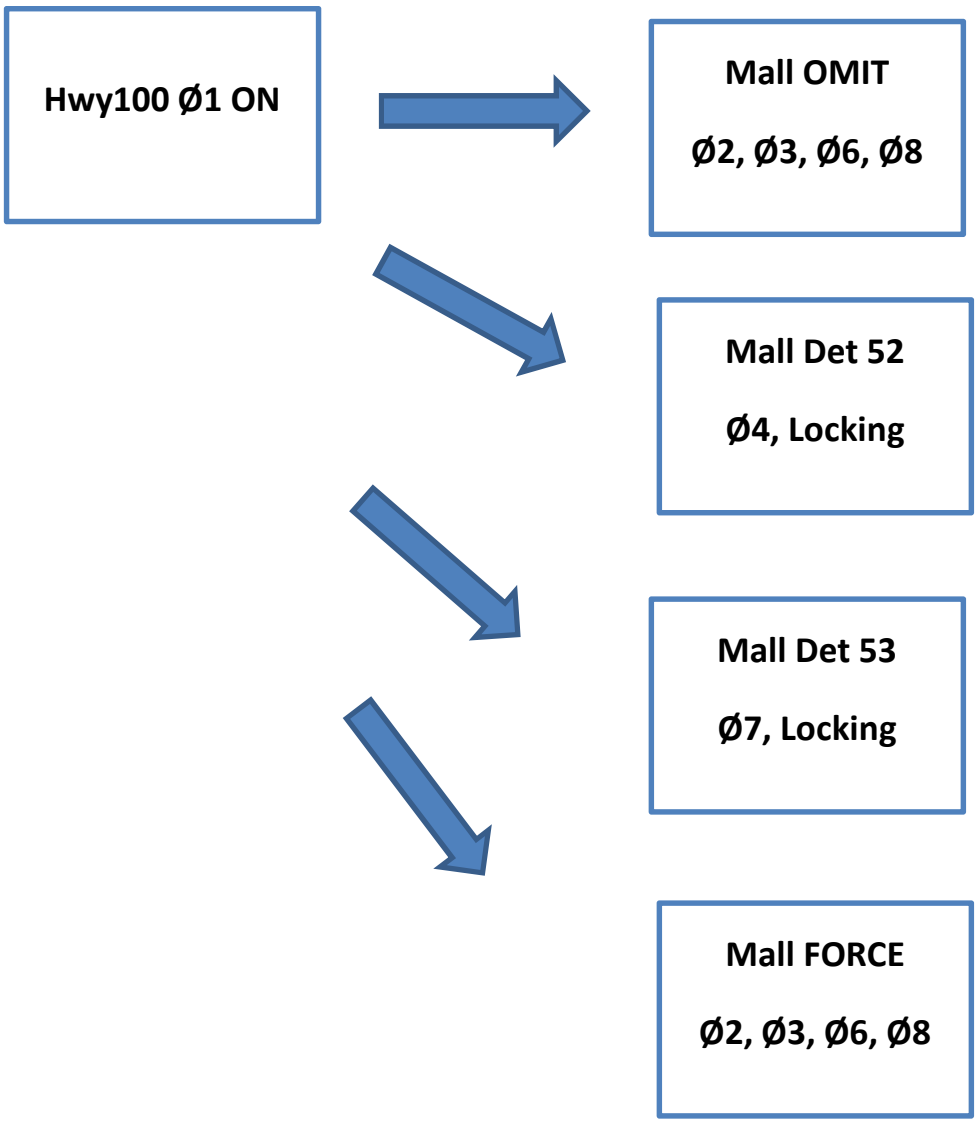


Peer-to-Peer Sample Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Logic Step 2



Peer-to-Peer Sample Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Programming P2P Into the X.57 Linux Controller

There are four steps to programming the P2P data into the Siemens X.57 firmware Linux controller.

1. Name/Address the controller [Both controllers].
2. At Follower Intersection [Mall], Chose Unit Bank that will employ Peer-to-Peer functions
    a. Unit Bank 1 if P2P 24/7
    b. Unit Bank 2 if P2P by TOD
    c. Program the TOD Steps if changing Unit Banks by TOD
3. At Follower Intersection [Mall], identify the Peer-to-Peer players.
4. At Follower Intersection [Mall], program the P2P Functions.

Refer to document, ***Peer to Peer Programming MoboTrex***, for explanation of Steps 1 to 3.

Peer-to-Peer Sample Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Program Step 4

Logic Step 1

```
ASSIGN FUNCTIONS
   PEER      SRCFCN  IDX INFCN   IDX FAIL
01 HWY100    PhsNxt    1 Omit      2    0
02 HWY100    PhsNxt    1 Omit      3    0
03 HWY100    PhsNxt    1 Omit      6    0
04 HWY100    PhsNxt    1 Omit      8    0

 UP/DOWN TO SCROLL            E-EDIT [1]
```

Logic Step 1 Ending; Logic Step 2 Beginning

```
ASSIGN FUNCTIONS
   PEER      SRCFCN  IDX INFCN   IDX FAIL
05 HWY100    PhsNxt    1 VehDet   50    0
06 HWY100    PhsNxt    1 VehDet   51    0
07 HWY100    PhsOn     1 Omit      2    0
08 HWY100    PhsOn     1 Omit      3    0

 UP/DOWN TO SCROLL            E-EDIT [1]
```

Peer-to-Peer Sample Programming

By Michael Grunewald

Firmware Version – X.57
Tactics Version 3.1.0 – or newer required

Logic Step 2 (cont.)

```
ASSIGN FUNCTIONS
   PEER     SRCFCN  IDX INFCN   IDX FAIL
09 HWY100   PhsOn     1 Omit      6    0
10 HWY100   PhsOn     1 Omit      8    0
11 HWY100   PhsOn     1 VehDet   52    0
12 HWY100   PhsOn     1 VehDet   53    0

 UP/DOWN TO SCROLL           E-EDIT [1]
```

Logic Step 2 (cont.)

```
ASSIGN FUNCTIONS
   PEER     SRCFCN  IDX INFCN   IDX FAIL
01 HWY100   PhsNxt    1 Omit      2    0
02 HWY100   PhsNxt    1 Omit      3    0
03 HWY100   PhsNxt    1 Omit      6    0
04 HWY100   PhsNxt    1 Omit      8    0

 UP/DOWN TO SCROLL           E-EDIT [1]
```

Summary

P2P programming is pretty easy. The difficult part is identifying the logic steps. Once you have the logic steps identified, the programming falls into place.