Operations Manual

For

Transit Priority

Using TS2 controllers with Version 61.x Firmware

Spl-1	Ø..1	...2	...3	...4	...5	...6	...7	..8 ->
Time	25	25	25	25	25	25	25	25
Coor-Ø	.	X	.	.	.	.	.	.
Mode	NON	MAX	NON	NON	NON	MAX	NON	NON
MaxReduc	5	0	5	5	5	0	5	5
MaxExtnd	0	15	0	0	0	15	0	0

April 2021

522 Gillingham
Sugar Land, Texas 77478
Phone: (281) 240-7233
Fax: (281) 240-7238

Terms of Use Agreement

USE OF THIS SOFTWARE MANUAL IS SUBJECT TO THESE TERMS OF USE.

Agreement

Please read this *User Agreement* carefully before using the information provided in this *Manual*. This Agreement explains the terms and conditions governing the use of this *Manual* ("Terms of Use"), and it is your responsibility to read and understand them. By using this *Manual*, you expressly agree to be bound by these *Terms of Use* and to follow them as well as all applicable laws and regulations governing the *Manual*. If you do not agree to be bound by these *Terms of Use*, you may not access or use this *Manual*. Cubic | Trafficware reserves the right to change these *Terms of Use* at any time, effective immediately upon posting the *Manual* from our company website. By continuing to use the *Manual* after we post any such change, you accept the revised *Terms of Use*. If you violate these *Terms of Use*, Cubic | Trafficware may terminate your use of the *Manual*, bar you from future use of the *Manual*, and take appropriate legal action against you.

Permitted Use

You agree that you are only authorized to read, view and retain a copy of pages of this *Manual* for your own personal use, and that you will not duplicate, download, publish, modify or otherwise distribute the material on this *Manual* for any purpose other than to review product information for personal use or the use of a government or non-profit organization.

No Commercial Use

Users may not use this *Manual* for any commercial purposes such as to sell merchandise or services of any kind. You must obtain our prior written consent to make commercial offers of any kind, whether by advertisements, solicitations, links, or by any other form of communication. Cubic | Trafficware will investigate and take appropriate legal action against anyone who violates this provision.

Copyright

All content included on this *Manual*, including text, graphics, logos, icons, images, and software is the property of Cubic ITS Inc. or its content suppliers and is protected by United States and international copyright laws. This compilation (that is, the collection, arrangement and assembly) of all content on this *Manual* is the exclusive property of Cubic ITS Inc. and is protected by U.S. and international copyright laws. Cubic ITS Inc. reserves the right to revise the pages of the *Manual* or withdraw access to them at any time.

Trademarks

The logo and trademarks that appear throughout the *Manual* belong to Cubic ITS Inc., its affiliates or third-party trademark owners, and are protected by U.S. and international trademark laws. Without express prior written permission, you may not display or use in any manner, the logos or trademarks. Nothing in this *Manual* shall be construed as granting any license or other right to the intellectual property or other proprietary rights of Cubic ITS Inc., its affiliates or any third party, whether by estoppels, implication or otherwise. All contents of the *Manual* are: © Copyright 2021 Cubic ITS Inc. or its licensors. All Rights Reserved.

Disclaimer of Warranty

YOU UNDERSTAND AND EXPRESSLY AGREE THAT YOUR USE OF THE MANUAL AND THE INFORMATION FOUND THERE IS ENTIRELY AT YOUR RISK. CUBIC | TRAFFICWARE AND ITS AFFILIATES AND LICENSORS MAKE NO WARRANTIES OR ANY REPRESENTATIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF TITLE OR NON-INFRINGEMENT OR IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT OR OTHER VIOLATION OF RIGHTS IN RELATION TO THE AVAILABILITY, ACCURACY, VALIDITY, COMPLETENESS, RELIABILITY OR CONTENT OF THESE PAGES AND/OR THE MANUAL. CUBIC | TRAFFICWARE SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, LOST PROFITS OR FOR BUSINESS INTERRUPTION ARISING OUT OF THE USE OF OR INABILITY TO USE THIS MANUAL, EVEN IF CUBIC | TRAFFICWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. TO THE MAXIMUM EXTENT PERMITTED BY LAW, YOU HEREBY RELEASE AND FOREVER WAIVE ANY AND ALL CLAIMS YOU MAY HAVE AGAINST CUBIC | TRAFFICWARE, ITS AFFILIATES AND LICENSORS FROM LOSSES OR DAMAGES YOU SUSTAIN IN CONNECTION WITH YOUR USE OF THE MANUAL.

SOME JURISDICTIONS DO NOT ALLOW EXCLUSION OF CERTAIN WARRANTIES OR LIMITATIONS OF LIABILITY, SO THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU. THE LIABILITY OF CUBIC | TRAFFICWARE WOULD IN SUCH CASE BE LIMITED TO THE GREATEST EXTENT PERMITTED BY LAW.

Applicable Law

These *Terms of Use* and all legal issues related to the *Manual* shall be governed by the laws of the State of Texas, without regard to conflict of laws principles. You agree that any dispute involving these terms of use or this *Manual* will be heard in a court with jurisdiction in Fort Bend County, Texas. You further agree that the prevailing party in any legal action will be entitled to recover its reasonable attorney's fees incurred in connection with that action. If for any reason a court of competent jurisdiction finds any provision of these Terms of Use to be unenforceable, the remaining terms will continue in full force and effect.

Entire Agreement

These *Terms of Use* are the final and entire agreement between you and Cubic | Trafficware with respect to this subject and replace any and prior or contemporaneous understandings or agreements, written or oral, regarding the subject matter. Any waiver of any provision of these Terms of Use shall be effective only if in writing and signed by an authorized representative of Cubic | Trafficware.

Table of Contents

1 Transit Priority Overview

Cubic | Trafficware controllers provide six high-priority and four low-priority preempt requests. In addition, the four low-priority requests may be programmed for transit priority service. Preemption is documented in Chapter 8 of the Cubic | Trafficware controller manual. The purpose of this manual is to supplement Chapter 8 for users wishing to apply priority service rather than preemption for transit operations.

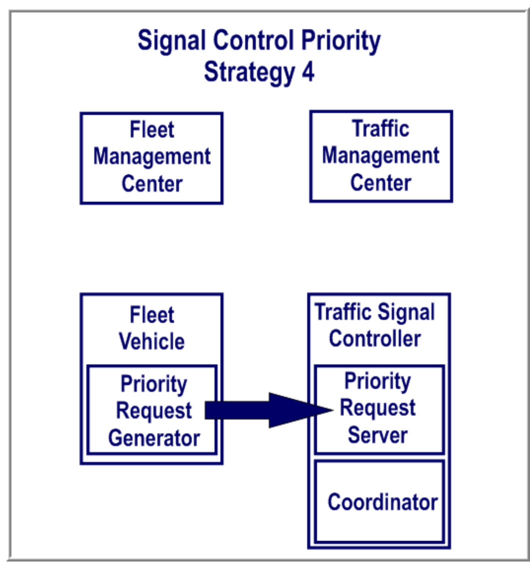
1.1 Preemption Compared With Priority Service

Preemption is defined by NEMA TS2-2003 and NTCIP 1202 – "Object Definitions for Actuated Traffic Signal Controllers". When a preempt request is received, the controller terminates the active phase to service any track clearance intervals associated with the preempt. The controller then moves to the dwell state in flash, free or coordinated operation to service the programmed dwell phase(s) until the preempt input is released.

Transit priority is described in the latest draft of NTCIP 1211- "Object Definitions for Signal Control and Prioritization" available from http://www.ntcip.org. Priority service differs from preemption in that the controller never leaves coordination and phase skipping is optional based on a user defined strategy used to service the priority request. In addition, Cubic | Trafficware goes beyond the operation described in NTCIP 1211 by providing priority service in free operation as well as during coordination.

1.2 NTCIP 1211 Signal Control Priority Scenario 4

Cubic | Trafficware controllers implement NTCIP *Signal Control Priority Scenario 4*. In this case, the *Priority Request Server (PRS)* is embedded within the local controller logic and no data exchange takes place between the *PRS*, fleet management and the traffic management center. In other scenarios, requests are forwarded to a central system before priority service is granted at the local level. This is often done to identify the vehicle and determine if it is behind schedule before granting the priority request.



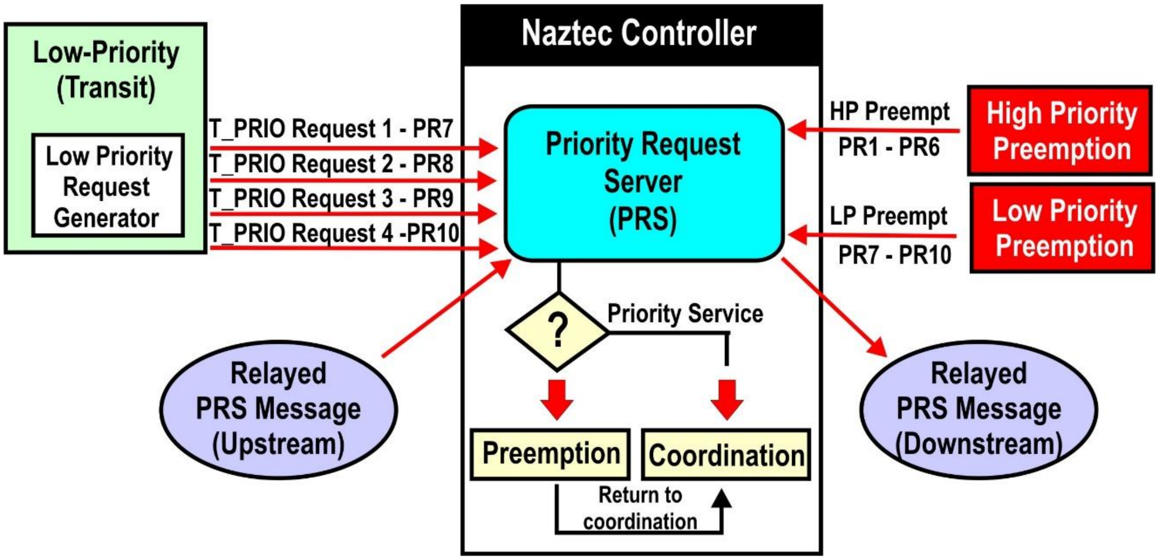
The *Priority Request Server* is tightly coupled within the Cubic | Trafficware controller logic so decisions can be made in real-time to service the transit vehicle. This approach does not impose any time latencies compared with other scenarios which expect a reply message before priority service can be granted at the local level.

The disadvantage of Scenario 4 is that the decision to service the request is not conditioned on whether the transit vehicle is "early" or "late" compared to a predefined transit schedule. Many agencies are now placing this decision within the *Fleet Vehicle* and conditioning the *Priority Request Generator* to initiate the request only if the vehicle is behind schedule.

Cubic | Trafficware does allow priority requests from one controller to be relayed to controllers downstream. This can be accomplished either through a dedicated hardwire interconnect or as a Cubic | Trafficware Central (ATMS.now or StreetWise) GENERATED software trigger to allow the central system to act as the *Priority Request Generator*. An example is provided in Chapter 8 to show how the Cubic | Trafficware Central software triggers can forward priority service requests (PRS Messages) for light rail operations.

1.3 Cubic | Trafficware Preemption / Transit Priority Model

The *Cubic | Trafficware Preemption / Transit Priority Model* below extends the *Priority Request Server* from NTCIP 1211 to include preemption as well as transit priority.



- High-priority inputs PR1-PR6 are reserved for rail and emergency vehicle preemption
- Low-priority inputs PR7-10 may be assigned to low-priority preemption or transit priority
- Low-priority inputs PR7-10 activate transit priority if **LP_Type** is set to **T_PRIO**
- Low-priority inputs PR7-10 activate preemption if **LP_Type** (PR7-10) is set to **EMERG** (emergency vehicle preemption or **T_PRMPT** (transit preemption)
- Higher priority preempts always override lower priority preempt requests (PR1 overrides PR2)
- PR1 and PR2 always override lower priority requests PR3-PR6 and PR7-PR10
- Requests within a priority group (PR3-PR6 or PR7-PR10) are handled on a first-come first-served basis; however, there is an option to disable this for PR3-PR6
- The controller returns from preemption to the programmed exit phase(s) or to the phase currently being serviced in the coordination background cycle (if COOR+PRE is set)
- The controller never leaves coordination during priority service (**T_PRIO**)
- **T_PRIO** is built to work in association with coordination. The Coord Phase is always served in the cycle.
- The *NTCIP* method provides an early return or extension of the *priority service phase.*
- The TSD and TED provide a window for which the transit phase will try to be served. Consider the end of green coordination reference (ENDGRN). Once the coord phase is released by the coordinator, the TSP will try to reduce all phases before the transit phase to get transit in into the green window. If not, it will try to extend the window or do the early return. Always keep in mind that the coord phase is sacred.
- *EZ Transit* applies longway and shortway transition to achieve early return and extension
- Preemption and priority messages may be relayed between intersections using an external hardwire interconnect or from a StreetWise or ATMS.now generated (software) trigger.

The NTCIP Transit Priority Method

A *Priority Service Request* is an oscillating 6.25 Hz signal applied to inputs PR7-PR10 when the emitter from a transit vehicle is detected. When the request is received, the *Priority Request Server (PRS)* initiates countdown timers to project the arrival and departure of the vehicle at the intersection. The *Priority Request Server* will then select one of the following actions to modify the *Coordinator* and service the projected arrival of the transit vehicle.

1. Reduce phases to provide an early return to the priority service phase
2. Do nothing, or
3. Extend the priority service phase to service a later arrival of the vehicle

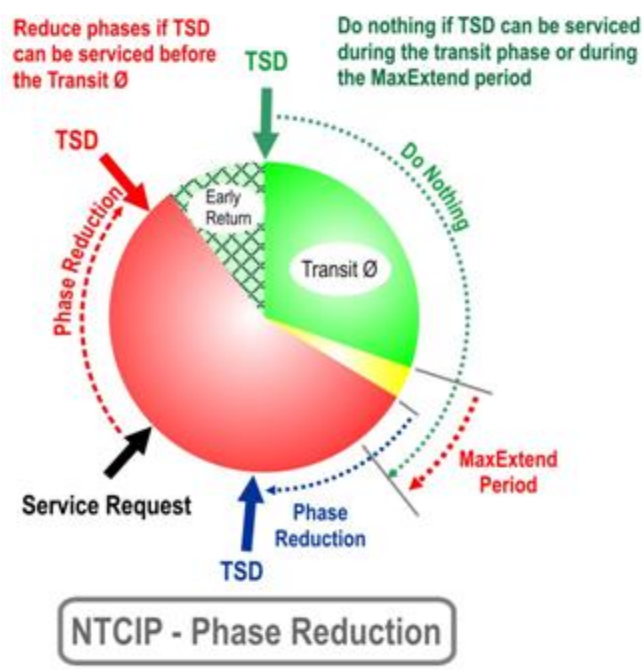
1.3.1 NTCIP Phase Reduction (Early Return)

The *Time of Service Desired (TSD)* is the arrival time of the transit vehicle at the stop-bar after it is first detected by the *Priority Request Server*. The *TSD* includes any dwell time to discharge and load passengers at a nearside stop and any expected congestion delay in the estimate of the arrival time. The *TSD* counter begins counting down when the vehicle is first detected. The vehicle is expected to be at the stop bar at *TSD* = 0.

An *early return* is provided if the *TSD* is projected prior to the start of the priority service phase. If the *TSD* lies within the green portion of the priority phase, the response is "do nothing".

Early return is accomplished with user *MaxReduce* and *MaxExtend* values. NTCIP calls for these values to extend the *Split Table*. Cubic | Trafficware provides separate *MaxReduce* and *MaxExtend* values for every split time in each *Split Table*. *MaxReduce* and *MaxExtend* are used to calculate *Priority Max (PrMx)* times for each phase to reduce phases and provide an early return to the priority phase.

The controller insures that phases are not reduced below minimum phase requirements and that splits are balanced at the barriers. Therefore, it is not possible to fail the coordinator by specifying inappropriate *MaxReduce* and *MaxExtend* values. This is a major improvement over other methods that insert additional phases in the sequence using *MaxReduce* and *MaxExtend* times.



1.3.2 NTCIP Based Phase Extension

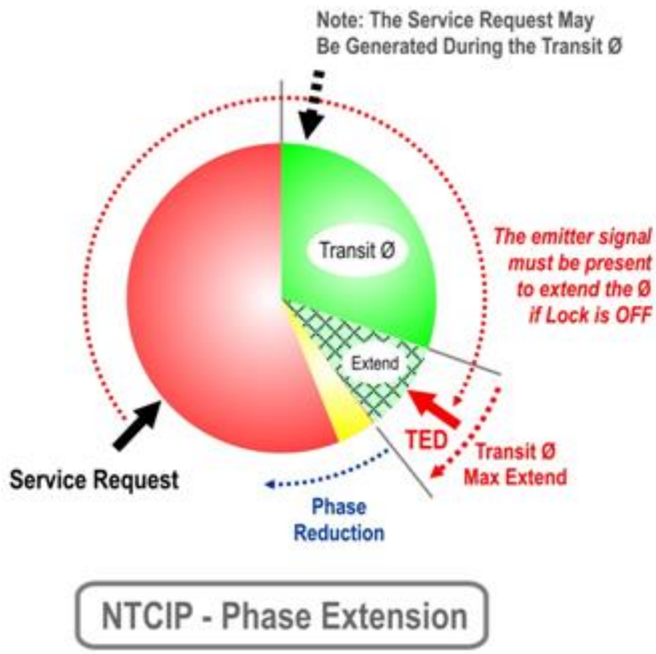
NTCIP defines *Time of Expected Departure (TED)* as the time required for the transit vehicle to clear the intersection after it is first detected (in seconds). The controller begins separate *TSD* and *TED* countdown timers when the service request is received. The priority service phase is extended by the *MaxExtend* value one second before the force-off if the following conditions are met:

1. the remaining *TED* counter is greater than zero and less than or equal to *MaxExtend*
2. the emitter input for the request is still active and the *Lock* parameter for the input is OFF

No extension will take place if the request is not "locked" and the emitter drops out because the vehicle has cleared the intersection. If the *TSD* is projected after the end of the *MaxExtend* period, the controller will provide an early return in the next signal cycle using phase reduction.

Note: As long as the preempt call does not go away or is "locked out", then transit will not complete. In other words, TED may expire, but a preempt call held through the end of the transit phase should still cause an extension of the transit phase.

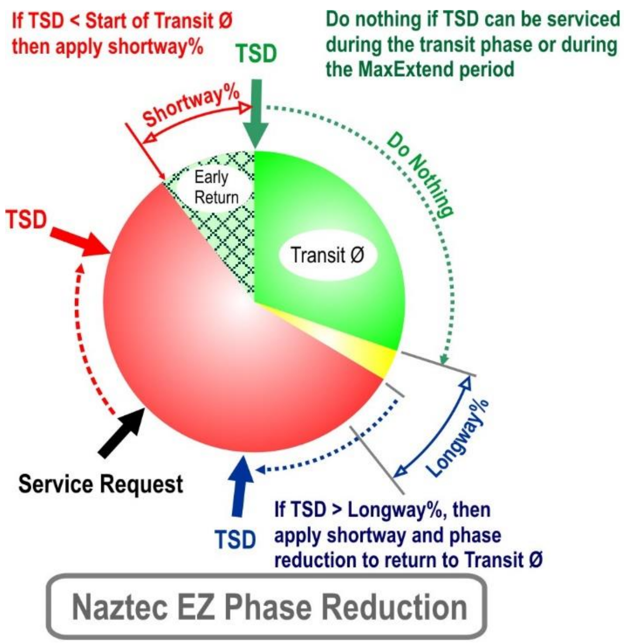
Many times the *TSD* and *TED* times are typically set to the same value because the arrival at the stop-bar and the departure time are essentially the same after the vehicle is detected.

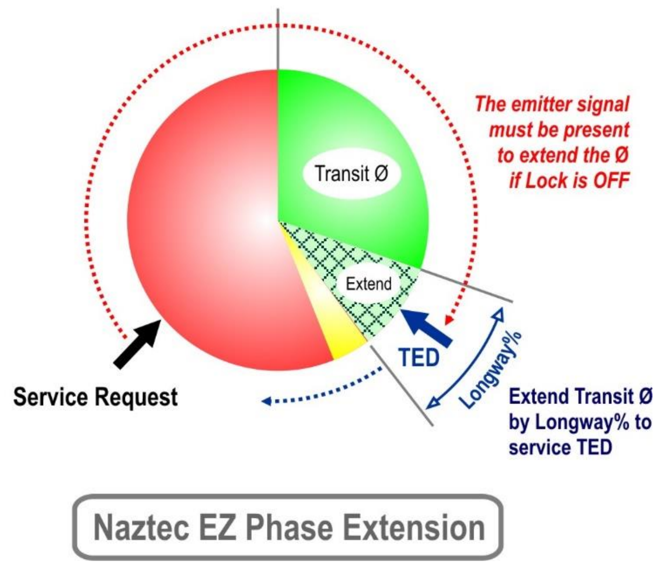


1.4 Cubic | Trafficware EZ Method

The *Cubic | Trafficware EZ* method uses shortway and longway transition to accommodate the arrival of transit vehicles during coordination. If the *TSD* arrival is expected before the priority service phase, the controller will apply shortway transition to reduce phases leading up to the priority phase. If the TSD cannot be serviced with shortyway% alone, *PrMax* will also be applied if *MaxReduce* values have been supplied by the user in the active split table.

If the *TSD* is projected within the green period of the priority phase, the response is "Do Nothing". However, if the *TED* can be accommodated by extending the transit phase, the controller will apply longway% until the *TED* timer expires or until the emitter drops out. If the controller cannot accommodate the late arrival, then *PrMx* and shortway% will be applied in the next signal cycle.





1.5 Additional References

Please refer to *"An Overview of Transit Signal Priority"* published by ITS America. This free publication may be downloaded from the ITS America web site using the following address:

http://itsa.org/resources.nsf/Files/FinalTSPOverviewUpdated/$file/FinalTSPOverviewUpdate.pdf

2 Transit Priority Programming

2.1 High-Priority and Low-Priority Inputs

Transit priority and low-priority inputs are shared with high-priority preempts 3-6 (see chart below). The controller recognizes high-priority as a steady ground true input and low-priority as an oscillating 6.25 Hz input on these inputs. This is an industry standard set by *3M Corporation.*

Preempt #	Preempt Input	Type (typical)
1	1 (steady low)	RAIL
2	2 (steady low)	RAIL
3	3 (steady low)	RAIL or EMERG – H Prior
4	4 (steady low)	RAIL or EMERG – H Prior
5	5 (steady low)	RAIL or EMERG – H Prior
6	6 (steady low)	RAIL or EMERG – H Prior
7	3 (oscillating)	EMERG, T_PRIO, T_PRMT
8	4 (oscillating)	EMERG, T_PRIO, T_PRMT
9	5 (oscillating)	EMERG, T_PRIO, T_PRMT
10	6 (oscillating)	EMERG, T_PRIO, T_PRMT

2.2 Enabling Transit Priority (MM->3->7)

Enter Transit by choosing selection #3 , Preemption from the main menu. Then choose the appropriate Low priority preemption number, PR7-PR10.

```
# 7      Low-priority / Platoon
1.Low-Priority
2.Platoon Progres'n
```

Sub-menu # 1 is used to program the Low priority Times. The controller menu below enables transit priority for preempt # 7 by setting ***LP_Type*** to T_PRIO.

```
# 7  Low-Priority Times      Phases
 LP_Type T_PRIO Min    0 Dwel  0  0  0  0
Coor+Pre OFF    Max    0
    Lock ON  MaxL/O 255  --- Transit ---
Cond L/O OFF            Headway tm     0
Begin Mode SKIP         Headway Grp OFF
                        Queue Jump   OFF
```

Sub-menu #2 is used to program Platoon progression.

```
# 7   Platoon       Times       Phases
 Plat.Rx OFF  RxDel    0
               Hold    0
 Plat Tx OFF  TxDel    0   Tx  0  0  0  0
```

2.2.1 Enable Parameter (OFF / ON / ENERG / T_PRMT / T_PRIO)

The ***Enable*** parameter must be set to **ON** to enable bus preemption or **OFF** to disable the preemption. The parameter may also be set to **EMERG** to enable a low-priority emergency vehicle preemption, **T_PRMT** or **T_PRIO** for a Transit preemption variable.

```
#  7  Low-Priority Times       Phases
 LP_Type T_PRIO Min    0 Dwel  0  0  0  0
Coor+Pre OFF    Max    0
    Lock ON  MaxL/O 255  --- Transit ---
Cond L/O OFF             Headway tm    0
Begin Mode SKIP          Headway Grp OFF
                         Queue Jump  OFF
```

The primary difference between the **ON** (bus preempt) option, the **EMERG** (low-priority emergency vehicle) or **T_PRMT** and **T_PRIO** options lies in the preempt response during coordination. To run transit preemption the user will select **T_PRIO**.

2.2.2 Coor+Preempt

The Coord+Preempt parameter allows coordination to proceed in the background during the preempt sequences. This allows the controller to return to the phase(s) currently active in the background cycle rather than the next phases in rotation. This option allows the controller to return from preemption to coordination in SYNC without going through a transition period to correct the offset. Many agencies utilize the Coor+Preempt option when coordination is interrupted frequently by preemption.

Please note that because preemption is an emergency operation, there are times that the coordinator must go FREE to insure the safety of the motoring public. One example is during railroad preemption track clearance phase timing. If Track Clearance phases and timing are programmed, the coordinator will go free to insure that the vehicles will move off the track. Once the dwell phases begin timing, the coordinator will begin to transition back to being in SYNC.

2.2.3 Lock (Max Lockout Type) Parameter (ON/OFF)

The ***Lock*** parameter only applies to low-priority requests. This locks out any other low pre call. The ***Lock*** will tell how the controller uses the ***Lock*** (lockout) timer. Selecting it to OFF will not lock out any new low priority requests. Selecting it to **ON** will lock out low priority requests based on the ***Lock*** time and demand. With ***Lock*** set to **ON**, a ***Lock*** time greater than zero will inhibit a new service request until the lock out period expires **or** all phases with demand when the lockout period begins have been serviced. In other words, a ***Lock*** set to **ON** is provided to insure that all demand phases have been serviced before a new request is serviced.

Please set *Lock* to ON for the examples in this manual to make sure the service request does not drop out before the controller returns to the transit phase.

2.2.4 MaxL/O (Max Lockout Time) Applied to Transit Priority

```
# 7  Low-Priority Times       Phases
 LP_Type T_PRIO Min    0 Dwel  0   0   0   0
Coor+Pre OFF     Max   0
    Lock ON  MaxL/O 255  --- Transit ---
Cond L/O OFF             Headway tm     0
Begin Mode SKIP          Headway Grp OFF
                         Queue Jump  OFF
```

The **MaxL/O** time period (0-999 seconds) limits the duration of the lockout period following any preempt or priority service. A value of zero disables the lockout, thereby allowing a new priority request to be serviced 3" after another preemption or priority service ends. This inherent 3" lockout insures that the last service is complete and all affected values, including status screens have been updated before initiating the new service request. This timer is used in association with the ***Lock*** parameter.

A ***MaxL/O*** greater than zero will inhibit a new service request until the lockout period expires or all phases with demand when the lockout period begins have been serviced. ***MaxL/O*** is provided to insure that all demand phases have been serviced before a new request is serviced.

2.2.5 Cond L/O (ON/OFF)

Setting the conditional lockout parameter will lock out all low priority requests for the duration of the ***Lock*** time. This parameter should be set to **OFF** for Transit priority .

2.2.6 Begin Mode (MIN/SKIP)

Setting ***Begin Mode*** to **MIN** services only the minimum times for all phases with calls prior to serving the transit phase(s). Think of it as "a poor man's transit" because in effect, it reduces each phase to the phase minimum prior to serving the transit phase(s). Based on when the call occurs, as well as the sequence and concurrency that is currently running, the algorithm will move to the LP phases as soon as it can. This setting does **not** guarantee that all phases run prior to rotating to the LP preemption phase(s) but allows other phases to run prior to serving LP phases. Setting this to **SKIP**, when the controller receives a Low-Priority preempt, the controller will service the begin phases normally and then move directly to the Priority phases, skipping any other phases in sequence that may have calls. This parameter has no effect when ***Enable*** is set to **T_PRIO.**

2.2.7 Transit Priority Min and Max Times

The ***Min*** time (0-255 sec) insures that the priority request is active for the minimum period specified even if the oscillating input drops before the end of the period. This feature is useful to mask calls from an emitter that drops in and out when the phase selector is set to maximum sensitivity.

The ***Max*** time (0-255 sec) limits the time that a transit service can be active. If ***Max*** is zero, then no maximum limit is applied. The priority service will end after the ***Max*** time and will not reservice until the max lockout period ends to insure all phases with demand have been serviced.

Max is normally set to TSD plus the cycle length to prevent interrupting the controller with a continuous priority request. This feature also comes into play when queuing or spillback prevents the transit vehicle from being serviced within one cycle after the request is received.

2.2.8 Prior Phases

For low priority preemption types **EMERG** or **ON**, whenever a 6.25 Hz oscillating signal is applied to high priority inputs 3-6 (PR7-10), the controller will either dwell in the Prior *Phases* specified if these phases are active, or move immediately to the *Prior Phases* without violating the min times and pedestrian times of the phases currently being serviced With the exception of ***FreeMode*** being set to **ON**, the user typically does not program this variable for transit priority (**T_PRIO**) because the priority phase programming will be done in the strategy tables. However, it is advised to program the transit phase in case the **Enable** type is modified by the user to a type other than **T_PRIO**. The next chapter will discuss programming of the priority phases.

Please ensure if **LP_Type** is set to ON, T_PRMT, EMERG or T_PRIOR that at least one non-zero priority phase is programmed.

2.2.9 Headway tm (Maximum headway Time) (0-255 minutes)

Each low priority preemption has an independent internal headway timer which counts up from zero whenever a low priority preempt input occurs. While this timer is running, the low priority preempt in question is "locked out" until the headway timer exceeds the time programmed under the ***Headway tm*** parameter. It is used in association with the ***GrpLock*** parameter.

2.2.10 Headway Grp (ON / OFF)

The ***Headway Grp*** parameter is used in association with the headway timer. When ***GrpLock*** is **OFF**, the specific headway timer for the existing low priority preemption will be run and not allow any new preemption call for the current running low priority preemption to occur until the maximum headway time is reached. When ***GrpLock*** is **ON** the specific headway timer for the existing low priority preemption will be run and will not allow a new preemption call for **any** low priority preemption to occur until the maximum headway time is reached for the current running preemption.

2.2.11 QJmp (ON / OFF)

This parameter is used with bus preemption and should be set to **OFF** for Transit priority . It enables a transit overlap output (sign or indication) to display a Queue Jump signal (output) to the public.

In summary, each low priority preemption type allows the above programming features to be enabled as shown below.

FEATURE	TRANS	EMERG	ON	OFF
Coor+Pre			x	
Lock Mode / Lock Time			x	
Begin			x	
Queue Jump			x	
Min			x	
Max			x	
Prior Phases			x	
Headway tm	x			
Headway Grp	x			

2.3 Transit Priority Parameters

All programming required for transit priority service is accessed from coordination menu MM->2.

```
Coord Parms |    Pattern       |
1.Modes,+    4.Pattern Tbl   7.Splits
2.Priority   5.Trans,CoorP+  8.Status/Util
3.PatOftInh  6.Alt Tables+   9.More
```

Eight *Priority Strategy Tables* are programmed under MM->2->2 above. Each table allows the user to assign the priority strategy phase and any vehicle or pedestrian phases omitted while the strategy table is active. The eight *Priority Strategy Tables* may be assigned to any of the 4 priority requests (PR7-10) under the *Split* programming.

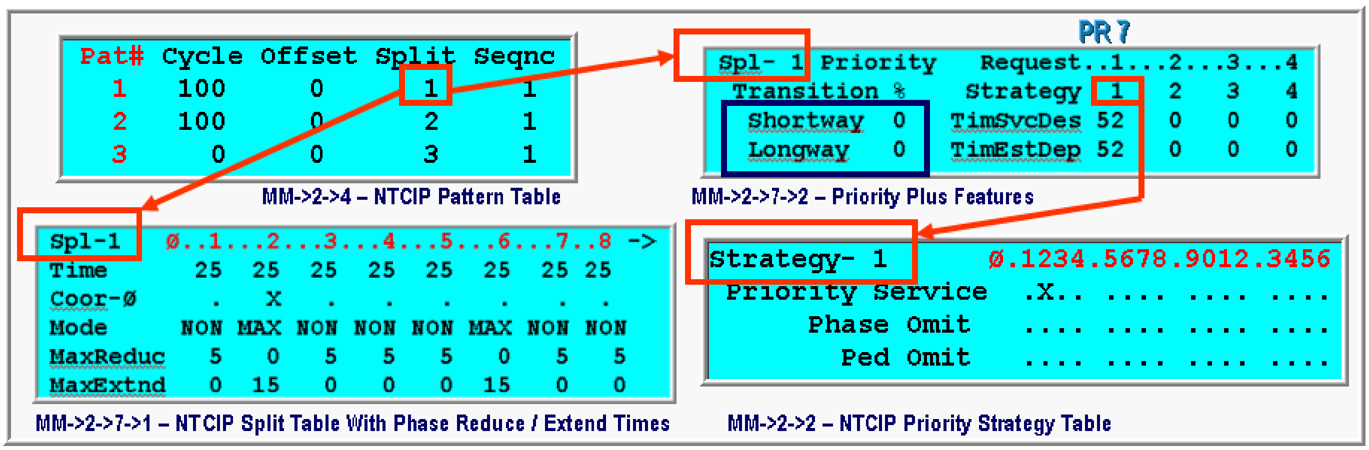
NTCIP calls for the ***MaxExtend*** and ***MaxReduce*** times to be an ***extension*** of the split tables, so these values are programmed under MM->2->7 above. Cubic | Trafficware also provides a split "plus" feature that extends the functionality of transit priority as discussed in the next section. This approach allows the user to vary the priority strategies and parameters by pattern and time-of-day.

Please note that because the ***MaxExtend*** and ***MaxReduce*** times are an ***extension*** of the split tables, the reduction or extension is a function of the SPLIT itself. In other words, the controller cannot predict demand on a phase in the future, so it must assume that the phase will have worse case service, and reduce the split accordingly. For instance, if you are running fixed coordination, reaching a phase prior to the force off of the preceding phase does not insure arrival early for phases that follow the phase that was returned to early. So, in the event of light service, it is possible to get to the transit phase earlier than needed.

NOTE: Max Reduction cannot be programmed for the COORD Phase (or the Pseudo-COORD phase) when running end-of-green coordination.

2.3.1 Programming the NTCIP Method

The programming illustration below shows the relationship between the *Split Table* and *"Plus" Features* associated with the table and any *Priority Strategy Tables* assigned to the split table.



MaxReduce and ***MaxExtend*** values are in seconds. Controller transit diagnostics insure that the sum of the ***MaxExtend*** times equal the sum of the ***MaxReduce*** times in the same ring. A worksheet is also provided in StreetWise to assist the user in programming these values.

The *Priority Plus* menu above assigns a separate *Priority Strategy Table* to ***Request*** 1-4 (PR7-10). A separate ***TimSvcDes*** (*TSD)* and ***TimeEstDep*** (*TED)* value (in seconds) may be assigned to each request. If you want to disable a transit priority request while a split table is active, simply program a zero *Priority Strategy Table* for the request.

A separate *Priority Plus* menu is provided for each *Split Table* in the controller. This allows the user to vary priority service by pattern under time-of-day, traffic responsive or adaptive control.

Adjusting Early Yield to Reduce the Coord Phase

All of the examples in this manual assume that the *priority service phase* and the *coord phase* are the same. While this is generally the case for most transit operations, you may also wish to assign a priority phase to a non-coordinated movement. If you desire to reduce the coordinated split time, you must adjust the NTCIP yield points of the non-coordinated phases to allow the controller to leave before the force-off of the coord phase. This can be accomplished through the *Early Yield* value assigned to each pattern under MM->2->5 (right menu)

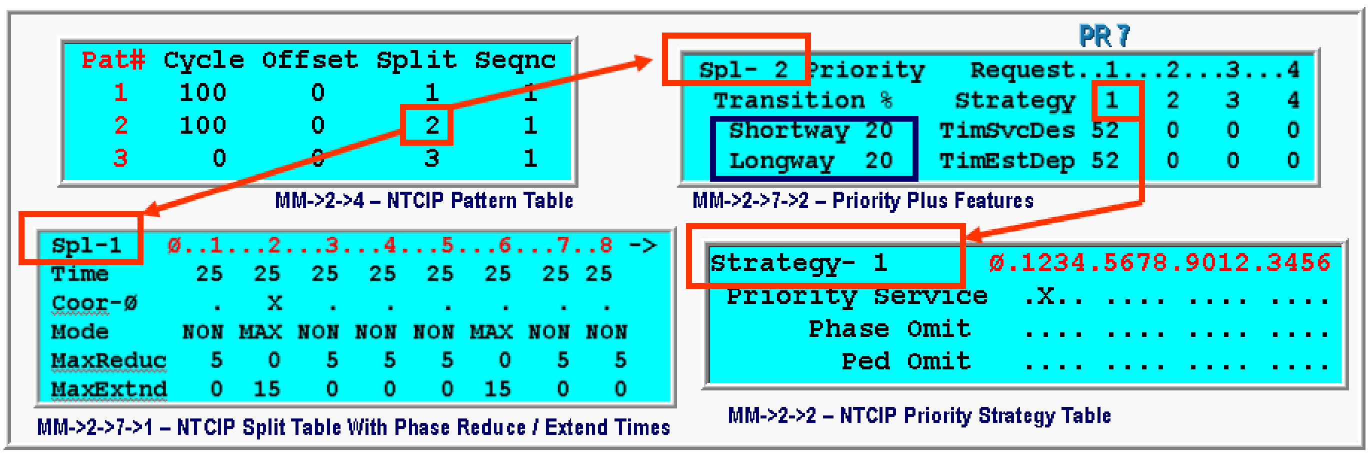
<- #	ErlyYld	Offset	RetHld	Flt	MinPermV/P
1	0	BegGRN	X	.	. .
2	25	EndGRN	.	.	. .
3	0	BegGRN	.	.	. .
....					
48	0	BegGRN	.	.	. .

MM->2->5: Early Yield Adjustment for Non-Coord Phase Yields

Set the *Early Yield* equal to the sum of the *MaxReduc* times applied prior to the force-off of the designated coord phase. For example, if *Pattern# 2* is active during a priority service request, the coord phase can leave 25" prior to the force-off if the *ErlyYld* parameter is assigned as shown above. If *Early Yield* is not assigned, then the *MaxReduc* times applied prior to the force-off of the coord phase will not allow the coord phase to leave early.

2.3.2 Programming the Cubic | Trafficware EZ Method

EZ Transit is enabled when ***Shortway*** and ***Longway*** values in *Priority Plus Features* are non-zero. Under the Cubic | Trafficware EZ method, ***MaxReduce*** and ***MaxExtend*** times are optional, but the strategy table, TSD and TED values are the same as the *NTCIP* method. Program the transit priority parameters below for pattern# 2 to follow the examples in Chapter 5.



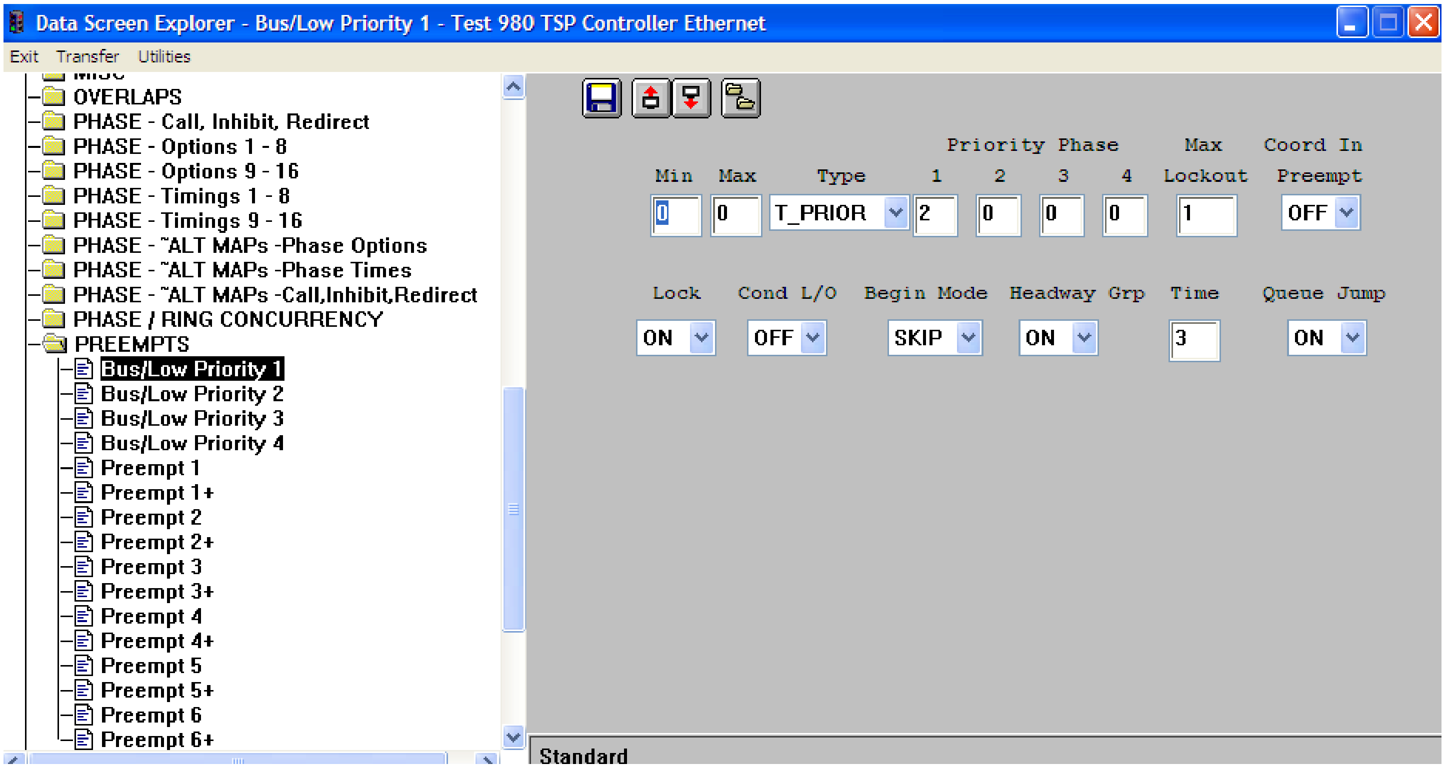
TSD and *TED* arrival times can vary greatly by time-of-day because approach speeds, dwell times and queuing delays are not uniform throughout the day. The *Priority Plus* features above allow the user account for these varying arrival times throughout the day. *TSD* and *TED* estimates are discussed in the next section.

NOTE: Care must be taken when programming the Shortway and Longway parameters to insure that the controller can transition properly to the priority phase. If there isn't adequate times for short or long, then strategy table 99 may be reported which has the effect of not running Transit Priority due to errors in user programmable data.

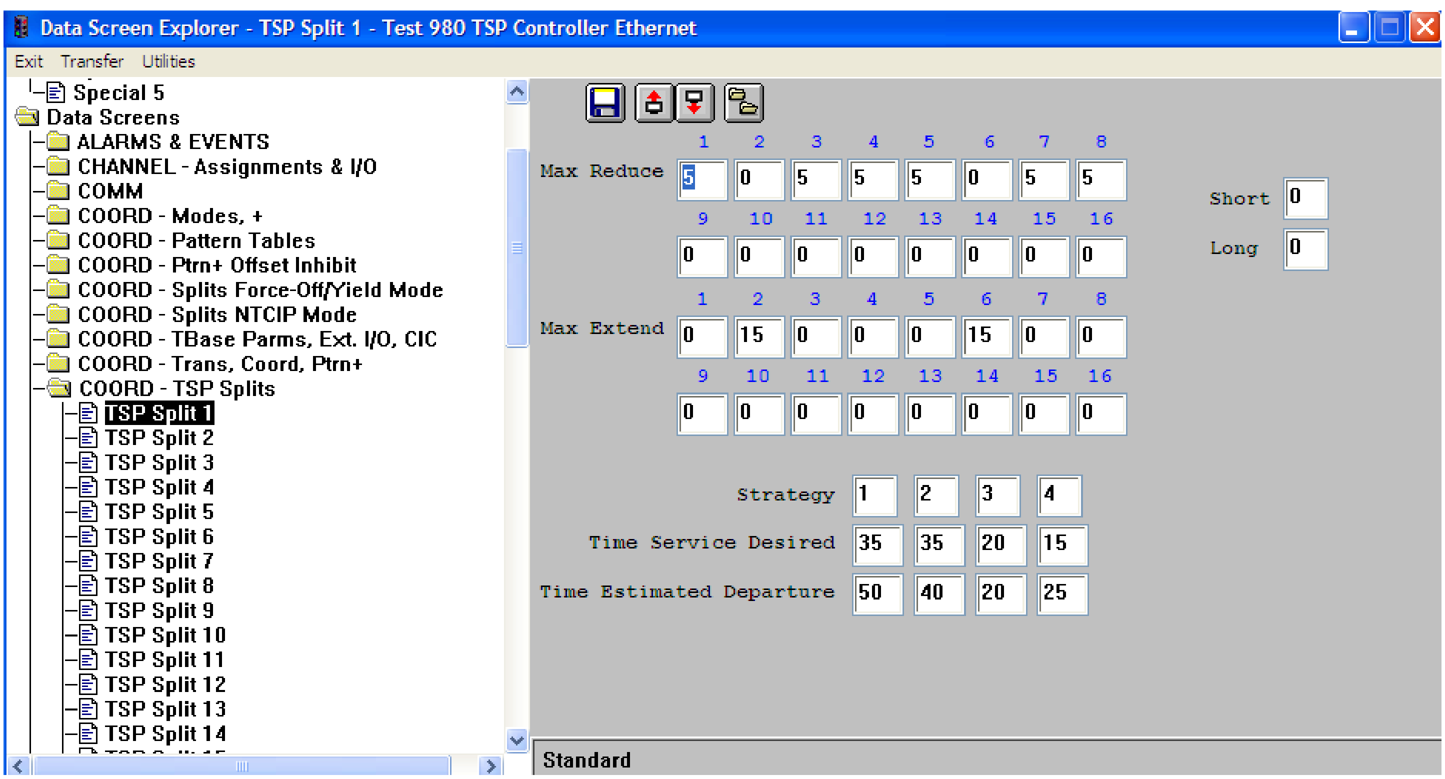
2.4 Streetwise Programming Screens

The following screens are displayed for user information

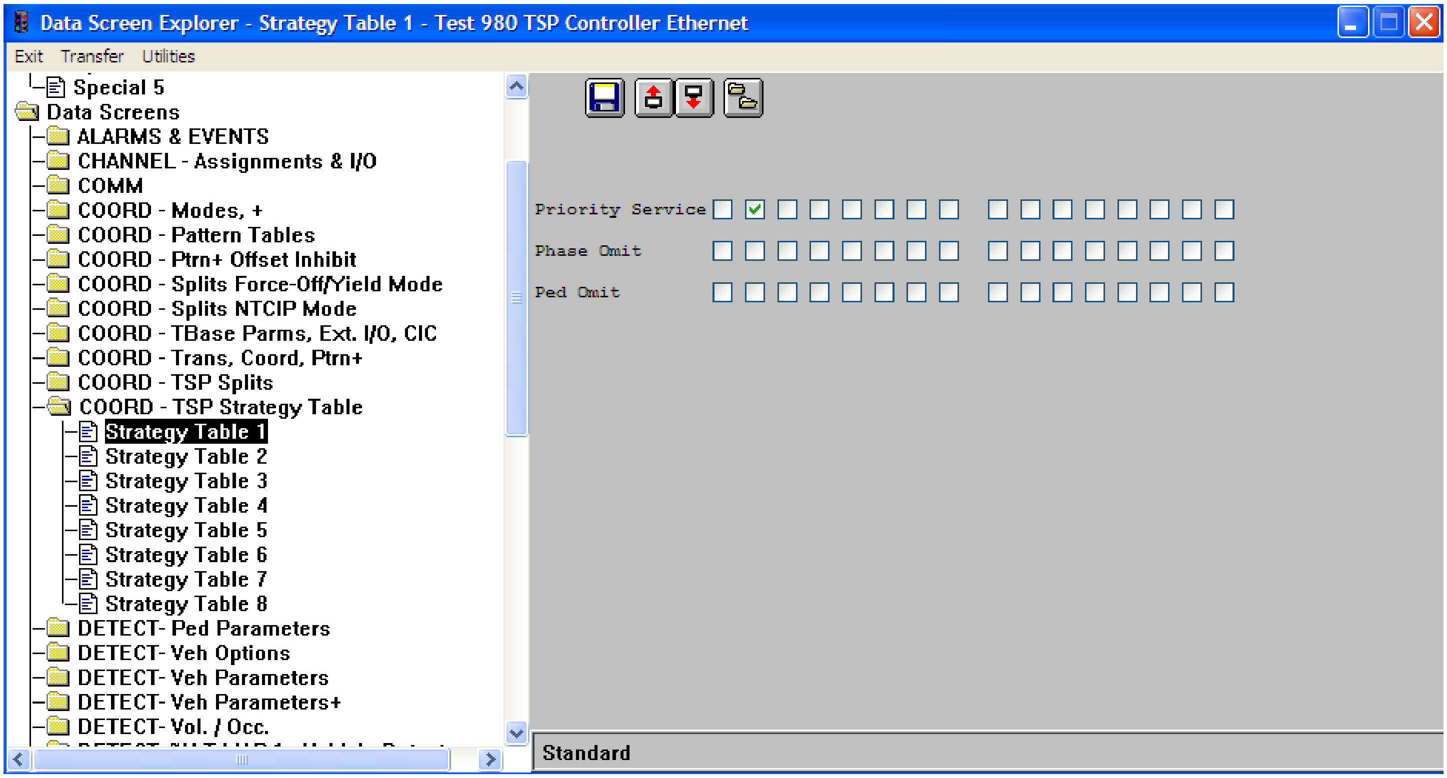
2.4.1 Low Priority Preemption Programming



2.4.2 TSP Split screen



2.4.3 TSP Strategy tables



2.5 ATMS.now Programming screens

The following screens are displayed for user information

2.5.1 Low Priority Preemption Programming

Controller Database Editor - Preempt LP

Preempt	Min	Max	Type	Platoon Rx	Cond Lockout	Coord in Preempt	Platoon Tx	Lock	Begin Mode	Priority P1	Priority P2	Priority P3	Priority P4	Max Lockout
Preempt 1	0	0	T_PRIOR	OFF	OFF	OFF	OFF	ON	SKIP	2	0	0	0	1
Preempt 2	0	0	T_PRIOR	OFF	OFF	OFF	OFF	ON	SKIP	2	6	0	0	1
Preempt 3	0	0	T_PRIOR	OFF	OFF	OFF	OFF	ON	SKIP	2	0	0	0	1
Preempt 4	0	0	T_PRIOR	OFF	OFF	OFF	OFF	ON	SKIP	4	8	0	0	1

Table 1

Preempt LP Alt | Preempt LP | Preempt+ | Preempt

Database Configuration: Standard | Filter: Preemption | Apply | OK

y Controller | By Entry

Controller Database Editor - Preempt LP

Preempt	Ext Dwell	Ant Arrival	Max Grn 1	Max Grn 2	Max Grn 3	Max Grn 4	Max Grn 5	Max Grn 6	Max Grn 7	Max Grn 8	Max Grn 9	Max Grn 10	Max Grn 11	Max Grn 12
Preempt 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Preempt 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Preempt 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Preempt 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 1

Preempt LP Alt | Preempt LP | Preempt+ | Preempt

Database Configuration: Standard | Filter: Preemption | Apply | OK

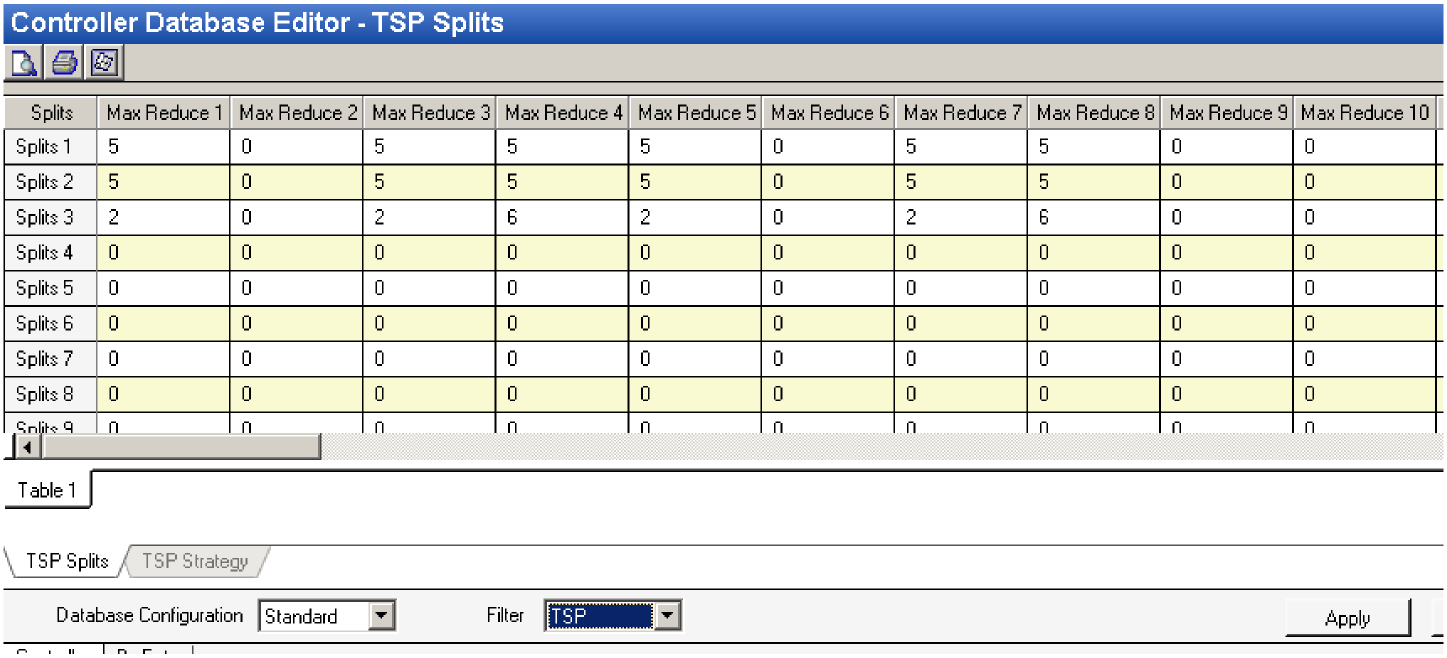
Controller Database Editor - Preempt LP

Preempt	Max Grn 13	Max Grn 14	Max Grn 15	Max Grn 16	Headway Group	Queue Jump	Headway Time	TX Time	PP Hold Time	PP Tx Phase 1	PP Tx Phase 2	PP Tx Phase
Preempt 1	0	0	0	0	ON	ON	3	0	0	0	0	0
Preempt 2	0	0	0	0	ON	ON	3	0	0	0	0	0
Preempt 3	0	0	0	0	ON	ON	3	0	0	0	0	0
Preempt 4	0	0	0	0	ON	ON	3	0	0	0	0	0

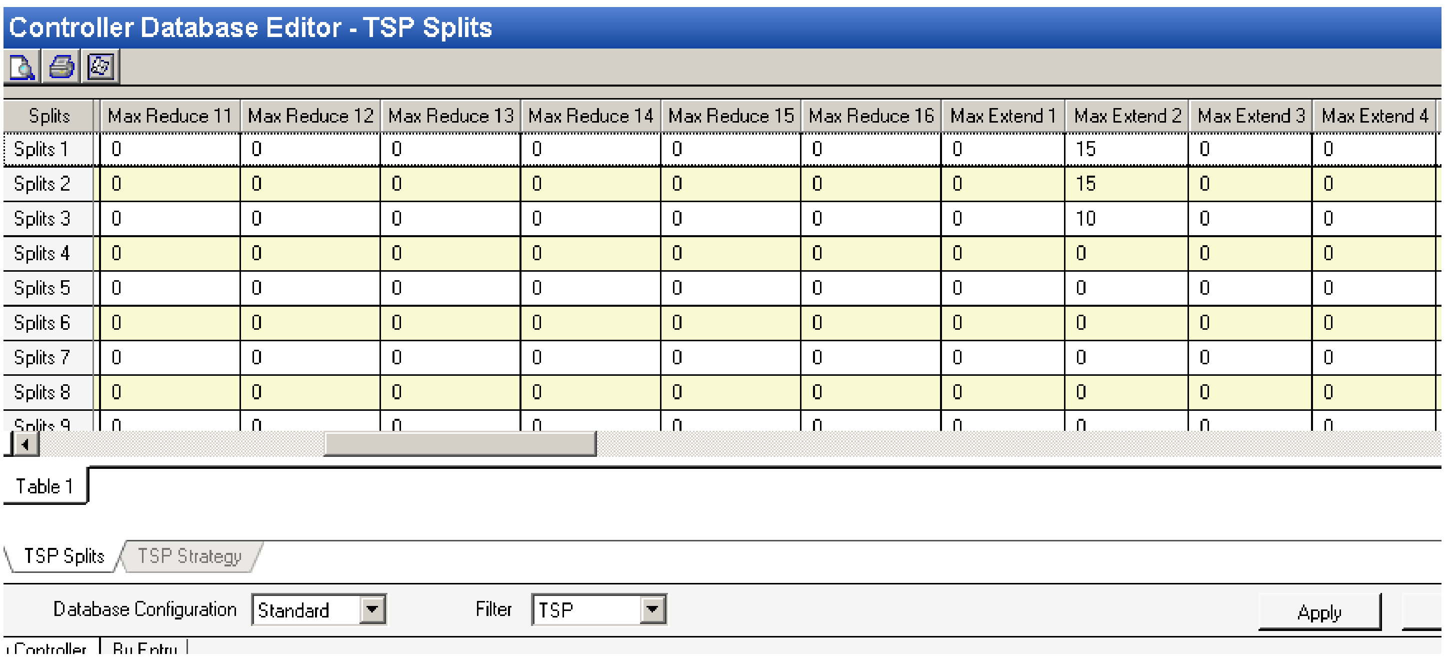
Table 1

Preempt LP Alt | Preempt LP | Preempt+ | Preempt

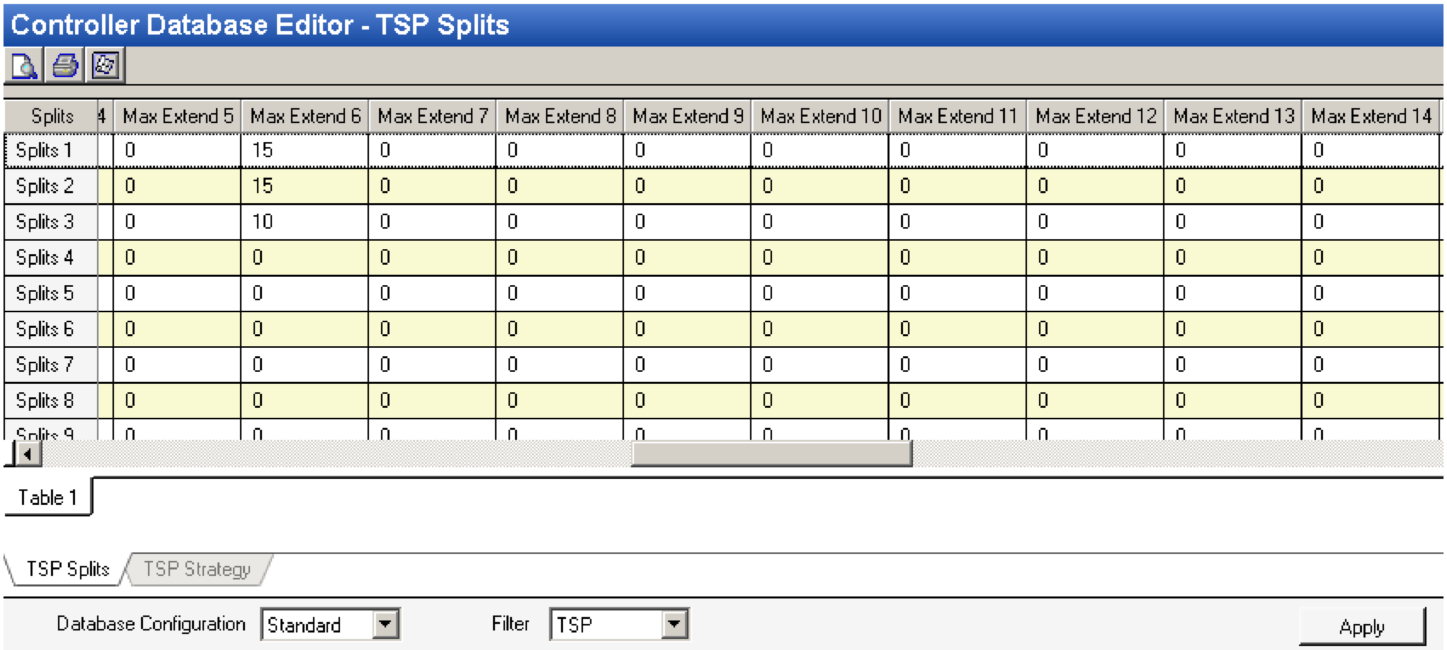
2.5.2 TSP Split screen



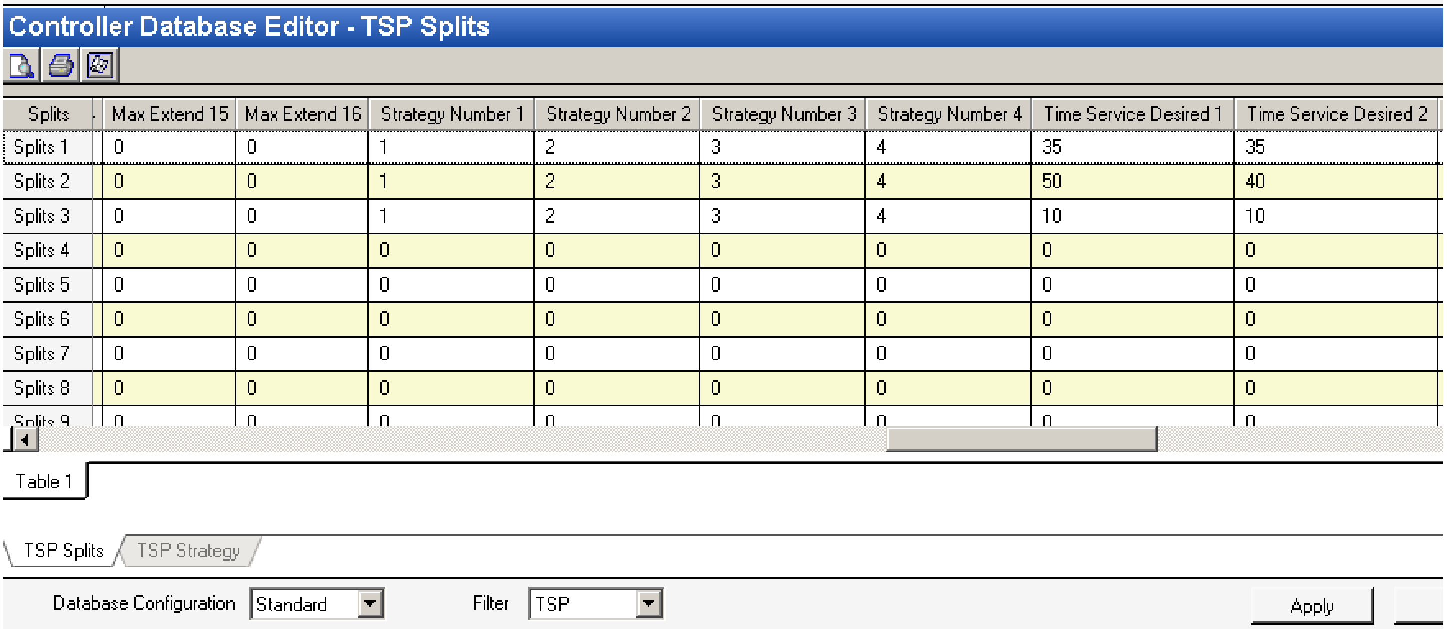
Splits	Max Reduce 1	Max Reduce 2	Max Reduce 3	Max Reduce 4	Max Reduce 5	Max Reduce 6	Max Reduce 7	Max Reduce 8	Max Reduce 9	Max Reduce 10
Splits 1	5	0	5	5	5	0	5	5	0	0
Splits 2	5	0	5	5	5	0	5	5	0	0
Splits 3	2	0	2	6	2	0	2	6	0	0
Splits 4	0	0	0	0	0	0	0	0	0	0
Splits 5	0	0	0	0	0	0	0	0	0	0
Splits 6	0	0	0	0	0	0	0	0	0	0
Splits 7	0	0	0	0	0	0	0	0	0	0
Splits 8	0	0	0	0	0	0	0	0	0	0
Splits 9	0	0	0	0	0	0	0	0	0	0



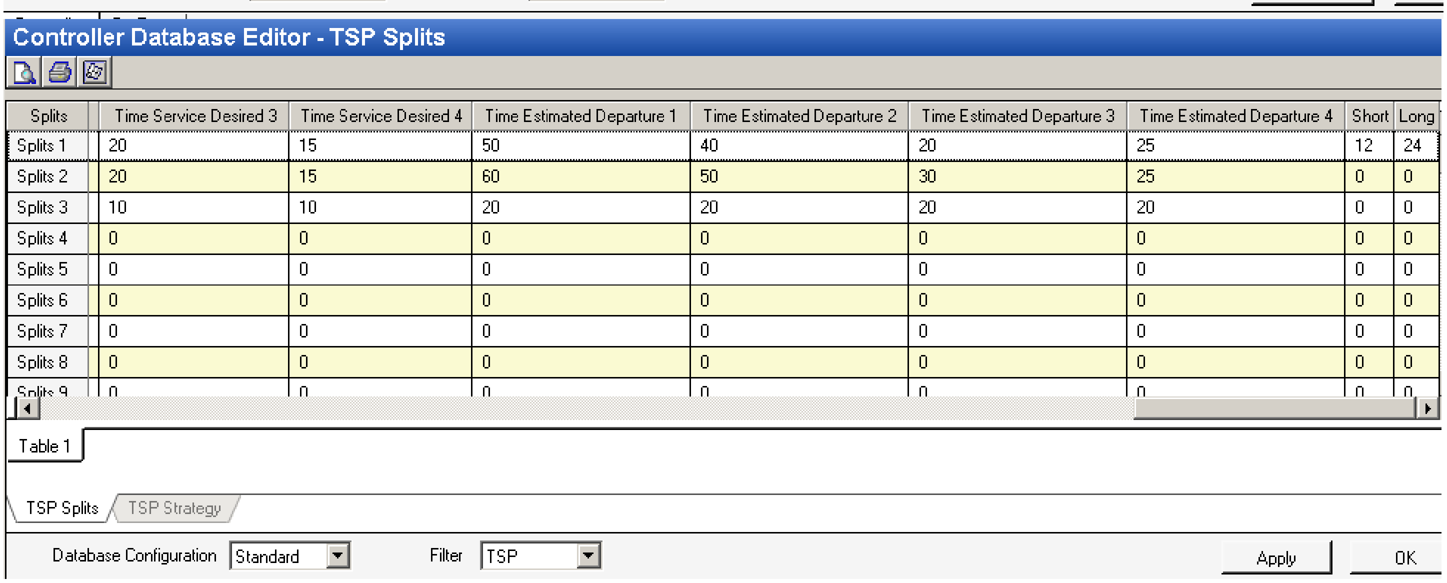
Splits	Max Reduce 11	Max Reduce 12	Max Reduce 13	Max Reduce 14	Max Reduce 15	Max Reduce 16	Max Extend 1	Max Extend 2	Max Extend 3	Max Extend 4
Splits 1	0	0	0	0	0	0	0	15	0	0
Splits 2	0	0	0	0	0	0	0	15	0	0
Splits 3	0	0	0	0	0	0	0	10	0	0
Splits 4	0	0	0	0	0	0	0	0	0	0
Splits 5	0	0	0	0	0	0	0	0	0	0
Splits 6	0	0	0	0	0	0	0	0	0	0
Splits 7	0	0	0	0	0	0	0	0	0	0
Splits 8	0	0	0	0	0	0	0	0	0	0
Splits 9	0	0	0	0	0	0	0	0	0	0



Splits	4	Max Extend 5	Max Extend 6	Max Extend 7	Max Extend 8	Max Extend 9	Max Extend 10	Max Extend 11	Max Extend 12	Max Extend 13	Max Extend 14
Splits 1		0	15	0	0	0	0	0	0	0	0
Splits 2		0	15	0	0	0	0	0	0	0	0
Splits 3		0	10	0	0	0	0	0	0	0	0
Splits 4		0	0	0	0	0	0	0	0	0	0
Splits 5		0	0	0	0	0	0	0	0	0	0
Splits 6		0	0	0	0	0	0	0	0	0	0
Splits 7		0	0	0	0	0	0	0	0	0	0
Splits 8		0	0	0	0	0	0	0	0	0	0
Splits 9		0	0	0	0	0	0	0	0	0	0



Splits	Max Extend 15	Max Extend 16	Strategy Number 1	Strategy Number 2	Strategy Number 3	Strategy Number 4	Time Service Desired 1	Time Service Desired 2
Splits 1	0	0	1	2	3	4	35	35
Splits 2	0	0	1	2	3	4	50	40
Splits 3	0	0	1	2	3	4	10	10
Splits 4	0	0	0	0	0	0	0	0
Splits 5	0	0	0	0	0	0	0	0
Splits 6	0	0	0	0	0	0	0	0
Splits 7	0	0	0	0	0	0	0	0
Splits 8	0	0	0	0	0	0	0	0
Splits 9	0	0	0	0	0	0	0	0



Splits	Time Service Desired 3	Time Service Desired 4	Time Estimated Departure 1	Time Estimated Departure 2	Time Estimated Departure 3	Time Estimated Departure 4	Short	Long
Splits 1	20	15	50	40	20	25	12	24
Splits 2	20	15	60	50	30	25	0	0
Splits 3	10	10	20	20	20	20	0	0
Splits 4	0	0	0	0	0	0	0	0
Splits 5	0	0	0	0	0	0	0	0
Splits 6	0	0	0	0	0	0	0	0
Splits 7	0	0	0	0	0	0	0	0
Splits 8	0	0	0	0	0	0	0	0
Splits 9	0	0	0	0	0	0	0	0

2.5.3 TSP Strategy Tables

Controller Database Editor - TSP Strategy

Strategy	Serv Ph 1	Serv Ph 2	Serv Ph 3	Serv Ph 4	Serv Ph 5	Serv Ph 6	Serv Ph 7	Serv Ph 8	Serv Ph 9	Serv Ph 10	Serv Ph 11	Serv Ph 12	Serv Ph 13	Serv Ph 14	Serv Ph 15	Serv Ph 16	Omit Ph 1
Strategy 1		YES															
Strategy 2		YES				YES											YES
Strategy 3		YES															
Strategy 4						YES											
Strategy 5																	
Strategy 6																	
Strategy 7																	
Strategy 8																	

Table 1

TSP Splits | TSP Strategy

Database Configuration: Standard — Filter: TSP — Apply — OK

Controller Database Editor - TSP Strategy

Strategy	Omit Ph 2	Omit Ph 3	Omit Ph 4	Omit Ph 5	Omit Ph 6	Omit Ph 7	Omit Ph 8	Omit Ph 9	Omit Ph 10	Omit Ph 11	Omit Ph 12	Omit Ph 13	Omit Ph 14	Omit Ph 15	Omit Ph 16	Omit Ped 1
Strategy 1																
Strategy 2				YES												
Strategy 3																
Strategy 4																
Strategy 5																
Strategy 6																
Strategy 7																
Strategy 8																

Table 1

TSP Splits | TSP Strategy

Database Configuration: Standard — Filter: TSP — Apply

Controller Database Editor - TSP Strategy

Strategy	mit	Omit Ped 2	Omit Ped 3	Omit Ped 4	Omit Ped 5	Omit Ped 6	Omit Ped 7	Omit Ped 8	Omit Ped 9	Omit Ped 10	Omit Ped 11	Omit Ped 12	Omit Ped 13	Omit Ped 14	Omit Ped 15	Omit Ped 16
Strategy 1																
Strategy 2																
Strategy 3																
Strategy 4																
Strategy 5																
Strategy 6																
Strategy 7																
Strategy 8																

Table 1

TSP Splits | TSP Strategy

Database Configuration: Standard — Filter: TSP — Apply — OK

3 Estimating Transit Vehicle Arrival Times

The key to implementing an efficient transit priority system is accurate arrival time estimates of the transit vehicle. These estimates can vary greatly by time-of-day, especially if a nearside transit stop is included in the estimate. Arrival times should be estimated during coord pattern development because transit priority programming is simply an extension of the pattern.

3.1 Time of Service Desired (TSD)

NTCIP defines *Time of Service Desired (TSD)* as the arrival time of the transit vehicle at the stop bar. This arrival time is compared with the start of the next transit priority phase to determine if phases should be reduced to provide an *early return* to the transit phase. The *TSD* estimate is based on the free flow speed of the transit vehicle and any deceleration, lost-time or dwell time expected prior to the vehicle arrival at the stop bar.

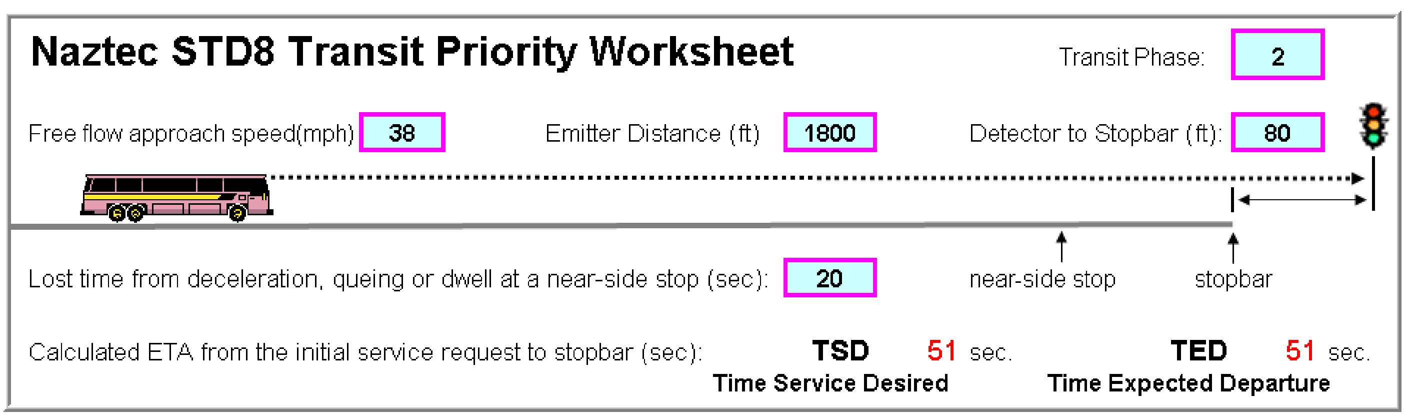
3.2 Time of Estimated Departure (TED)

NTCIP defines *Time of Estimated Departure (TED)* as the departure time of the vehicle clearing the intersection. This is also the point in the cycle when the emitter signal used to detect the transit vehicle is expected to drop out. The *TED* timer begins counting down when the priority request is received and this timer along with the presence of the emitter signal is used to make a decision to extend the transit phase at the force-off point.

The *TSD* and *TED* estimates are typically set to the same value because the decision to extend the priority phase is made one second before the force-off. If the *TSD* and *TED* countdown timers have timed to zero before this decision point, the transit phase does not need to be extended. However, if the *TED* countdown timer is greater than zero and less than *MaxExtend*, the transit phase should be extended because the vehicle has not reached the stop bar.

3.3 STD8 Transit Priority

The example below estimates *TSD* and *TED* arrival times for a transit vehicle approaching the intersection at 38 mph with an emitter distance of 1800 ft. The vehicle is expected to either dwell at a nearside stop or experience 20 seconds of delay before entering the intersection.



For the example above, the following calculations produced a TSD equal to 51 seconds.

a) 1 mi/hr is equivalent to 1.4667 ft/second, therefore a speed of 38 mph is equal to 55.73 ft/second.

b) The travel distance to the stop bar, once the transit priority preemption is detected, is 1800 ft – 80 ft or 1720 ft.

c) Therefore the travel time to the stop bar is 1720ft divided by 55.73 ft/second or 30.86 seconds

d) Add the lost time of 20 seconds and round up to get the TSD of 51 seconds. TED can be the same or greater based on agency practices and demands.

The calculated *TSD* and *TED* arrival times can be associated with the priority request (PR7-10) that use transit phase 2. *TSD* and *TED* times can be varied by pattern and by time-of-day because a separate *Priority Plus Features* table is provided with each *Split Table*.

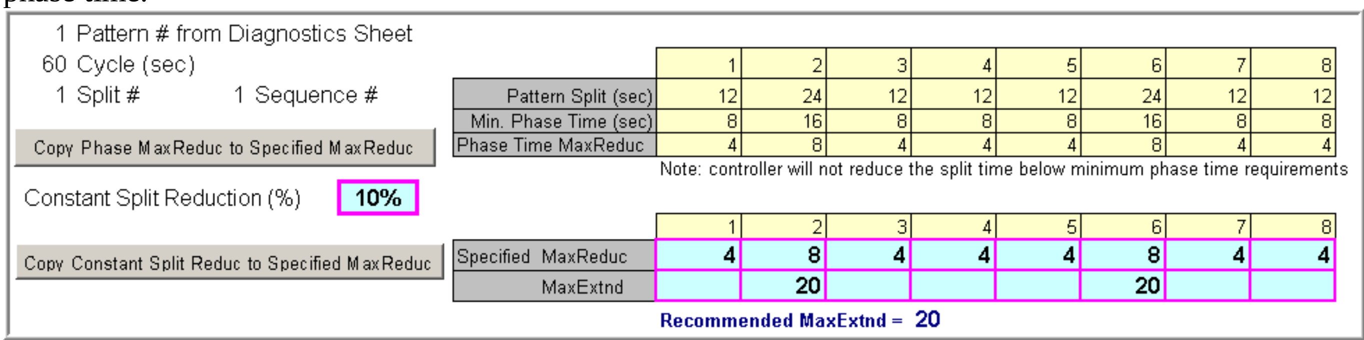
4 Modeling MaxReduce and MaxExtend Times

The next step in the evaluation is to model *MaxReduce* and *MaxExtend* and insure that transit diagnostics pass when the transit strategy table is requested.

4.1 Maximum Phase Reduction and Extend Times

The controller's internal *Coord Diagnostics* calculates minimum vehicle and pedestrian times shown below as *Min Phase Times(sec)*. These *min times* are guaranteed during priority service even if *MaxReduce* values exceed these minimums. However, the user should attempt to set *MaxReduce* that do not violate the *Min Phase Times* to understand the true *early return* and *extension* for each pattern. *MaxExtend* and *MaxReduce* must also balance to satisfy the transit diagnostics (section 7.4).

In the STD8 example below, a 60" cycle is programmed for pattern# 1 and split table #1. The coord phase and priority service phase 2 associated with the arterial is given a 24" split. Split times for non-transit phase splits are 12". The *Min Phase Times* calculated by the spreadsheet are 16" for the transit/coord phase and 8" for the non-transit phases. The calculated *Phase Time MaxReduce* value is the split time minus the min phase time.



1 Pattern # from Diagnostics Sheet
60 Cycle (sec)
1 Split # 1 Sequence #
Copy Phase MaxReduc to Specified MaxReduc
Constant Split Reduction (%) 10%
Copy Constant Split Reduc to Specified MaxReduc

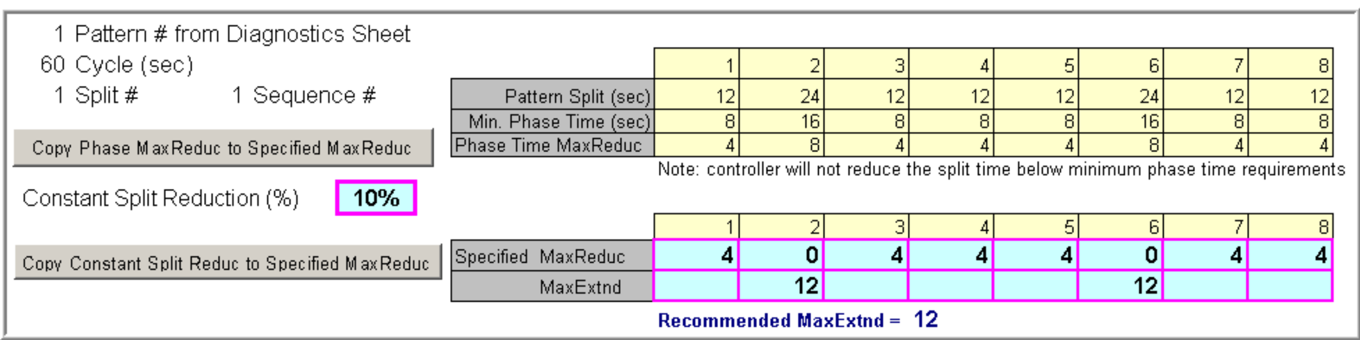
	1	2	3	4	5	6	7	8
Pattern Split (sec)	12	24	12	12	12	24	12	12
Min. Phase Time (sec)	8	16	8	8	8	16	8	8
Phase Time MaxReduc	4	8	4	4	4	8	4	4

Note: controller will not reduce the split time below minimum phase time requirements

	1	2	3	4	5	6	7	8
Specified MaxReduc	4	8	4	4	4	8	4	4
MaxExtnd		20				20		

Recommended MaxExtnd = 20

If you adjust the *Specified MaxReduce* for phases 2 and 6 to zero, notice below that the *Recommended MaxExtend* value changes from 20" to 12". In this case, *MaxExtend* for phase 2 and 6 should be set to this recommended value as shown below.



1 Pattern # from Diagnostics Sheet
60 Cycle (sec)
1 Split # 1 Sequence #
Copy Phase MaxReduc to Specified MaxReduc
Constant Split Reduction (%) 10%
Copy Constant Split Reduc to Specified MaxReduc

	1	2	3	4	5	6	7	8
Pattern Split (sec)	12	24	12	12	12	24	12	12
Min. Phase Time (sec)	8	16	8	8	8	16	8	8
Phase Time MaxReduc	4	8	4	4	4	8	4	4

Note: controller will not reduce the split time below minimum phase time requirements

	1	2	3	4	5	6	7	8
Specified MaxReduc	4	0	4	4	4	0	4	4
MaxExtnd		12				12		

Recommended MaxExtnd = 12

If the *Specified MaxReduce* value is less than the calculated *Phase Time MaxReduce,* then this is an error. Even though the controller will not reduce the split times below minimum phase time requirements, you should attempt to adjust *Phase Time MaxReduc* to correct any errors to understand the true benefit of transit priority.

This example is more complex than it would first appear because minimum phase time calculations take into consideration not only minimum phase times requirements, but, also shortway transition specified for the pattern, no-shortway phases and the stop-in-walk parameter.

4.2 "Practical" Phase Reduction and Extend Times

Maximum reduction times are rarely used in practice because the benefit of transit priority must be weighed against the reduction in capacity expected when non-transit split times are reduced. "Practical" phase reduction and extend times must balance the trade-off between transit delay reduction and driver delay associated with the non-transit phases.

Consider a *constant split reduction factor* which can be used as an aide in reducing split times by a constant value. This approach is helpful if you assume that the split times used equalize drive delay and you wish to penalize the non-transit phases equally.

In the example below, a 20% reduction was copied to *Specified MaxReduce* by clicking the button labeled *"Copy Constant Split Reduc to Specified MaxReduce"*.

1 Pattern # from Diagnostics Sheet
60 Cycle (sec)
1 Split # 1 Sequence #

	1	2	3	4	5	6	7	8
Pattern Split (sec)	12	24	12	12	12	24	12	12
Min. Phase Time (sec)	8	16	8	8	8	16	8	8
Phase Time MaxReduc	4	8	4	4	4	8	4	4

Copy Phase MaxReduc to Specified MaxReduc

Note: controller will not reduce the split time below minimum phase time requirements

Constant Split Reduction (%) **20%**

Copy Constant Split Reduc to Specified MaxReduc

	1	2	3	4	5	6	7	8
Specified MaxReduc	**2**	**0**	**2**	**2**	**2**	**0**	**2**	**2**
MaxExtnd		**6**				**6**		

Recommended MaxExtnd = 6

Early Return Calculation STD8 - Seq# 1

Note: Max Reduction will not violate Mins
Max Early Return Assumes Seq# 1

	1	2	3	4	5	6	7	8
MaxReduc Used (sec)	2	0	2	2	2	0	2	2
Max Early Return		6				6		
Capacity Reduction (% Split)	-17%		-17%	-17%	-17%		-17%	-17%

Note: Capacity Reduction (non-transit phases) vs. Early Retun is a tradeoff !!

STD8 FloatMx Calculations (MM->2->8->6 Status Screen)

Note: This is a simplied case for Seq# 1 when
the transit phase is 2/4/6/8

	1	2	3	4	5	6	7	8
Reduced Split (sec)	10	24	10	10	10	24	10	10
Yellow+All-red Clearance	3	3	3	3	3	3	3	3
Priority Based FloatMax	7	27	7	7	7	27	7	7

Specified MaxReduce values for coord phase 2 and 6 were manually adjusted to zero because in this case phase 2 is the priority service phase. *MaxExtend* for phase 2 and 6 were set to the *Recommended MaxExtend* value as in the last example. The maximum early return for this example is 6" assuming that the *TSD* can be projected far enough in advance of the priority phase to accumulate max reduction times from the non-transit phases.

Capacity Reduction is calculated for each phase based on reduced split times. In the example above, the 20% *Constant Split Reduction* yields a 17% *Capacity Reduction*. These values differ by 3% because the constant reduction% is rounded to the nearest second. *Constant Split Reduction* factor is only provided as an aide to assist the user while specifying *MaxReduce* and *MaxExtend* values.

Capacity reduction of the non-transit phases is a very complex issue that not only depends on the *Specified Max Reduc* times, but also the frequency of the priority requests and the accuracy of the transit arrival times. A "poorly timed" system can actually provide a negative impact to both transit operation and drivers serviced by the non-transit phases if the projected arrival times are inaccurate.

For example, suppose priority requests are generated every 10 minutes using a *TSD* that estimates a 30 second dwell time at a nearside transit stop, but, the transit vehicle only stops 25% of the time. In this case, a significant reduction in capacity for drivers serviced by the non-transit phases could be experienced with little benefit to the transit vehicle. The *Transit Priority Worksheet* was designed to provide a visual while assigning *MaxReduce* and *MaxExtend* values to the split table.

The *Transit Priority Worksheet* also calculates *Floating Max* (PrMx) times. These are for educational purposes only, because the controller automatically asserts priority max times as needed. Floating max, or *Priority Max (PrMx)* times are used to reduce or extend split times while a priority service request is active.

4.3 MaxReduce Times and Barrier Requirements

One final example is in order before we leave this section on *MaxReduce* and *MaxExtend* times. The illustration below shows that *Recommended MaxExtend* can never be more than the sum of the *MaxReduce* times with barrier constraints applied.

Suppose the following *MaxReduce* and *MaxExtend* times are applied to the 60" pattern below.

1 Pattern # from Diagnostics Sheet
60 Cycle (sec)
1 Split # 1 Sequence #
Copy Phase MaxReduc to Specified MaxReduc
Constant Split Reduction (%) 10%
Copy Constant Split Reduc to Specified MaxReduc

	1	2	3	4	5	6	7	8
Pattern Split (sec)	12	24	12	12	12	24	12	12
Min. Phase Time (sec)	8	16	8	8	8	16	8	8
Phase Time MaxReduc	4	8	4	4	4	8	4	4

Note: controller will not reduce the split time below minimum phase time requirements

	1	2	3	4	5	6	7	8
Specified MaxReduc	4	0	4	4	4	0	4	4
MaxExtnd		12				12		

Recommended MaxExtnd = 12

If *Min Phase Time (sec)* for phases 1 and 4 is raised from 8" to 12", the maximum reduction changes from 4" to 0". This changes the *Recommended MaxExtend* from 12" to 4". Even though non-transit phases in ring 2 can be reduced by 12", ring 1 limits the reduction to 4" because phase 1 and 4 can no longer be reduced.

1 Pattern # from Diagnostics Sheet
60 Cycle (sec)
1 Split # 1 Sequence #
Copy Phase MaxReduc to Specified MaxReduc
Constant Split Reduction (%) 10%
Copy Constant Split Reduc to Specified MaxReduc

	1	2	3	4	5	6	7	8
Pattern Split (sec)	12	24	12	12	12	24	12	12
Min. Phase Time (sec)	12	16	8	12	8	16	8	8
Phase Time MaxReduc	0	8	4	0	4	8	4	4

Note: controller will not reduce the split time below minimum phase time requirements

	1	2	3	4	5	6	7	8
Specified MaxReduc	0	0	4	0	4	0	4	4
MaxExtnd		4				4		

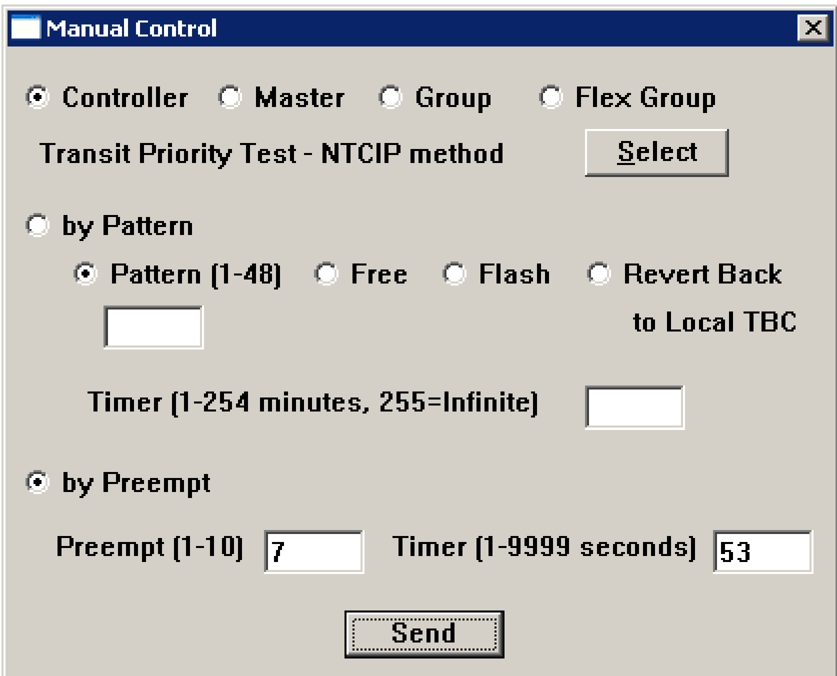
Recommended MaxExtnd = 4

The *Recommended MaxExtend* value calculated by the spreadsheet considers these barrier constraints. Therefore, the spreadsheet becomes a valuable aid when developing patterns because new *MaxReduce* and *MaxExtend* values should be considered whenever the split table is changed.

NOTE: Max Reduction cannot be programmed for the COORD Phase (or the Pseudo-COORD phase) when running end-of-green coordination.

5 Cubic | Trafficware Transit Priority During Coordination

The following examples demonstrate the *NTCIP* and *Cubic | Trafficware EZ* methods during coordination. Each example assumes the transit phase (priority service phase) and the coord phase are the same because this is the most common situation in practice. However, either method can assign the priority service phase to any phase in the sequence.



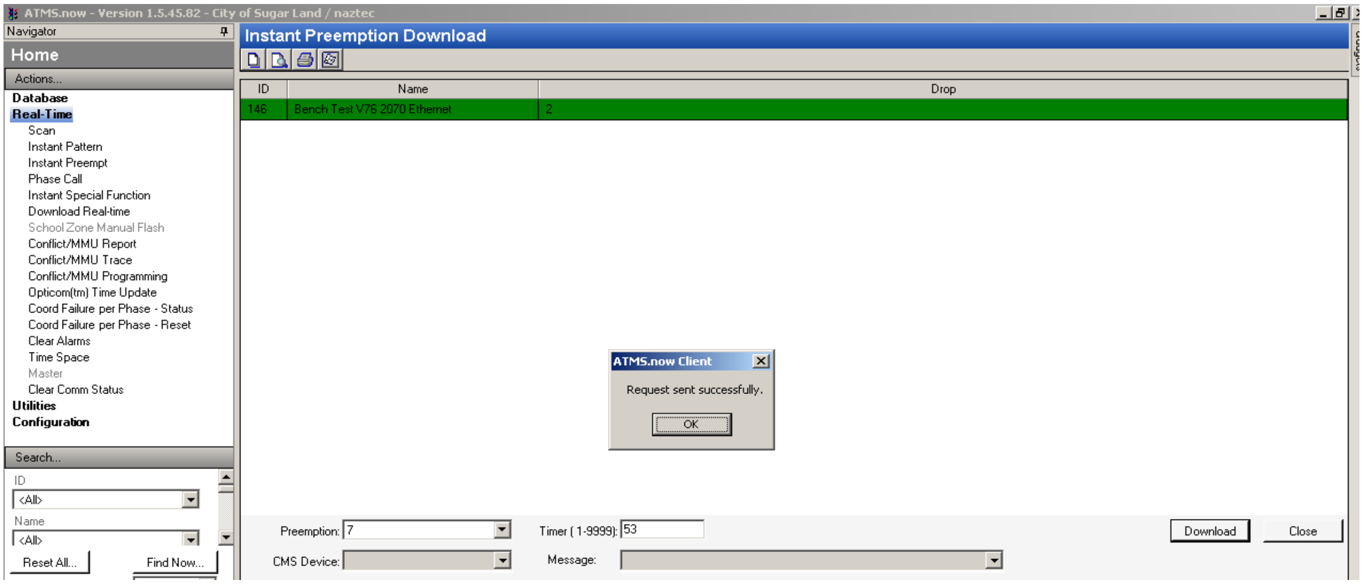
The user is encouraged to follow through the examples with a controller and observe the response when transit priority requests are generated from StreetWise or ATMS.now using *Manual Control*.

The *Manual Control* screen below is accessed from the StreetWise *Utilities* pull-down menu. In this example, service request PR7 is downloaded with a 53" duration timer to simulate an emitter that holds a low-priority call for 53".

Please set Lock to ON for the examples in this manual to make sure the service request does not drop out before the controller returns to the transit phase (see section 2.2.1).

For ATMS.now, the *Manual Control* screen is accessed from the Home module's Real-Time Action menu by accessing *Instant Preempt*. In this example, service request PR7 is downloaded with a 53" duration timer to simulate an emitter that holds a low-priority call for 53".

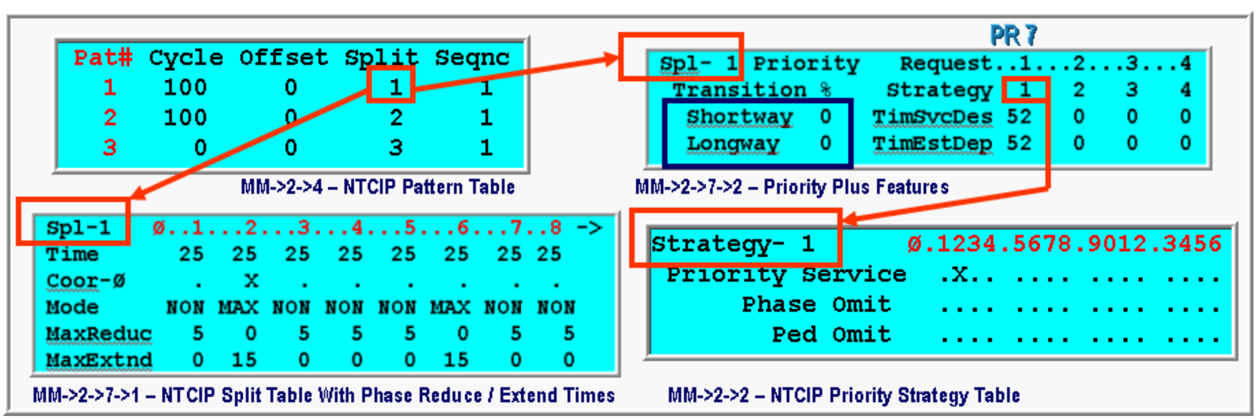
Please set Lock to ON for the examples in this manual to make sure the service request does not drop out before the controller returns to the transit phase.



5.1 Initialize the Test Controller

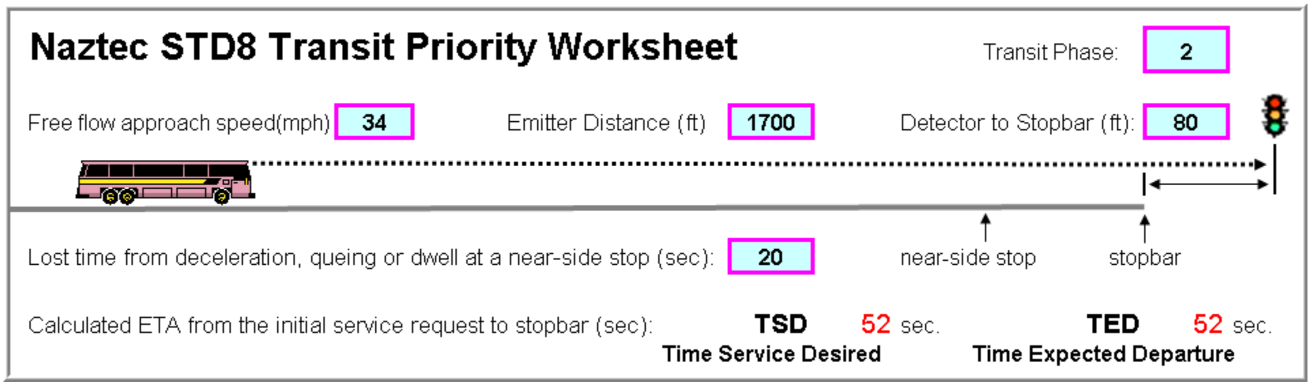
Please perform the following initialization to insure that the your results are consistent with those presented in this chapter.

1. Turn OFF the *Run-Timer* (MM->1->7) and initialize the controller as STD8 (MM->8->4)
2. Turn ON the *Run-Timer* (MM->1->7)
3. Set *Max Recall* on phases 1-8 under MM->1->1->2
4. Enter the values shown on the menus below. Make sure Coordination synch is referenced to BegGrn (MM->2->5) for pattern 1.
5. Set *Test OpMode* to 1 under MM->2->1. Verify that your controller is in pattern# 1 running a 100" cycle with max recalls on all phases from the MM->7->1 and 7->2 menus.
6. Define a controller in StreetWise with the same ID address under MM->6->1
7. Select this controller from the *Manual Control* screen above and verify that PR7 is received by looking at the controller timing status screen M->7->1 (this menu displays Pr7 in the lower right corner when PR7 becomes active)



5.2 NTCIP Priority Examples

In this example, assume that a transit vehicle is approaching at 34 mph and the emitter is detected at 1700 ft. There is also a 20" dwell time expected at a nearside bus stop. Therefore, the *TSD/TED* estimated time of arrival is 52" after the emitter is detected at the cabinet.

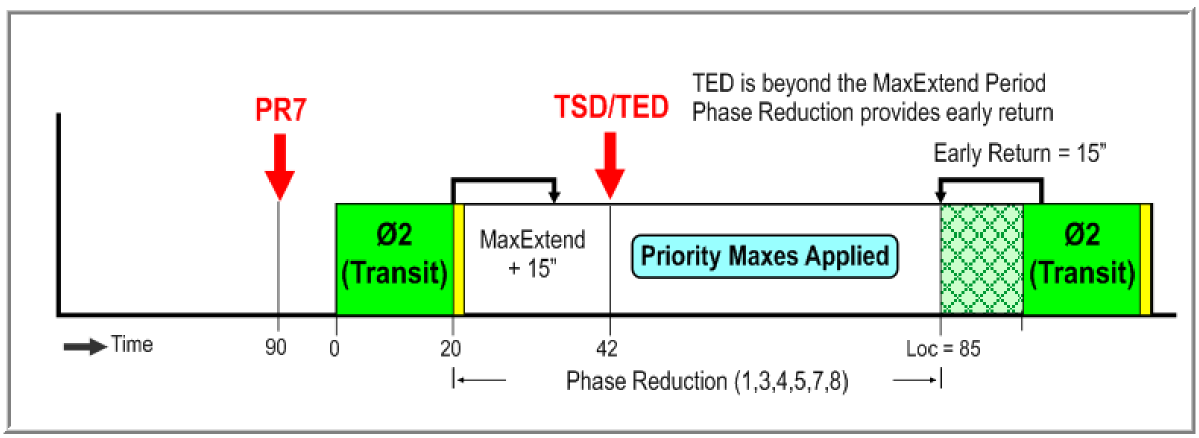


5.2.1 NTCIP Early Return Example (Priority Request in the Previous Cycle)

Please set LP Lock to ON for the examples in this manual to make sure the service request does not drop out before the controller returns to the transit phase (see section 2.2.1).

Issue a PR7 request from StreetWise so "Pr7" appears on MM->7->1 when the local cycle counter reaches Loc=90. You may need to click the "Send" button on the StreetWise *Manual Control* screen several seconds early depending on your communication setup to insure that "Pr7" appears at Loc=90.

Because the *TSD/TED* times are set to 52", the arrival time of the vehicle at the stop bar is projected to be at Loc=42 (90" + 52" – 100" = 42"). In this case, TSD/TED is projected into the next signal cycle.



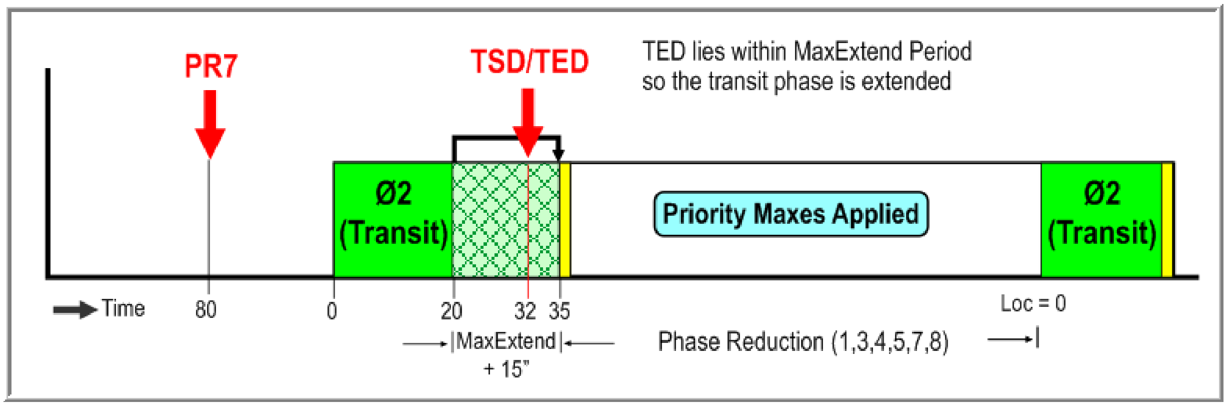
Phase Reduction (Cubic | Trafficware NTCIP Method) – Priority Max Times Provide Early Return to the Transit Ø

Note that the *TED* is projected after the ***MaxExtend*** period of the transit phase, so the non-transit phases must be reduced to provide an *early return* in the next signal cycle. Phase 2 normally begins at Loc=0 and ends at the force-off at Loc=22. However, during priority service, phase 2 returns 15" early at Loc=85 because the non-transit phases are reduced by applying *Priority Max (PrMx)* times.

Run this simulation several times observing the max times operating in each ring on status screen MM->7->1. Before PR7 is applied, you will notice *Max1* timing in each ring because this is the default max time in coordination under MM->2->1. However, when "Pr7" is active and phases are being reduced, you will see *PrMx* in effect which limits the phases from developing the 25" split time assigned in the *Split Table* for each phase. You can view the calculated *Priority Max* times under the *Trans Calcs* menu (MM->2->8->6, labeled as "*FloatMx"*).

5.2.2 NTCIP Phase Extension Example

Now, issue a PR7 request at Loc=80. The *TSD/TED* time is 52" and projects the vehicle arrival at Loc=32. The arrival is within the *MaxExtend* period as shown below. The controller recalculates the force-off and yield points at Loc=19 to extend the transit phase and reduce the phases that follow.



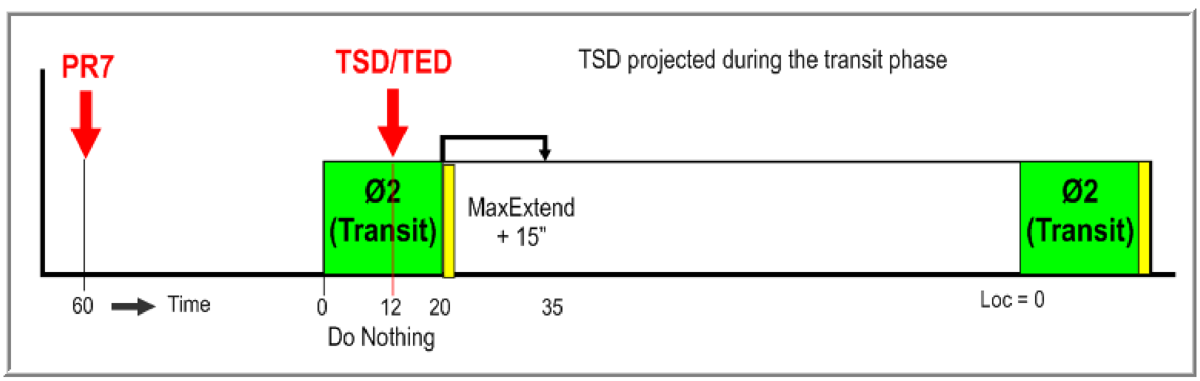
Phase Extension (Cubic | Trafficware NTCIP Method) – Force-offs Are Re-calculated to Extend the Transit Ø

Priority Max times (PrMx) are also applied to the phases that follow the transit phase to insure that any slack time developed in the non-transit phases is forwarded to coord phase.

In most cases, the transit phase is the same as the coordinated phase because transit typically operates on the major street. However, any phase may be assigned as the priority service phase. It is suggested that only one priority service phase be assigned to each request because if lead/lag left-turn phasing is in use, the begin and end points of the two phases may be at different points in the cycle.

5.2.3 NTCIP "Do Nothing" Example

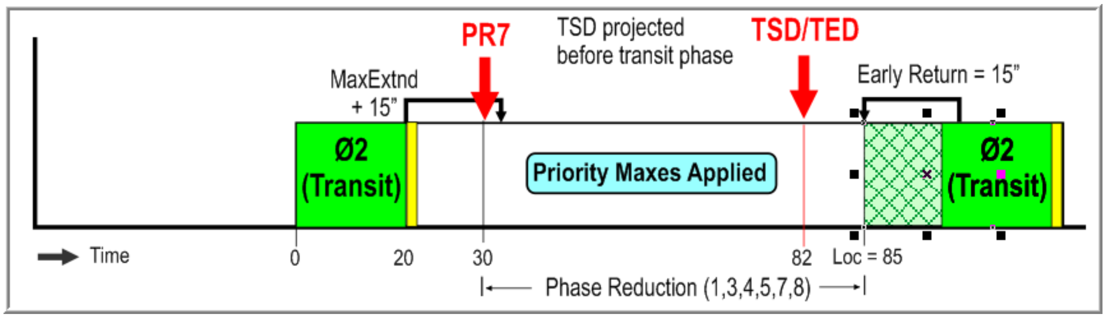
Next, issue a PR7 request at Loc=60 and verify that phase reduction and extension is not applied during the "Do Nothing" case below. Even if no action is taken, PR7 will lock out any new priority service request until the *TSD/TED* timers and the ***MaxL/O*** timer associated with the request have expired. The ***MaxL/O*** timer insures that all phases with calls at the time the request is received have been serviced before honoring another low-priority request (see section 2.2.3).



Do Nothing (Cubic | Trafficware NTCIP Method) – TSD/TED Projected During the Transit Ø

5.2.4 NTCIP Early Return Example – Service Request and TSD/TED in the Same Signal Cycle

Finally issue a PR7 request from StreetWise at Loc=30 and observe the early return to the transit phase at Loc=85. Note that *priority max times (PrMx)* are applied immediately when PR7 is detected even though phases 3 and 7 are already timing. Phase reduction provides an *early return* of 15" to the transit phase at Loc=85 to service a transit vehicle projected to arrive at Loc=82.



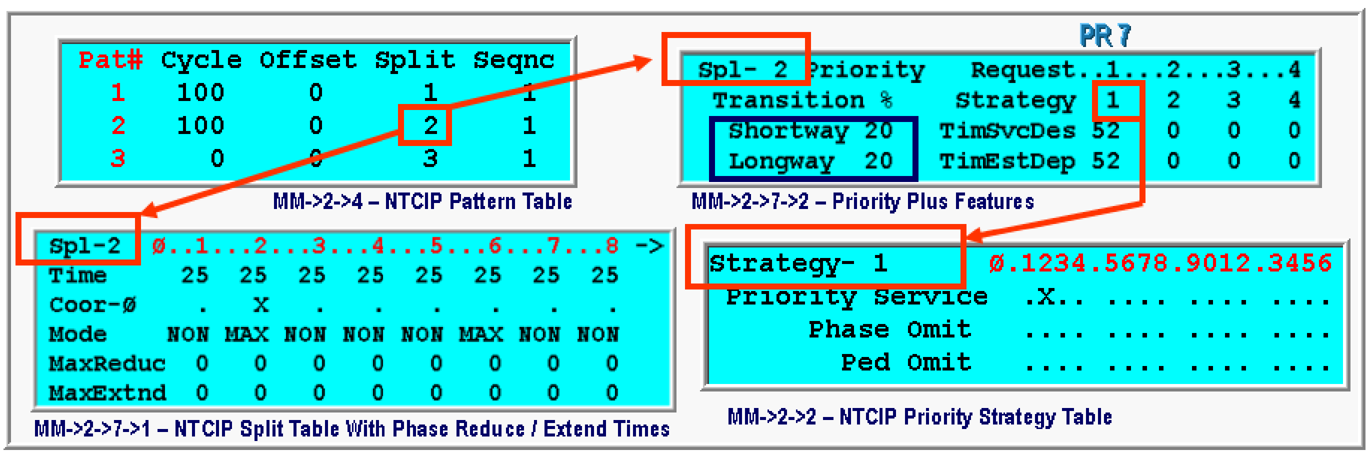
Phase Reduction (Cubic | Trafficware NTCIP Method) – Service Request and TSD/TED Occur Before Transit Ø

5.3 Cubic | Trafficware EZ Method Examples

In the last set of examples, you observed priority service running under the *NTCIP* method by watching the *Local* cycle counter from the timing status screen (MM->7->1).

To fully understand the *EZ Transit* method, you need to observe the coord status screen (MM->7->2) to observe the *Loc* and *Tbc* cycle counters and the offset transition method being applied.

Program a test controller using the values below and run pattern# 2 in TEST mode (MM->2->1). Notice that the ***MaxReduce*** and ***MaxExtend*** values in *Split Table* 2 are zero and that ***Shortway*** and ***Longway*** are 20% (the *Cubic | Trafficware EZ* method is enabled when ***Shortway*** and ***Longway*** are non-zero).



Please set LP Lock to ON for the examples in this manual to make sure the service request does not drop out before the controller returns to the transit phase (see section 2.2.1).

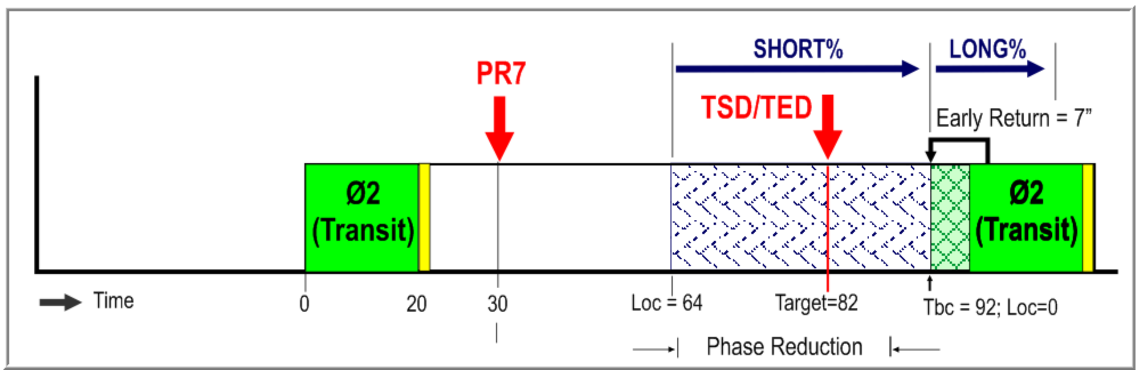
5.3.1 EZ Transit Phase Reduction Example – Service Request and TSD/TED in the Same Cycle

Repeat the last simulation by asserting PR7 at Loc=30 and observe the coord status (MM->7->2). Notice that the ***Target*** value (projected TSD/TED) displays 82" as soon at the request is received. The ***Transit*** value shown is the maximum *Early Return* (18") needed to move the start of the transit priority service phase in line with the *TSD/TED* arrival.

```
 OpModes.Src-TEST    Cycle     Ofst   0 7:44:35
 Sys-    0  Actv-   2  Loc-   30  Actu:   0 ACTIV
 Tbc-    0 Next-    2  Tbc-   30  Err:    0
 Ext-    0  Remo-   0  Prog-100  Targ: 82 SYNC
 Tod-    0 Test-    2         Transit: 18
Alt:.Opt.Time.Det.CIR
      0    0   0   0
```

EZ Transit - Coord Status (MM->7->2) When PR7 is First Applied

Under *EZ Transit*, the controller may apply ***Shortway%*** AND ***MaxReduce*** and *Priority Max* to achieve early return, but in this example, ***MaxReduce*** values are zero and only 20% shortway is used for phase reduction.



Phase Reduction (Cubic | Trafficware EZ Method) – Service Request and TSD/TED Before the Transit Ø

The controller enters shortway transition at Loc=64. The actual early return provided to phase 2 during the 100" cycle is 7", calculated as 0.2 * (100" – 64").

Re-assert PR7 at Loc=30 and observe the coord status menu (MM->7->2). At Loc=64, the coord status changes from SYNC to SHORT and you can see the *Loc* and *Tbc* cycle counters begin to move apart as the controller runs shortway. The controller enters phase 2 at *Tbc*=92 and *Loc*=0 and immediately switches to LONG transition to regain coordination before the end of phase 2.

```
 OpModes.Src-TEST     Cycle     Ofst  07:44:35
 Sys-    0  Actv-   2  Loc-    0  Actu:   0 ACTIV
 Tbc-    0 Next-    2  Tbc-   92  Err:+   7
 Ext-    0  Remo-   0  Prog-100  Targ: 82 LONG
 Tod-    0 Test-    2         Transit: 18
Alt:.Opt.Time.Det.CIR
      0    0   0   0
```

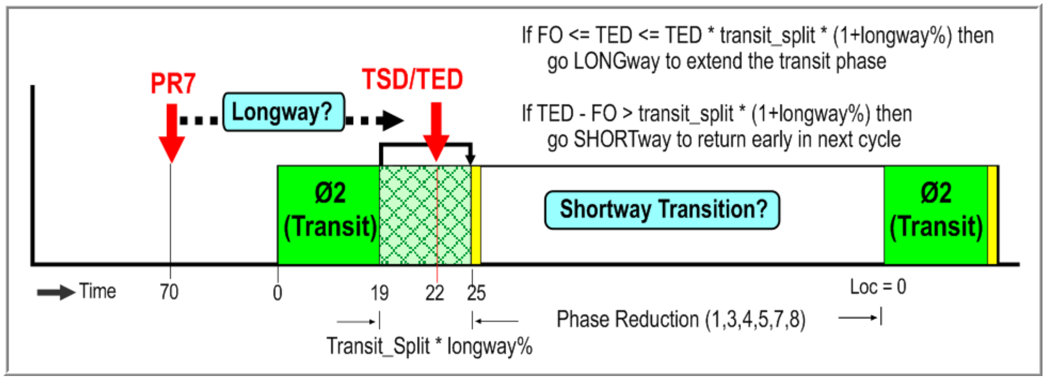
EZ Transit - Coord Status (MM->7->2) - Return to the Transit Phase

In this case, the *Cubic | Trafficware EZ* method provides a 7" early return to the transit phase using *Shortway%*. This compares with an early return of 15" under the *NTCIP* method in section 5.2.4. Both methods can be applied to STD8, QSeq, 8Seq and USER phase mode operation. In addition, *EZ Transit* simplifies coding phase reduce and extend times for complex USER mode sequences (16 phases assigned to 4 rings.

5.3.2 Easy Transit Phase Extension Example

Please set LP Lock to ON for the examples in this manual to make sure the service request does not drop out before the controller returns to the transit phase (see section 2.2.1).

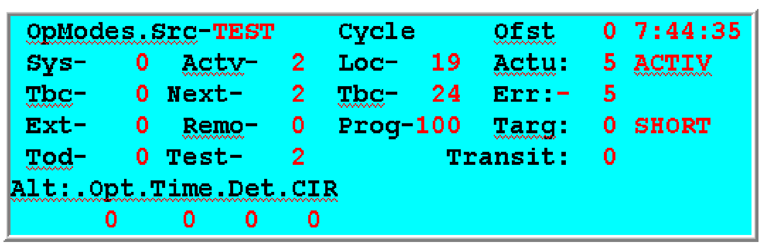
Assert a PR7 request at Loc=70 while observing the coord status screen (MM->7->2). The 52” *TSD/TED* projects the arrival of the transit vehicle during the extendable period of the transit phase as shown below. The maximum extension is the transit phase split time * longway% (25” * 0.20 = 5”).



Phase Extension (Cubic | Trafficware EZ Method) – Service Request and TSD/TED Occur Before Transit Ø

The coordinator looks to see if the *TED* lies within the extension of the transit phase one second before the force-off of phase 2 (Loc=19). If the *TED* lies within this extension and the input request from the emitter is still active, the local counter holds at the force-off (Loc=20) for 5 seconds and then releases. You may *Lock* the priority request as explained in section 2.2.1 if you wish to service the request even if the emitter input has dropped out. However, if *Lock* is OFF and the emitter drops out prior to Loc=19, the priority service phase will not be extended.

After the extension, the controller immediately enters transition by going shortway (SHORT). Run this simulation several times and observe the offset correction taking place on the coord status screen (MM->7->2). It is important to note that the controller returns to coord phase 2 in sync at Loc=0. This is essentially the same operation used to allow pedestrian clearance to overrun the force-off when the STOP-IN-WALK feature is set ON.



EZ Transit Phase Extension- Local Counter Stopped at Loc=19

Issue PR7 four seconds earlier at Loc = 66. The transit phase will not be extended, because the *TED* lies within the transit green period and no extension is needed. This is the “Do Nothing” case discussed in section 5.2.3 for the *NTCIP* method.

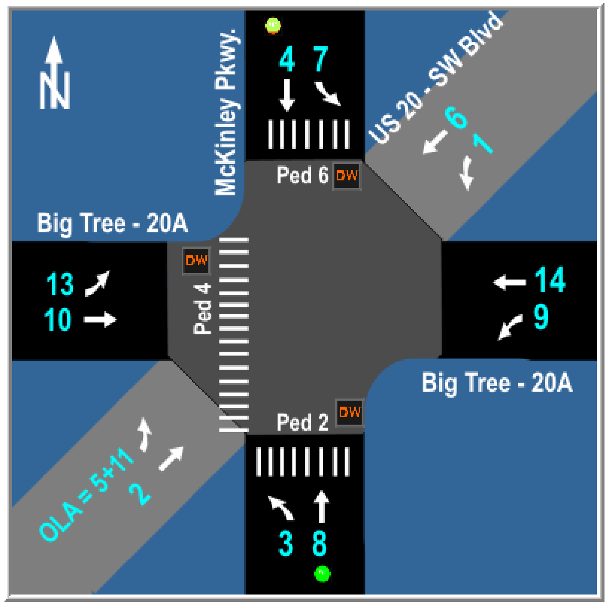
If you place the PR7 service request greater than Loc=73, the *TED* is projected into the next cycle and the coordinator provides an early return to the priority service phase using shortway% as discussed in section 5.3.1 above.

The *Cubic | Trafficware EZ* method applies a uniform split reduction to all non-transit phases, so the capacity reduction penalty is the same. This method must be used for USER phase mode configurations to provide an *early return* or phase *extension* to user defined phase sequences of 16 phases assigned to 4 rings.

5.3.3 Example of the Cubic | Trafficware EZ Method Applied to a USER Phase Sequence

The signal operation shown to the right exceeds the capability of STD8. The *Phase Mode* unit parameter (MM->1->2->1) must be set to USER to apply the following phase sequence:

Ring	Sequence of Phases							
1	1	2	0	4	3	9	10	0
2	5	6	11	7	8	14	13	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0



Suppose transit vehicles are operating on four of the six approaches to this intersection. If additional phases were required to implement *Maxextend* and *MaxReducee* times in this sequence, the application would require 26 phases and would be extremely complex to program in coordination.

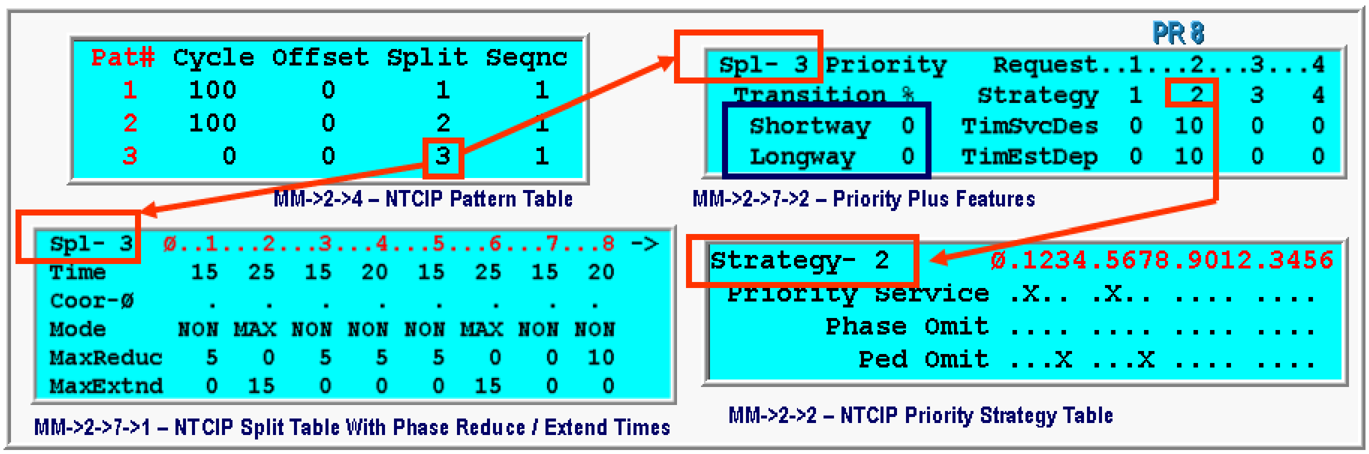
The *Cubic | Trafficware EZ* method allows the user to simply assign a *Priority Strategy Table* and *TSD/TED* arrival times for each request (PR7-PR10) in the plus features of each split table. *Shortway* and *Longway* percent can be used to provide an early return or extend the priority service phase associated with each request. Assigning additional phases to achieve transit priority in this case would be extremely difficult and would make coordination development overly complex.

6 Transit Priority in Free Operation

NTCIP 1211 defines transit priority only during coordinated operation. Cubic | Trafficware extends the *NTCIP* method during free operation using *Priority Max* times to provide an early return or extend the priority phase. In addition, the *Priority Strategy Table* may be used to selectively omit vehicle and/or pedestrian service. Even if you only plan to use transit priority during free operation, you should still familiarize yourself with this chapter to learn more about the *Priority Strategy Table* and the controller status displays related to transit priority.

6.1 Free Patterns and Priority Service

NTCIP 1202 – "ASC (Actuated Traffic Signal Controllers)" defines *free operation* as pattern# 254 and *automatic flash* as pattern# 255. In addition, any of the 48 controller patterns may be set to *free* by coding a 0" cycle length in the pattern table. This approach allows additional features associated with the pattern to be active during free operation. For example, transit priority is active during free pattern *# 3* below:



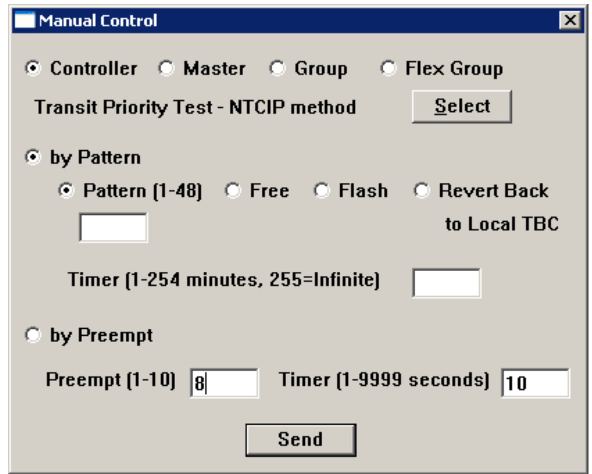
For these examples, set phases 1-8 on *Max Recall* and phases 4 and 8 on *Ped Recall* (MM->1->1->2).

Force the controller to *pattern # 254* in TEST mode (set *Test, OpMode* to 2 under MM->2->1). Then observe the operation of the controller under status screens MM->7->1 and MM->7->2. Note that the active *Max1* times for each phase match the *Max1* values programmed under MM->1->1->1. If you do not see ped recalls for phases 4 and 8, go back and program these recalls under MM->1->1->1.

Now, force the controller to *pattern # 3* in TEST mode (set *Test, OpMode* to 3 under MM->2->1). Note that the active *Max1* times change to the split times entered in *split table 2* under MM->7->1. During free operation, if a split time is zero, the *Max1* value from phase times (MM->1->1->1) is applied. However, if the split time is non-zero, the split value overrides the active max time while the split table is active.

This approach provides 32 combinations of *Max Times* (one combination for each split table) during free operation. You can still apply *Max1* or *Max2* using an external input to the controller or by setting *Max2* for the pattern (MM->2->6, right menu). However, setting max times through the split table is more powerful and lets you modify the max times for priority service under free operation.

6.2 Free Operation Priority Example



With the controller in free pattern# 3, place a priority service request for service request PR8 while the controller is in phase 2 yellow clearance.

Observe under MM->7->1 that the *Max1* times switch to *PrMx* when PR8 appears in the lower right hand corner of the menu. *PrMx* is applied until the controller returns to priority service phase 2. In this example, *Max1* times for phases 1, 3, 5 and 7 are reduced by their 5" ***MaxReduce*** values in *Split Table# 2.*

Pedestrian service is also skipped for phases 4 and 8 even though a ped recall exists for those phases. This can also provide an early return to *priority phase* 2.

Now, place a priority service request for PR8 from StreetWise during the last 10 seconds of green during phase 2 (click the "Send" button when the *Max1* timer of phase 2 is 010 on menu MM->7->1). Note that the *Max1* timer for phase 2 switches to *PrMx* =35" when PR8 is received. This extends the transit priority phase 10" to service a "late" request. After the priority phase is extended, the non-transit phases are reduced by their ***MaxReduce*** values until the controller returns to phase 2.

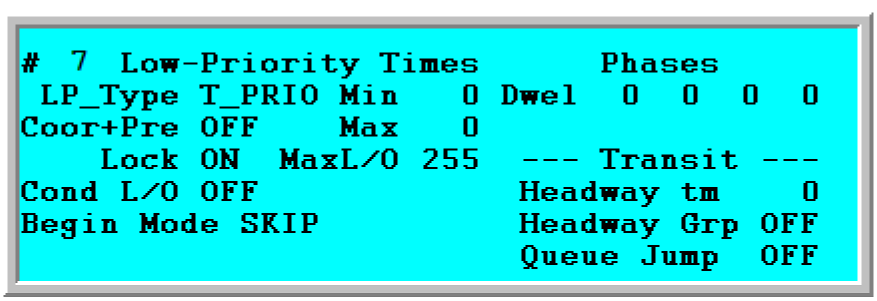
Think of *free operation* as a variable cycle length no greater than the sum of the active phase max times. If a split table is associated with a free pattern (cycle length 0"), the non-zero split values in the split table override the active max times. Transit priority extends or reduces these max times using ***MaxExtend*** and ***MaxReduce*** to provide an early return or extend the priority phase in free operation.

6.2.1 MaxL/O Parameter and Phase/Pedestrian Skipping

Vehicle phases and/or pedestrian service may be omitted during coordination or free operation using the NTCIP *Priority Strategy Table*. In the last example, *Priority Strategy Table #2 associated with PR8* omits pedestrian service on phases 4 and 8 while *pattern # 3* is active.

Eight separate *Priority Strategy Tables* are provided under MM->2->2. These eight tables may be assigned to priority Requests 1-4 (PR7-PR10) in any split table in the controller.

Section 2.2.3 defined the ***MaxL/O*** parameter for each service request. If phase skipping is used as a strategy during priority service, consider setting ***MaxL/O*** to a value larger than the greatest cycle length anticipated after a priority request. This will insure that all phases and pedestrian service skipped during a priority request will be serviced once before another priority request is serviced.



MaxL/O insures all phases with calls are serviced after a priority request

7 Priority Status Displays and Diagnostics

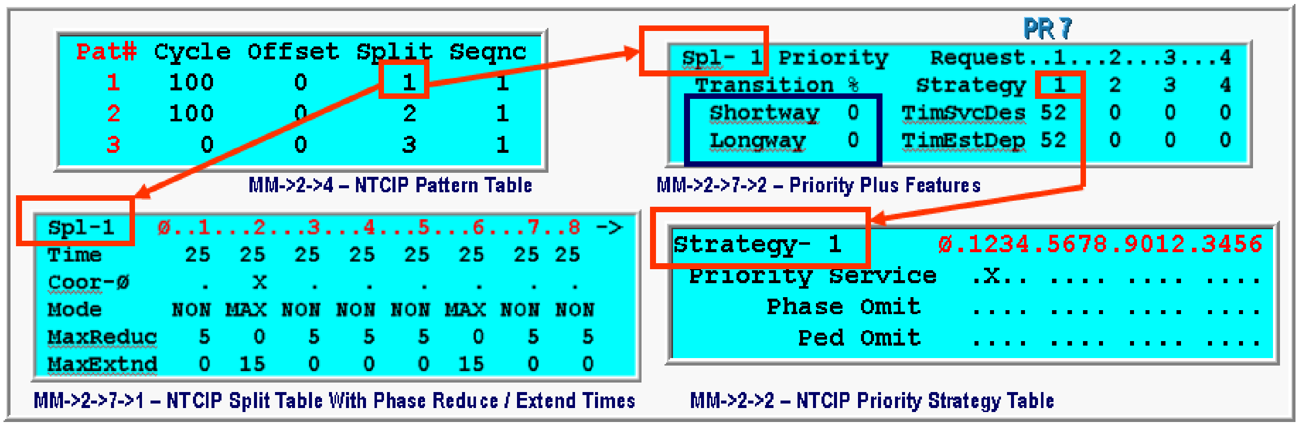
7.1 Transit Calcs Status Display

The *Easy Calcs* screen (MM->2->8->2) display force-off/yield calculations related to coordination.

The *Transit Calcs* screen (MM->2->8->6) displays priority service timers and calculations while a request is being serviced during coordination or free operation. The interpretation of this display varies with the method used.

7.2 Transit Calcs Under the NTCIP Method

The following programming was presented in the example in section 2.3.1.



The *Transit Calcs* below are displayed when pattern # 1 is active and PR7 is received at Loc=30. In this case, *TSD* is projected before the start of the priority phase in the same cycle as the request (section5.2.4).

Note that the coordinator is in ***SYNC*** with ***oft*** (offset) = 0" and ***er*** (offset error) = 0". If the coordinator was in transition when the priority request was received, the display would display ***SHORT***, ***LONG*** or ***DWELL*** and the ***er*** value (offset error) would indicate how many seconds the controller was out of ***SYNC***.

```
Pre#- 7   Loc-030 TBC-030 SYNC NO_ERROR
Sht-   0  ShSD-   0   Hold   0    0    0    0
Lng-   0  LgSD-   0 TSD/Mx  51    0    0    0
Er     0  Targ-   0    Ted  51    0    0    0
Oft-   0          PendMx     1    0    0    0
           .1...2...3...4...5...6...7...8
FloatMx    15   35   15   15   15   35   15   15
Split      25   25   25   25   25   25   25   25
Extend      0   15    0    0    0   15    0    0
Reduce      5    0    5    5    5    0    5    5
Veh Omt     .    .    .    .    .    .    .    .
Ped Omt     .    .    .    .    .    .    .    .
```

MM->2->8-> Transit Calcs

There are four columns corresponding with the four priority *requests* (PR7 – PR10).

The *TSD* and *TED* times are 52" and the ***TSD/Mx*** and ***Ted*** counters begin counting down to zero when PR7 is first received.

The ***FloatMx*** values are the *PrMx* times used to reduce the non-transit phases. These max times are active when the value ***PendMx*** is equal to 1. ***PendMx*** resets to zero after the priority phase is serviced and the service request is complete.

The ***Split***, ***Extend***, ***Reduce*** and ***Omit*** values are the programmed values for the active strategy.

You can also watch the force-offs recalculate during phase extension under the *NTCIP* method. Place a PR7 service request during pattern# 1 between Loc = 75 and Loc=79. This will cause the controller to extend priority service phase 2 one second before the force-off at Loc=20. If you watch *Easy Calcs* (MM->2->8->2), you can see the force-offs recalculate at Loc=19. The new force-offs extend the priority service phase and reduce the non-priority phases so the controller never leaves coordination. It is not necessary to review the *Easy Calcs* screen to apply the NTCIP method. This information is only presented to help you understand how the controller accomplishes phase extension under the *NTCIP* method.

7.3 Cubic | Trafficware EZ Method

The *Cubic | Trafficware EZ* method is enabled by setting *Shortway* and/or *Longway* in the *Priority Plus Features* table to non-zero values. Several values on the *Transit Calcs* display (MM->2->8->6) are unique to the *Cubic | Trafficware EZ* method. The example below assumes that *Shortway* and *Longway* are 20% and a PR7 request is issued at Loc=30 during pattern # 1.

```
Pre#- 7   Loc-030 TBC-030 SHRT NO_ERROR
Sht- 18   ShSD- 12    Hold    0    0    0    0
Lng- 82   LgSD- 12 TSD/Mx   51    0    0    0
Er  + 0   Targ- 82     Ted   51    0    0    0
Oft-  0           PendMx     1    0    0    0
           .1...2...3...4...5...6...7...8
FloatMx    15  35  15  15  15  35  15  15
Split      25  25  25  25  25  25  25  25
Extend      0  15   0   0   0  15   0   0
Reduce      5   0   5   5   5   0   5   5
Veh Omt     .   .   .   .   .   .   .   .
Ped Omt     .   .   .   .   .   .   .   .
```

MM->2->8->6: Transit Calcs

In this case, ***TSD/Mx*** and ***Ted*** countdown timers are identical with the *NTCIP method*. However, the ***Targ*** (target) value shown is the projected arrival based on the *TSD* (82 = 30" + 52"). You will also find this ***Targ*** value under MM->7->2:

```
 OpModes.Src-TEST     Cycle      Ofst  07:44:35
 Sys-    0  Actv-  1  Loc-  30  Actu: 00 ACTIV
 Tbc-    0 Next-   1  Tbc-  30  Err:  00
 Ext-    0  Remo-  0  Prog-100  Targ: 82 SYNC
 Tod-    0 Test-   1         Transit: 18
Alt:.Opt.Time.Det.CIR
      0    0   0   0
```

MM->7->2: Coordination Status

The controller compares ***Targ*** (TSD target) with the beginning of the priority service phase (in this case, Loc=0 is the start of phase 2). Since the vehicle arrives before the start of phase 2, the controller applies shortway from Loc=18 until Loc = 82 and applies longway at Loc=0 to correct the offset ***er*** back to 0 (SYNC). Transit priority is quite complex when you consider the decisions that the coordinator is making in real-time in response to a service request. Fortunately, the operation is transparent to the user, but to really understand the *Cubic | Trafficware EZ* method, you need to be able to interpret these displays.

7.4 Transit Priority Diagnostics

The *Transit Calcs* status screen (MM->2->8->6) provides a diagnostic check on the active transit priority request currently being serviced. If the request passes the transit priority diagnostics, the display in the top right corner of the screen displays NO_ERROR. Any other error message indicates the priority service request was ignored because the transit priority diagnostics failed.

```
Pre#- 7  Loc-030 TBC-030 SHRT NO_ERROR
Sht- 18  ShSD- 12   Hold   0   0   0   0
Lng- 82  LgSD- 12 TSD/Mx  51   0   0   0
Er   + 0  Targ- 82    Ted  51   0   0   0
Oft-  0          PendMx    1   0   0   0
          .1...2...3...4...5...6...7...8
FloatMx   15  35  15  15  15  35  15  15
Split     25  25  25  25  25  25  25  25
Extend     0  15   0   0   0  15   0   0
Reduce     5   0   5   5   5   0   5   5
Veh Omt    .   .   .   .   .   .   .   .
Ped Omt    .   .   .   .   .   .   .   .
```

MM->2->8->6: Transit Calcs

The following error messages indicate that a problem has been found in the active priority request.

1. NO_TRAN_PH

   No ***Priority*** phase is assigned in the *Priority Strategy Table* called by this request

2. TRAN_MAXEXTEND

   No ***MaxExtend*** values are specified (only applies to the *NTCIP* method)

3. RED/EXT

   No ***MaxReduce*** values are specified (only applies to the *NTCIP* method)

4. RINGS_BAL

   The sum of the ***MaxReduce*** times does not equal the sum of the ***MaxExtend*** times in a ring.

NOTE: Care must be taken when programming any Transit priority parameters to insure that the controller can transition properly to the priority phase. Strategy table 99 may be reported which has the effect of not running Transit Priority due to errors in user programmable data.

7.5 ATMS.Now Transit Report

ATMS.now provides a report known under the Controller category known as the Transit Priority Report. Two samples of this report is shown below.

Report prior to ATMS 2.7

Transit Priority Report

ID: 72
Name: Santa Rita Rd @ Stoneridge Dr (72)
Corridor:

1/25/2012

Start Date/Time	End Date/Time	Direction	Type	Secs	Headway	Red	Cycle
1/25/2012 6:34:00AM	1/25/2012 6:34:00AM	N	REDUCE	44		17	120
1/25/2012 6:35:00AM	1/25/2012 6:35:00AM	N	LOCKOUT	0	0' 3"	0	0
1/25/2012 7:03:00AM	1/25/2012 7:04:00AM	S	REDUCE	51		8	120
1/25/2012 7:22:00AM	1/25/2012 7:22:00AM	S	REDUCE	1	18' 12"	13	120
1/25/2012 8:38:00AM	1/25/2012 8:38:00AM	N	EXTEND	0	123' 13"	25	120
1/25/2012 9:03:00AM	1/25/2012 9:05:00AM	S	REDUCE	44	100' 34"	0	120

Report after ATMS 2.6

Transit Priority Report

Report Date: 12/5/2018

Time: 12/04/2018 00:00:00 **To** 12/05/2018 23:59:59

ID	Name	Start Date/Time	End Date/Time	Direction	Priority	Sec (s)	Headway (mm:ss)	Red Time (s)	Cycle (s)
15	188th St SW & SR 99	12/4/2018 6:36:19 AM	12/4/2018 6:36:34 AM	N	EXTEND	11	15:08	10	140
15	188th St SW & SR 99	12/4/2018 7:14:27 AM	12/4/2018 7:14:42 AM	N	NONE	0	37:53	10	180
15	188th St SW & SR 99	12/4/2018 7:24:27 AM	12/4/2018 7:25:17 AM	N	REDUCE	16	09:41	50	180
15	188th St SW & SR 99	12/4/2018 7:54:29 AM	12/4/2018 7:54:44 AM	N	NONE	0	29:16	10	131
15	188th St SW & SR 99	12/4/2018 8:35:54 AM	12/4/2018 8:36:08 AM	N	NONE	0	40:54	10	167
15	188th St SW & SR 99	12/4/2018 9:16:02 AM	12/4/2018 9:16:17 AM	N	EXTEND	3	39:45	10	120
15	188th St SW & SR 99	12/4/2018 9:28:04 AM	12/4/2018 9:28:19 AM	N	EXTEND	4	11:52	10	120
15	188th St SW & SR 99	12/4/2018 10:04:46 AM	12/4/2018 10:05:01 AM	N	REDUCE	7	36:27	10	136
15	188th St SW & SR 99	12/4/2018 10:13:43 AM	12/4/2018 10:13:58 AM	N	REDUCE	19	08:35	10	130
15	188th St SW & SR 99	12/4/2018 10:46:13 AM	12/4/2018 10:47:03 AM	N	REDUCE	1	32:14	50	130
15	188th St SW & SR 99	12/4/2018 10:49:07 AM	12/4/2018 10:49:34 AM	N	REDUCE	37	04:27	20	150
15	188th St SW & SR 99	12/4/2018 10:55:00 AM	12/4/2018 10:55:15 AM	N	REDUCE	31	05:44	10	150
15	188th St SW & SR 99	12/4/2018 10:59:30 AM	12/4/2018 11:00:25 AM	N	REDUCE	18	12:29	50	150
15	188th St SW & SR 99	12/4/2018 11:04:01 AM	12/4/2018 11:05:00 AM	S	REDUCE	36	01:44	50	150
15	188th St SW & SR 99	12/4/2018 11:08:58 AM	12/4/2018 11:09:13 AM	N	EXTEND	14	13:48	10	150
15	188th St SW & SR 99	12/4/2018 11:09:14 AM	12/4/2018 11:10:14 AM	S	REDUCE	36	04:58	60	150
15	188th St SW & SR 99	12/4/2018 11:16:09 AM	12/4/2018 11:16:15 AM	S	NONE	0	06:40	0	150
15	188th St SW & SR 99	12/4/2018 11:18:49 AM	12/4/2018 11:19:05 AM	N	REDUCE	6	18:49	10	130
15	188th St SW & SR 99	12/4/2018 11:28:00 AM	12/4/2018 11:28:15 AM	N	REDUCE	1	08:43	10	130
15	188th St SW & SR 99	12/4/2018 11:38:41 AM	12/4/2018 11:39:30 AM	N	REDUCE	16	10:32	40	130
15	188th St SW & SR 99	12/4/2018 11:45:49 AM	12/4/2018 11:46:11 AM	N	REDUCE	11	06:10	20	130
15	188th St SW & SR 99	12/4/2018 12:11:15 PM	12/4/2018 12:11:21 PM	S	NONE	0	18:57	0	150
15	188th St SW & SR 99	12/4/2018 12:22:42 PM	12/4/2018 12:22:48 PM	S	REDUCE	24	11:17	0	150
15	188th St SW & SR 99	12/4/2018 12:23:53 PM	12/4/2018 12:24:08 PM	N	NONE	0	37:21	10	160
15	188th St SW & SR 99	12/4/2018 1:19:17 PM	12/4/2018 1:19:32 PM	N	REDUCE	3	55:06	10	144

The report displays the following information.

ID is the controller ID number

Name is the ATMS defined controller name

Start Date/Time is the actual date and time that the transit priority request occurred.

End Date/Time is the actual date and time that the transit priority request ended.

Direction The direction of the transit vehicle when the transit priority request occurred

Priority (Type) Type describes the chosen method to serve the transit vehicle within the TSD/TED parameters. The following are the reported types:

NONE No phase reductions or extensions were required to serve the transit priority Phase(s)

REDUCE Reduction of non-transit phases occurred. The Reduce time on the TSP report is based on the actual start time of the transit phase. This includes any reduction for TSP, as well as any time due to previous phases gapping out or getting skipped that cycle.

EXTEND Extending the transit phases occurred. Please note that an EXTEND time of zero seconds indicates that the transit vehicle was successfully serviced during the expected window in the current cycle (i.e. **DO NOTHING** service).

LOCKOUT Transit priority is not run because the headway timer programmed has not been exceeded.

FREE The controller was in free operation when the transit call occurred.

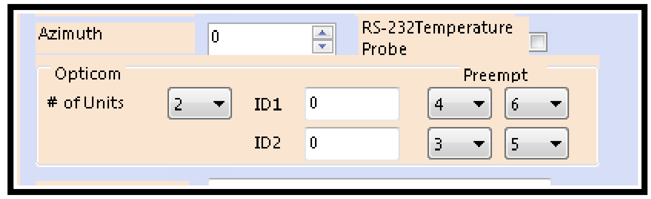
Sec is the time that was reduced or extended in seconds

Headway this time is displayed in minutes and seconds and indicates the time that occurred between transit priority actuations.

Red Time(Duration) is the amount of time that occurred when the transit call was received by the controller and when the controller actually started running the transit phase. It basically displays how much time it took to get to the transit phases once the call occurred. This is displayed in seconds.

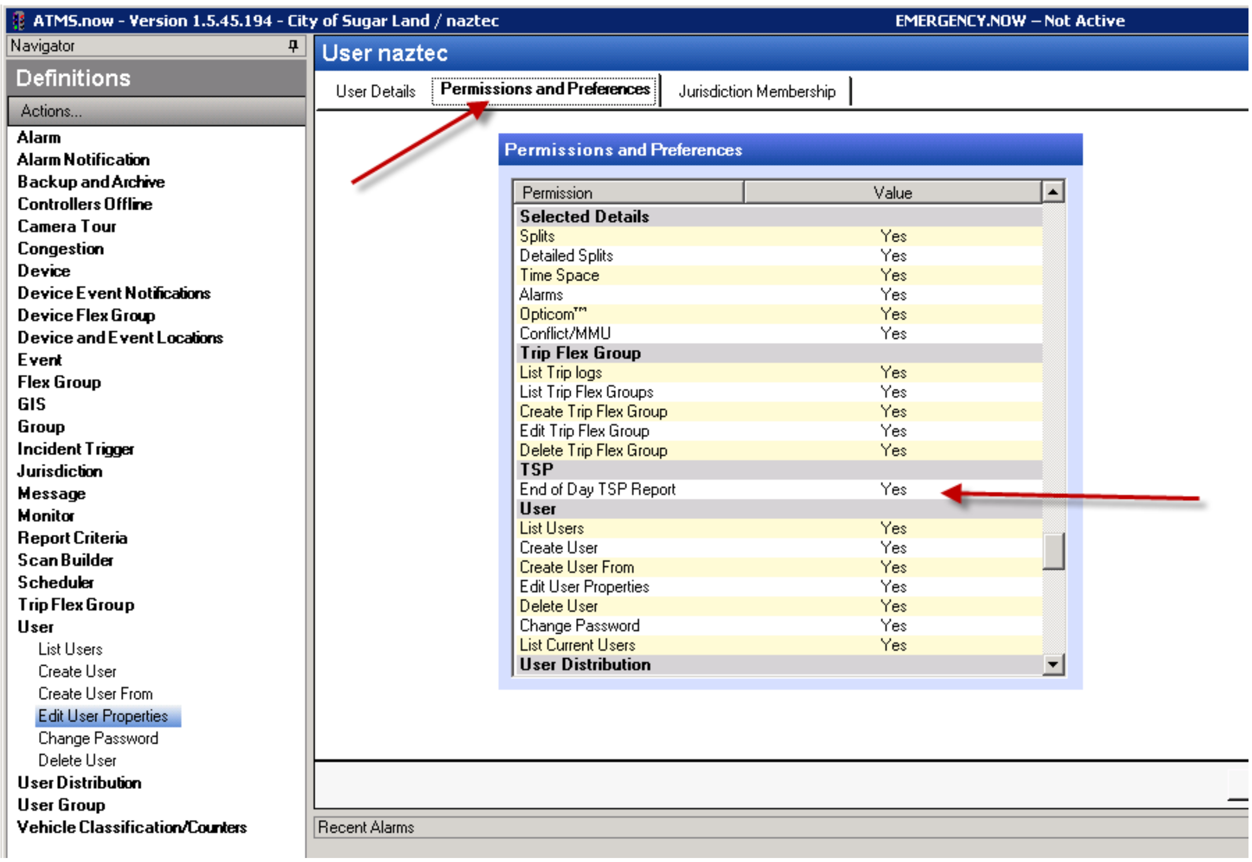
Cycle (Length) is the coordination cycle length that was running when the transit priority request occurred. When running in FREE operation a cycle length of 30 seconds will be displayed. This is displayed in seconds.

7.6 ATMS.now End of Day TSP Report



ATMS.now also has the capability of sending designated users a daily TSP report via e-mail **if the agency utilizes Opticom Discriminator Serial cards (Models 752 / 754)**. The opticom device(s) is set up in ATMS using the controller's definition as where each unit ID and preemption direction is set.

To generate this report, the ATMS.now administrator must turn on *The End of Day TSP Report* function under the designated user's Permission and preferences as shown below:



Once this occurs the following report will be sent via e-mail to the designated user:

Transit Report – Mon Jan 30 23:30:22 2012

Bus ID 314

Intersection	Start Time	End Time	Date	Dir	Type	Secs	Headway	Red Time	Cycle
First St @ Ray – Vineyard (93)	20:52:06	20:52:39	01/30/2012	N	FREE	0	96'26"	00	030
First St @ Neal St (95)	20:49:40	20:50:57	01/30/2012	W	FREE	0	67'48"	05	030
First St @ Kottinger (94)	19:58:11	19:58:44	01/30/2012	S	FREE	0	88'51"	00	030
First St @ Kottinger (94)	16:55:30	16:55:30	01/30/2012	S	LOCKOUT	0	21'55"	00	000
Main St @ Ray St-St John St (79)	16:55:28	16:55:28	01/30/2012	S	LOCKOUT	0	0'3"	00	000
Foothill Rd @ Laurel Creek Way (3)	16:07:51	16:08:10	01/30/2012	S	FREE	0	14'4"	00	030
Foothill Rd @ Stoneridge Dr (4)	16:07:33	16:07:50	01/30/2012	W	FREE	0	14'8"	00	030
Santa Rita Rd @ Amador HS (77)	16:07:01	16:07:01	01/30/2012	S	LOCKOUT	0	0'3"	00	000
Santa Rita Rd @ Amador HS (77)	16:06:48	16:06:59	01/30/2012	S	FREE	0	185'24"	00	030
Santa Rita Rd @ Stanley-Del Valle (78)	16:06:48	16:06:58	01/30/2012	N	FREE	0	65'33"	00	030
Stoneridge Mall Rd @ Canyon Way (9)	15:59:38	16:00:19	01/30/2012	S	FREE	0	15'45"	00	030
First St @ Kottinger (94)	15:59:48	16:00:07	01/30/2012	S	REDUCE	1	63'55"	18	120

Bus ID 320

Intersection	Start Time	End Time	Date	Dir	Type	Secs	Headway	Red Time	Cycle
Santa Rita Rd @ Amador HS (77)	18:30:23	18:30:35	01/30/2012	N	FREE	0	37'38"	00	030
Santa Rita Rd @ Black Ave (76)	18:29:58	18:30:09	01/30/2012	S	FREE	0	37'50"	00	030
First St @ Kottinger (94)	18:29:27	18:29:27	01/30/2012	S	LOCKOUT	0	0'6"	00	000
Santa Rita Rd @ Valley Ave (74)	18:28:28	18:29:37	01/30/2012	S	REDUCE	4	38'15"	17	120
First St @ Kottinger (94)	18:29:03	18:29:20	01/30/2012	S	EXTEND	0	115'28"	17	120
First St @ Ray – Vineyard (93)	18:28:41	18:29:07	01/30/2012	E	REDUCE	1	115'21"	00	120
First St @ Old Stanley (92)	18:27:51	18:28:18	01/30/2012	S	FREE	0	115'26"	00	030
Stanley Blvd @ California – Reflection	18:27:15	18:27:33	01/30/2012	W	FREE	0	115'36"	00	030
Owens Dr @ W Las Positas Blvd (65)	17:20:54	17:22:01	01/30/2012	S	FREE	0	15'54"	00	030
First St @ Ray – Vineyard (93)	17:19:07	17:19:34	01/30/2012	N	EXTEND	0	41'25"	01	120
Santa Rita Rd @ Valley Ave (74)	17:21:50	17:23:19	01/30/2012	N	REDUCE	30	18'22"	12	120
Owens Dr @ East Bart Ent (46)	17:14:03	17:14:03	01/30/2012	E	LOCKOUT	0	0'3"	00	000
Owens Dr @ East Bart Ent (46)	17:13:45	17:14:01	01/30/2012	E	FREE	0	19'50"	15	030
Owens Dr @ Rosewood Dr (60)	17:19:51	17:20:09	01/30/2012	E	FREE	0	17'8"	00	030
First St @ Neal St (95)	17:17:25	17:17:25	01/30/2012	W	LOCKOUT	0	0'0"	00	000
Stanley Blvd @ California – Reflection	17:20:31	17:20:31	01/30/2012	E	LOCKOUT	0	25'23"	00	000
Santa Rita Rd @ Black Ave (76)	17:20:42	17:20:42	01/30/2012	N	LOCKOUT	0	9'1"	00	000
Santa Rita Rd @ Black Ave (76)	17:20:42	17:20:43	01/30/2012	N	FREE	0	0'0"	00	030
Santa Rita Rd @ Black Ave (76)	17:20:28	17:20:42	01/30/2012	N	FREE	0	17'59"	00	030
Foothill Rd @ Deodar Way (2)	17:20:27	17:20:27	01/30/2012	N	LOCKOUT	0	0'3"	00	000
Foothill Rd @ Deodar Way (2)	17:20:15	17:20:24	01/30/2012	N	FREE	0	20'29"	00	030
Foothill Rd @ Laurel Creek Way (3)	17:20:12	17:20:12	01/30/2012	S	LOCKOUT	0	20'4"	00	000
Foothill Rd @ Laurel Creek Way (3)	17:20:12	17:20:12	01/30/2012	S	FREE	0	0'0"	00	030
Foothill Rd @ Laurel Creek Way (3)	17:19:51	17:20:11	01/30/2012	S	FREE	0	20'22"	00	030
Santa Rita Rd @ Amador HS (77)	17:19:59	17:19:59	01/30/2012	S	LOCKOUT	0	0'3"	00	000
Santa Rita Rd @ Amador HS (77)	17:19:45	17:19:57	01/30/2012	S	FREE	0	19'9"	00	030
First St @ Old Stanley (92)	17:19:56	17:19:56	01/30/2012	N	LOCKOUT	0	0'3"	00	000
First St @ Old Stanley (92)	17:19:41	17:19:54	01/30/2012	N	FREE	0	113'36"	09	030
Foothill Rd @ Stoneridge Dr (4)	17:19:14	17:19:49	01/30/2012	W	FREE	0	11'30"	00	030
Hacienda Dr @ Owens Dr (55)	17:18:06	17:19:21	01/30/2012	E	FREE	0	15'13"	07	030
Santa Rita Rd @ Stanley-Del Valle (78)	17:19:10	17:19:10	01/30/2012	N	LOCKOUT	0	0'3"	00	000
Santa Rita Rd @ Stanley-Del Valle (78)	17:18:56	17:19:07	01/30/2012	N	FREE	0	18'52"	00	030
First St @ Kottinger (94)	17:18:13	17:18:52	01/30/2012	N	REDUCE	1	41'17"	12	120
Owens Dr @ Oracle (47)	17:17:30	17:17:50	01/30/2012	E	FREE	0	15'24"	05	030
First St @ Neal St (95)	17:17:07	17:17:20	01/30/2012	W	FREE	0	18'41"	00	120
Stoneridge Mall Rd @ Canyon Way (9)	17:14:03	17:14:03	01/30/2012	S	LOCKOUT	0	0'0"	00	000
Stoneridge Mall Rd @ Canyon Way (9)	17:13:25	17:13:25	01/30/2012	S	LOCKOUT	0	0'0"	00	000
Main St @ Ray St-St John St (79)	17:13:50	17:13:50	01/30/2012	S	LOCKOUT	0	0'2"	00	000
Main St @ Ray St-St John St (79)	17:13:28	17:13:48	01/30/2012	S	FREE	0	18'3"	19	030
Santa Rita Rd @ Stanley-Del Valle (78)	17:13:10	17:13:27	01/30/2012	S	FREE	0	19'11"	00	030
Santa Rita Rd @ Amador HS (77)	17:12:32	17:12:54	01/30/2012	N	FREE	0	19'22"	00	030
Owens Dr @ W Las Positas Blvd (65)	16:36:17	16:36:31	01/30/2012	S	FREE	0	16'34"	00	030
First St @ Kottinger (94)	16:36:20	16:36:55	01/30/2012	N	REDUCE	1	76'24"	09	120
First St @ Kottinger (94)	16:34:10	16:34:10	01/30/2012	S	LOCKOUT	0	0'35"	00	000
First St @ Neal St (95)	16:29:43	16:35:56	01/30/2012	W	FREE	0	44'45"	00	120
Owens Dr @ Rosewood Dr (60)	16:34:30	16:34:41	01/30/2012	E	FREE	0	16'22"	00	030
First St @ Kottinger (94)	16:33:14	16:33:34	01/30/2012	S	EXTEND	0	33'7"	20	120
First St @ Ray – Vineyard (93)	16:32:46	16:33:20	01/30/2012	E	REDUCE	1	97'10"	07	120
First St @ Old Stanley (92)	16:31:56	16:32:24	01/30/2012	S	FREE	0	89'3"	08	030
Stanley Blvd @ California – Reflection	16:31:23	16:31:40	01/30/2012	W	FREE	0	89'19"	00	030
Stoneridge Mall Rd @ Fabian Ct (10)	15:27:51	15:29:35	01/30/2012	E	FREE	0	12'17"	04	030
Foothill Rd @ Deodar Way (2)	15:26:59	15:26:59	01/30/2012	N	LOCKOUT	0	0'3"	00	000

8 Combining Preemption and Priority Service for Light Rail Operation

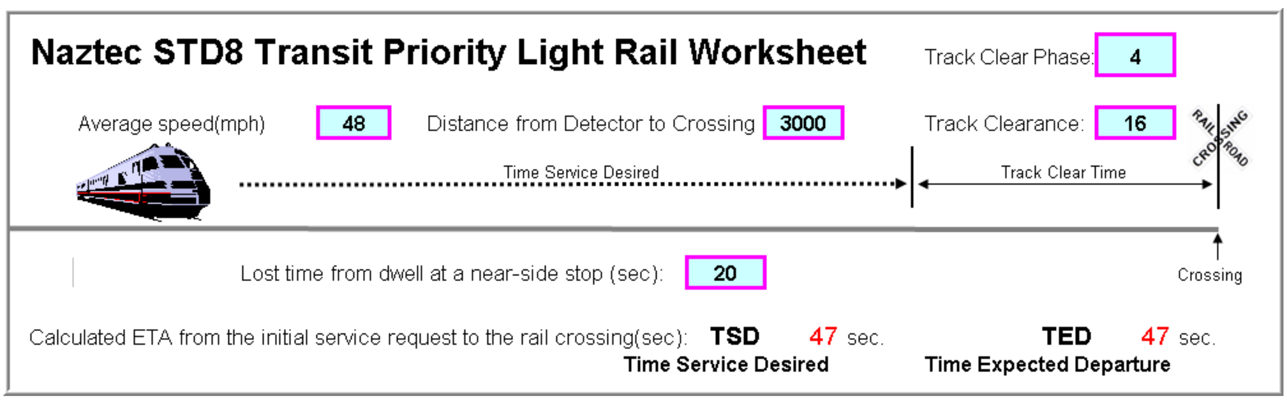
Light rail facilities must run concurrently with a coordinated signal system. Often, gates are provided to maintain the right-of-way for the light rail vehicle without stopping. If queuing from the signal extends over the tracks, then a *Track Clear* interval must be provided to preempt the traffic signal and clear the track area before the gates come down.

Preemption skips all phases to move to track clearance as soon as the light rail vehicle is detected and returns to a programmed *Exit Phase* after the vehicle clears the crossing. Cubic | Trafficware controllers also provide the ability to return to coordination without a transition penalty with the COOR+PRE feature. Preemption is acceptable for heavy rail where crossings are infrequent (8-10 events per day) and the duration of each event is long. However, preemption is not desirable for Light rail operations where crossings are frequent (5 or more events per hour) and the duration of each event is relatively short.

Transit priority service can be combined with preemption to reduce phase skipping prior to a preempt event. In this case, the *priority service phase* is the *track clearance phase* of the preempt that overrides priority service and clears the tracks prior to the arrival of the vehicle.

8.1 Estimating TSD and TED Times for Light Rail Operation

The *TSD* at the beginning of track clearance phase is simply the estimated time of arrival at the crossing minus the track clearance time. A worksheet is provided in StreetWise to assist the use in this calculation. The derivation of *TSD* and *TED* times is similar to the calculations presented in *Chapter 3 Estimating Transit Vehicle Arrival Times.*



This worksheet includes an estimate for dwell times at a nearside station prior to the crossing. Note that, dwell time for light rail operation is generally more predictable than for transit bus operation.

The *Track Clearance* time is the *Track Grn* + vehicle clearance of the track clearance phase serviced at the beginning of preemption. The goal of this Light rail priority strategy is to:

1. Provide an early return to the track clearance phase without skipping phases
2. Extend the track clearance phase to accommodate a late arrival during coordinated operation

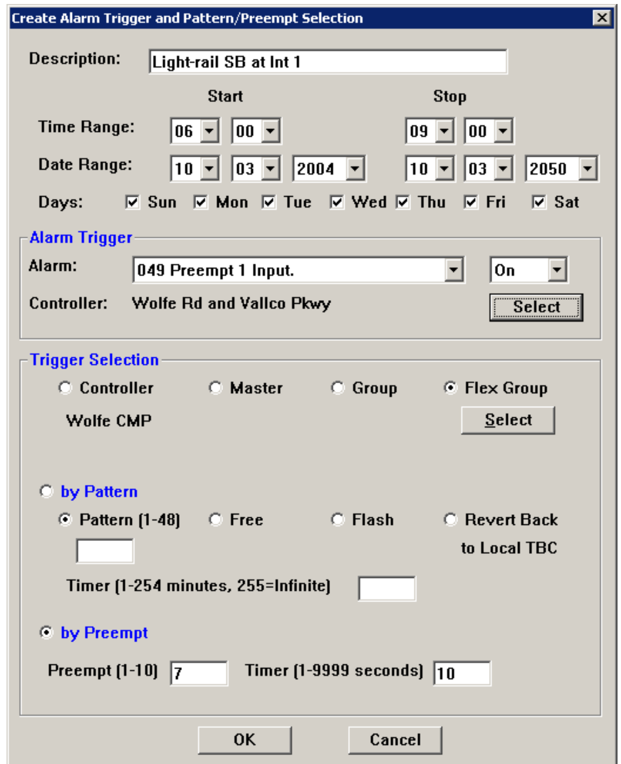
All other concepts discussed in the previous chapters are applicable for priority service during light rail operations. Some agencies omit pedestrian service during Light rail events. This can be easily accommodated in the *Priority Strategy Table* as discussed in section 6.1 to help insure an early return to the track clearance phase without phase skipping.

8.2 Relaying Transit Priority Requests to Downstream Intersections

PRS Messages were identified in the preempt/priority controller model presented in section 1.3.

When a Light rail vehicle activates a preempt request, an alarm is generated in the controller that can be retrieved by the StreetWise or ATMS.now central system. "Continuous" alarm polling by the central computer can detect these alarms and activate a software "trigger" to forward the *PRS Message* to downstream intersections in the system. Ideally, a fiber based TCP/IP network is used to minimize time latencies associated with forwarding these requests.

Suppose the head-end station of a light rail line is at the intersection of Wolfe Rd and Vallco Pkwy adjacent to a traffic signal. When the light rail departs the station, a preempt request is received at this signal to clear the tracks and lower the gates.



If the central is configured to continuously monitor alarms at this intersection, Alarm 49 (Preempt 1 Input) will be received within 3 seconds after track clearance begins. This alarm can activate a software "trigger" to forward a PR7 request to flex-group "Wolfe CMP" as shown to the right.

These priority requests are forwarded with a timer (10 seconds in this case) which is long enough for the PRS to initiate a LOCKed request. Each controller receiving the request will begin timing the *TSD* and *TED* times associated with the active strategy currently for PR7 (Request 1).

Software triggers may be activated or de-activated through the StreetWise or ATMS.now scheduler by time-of-day and day-of-week. This allows the user to activate priority service by time-of-day at the system level.

In the example above, Alarm 49 was used to detect when preempt # 1 becomes active at a signal. However, light rail applications require separate messages for each rail line and for each direction of travel. This can be easily accomplished by interfacing the controller external alarm inputs with the railroad circuits and using each alarm input to activate a separate trigger. It is important in the design of the rail system to insure that a separate input can be provided for each direction and each rail line. A dual track system will typically require 4 separate external alarm inputs to differentiate the rail line and direction for each preempt event.

In summary, preemption status for each rail line and direction can be detected immediately at central as a status alarm which can be used to trigger a priority request to a group of signals downstream. This approach is similar to NTCIP 1211 (Scenario 1-3); however, no additional bandwidth is required because StreetWise gathers continuous alarms as fast as the communication bandwidth will allow.

NTCIP 1211 *Scenario 4* allows the *Priority Request Server* to be embedded in the controller logic and tightly coupled with the coordinator logic. This scenario allows the coordinator to make decisions one second before a phase is forced-off without relaying the request to a *Traffic Management Center*. Cubic | Trafficware enhances *Scenario 4* using a message scheme that does not require a separate communication path to provide the fastest response possible within an existing communication system.

9 Summary

Cubic | Trafficware controllers implement an *NTCIP* method based on *Scenario 4* of the latest draft of NTCIP 1211- “Object Definitions for Signal Control and Prioritization”. NTCIP based phase reduce and extend times extend each split table to accomplish priority phase reduction and extension for STD8, QSeq, 8Seq and USER phase sequences. In addition, the *EZ Transit m*ethod provides an “easy” alternative for USER mode applications by applying uniform reduction and extension times using longway and shortway transition.

The *Priority Strategy Table* can be applied to either method to assign the priority service phase for each priority request and to selectively omit phases and/or pedestrian service. *TSD* (Time Service Desired) and *TED* (Time of Estimated Departure) project the arrival of the transit vehicle. *TSD*, *TED* and the *Priority Strategy Table* are assigned to each request through the split table. This approach allows the strategy to vary by pattern and by time-of-day.

Cubic | Trafficware identified 5 primary goals in the design of a transit priority system:

1. Provide the NTCIP method to reduce and extend phases with the ability to omit vehicle phases and pedestrian service from the *Priority Strategy Table*. Also, provide the ability to vary these parameters by pattern and by time-of-day.
2. Provide an *EZ* method to simplify coding for USER configurations (16 phases assigned to 4 rings)
3. Insure that minimum phase times and ring/barrier requirements for STD8, QSeq or 8Seq phase modes are not compromised without placing the responsibility for these checks on the user
4. Provide transit priority during free operation as well as coordination
5. Provide a mechanism to forward requests through the central system without adding to the communication overhead of the system

The Cubic | Trafficware priority system accomplishes these goals by building upon the objectives of NTCIP 1211 *Scenario 4* and providing priority service during free operation as well as coordination. Phase reduce and extend times are provided as an extension to the split table rather than assigning additional phases in the sequence. The Cubic | Trafficware design insures that transit priority values supplied by the user do no fail the active coordination pattern under coordination. In addition, priority requests may be relayed to the central ATMS (StreetWise) and forwarded to downstream intersection using software triggers without adding to the communication bandwidth.

We at Cubic | Trafficware, Inc. believe that the transit priority features in our controllers are easy to setup and configure, but also flexible enough to handle any application the user may require during coordination or free operation.

10 Annex

The following is excerpted from NTCIP 1211, Annex E - NTCIP Tutorial

In a Signal Control and Prioritization System, a Fleet Vehicle, Fleet Management, or Traffic Management initiates priority service by instructing the Priority Request Generator to send a Priority Request to a Priority Request Server. The Priority Request Server prioritizes and sorts all requests based upon Class Type and Class Level.

It then sends its queue of requests to the Coordinator that resides in a Traffic Signal Controller. Based upon the strategy (or strategies based upon the implementation), the Coordinator will attempt to adjust the split timing in the Coordinator to accommodate the service request(s).

Based upon pre-programmed entries, the Coordinator applies phase and ped omits, and increases or decreases non-priority phase splits to extend the priority split time up to some maximum. The following figures illustrate the operation:

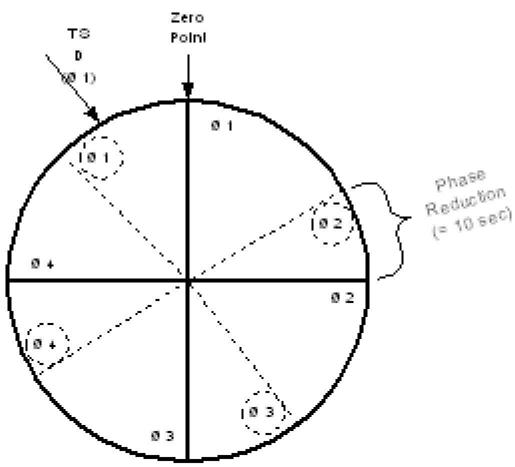


Figure E-1
Early Return to Coordinated Phase

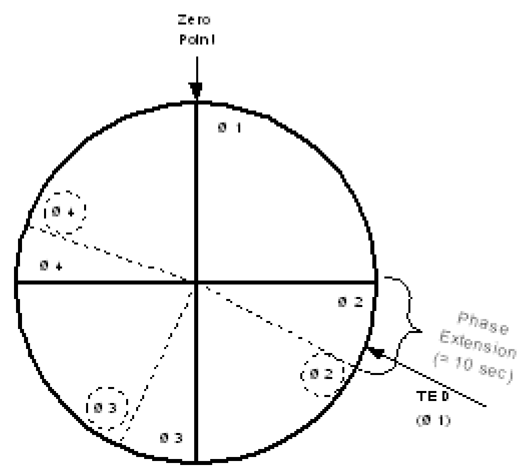


Figure E-2
Late Departure from Coordinated Phase

In the first figure, a priority request message for priority service on Phase 1 is received at some point prior to the "zero point". Projecting the priorityRequestTimeOfServiceDesired (TSD) into the anticipated timing sequence, the phase 1, 2, 3 and 4 split times are reduced by the priorityStrategyMaximumReductionTime to ensure an early return to Phase 1 green at TSD. [The split times for phases 2, 3, and 4 could have also been reduced to zero (0) by application of priorityStrategyPhaseOmits and/or priorityStrategyPedOmits.]

All coordinator calculations that project TSD and TED into a future timing position shall be based upon the current pattern. The calculations are not required to take into consideration a future change in pattern and any transitioning to that new pattern.