**NCHRP Project 03-122, “Performance-Based Management of Traffic Signals”**

# 

Data Dictionary

*Prepared for the*

**NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM (NCHRP)**

*by*

Chris Day, *Purdue University*Howell Li, *Purdue University*Darcy Bullock, *Purdue University*

|  |
| --- |
| **LIMITED USE DOCUMENT**  This report is furnished only for review by members of the NCHRP project panel and is regarded as fully privileged. Dissemination of information included herein must be approved by NCHRP. |

Draft 1.0

**May 2017**

Table of Contents

[Working Paper No. 1 1-1](#_Toc526524526)

[1.1 Purpose of Document 1-4](#_Toc526524527)

[1.2 Background 1-4](#_Toc526524528)

[1.3 Appraisal of Existing and New Event Codes 1-8](#_Toc526524529)

[1.4 Review of Signal Functions and Event Descriptions 1-9](#_Toc526524530)

[1.5 Candidates for New Event Codes 1-23](#_Toc526524531)

[1.6 Summary 1-26](#_Toc526524532)

[Appendix A: Existing Event Definitions 1-27](#_Toc526524533)

[Appendix B: New Event Definitions By Vendor 1-34](#_Toc526524534)

Table of Figures and Tables

[Table 1‑1. Existing and emerging data sources in traffic signal control systems. 1-5](#_Toc526524475)

[Figure 1‑1. High-resolution flat files in the traffic signal data ecosystem. 1-6](#_Toc526524476)

[Figure 1‑2. Vehicle displays and internal states for phase-driven outputs. 1-10](#_Toc526524477)

[Figure 1‑3. Vehicle displays and internal states for overlap-driven outputs. 1-11](#_Toc526524478)

[Figure 1‑4. Vehicle displays and internal states for flashing yellow arrow displays. 1-12](#_Toc526524479)

[Figure 1‑5. Pedestrian output display. 1-13](#_Toc526524480)

[Figure 1‑6. Basic actuated phase logic based on timers, in absence of gap-reduction logic, and existing event codes. 1-15](#_Toc526524481)

[Figure 1‑7. Added initial timing and existing event codes. 1-16](#_Toc526524482)

[Figure 1‑8. Gap reduction timing and existing event codes. 1-17](#_Toc526524483)

[Figure 1‑9. Adaptive maximum timing and existing event codes. 1-18](#_Toc526524484)

[Figure 1‑10. Pattern transition events. 1-19](#_Toc526524485)

[Figure 1‑11. Railroad preemption timing and existing event codes. 1-20](#_Toc526524486)

[Figure 1‑12. Transit signal priority (TSP) / Signal control priority (SCP) timing and existing event codes. 1-21](#_Toc526524487)

[Figure 1‑13. Adaptive changes to cycle, offset, and splits and existing event codes. \*Event 150 will be accompanied by different parameters to mark the start and end of transition. 1-23](#_Toc526524488)

## Purpose of Document

High-resolution data is a record of events that take place at signalized intersections. The “high-resolution” name reflects that the data is intended to be captured at the time resolution of a traffic signal controller (typically 0.1 second), with finer resolution being acceptable. At the time of writing, high-resolution data has become widely used, with many applications for performance measurement.

This document contains a listing of events that are not currently reflected in the enumerations for high-resolution signal controller data. The document begins by providing a brief description of where high-resolution data fits into the overall ecosystem of traffic signal data, and how it differs from other data sources that contain similar pieces of information. Next, a review of several signal functions is carried out to examine potential sources of signal operations information that are not currently reflected in the existing enumerations. Finally, this is combined with input from vendors to provide a listing of candidate events that are recommended for consideration in a future version of the enumerations.

## Background

Over the years, the application of improved technologies and process control methods to traffic signal infrastructure has led to the development of numerous data sources. Recently, these have been joined by automated methods of traffic data collection that measure user activity. This group of data sources can be used to monitor the performance of signalized intersections. This section describes the evolution of these data sets, and the position that high-resolution data has developed in the traffic signal data space, with the set of related data sources referred to as the data “ecosystem.”

### Summary of Data Sources

Table 1‑1 shows a list of existing and emerging data sources that are pertinent to traffic signals. The most fundamental data, sourced directly from terminal strips and pins, are generated by electrical input and output signals from contact closures, and can be captured by data acquisition equipment. The Synchronous Data Link Control (SDLC) bus communicates the same type of information digitally through a single bus shared among equipment within the signal cabinet, and that information can be captured by reading the digital signal passed through the SDLC bus. These two options for data collection require external hardware interfaces to capture and communicate the data beyond the cabinet.

The need to maintain networks of coordinated traffic signals led to the development of Advanced Traffic Management Systems (ATMSs), requiring communication protocols to local controllers. Data was also developed to facilitate traffic-responsive operation, specifically detector volume and occupancy aggregated to time intervals of several minutes. Various architectures for traffic-responsive control were developed; common schemes include the use of a central server or field master controllers. Adaptive control systems have also been developed within these architectures, with their own proprietary data objects, which are not usually directly accessible to the user (with some exceptions).

Table 1‑1. Existing and emerging data sources in traffic signal control systems.

| Data Source | Description | Data Objects |
| --- | --- | --- |
| Reading from Terminal Strip and Pins | Used to connect to detectors and load switches in older cabinets. Each wire corresponds to a particular connection. | Load Switch, Detector. |
| Reading from Synchronous Data Link Control (SDLC) Bus | Used to connect to detectors in newer cabinets. Data is encoded and collectively shared via serial connection among several components within the cabinet. | Load Switch, Detector. |
| Advanced Traffic Management System (ATMS) Data | Aggregated detector and occupancy data collected at regular intervals (e.g., 5 minutes) and communicated to an ATMS server for monitoring and advanced operation. | Detector volume and occupancy. |
| National Transportation Communications for ITS Protocol (NTCIP) | Used to remotely transmit messages between the controller and external devices or central system using the Simple Network Management Protocol (SNMP). Messages may include commands and status requests. Any aspect of the controller state can be accessed, but only in real time. Requires constant communication to collect detector and phase states. | All control settings; Status of phases, overlaps, detectors. |
| High-Resolution Data | Used to store specific events locally, at a minimum  0.1-second resolution, which are collected at various intervals by an external data collection system. | All specified state changes that generate an event. |
| SAE J2735 Messages via Dedicated Short Range Communications (DSRC) | Used to communicate with connected vehicles. Messages include Signal Phase and Timing (SPaT), which intersections broadcast to vehicles, and the Basic Safety Message (BSM), which vehicles broadcast to other vehicles and to the intersection. | Status of phases and overlaps and predicted state change times; Vehicle position and other detailed information. |
| Probe Vehicle Data | Used to track individual vehicles using GPS-instrumented mobile devices. The data can be sourced from software developed by third-party commercial traffic data providers, or collected in-house using dedicated devices. | Waypoints in the form of vehicle location and timestamp; or representative speeds on aggregated segments of roadway. |
| Other/Proprietary | Used for various purposes by different vendors. | May include any conceivable data objects. |

The National Transportation Communications for ITS Protocol (NTCIP) was developed to meet the need for interoperability between different manufacturers’ equipment and to standardize communication between traffic control equipment. The standard operates on the high-level application layer and is dependent on network-level communications such as serial or Ethernet connections. Many vendors have implemented NTCIP messaging using the Simple Network Management Protocol (SNMP). However, the standard does not prescribe whether the data should be recorded (or made recordable) by the equipment that transmits it. Thus, vendors have typically implemented their own formats for system control and to output measurements of performance. Examples of system data include some advanced control or adaptive systems that would make use of near-real-time cyclic volume data to make control decisions but would also provide “system” data in aggregated, nonstandardized formats as outputs. Some of this data may include volume and occupancy information at pre-defined time bins. Other emerging data standards include the SAE J2735 messaging specification where communication occurs over the Dedicated Short Range Communication (DSRC) 5.9 Ghz channel between Onboard Units (OBU) in the vehicles and Roadside Equipment (RSE), and probe data where vehicles are tracked through the system via GPS-instrumented devices.

The key benefit of high-resolution data is that, unlike other data sources that provide information in streams and thus must be intercepted, high-resolution data captures and logs all phase, detection, and other cabinet events locally in the controller in a consistent and open format. The data is stored as unstructured files, using a binary-packed or plain-text format that can be transmitted, decoded, and read at a later time. Such files may be compressed to minimize bandwidth requirements for transmission. High-resolution data has been implemented in six major controller manufacturers’ products and has since been instrumental in the research and development of automated traffic signal performance measure applications.

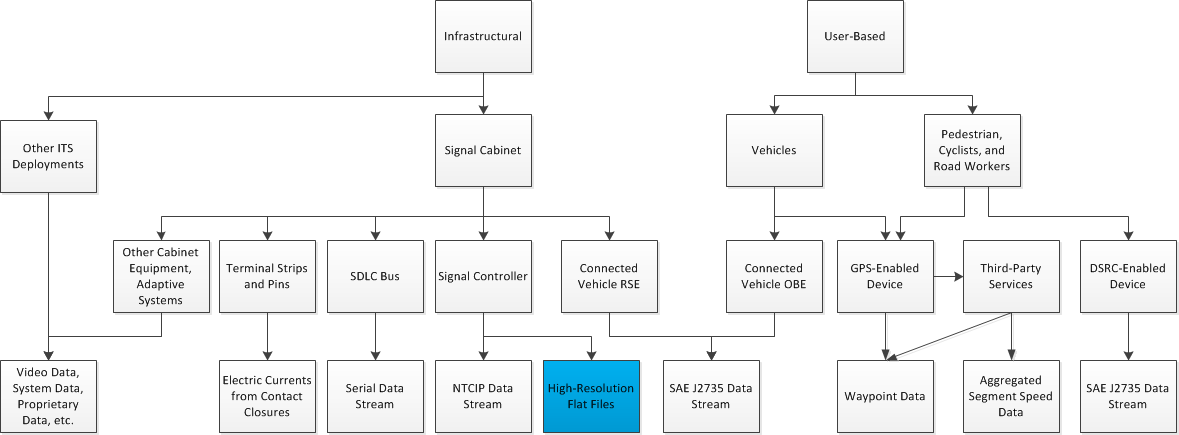


Figure 1‑1. High-resolution flat files in the traffic signal data ecosystem.

### The Traffic Signal Data Ecosystem

Figure 1‑1 shows high-resolution flat files in the context of all current and emerging traffic signal data. The first level from the top describes the two major areas where traffic signal data is generated – one from infrastructure and the other from the road user. The second level separates how the data is physically contained. In the case of the infrastructure, this pertains to the signal cabinet or other ITS deployments, such as red-light enforcement systems and video systems, which may use proprietary data formats. In the case of the user, this pertains to whether the data generator is located on a vehicle, or carried by a pedestrian, cyclist, or road worker. The third level describes the equipment that generates the data such as the signal controller or connected vehicle OBE. Finally, the fourth layer describes the generated data itself.

Some data sources are transient in nature. The electrical currents from contact closures, the serial data stream on the SDLC bus, and the stream of J2735 messages are constantly changing with the changing states of the traffic signal controller. It is not possible to query these data sources for a previous system state. They are not intended for data logging so much as data transmission for immediate use. External components would need to be added to capture the states over time.

Probe vehicle data, also known as “automatic vehicle data,” “trip data,” among other names, captures vehicle positions over time. Third-party traffic service providers collect the raw data from individual probe vehicles through the use of mobile phone apps or navigation devices. The data is then anonymized and provided in the form of aggregated average speeds at 1-minute intervals. The National Performance Measures Research Data Set (NPMRDS) is an example; this data set is aggregated at 5-minute intervals. The location data (latitudes and longitudes) of individual vehicles can also be obtained from third-party providers. These data do not contain any information about the signal control itself, such as phasing and decision-making.

Out of all the data sources described, only the high-resolution flat files are able to produce static data that describes phase output, decisions, and status, *and* accesses vehicular data in the same data scope.

### Looking Ahead

The future of high-resolution data must take into consideration the extensibility of the data to accommodate new events and detection sources, scalability of the data in large quantities, security of the data, and flexibility to interface with various existing and emerging data transmission protocols.

In terms of extensibility, high-resolution data is currently limited to 256 distinct event types due to having one byte assigned to describing the event, called the “event code.” In addition, for each event code, another byte is used to specify the value associated with the event itself, called the “event parameter.” This value can be a phase or a detector channel number, again up to 256 distinct values. This allows for 65,536 possible combinations of event codes and parameters. As emerging data standards such as J2735 become more widely adopted, it is anticipated that new events and enumerations to store values will exhaust the available pool.

Large volumes of high-resolution data can accumulate as data is collected over time at many locations, bringing some practical considerations. The storage in controller platforms may not be adequate and will require a network to transfer the data to a storage system. The network throughput must be at least as great as the rate of data being collected at the intersection. Additionally, for developing and visualizing performance measures, additional computational cycles are required to perform the calculations and visualizations and the efficiency of these processes are contingent on the way the data is stored. Increases in performance may be seen by adopting relational databases, messaging queues, key-value or column stores, or other data storage technologies. These data management systems can reside in a single system or distributed networks.

The way data is currently accessed is through a File Transfer Protocol (FTP) or Secure FTP (SFTP). The former technology contains no encryption mechanism and the data can be intercepted as plain text. In future implementations, greater security measures should be taken by increasing the adoption of SFTP, Transport Layer Security and Secure Sockets Layer (SSL/TLS) protocols, and Internet Protocol Security (IPSec) technologies. As both command-and-control systems, data store and cloud systems continue to mature and become more robust and distributed, there will be an increased need for controllers to interface with these systems. In time, this will necessitate that high-resolution data be accessible via application-layer protocols on which these systems operate. Since these systems are varied and can be agnostic as to how the structure of the enumerations are defined, the data may need to be self-describing so that systems can be aware of its context. Recently developed protocols such as the Javascript Object Notation (JSON) can facilitate these implementations.

## Appraisal of Existing and New Event Codes

As part of developing this data dictionary, the research team examined both the existing event codes to identify whether these captured the mechanisms of different methods of signal control, such as actuation, preemption, and adaptive control. A few potential new event codes were identified in that analysis. The following section presents illustrative definitions of the event codes relative to the state changes involved in these signal control strategies. The existing enumerations are listed in Appendix A.

In addition, some vendors had added new event codes to their enumerations since adopting high-resolution data. At the request of the NCHRP 03-122 panel, the research team compiled and synthesized these events and reconciled them against existing events to determine candidates for addition to the list of enumerations.

In April 2017, the research team sent requests to the six vendors to send listings of the new event codes that they had added or were thinking of implementing. As of May 25, 2017, three of the vendors had responded with event codes. Those are listed in Appendix B. These can be broken into four main categories:

* New and changed events for controller management, such as database upload/download and clock updates.
* New events for dynamic signal timing such as transition logic and adaptive control.
* New events for transit signal priority (TSP), particularly for priority detector diagnostics.
* New events for signal control and prioritization (SPC) requests using NTCIP 1211.

Altogether, 52 new events were compiled from these responses. Some of these overlapped and needed to be reconciled. To do so, this paper considers those events in the context of common features of signal operation to determine whether the key characteristics of such operation are captured in the current and proposed enumerations. The next section provides details of that analysis. Section 1.4 synthesizes the results and presents candidates for new event codes.

## Review of Signal Functions and Event Descriptions

### Signal Outputs

Many aspects of the basic actuated signal controller operation are described by the existing enumerations. Every detector input and every phase and overlap state, both vehicle and pedestrian, is captured in those enumerations.

Phase-driven outputs are used for protected vehicle movements (with three-section signal heads) and some permitted movements (typically two sections of a five-section signal head). The events related to these are shown in Figure 1‑2. The chart shows both the visible output display and the internal phase states, which may go through several intervals within the controller. Certain controllers may have other intervals or may not support some of these. Some of these states are optional. The following discussion talks through these:

* The “Last Phase” indicates that the phase is currently inactive, while the controller is serving another phase on the appropriate ring. The output display is red. The start of service on the current phase is marked by Event 0.
* Red Revert – this is an optional interval used to provide extra red time when the controller returns to serve a phase that just finished terminating. The end of the Red Revert interval (“A”) is not captured by a particular event besides the beginning of the minimum green.
* Delayed Green – this is used to establish a leading pedestrian interval by delaying the start of green by a few seconds. The start of the interval is not necessarily captured by an event, other than the start of the phase service (Event 0). It is not distinguishable from red revert (if any is used). The start of minimum green (Event 1) would typically mark the end of the delayed green interval.
* Min Green – the minimum amount of green time as controlled by a “phase table” entry in the controller. The boundaries of the interval are marked by events 1 and 3.
* Initial Green – an amount of green time that accumulates when added initial timing is used. Typically, the initial green interval starts at the same time as the minimum green. The initial green may sometimes be longer if there are enough actuations to increase the initial green beyond the minimum green value. A transition from minimum to initial green would be marked by Event 3, but the end of initial green (when transitioning to another type of green) is not noted by any existing event.
* Extended Green – this is used when a passage detector is used to lengthen the duration of green while the passage timer is active. There is no direct measurement of when the extended green begins or ends (“B” and “C”), but it may be possible to infer it from detectors.
* Green Rest – this is used for phases where the controller rests in green. Typically, this happens when there is no call or any conflicting phase, or the phase is at the barrier and waiting for phases in other rings to expire. Within the current event codes, there is no differentiation between this green and other green intervals after the minimum. The end of the green rest will usually be marked by phase termination, for which several events are defined. Transitions back to extended green, as may occur when a simultaneous gap is used, are not recorded by events but can be inferred from detector events.
* Guaranteed Passage – this is a green interval occurring after the phase is terminated, but the green display is maintained for a few moments to allow for vehicles to finish traveling through the intersection before changing the display. It is not clear whether the “termination type” events 4, 5, 6 would typically be written before or after the start of the guaranteed passage. Event 7 is specified as being written at the end of the green indication, regardless of which green state was active at that time.
* The yellow clearance interval corresponds to the yellow display and is governed by its own timer. The event would typically be demarcated by events 7 and 8 for the start and 9 and 10 for the end.
* The red clearance interval corresponds to the beginning of red on the phase, where the phase continues to time before releasing to the next phase. This interval is demarcated by events 9 and 10 for the start, and events 11 and 12 for the end.
* There are no explicit events that would mark a “red clearance extension” as might be used to prevent red-light running crashes under certain configurations, to differentiate from other types of red clearance. It may be able to infer this from detector activity. (This is not shown in Figure 1‑2).



Figure 1‑2. Vehicle displays and internal states for phase-driven outputs.

Overlap-driven outputs are used for a variety of purposes; one of the more common types of uses is for right-turn arrows which display a green indication at the same times as the adjacent through or a nonconflicting left turn are active. Regardless of their use, overlaps typically have fewer interval types than phases, mainly because they inherit their displays from parent phases. There are, however, a few timed intervals related to leading and trailing intervals. Figure 1‑3 shows interval states.

* The overlap typically becomes active when one of its parent phases turns green. In some cases, a leading green interval may be used to cause the overlap to turn green prior to the parent phase turning green. This might be achieved with the use of a “delayed green” interval on the phase. After the leading interval, the overlap green begins. Event 61 marks the beginning of green regardless of whether a leading interval is used; the change from leading green to overlap green (“D”) is not marked by any specific event, although it can be inferred from the parent phase states.
* Another event is that of trailing green, which is used for some specific locations where the overlap needs to display a green longer than the parent phase. The trailing green interval is demarcated by events 62 and 63.
* The overlap yellow clearance follows the end of green. The yellow time is typically inherited from the parent phase that is currently terminating. The yellow interval is demarcated by events 63 and 64.
* The overlap red clearance interval follows the yellow. As before, this amount of time is typically inherited from the parent phase. The red clearance interval is demarcated by events 64 and 65. The overlap becomes inactive again after the overlap is no longer active.



Figure 1‑3. Vehicle displays and internal states for overlap-driven outputs.

Flashing yellow arrow (FYA) displays are often an application of overlaps, but they are a bit more complicated. Their programming may also vary with the controller type and site configuration. Figure 1‑4 shows the states of a protected-permitted left turn (PPLT) with a 4-section signal head configuration including a flashing yellow section. The associated events are shown at the bottom for the protected left-turn phase (φL), opposing through phase (φL), and the overlap controlling the permitted output.

* The protected phase events for the protected left-turn phase are covered by those for individual phases, as explained earlier (Figure 1‑2). The resulting left-turn signal head display will be a sequence of green, yellow, and red clearance intervals which can be determined by the same events as for protected phases.
* In some controllers, an optional delay interval may be used to prevent the FYA from starting immediately after the end of the protected phase. The beginning of this interval should coincide with events 11 and 12 of the protected left-turn phases and Event 1 of the opposing through phase. The end of the interval should coincide with Event 61.
* The FYA display is provided by the overlap and should be displayed as long as the opposing through is green. The end of the display will be marked by Event 63.
* The (solid) yellow clearance and red clearance intervals are displayed after the end of the FYA and will be marked by the same events at an ordinary overlap output (Figure 1‑3).



Figure 1‑4. Vehicle displays and internal states for FYA displays.

Pedestrian outputs are illustrated in Figure 1‑5.

* Typically, the Walk interval will come up immediately when the phase is in service, but it is possible to delay this in some controllers with the use of a delayed walk interval. At present, there is no event directly corresponding to a delayed walk, but the duration of such an interval could be inferred from the start of the phase (Event 0) and the start of walk (Event 21).
* Under typical operation, the walk will be displayed with a sequence of fixed-length Walk and Ped Clearance intervals. However, under some circumstances, a “Walk Hold” interval will be used. This is typically used for “Call to Non-Actuated” phases operating under coordination or fixed-time modes so that the Walk display is extended as long as possible when the associated vehicle phase is unable to terminate early (because of coordination or fixed-time operation). In some cases, the controller may rest in walk on a phase (e.g., when there is no call on other phases). These are both distinct from a Walk interval which is actively timed. A timed Walk usually must follow the Walk Hold or Walk Rest in order to ensure that the Walk display was shown for the required duration. At present, there are no events to mark the end of Walk Hold (“E”) or Walk Rest (“F”).
* The Ped Clearance interval is usually an interval of fixed duration coinciding with the flashing Don’t Walk display. Some advanced controller logic may be able to extend it in response to a pedestrian presence in the crosswalk. This is not a standard controller feature. The start and end of ped clearance are marked by events 22 and 23.
* After the end of ped clearance, the ped phase goes into a Don’t Walk state, of which different types may exist in different controllers. In some cases, a timed “Don’t Walk” interval may be used to guarantee that the Don’t Walk display remains up for a minimum amount of time. Where such an interval is used, there is no way to note its end (“G”). The end of the phase would be marked by Event 12 for the associated vehicle phase.



Figure 1‑5. Pedestrian output display.

The above analysis identifies the following signal display elements that are not currently captured:

* Certain interval transitions related to vehicle phases are not directly captured, including the end of a Red Revert interval and the beginning or ending of a green extension. It is probably not necessary to explicitly denote green extensions because that can be inferred from associated detectors or passage timer events (see next section). It may be useful to mark the end of initial green.
* The end of a leading green interval for overlaps.
* The beginning or end of Walk Hold and Walk Rest intervals for pedestrian phases, or any “timed” Don’t Walk intervals if they exist.

In addition, the following are also not captured by the current enumerations:

* Signal outputs not directly controlled by phase or overlap timing (such as a load switch output set directly by other logic).
* Information pertaining to pedestrian countdown displays. The numerical display will usually change predictably, and the total time can be inferred from the start and end times of the interval. However, unexpected changes in the numerical display would not be captured. This does not occur in current signal control, but may be a consideration at some future time.

### Basic Actuation

Basic actuation is captured by the existing event codes. The current enumerations do not show individual extensions of active phases but must be inferred from the detector states. Individual phase calls are included (events 43 and 44).

Figure 1‑6 shows a view of passage timer operation and its representation in the current enumerations.

* The start of passage timer countdown and times where its value is reset are not directly captured but can be inferred from detector states.
* The first conflicting call time is marked by Event 2 (phase check).
* The expiration of a passage timer (“H”) is not explicitly captured in current enumerations. It can be inferred from detector states if the passage timer value is known. It may be rather complex to capture and analyze passage timer expiration, because several passage timers may exist. A phase passage timer may be used for all detectors mapped to the phase, or individual detector-specific passage timers (for lane-by-lane extension).
* The expiration of the maximum green timer (“I”) would typically be written down when it coincides with the end of green during a max out. However, if the phase cannot terminate for some reason (e.g., the phase is at a barrier and other rings are not ready), the event might not be written.
* In addition, detectors operating with a delay will not place calls on phases until after the delay timer expires. Delay timer activity is not captured in the current enumerations but can be inferred from the on/off states and Event 43 (phase call).

Other actuation-related functions such as gap reduction timing are discussed in later sections.



Figure 1‑6. Basic actuated phase logic based on timers, in absence of gap reduction logic, and existing event codes.

### Added Initial Timing

Added initial timing logic is typically used for actuation when advance detectors are present and stop bar detectors are not present for extension. Figure 1‑7 shows a view of the logic for added initial timing. The individual actuations are already captured in the existing detector on/off event codes (particularly Event 82 for detector “on”), so any other events would likely be redundant. The initial timing (“J”), shown at its maximum value in Figure 1‑7 but which could be at a lower value, is not reflected in the current events. However, it is probably not necessary to write this event. Rather than recording changes to the added initial value, it may be better to include an event to reflect the expiration of the initial green timer, similar to how the end of minimum green is written in the current enumerations.



Figure 1‑7. Added initial timing and existing event codes.

### Gap Reduction Timing

“Gap reduction” timing is used to change the passage time during green. Figure 1‑8 illustrates the logic. At present, the beginning of the “time before reduction” can be inferred from the “phase check” event (2), which marks the first conflicting phase call. The start (“K”) and end (“L”) of the “time to reduce” interval and the value of the minimum gap all require that the user know the controller settings. Gap reduction timing is not reflected in the current enumerations. The start and end of the “time to reduce” intervals seem like appropriate events to add; the gap value itself can be inferred from data.



Figure 1‑8. Gap reduction timing and existing event codes.

### Adaptive/Dynamic Maximum

Some controllers feature the ability to adjust maximum times in a dynamic or adaptive fashion. Figure 1‑9 illustrates the logic. After a certain number of max outs occur, the controller increases the maximum time, up to an ultimate dynamic maximum. It then resets after a gap out occurs.

At present, the initial termination events are captured with events 4 and 5 (gap out / max out), but the change of the maximum value in effect is not known. Increments of the maximum time (“M”), attainment of the greatest possible maximum (“N”), and reset to the initial value (“O”) are not captured in the events.

The actual maximum value in effect can likely be inferred from the data. However, the dynamic increment and decrement of the maximum timer setting could be captured by events.



Figure 1‑9. Adaptive maximum timing and existing event codes.

### Coordination and Transition

The existing enumerations capture several events pertaining to coordination: pattern change (cycle, offset, split) information in events 131–149, and passage of the yield point (Event 151). Figure 1‑10 illustrates events related to the transition between patterns. Event 131 marks the adoption of a new pattern. Event 150 serves several purposes with the use of different parameters, including the passage of the local zero (parameter 5), including entry into transition, as well as current transition modes (add or subtract using parameters 2 and 3 respectively) or when the controller is successfully synchronized (parameter 1). Parameter 6 represents the “pickup” of coordination, the meaning of which is not made explicitly clear in the existing enumerations but may be used to mark when the coordinator becomes active.

The current enumerations do not capture some events, including the following:

* Passage of the system zero (“P”).
* Any “active” cycle/offset/split data relevant to managing transition.
* The reason for entry into transition other than pattern changes—e.g., due to force off errors.

Vendor input included several new events for transition, which are listed in Appendix B.



Figure 1‑10. Pattern transition events.

### Preemption

The existing enumerations include events for preemption. Figure 1‑11 describes the logic for railroad preemption, which is generally more complex than emergency vehicle preemption. The start and end of most of the events relevant to the signal control are captured in the existing logic, including events marking the receipt of advance warning (101) and the preempt call (102), gate down confirmation (103), end of preempt delay (105), start of the track clearance interval (106), the beginning of limited service in the “dwell” state (107), the end of the preempt input (104), and end of limited service (108). Preemption sometimes ends with the placement of the controller into an exit pattern; the duration of the exit pattern can likely be inferred from other events (some controllers might maintain this state for a time while others may simply go to the state and release the controller).

Often, there is a limited amount of interconnect between devices at the grade crossing and at a preempted signal. The start and end of the railroad warning activity (lights flashing and gates descending) are not recorded in current events (“Q” and “R”) but can be inferred from the preempt call times. It may not be realistic to attempt to capture those events directly as the railroad warning equipment does not typically transmit its states to the signal control equipment, other than in places where a gate down circuit is used. The gate down event is specifically defined (Event 103).

In summary, it is not considered necessary at this time to add further events for preemption *timing*. Some new events for preemption detection are shared with priority request detection, as discussed in the next section.



Figure 1‑11. Railroad preemption timing and existing event codes.

### Priority

The existing enumerations contain four events for priority request operation. These include check-in and check-out events (112 and 115) and indications that the request is accommodated by early green or extended green (113 or 114). Figure 1‑12 shows some examples of priority operation when the request ends due to check-out (Figure 1‑12a) or due to maximum time-out when no check-out is received (Figure 1‑12b). The end of active priority control is not explicitly included in the existing enumerations. It may be reflected by Event 115 although it is not clear that this would occur in the case of a time-out. After finishing service to a request, the controller may inhibit new priority requests for a time. At present, the start (“S”) or end (“T”) of a period of priority request inhibition is not reflected in the event data.



(a) TSP/SCP request ends due to check-out



(b) TSP/SCP request ends due to time-out

Figure 1‑12. Transit signal priority (TSP)/signal control and prioritization (SCP) timing and existing event codes.

Several additional events have been identified for priority requests. These are based primarily on vendor input, with details in Appendix B.

* Specific phases involved in a request
* The priority level of a request
* Priority vehicle ETA, arrival, and time-out
* Delay and extension times relevant to priority requests
* Pattern and sequence information relative to a request
* Priority and preemption detector fault events

### Responsive, Adaptive, and Real-Time Control

There are numerous advanced signal control methods, including traffic-responsive pattern selection, adaptive adjustment, and real-time control. The workings of some of these are accommodated in the present enumerations, while others are not represented.

* Traffic-responsive pattern selection activity is captured in coordination events pertaining to pattern change (131–149).
* Adaptive changes to coordination status due, such as decisions whether to capture or release controllers from coordinated groups, could be deduced from existing events.
* Some adaptive systems make changes to cycle, offset, splits, and sequence.
  + Figure 1‑13 shows a schematic of this type of operation. Over time, the controller is put into a series of new patterns (here shown as *i* and *i* + 1).
  + Adaptive changes to cycle, offset, and splits are captured in existing events (132–149). The present data does not differentiate between TOD control or traffic-responsive pattern changes and changes due to adaptive adjustment. It is not immediately clear whether those events will be written on receipt of the command by the controller, or implementation by the coordinator.
  + Transitions related to pattern changes would be identifiable from Event 150 records.
  + The current enumerations do not include specific provisions for adaptive sequence changes (although the selected sequence could be inferred from the observed phase order). There is some overlap between adaptive adjustments and changes made to these values during transition; thus the two are combined in the recommendations to conserve the total number of events. In the future, the events can probably be separated.
* Real-time control decisions are not reflected in the current enumerations. The outcome of those decisions may be captured by other events (such as phase holds and omits). However, the internal events relative to real-time control are not directly written down.

Of the above, real-time control is probably the most difficult to arrive at a set of common events, because of the variety of processes comprising such control. For example, some real-time methods compute phase durations in advance, and may update these continually based on the changing arrangement of traffic on the approaches to the intersection; other real-time methods make phase extension or termination decisions based on those conditions and may update those continually. These internal processes will likely rely on variables and interval types that extend beyond current common definitions. It might be possible to define events such as when the control is “committed” to a particular decision. However, it remains to be seen, for analysis purposes, whether such events would actually provide useful information beyond what is already written by the existing events. Further research is likely needed to identify events that may be common to many such systems.



Figure 1‑13. Adaptive changes to cycle, offset, and splits and existing event codes. \*Event 150 will be accompanied by different parameters to mark the start and end of transition.

### Analysis Summary

The preceding analysis identified 20 specific events (A…T) in graphics, as well as a few additional types of events, which are summarized below along with recommendations on whether or not such events should be included. This list does not include numerous additions to priority control that were previously mentioned, as they largely are driven by vendor input described in Appendix B.

Item Figure Callout Description Recommendation

| **Item** | **Figure** | **Callout** | **Description** | **Recommendation** |
| --- | --- | --- | --- | --- |
|  | Figure 1‑2 | A | End of Red Revert | Event should be included. |
|  | Figure 1‑2 | B | End of Initial Green | Event should be included. |
|  | Figure 1‑2 | C | End of Extended Green | Not necessary as extension could be inferred from detector states or passage timer states. |
|  | Figure 1‑3 | D | End of Overlap Leading Green | Event should be included. |
|  | Figure 1‑5 | E | End of Walk Hold interval | Beginning and end of Walk Hold should be included. |
|  | Figure 1‑5 | F | End of Walk Rest interval | Beginning and end of Walk Rest should be included. |
|  | Figure 1‑5 | G | End of a timed Don’t Walk interval | Probably not necessary as few controllers directly control the don’t walk state in this manner. |
|  | Figure 1‑6 | H | Passage timer expiration (independent of detection) | May not be necessary due to ability to infer extensions from detector data and gap reduction event data (if included). However, it may be useful as an easier way to examine extensions. More work needs to be done to specify how it should be written (does the parameter refer to the associated phase or detector, or are they separate event codes?) |
|  | Figure 1‑6 | I | End of maximum green time when the phase cannot be terminated | Not necessary as this can be inferred from the actual duration of green. |
|  | Figure 1‑7 | J | When maximum added initial time is reached | Not necessary as it can be inferred from detector states. Duration of initial green could be inferred from the addition of an “end of Initial Green” event. |
|  | Figure 1‑8 | K | End of Time Before Reduction | Event should be included. |
|  | Figure 1‑8 | L | End of Time To Reduce | Event should be included. |
|  | Figure 1‑9 | M | Dynamic max increment | Event should be included. |
|  | Figure 1‑9 | N | Dynamic max ceiling reached | Event should be included. |
|  | Figure 1‑9 | O | Dynamic max reset | Event should be included. |
|  | Figure 1‑10 | P | System Zero | Not necessary for inclusion as it can be inferred from the cycle length in effect. |
|  | Figure 1‑11 | Q | Beginning of active railroad warning (lights, gates) | Additional analysis needed to determine whether such events can readily be measured by a signal controller and if there is value to writing them. |
|  | Figure 1‑11 | R | End of active railroad warning (lights, gates) |
|  | Figure 1‑12 | S | Begin priority request inhibition | Event should be included. |
|  | Figure 1‑12 | T | End priority request inhibition | Event should be included. |
|  | N/A | N/A | Begin, end, or change of signal output not controlled by phase or overlap timing | More analysis needs to be done to identify events that could trigger output changes. One such event is logic processor events. The ability of logic processor flags to be logged would be useful. |
|  | N/A | N/A | Changes in pedestrian countdown timing | Not necessary for inclusion at this time because this type of operation does not currently exist. |
|  | N/A | N/A | Detector delay timer activity | Not necessary because the start and end of delay can be inferred from detector and phase call events. |
|  | N/A | N/A | Reason for Entry into Transition | Event should be included. Can probably be integrated as other parameters under existing Event 150. |
|  | N/A | N/A | Sequence changes | Event should be included. |
|  | N/A | N/A | Real-time phase control decisions | More analysis is needed to identify potential events that could not be included in current enumerations. |

## Candidates for New Event Codes

The following table lists the events that are recommended for consideration as new event codes in the enumerations. These combine the previous analysis with comments received from vendors (summarized in Appendix B).

| **ID** | **Refer-ences** | **Name** | **Parameter** | **Description** |
| --- | --- | --- | --- | --- |
|  | I1 | Active Split (for Phase #)  (up to 16 separate event codes) | Value in Seconds | The current value of a dynamic split as set by adaptive control, transition, or other dynamic process. Existing events 134–149 would be used only to reflect pattern changes. |
|  | I2 | Active Cycle Length | Value in Seconds | The current value of a dynamic cycle length as set by adaptive control, transition, or other dynamic process. Existing Event 132 would be used only to reflect pattern changes. |
|  | I4, E2, U3 | Active Offset | Value in Seconds | The current value of a dynamic offset as set by adaptive control, transition, or other dynamic process. Existing Event 133 would be used only to reflect background pattern changes. |
|  |  | Active Sequence | Sequence Number | The current value of a dynamic sequence change as set by adaptive control, transition, or other dynamic process. |
|  |  | Sequence | Sequence Number | Gives the ID of the “background” pattern in effect, or when this is changed. This would be similar to existing events 131–149. |
|  | I5 | Sequence Change Request | Sequence Number | Logs a requested sequence number. |
|  | I6 | System Zero | To be determined | To be written when the system (master cycle) zero is encountered, similar to local zero as part of existing Event 150. |
|  | E7 | SCP Request Phase | Phase Number | Gives the number of a priority requested phase. |
|  | E8 | SCP Request Class Level | To be determined | Gives the level of a priority (SCP) request. The requested phase would need to be reconciled with the level, possibly by pairing with Event A9. |
|  | E9 | SCP Request ETA | Phase Number | Provides the estimated time of arrival for a new request (not intended to be updated in real time). |
|  | E10 | SCP Vehicle Arrival | Phase Number | Indicates the arrival of a vehicle at an intersection. |
|  | E11 | SCP Phase Cancel | Phase Number | Indicates that an SCP request for a phase is canceled. |
|  | E12 | SCP Begin Early Green | Phase Number | The controller is serving SCP with early green on the indicted phase. Analogous to existing Event 113 for TSP. |
|  | E13 | SCP Begin Green Extend | Phase Number | The controller is serving SCP with a green extension on the indicated phase. Analogous to existing Event 114 for TSP. |
|  | I8 | TSP Delay Time | Amount of time | Tracks the time from TSP arrival to when TSP phase service begins. |
|  | I9 | TSP Coord Pattern Select | Pattern number | Indicates the selection of coordination pattern during a TSP request. |
|  | I15 | TSP Sequence Entry | Sequence number | Indicates the selection of sequence during a TSP request. |
|  | I25 | TSP Begin Service | Prioritor | Indicates beginning of service of specified TSP request. |
|  | I26 | TSP End Service | Prioritor | Indicates end of service of specified TSP request. |
|  | I10 | TSP Hold Time | Phase hold time (deciseconds) | The amount of time the TSP request held the active phase past the programmed force off or max green time. |
|  | I11 | TSP Early Force Off Time | Phase hold time (deciseconds) | The amount time the TSP request terminated the active phase before the programmed force off or max green time. |
|  | I12 | TSP Delay Begin | Prioritor | Set when the TSP delay interval begins. |
|  | I13 | TSP Delay End | Prioritor | Set when the TSP delay interval ends. |
|  | I14 | TSP Arrival Time Expired | Prioritor | Set when the TSP arrival time reaches 0 |
|  |  | Begin Inhibited TSP | Prioritor | Set when TSP inputs are inhibited on the prioritor. |
|  |  | End Inhibited TSP | Prioritor | Set when TSP inputs are no longer inhibited on the prioritor. |
|  | I16 | Priority/Preemption Detector Low – On | Priority/Preemption Detector Number | Set when pri/pre detector low call is received. |
|  | I17 | Priority/Preemption Detector Low – Off | Priority/Preemption Detector Number | Set when pri/pre detector low call is cleared. |
|  | I18 | Priority/Preemption Detector High – On | Priority/Preemption Detector Number | Set when pri/pre detector high call is received. |
|  | I19 | Priority/Preemption Detector High – Off | Priority/Preemption Detector Number | Set when pri/pre detector high call is cleared. |
|  | I20 | Priority/Preemption Detector Fault – Open Loop | Priority/Preemption Detector Number | Priority Detector Diagnostics: Set when "No Activity" on detector (See NTCIP for Veh Det Diagnostics). |
|  | I21 | Priority/Preemption Detector Fault – Shorted | Priority/Preemption Detector Number | Priority Detector Diagnostics: Set when "Max Presence" on detector (See NTCIP for Veh Det Diagnostics). |
|  | I22 | Priority/Preemption Detector Fault – Excessive Charge | Priority/Preemption Detector Number | Priority Detector Diagnostics: Set when "Erratic Count" on detector (See NTCIP for Veh Det Diagnostics). |
|  | I24 | Priority/Preemption Detector Fault – Other | Priority/Preemption Detector Number | Priority Detector Diagnostics: Used for other error types. |
|  | I23 | Priority/Preemption Detector Fault – Detector Restored | Priority/Preemption Detector Number | Priority Detector Diagnostics: Set when detector fault on detector cleared (See NTCIP for Veh Det Diagnostics). |
|  | E1, U1 | Gap Reduction Start | Phase Number | Indicates beginning of gap reduction. |
|  | E1, U2 | Gap Reduction End | Phase Number | Indicates end of gap reduction. |
|  |  | End Red Revert | Phase Number | Indicates expiration of a red revert interval. |
|  |  | End Initial Green | Phase Number | Indicates expiration of initial green timer (used with added initial timing). |
|  |  | Dynamic Maximum Increment | Phase Number | Indicates that max timer value is incremented. |
|  |  | Dynamic Maximum Ceiling | Phase Number | Indicates that max timer value has been incremented to the maximum possible value. |
|  |  | Dynamic Maximum Reset | Phase Number | Indicates that max timer value is reset to initial value. |
|  |  | Detector Gap Out | Detector Number | For lane-by-lane extension of an active phase. This indicates that a passage timer specific to the detector is expired. Do not write the event if it is coincident with “detector off” (i.e., the passage time is set to zero). |
|  |  | Detector End Delay | Detector Number | Logs the expiration of a delay timer associated with a detector. Event 81 (“detector on”) would mark the beginning of the delay interval. Do not write if there is no delay time. |
|  | I7 | Force Off Error: Pedestrian Intervals Still Timing |  | Logs when coordination is put into transition because a phase cannot be forced off terminated due to active pedestrian timing or other reasons. These might be new parameters for existing Event 150. |
|  |  | Force Off Error: Minimum Green Still Timing |  |
|  |  | Force Off Error: Other |  |
|  | E4 | External/NTCIP Hold | Phase Number | These three events would write phase holds, force-offs, and omits based on external control. If included, existing events 6, 41, 42, 46, and 47 should be revised so that they are not written at the same time as these events. |
|  | E5 | External/NTCIP Force Off | Phase Number |
|  | E6 | External/NTCIP Omit | Phase Number |
|  |  | Logic Processor Active | Logic function ID | Indicates that the logic function is active. |
|  |  | Logic Processor Inactive | Logic function ID | Indicates that the logic function is inactive. |
|  |  | Logic Processor True | Logic function ID | Indicates that the logic function has a “true” (1) output state. |
|  |  | Logic Processor False | Logic function ID | Indicates that the logic function has a “false” (0) output state. |
|  | E14, U5 | Controller Database Upload | To be determined | These events note that changes to the controller database have been made by various sources. The parameter might be used to indicate the type of change made, or these events might be separate parameters of a single new event code. |
|  | U4 | NTCIP Set Command | To be determined |
|  |  | Front Panel Change | To be determined |
|  |  | Critical Database Change |  | This event would log when the controller seeks an all-red transition or other default state because of a critical change to the control settings. The event is written at the time when the critical change is made. |
|  | M1, M5 | Clock Update Old Time | Reference Number | These events are used to mark clock changes with the old time and new time. The correction can be calculated from the two timestamps. The Reference Number is proposed as a way to link the two numbers in case they are not written sequentially or have to be matched by a query from events in a table. The Reference Number would be unique within a 24-hour period. |
|  | M1, M5 | Clock Update New time | Reference Number |
|  | M3 | Data Logger Enabled | To be determined | Indicates the time when the data logger is enabled. |
|  | M4 | Data Logger Disabled | To be determined | Indicates the time when the data logger is disabled. |
|  | M2 | Buffer Overflow | N/A | Indicates memory issues leading to loss of event data. |

## Summary

This document presents a series of candidate events to be considered for inclusion in high-resolution data as new enumerations. The document begins by providing background on high-resolution data and where it fits in the overall ecosystem of traffic signal information. High-resolution data offers one key difference that reveals the niche it fills in this ecosystem: it acts as an interoperable record of past events at an intersection. All of the other data sources either (1) only provide the *current* intersection states to external data systems; (2) aggregate past states, obscuring useful details; or (3) are fully proprietary and consequently not interoperable. The only way to get the equivalent data would be to duplicate the data logger concept.

Next, the document reviews a series of proposed events and synthesizes these to form a list of candidate events for consideration. At present, there are 146 reserved events. Three vendors and one agency provided input, with 52 events identified in total. Some of these were overlapping and there were potential redundancies with existing events. A few more events were identified by examining some basic signal functions and examining whether key events were reflected in current enumerations. These candidate events were reconciled and distilled into a list of 51 potential events for consideration. These add additional detail, especially for transit signal priority (TSP) and signal control and prioritization (SCP) types of control, as well as a few additional details for actuation, transition, and basic signal input/output functions. Additional dialog is needed with users of the high-resolution data before moving forward with new events, but this document may serve as a starting point to begin such dialog.

Appendix A: Existing Event Definitions

A.1. Phase Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 0 | Phase On | Phase # (1–16) | Set when NEMA Phase On becomes active, either upon start of green or walk interval, whichever comes first. |
| 1 | Phase Begin Green | Phase # (1–16) | Set when either solid or flashing green indication has begun. Do not set repeatedly during flashing operation. |
| 2 | Phase Check | Phase # (1–16) | Set when a conflicting call is registered against the active phase. (Marks beginning of max timing). |
| 3 | Phase Min Complete | Phase # (1–16) | Set when phase min timer expires. |
| 4 | Phase Gap Out | Phase # (1–16) | Set when phase gaps out, but may not necessarily occur upon phase termination. Event may be set multiple times within a single green under simultaneous gap out. |
| 5 | Phase Max Out | Phase # (1–16) | Set when phase max timer expires, but may not necessarily occur upon phase termination due to last car passage or other features. |
| 6 | Phase Force Off | Phase # (1–16) | Set when phase force off is applied to the active green phase. |
| 7 | Phase Green Termination | Phase # (1–16) | Set when phase green indications are terminated into either yellow clearance or permissive (FYA) movement. |
| 8 | Phase Begin Yellow Clearance | Phase # (1–16) | Set when phase yellow indication becomes active and clearance timer begins. |
| 9 | Phase End Yellow Clearance | Phase # (1–16) | Set when phase yellow indication becomes inactive. |
| 10 | Phase Begin Red Clearance | Phase # (1–16) | Set only if phase red clearance is served. Set when red clearance timing begins. |
| 11 | Phase End Red Clearance | Phase # (1–16) | Set only if phase red clearance is served. Set when red clearance timing concludes. This may not necessarily coincide with completion of the phase, especially during clearance of trailing overlaps, red revert timing, red rest, or delay for other ring terminations. |
| 12 | Phase Inactive | Phase # (1–16) | Set when the phase is no longer active within the ring, including completion of any trailing overlaps or end of barrier delays for adjacent ring termination. |
| 13–20 | Reserved |  |  |

A.2. Pedestrian Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 21 | Pedestrian Begin Walk | Phase # (1–16) | Set when walk indication becomes active. |
| 22 | Pedestrian Begin Clearance | Phase # (1–16) | Set when flashing don’t walk indication becomes active. |
| 23 | Pedestrian Begin Solid Don’t Walk | Phase # (1–16) | Set when don’t walk indication becomes solid (nonflashing) from either termination of ped clearance, or head illumination after a ped dark interval. |
| 24 | Pedestrian Dark | Phase # (1–16) | Set when the pedestrian outputs are set off. |
| 25–30 | Reserved |  |  |

A.3. Barrier/Ring Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 31 | Barrier Termination | Barrier # (1–8) | Set when all active phases become inactive in the ring and cross barrier phases are next to be served. |
| 32 | FYA – Begin Permissive | FYA # (1–4) | Set when flashing yellow arrow becomes active. |
| 33 | FYA – End Permissive | FYA # (1–4) | Set when flashing yellow arrow becomes inactive through either clearance of the permissive movement or transition into a protected movement. |
| 34–40 | Reserved |  |  |

A.4. Phase Control Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 41 | Phase Hold Active | Phase # (1–16) | Set when phase hold is applied by the coordinator, preemptor, or external logic. Phase does not necessarily need to be actively timing for this event to occur. |
| 42 | Phase Hold Released | Phase # (1–16) | Set when phase hold is released by the coordinator, preemptor, or external logic. Phase does not necessarily need to be actively timing for this event to occur. |
| 43 | Phase Call Registered | Phase # (1–16) | Call to service on a phase is registered by vehicular demand. This event will not be set if a recall exists on the phase. |
| 44 | Phase Call Dropped | Phase # (1–16) | Call to service on a phase is cleared by either service of the phase or removal of call. |
| 45 | Pedestrian Call Registered | Phase # (1–16) | Call to service on a phase is registered by pedestrian demand. This event will not be set if a recall exists on the phase. |
| 46 | Phase Omit On | Phase # (1–16) | Set when phase omit is applied by the coordinator, preemptor, or other dynamic sources. Phase does not necessarily need to be actively timing for this event to occur. This event is not set when phase is removed from the active sequence or other configuration-level change has occurred. |
| 47 | Phase Omit Off | Phase # (1–16) | Set when phase omit is released by the coordinator, preemptor, or other dynamic sources. Phase does not necessarily need to be actively timing for this event to occur. This event is not set when phase is added from the active sequence or other configuration-level change has occurred. |
| 48 | Pedestrian Omit On | Phase # (1–16) | Set when ped omit is applied by the coordinator, preemptor, or other dynamic sources. Phase does not necessarily need to be actively timing for this event to occur. This event is not set when phase is removed from the active sequence or other configuration-level change has occurred. |
| 49 | Pedestrian Omit Off | Phase # (1–16) | Set when ped omit is released by the coordinator, preemptor, or other dynamic sources. Phase does not necessarily need to be actively timing for this event to occur. This event is not set when phase is added from the active sequence or other configuration-level change has occurred. |
| 50–60 | Reserved |  |  |

A.5. Overlap Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 61 | Overlap Begin Green | Overlap # (as number A=1 B=2, etc) | Set when overlap becomes green. Do not set repeatedly when overlap is flashing green. Note that overlap colors are consistent to the GYR intervals resultant from the controller programming and may not be indicative of actual signal head colors. |
| 62 | Overlap Begin Trailing Green (Extension) | Overlap # | Set when overlap is green and extension timers begin timing. |
| 63 | Overlap Begin Yellow | Overlap # | Set when overlap is in a yellow clearance state.  Note that overlaps which drive yellow field indications during a dwell state may be reported as green or inactive. (common to midblock signals). |
| 64 | Overlap Begin Red Clearance | Overlap # | Set when overlap begins timing red clearance intervals. |
| 65 | Overlap Off (Inactive with red indication) | Overlap # | Set when overlap has completed all timing, allowing any conflicting phase next to begin service. |
| 66 | Overlap Dark | Overlap # | Set when overlap head is set dark (no active outputs). The end of this interval shall be recorded by either an overlap off state or other active overlap state. |
| 67 | Pedestrian Overlap Begin Walk | Overlap # | Set when walk indication becomes active. |
| 68 | Pedestrian Overlap Begin Clearance | Overlap # | Set when flashing don’t walk indication becomes active. |
| 69 | Pedestrian Overlap Begin Solid Don’t Walk | Overlap # | Set when don’t walk indication becomes solid (nonflashing) from either termination of ped clearance, or head illumination after a ped dark interval. |
| 70 | Pedestrian Overlap Dark | Overlap # | Set when the pedestrian outputs are set off. |
| 71–80 | Reserved |  |  |

A.6. Detector Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 81 | Detector Off | DET Channel # (1–64) | Detector on and off events shall be triggered post any detector delay/extension processing. |
| 82 | Detector On | DET Channel # (1–64) |  |
| 83 | Detector Restored | DET Channel # (1–64) | Detector restored to nonfailed state by either manual restoration or re-enabling via continued diagnostics. |
| 84 | Detector Fault- Other | DET Channel # (1–64) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 85 | Detector Fault- Watchdog Fault | DET Channel # (1–64) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 86 | Detector Fault- Open Loop Fault | DET Channel # (1–64) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 87 | Detector Fault- Shorted Loop Fault | DET Channel # (1–64) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 88 | Detector Fault- Excessive Change Fault | DET Channel # (1–64) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 89 | PedDetector Off | DET Channel # (1–16) | Ped detector events shall be triggered post any detector delay/extension processing and may be set multiple times for a single pedestrian call (with future intent to eventually support ped presence and volume). |
| 90 | PedDetector On | DET Channel # (1–16) |  |
| 91 | Pedestrian Detector Failed | Ped Det # (1–16) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 92 | Pedestrian Detector Restored | Ped Det # (1–16) | Detector failure logged upon local controller diagnostics only (not system diagnostics). |
| 93–100 | Reserved |  |  |

A.7. Preemption Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 101 | Preempt Advance Warning Input | Preempt # (1–10) | Set when preemption advance warning input is activated. |
| 102 | Preempt (Call) Input On | Preempt # (1–10) | Set when preemption input is activated (prior to preemption delay timing). May be set multiple times if input is intermittent during preemption service. |
| 103 | Preempt Gate Down Input Received | Preempt # (1–10) | Set when gate down input is received by the controller (if available). |
| 104 | Preempt (Call) Input Off | Preempt # (1–10) | Set when preemption input is de-activated. May be set multiple times if input is intermittent preemption service. |
| 105 | Preempt Entry Started | Preempt # (1–10) | Set when preemption delay expires and controller begins transition timing (force off) to serve preemption. |
| 106 | Preemption Begin Track Clearance | Preempt # (1–10) | Set when track clearance phases are green and track clearance timing begins. |
| 107 | Preemption Begin Dwell Service | Preempt # (1–10) | Set when preemption dwell or limited service begins or minimum dwell timer is reset due to call drop and reapplication. |
| 108 | Preemption Link Active On | Preempt # (1–10) | Set when linked preemptor input is applied from active preemptor. |
| 109 | Preemption Link Active Off | Preempt # (1–10) | Set when linked preemptor input is dropped from active preemptor. |
| 110 | Preemption Max Presence Exceeded | Preempt # (1–10) | Set when preemption max presence timer is exceeded and preemption input is released from service. |
| 111 | Preemption Begin Exit Interval | Preempt # (1–10) | Set when preemption exit interval phases are green and exit timing begins. |
| 112 | TSP Check-In | TSP # (1–10) | Set when request for priority is received. |
| 113 | TSP Adjustment to Early Green | TSP # (1–10) | Set when controller is adjusting active cycle to accommodate early service to TSP phases. |
| 114 | TSP Adjustment to Extend Green | TSP # (1–10) | Set when controller is adjusting active cycle to accommodate extended service to TSP phases. |
| 115 | TSP Check-Out | TSP # (1–10) | Set when request for priority is retracted. |
| 116–130 | Reserved |  |  |

A.8. Coordination Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 131 | Coord Pattern Change | Pattern # (0–255) | Coordination pattern that is actively running in the controller. (Highest priority of TOD, System or manual command). This event will not be reapplied if coordination is temporarily suspended for preemption or other external control. |
| 132 | Cycle Length Change | Seconds (0–255) | This event shall be populated upon selection of a new coordination pattern change that selects a new cycle length. Cycle lengths in excess of 255 shall record this event with a 255 parameter, requiring controller database lookup for this actual value. |
| 133 | Offset Change | Seconds (0–255) | This event shall be populated upon selection of a new coordination pattern change that selects a new cycle length. Offsets in excess of 255 shall record this event with a 255 parameter, requiring controller database lookup for this actual value. |
| 134–149 | Split 1–16 change | New Split Time in Seconds (0–255) | Split change events shall be populated upon selection of a new coordination pattern as well as during a split change to an active pattern via ACS Lite or other adaptive control system. |
| 150 | Coord cycle state change | 0 = free  1 = in step  2 = transition (add)  3 = transition (subtract)  4 = transition (dwell)  5 = local zero  6 = pickup | Set when the appropriate coordinator event occurs. |
| 151 | Coordinated phase yield point | Phase # (1–16) | Set when the coordinator reaches the yield point for the specified phase. |
| 152–170 | Reserved |  |  |

A.9. Cabinet/System Events

| **Id** | **Name** | **Parameter** | **Existing Description** |
| --- | --- | --- | --- |
| 171 | Test Input on | Test Input # (as number A=1 B=2, etc) | Cabinet test or special function input as defined by the local controller. |
| 172 | Test Input off | Test Input # (as number A=1 B=2, etc) |  |
| 173 | Unit Flash Status change | NTCIP Flash state # (0-255) | See NTCIP 1202 - 2.4.5 for definition. |
| 174 | Unit Alarm Status 1 change | NTCIP Alarm Status 1# (0-255) | See NTCIP 1202 - 2.4.5 for definition. |
| 175 | Alarm Group State Change | NTCIP Alarm Group State (0-255) | See NTCIP 1202 - 2.4.5 for definition. |
| 176 | Special Function Output on | Special Function # (0-255) | Special function output as defined by the local controller. |
| 177 | Special Function Output off | Special Function # (0-255) | Special function output as defined by the local controller. |
| 178 | Manual control enable off/on | Manual control enable off/on # (0,1) | Set when manual control input is applied or removed. |
| 179 | Interval Advance off/on | Interval Advance off/on # (0,1) | Manual signal control input: leading edge on (1),  lagging edge (0) optional. |
| 180 | Stop Time Input off/on | Stop Time Input Advance off/on # (0,1) | Set when stop time input is applied or removed, regardless of source. |
| 181 | Controller Clock Updated | Optional parameter: Time correction in Seconds (0-255) | Set when the controller OS clock is adjusted via communications, OS command, or external input. |
| 182 | Power Failure Detected | True (1) | Line voltage drops between 0-89 volts AC for more than 100 ms. |
| 183 | Reserved |  |  |
| 184 | Power Restored | True (1) | Line voltage applied/reapplied greater than 98 volts AC. |
| 185 | Vendor-Specific Alarm | Vendor defined parameter | Placeholder for generic failure/alarm types as defined by vendor. |
| 186–199 | Reserved |  |  |
| 200–255 | Reserved |  |  |

Appendix B: New Event Definitions By Vendor

The “Type” column in these tables lists “New” and “Changed” definitions. “New” events are those that are provided (or proposed to be provided) in addition to the existing events. “Changed” events are an existing event that was implemented differently.

B.1. Events Not Included in Main List

The following events were not included in the proposed candidate list, or their meaning was changed from the vendor definition. The following discussion lists a rationale for these changes or exclusions.

* Event I1 was intended to log the actual split time (sum of actual green, yellow, and red times given). One potential issue with the proposed definition is that there could be additional intervals comprising the split (such as a leading pedestrian interval). Further, because the split time can be calculated from the time between events, this event could instead be used to write down a background split time for a method that changes it from cycle to cycle, such as a transition method or adaptive control method.
* Event I3 was proposed to log a natural cycle time. Because the meaning of natural cycle was certain, this event was not included in the main list.
* Event E3 was proposed to log an active offset change in percent. Another event was proposed to log the same data in seconds. Because cycle length is also logged, and offsets are not given in both seconds and percent in the existing enumerations (Event 133), this event was not included in the main list.
* Event I24 was proposed to log priority/preemption detector failures for any reason. This was changed to failure for reasons other than no activity, max presence, or erratic count, which are covered in events I20, I21, and I22.
* Events E1, U1, and U2 are related to gap reduction logic. These were reconciled to a pair of events that would mark the beginning and end of a gap reduction interval.

B.2. Intelight

| **ID** | **Name** | **Parameter** | **Description** |
| --- | --- | --- | --- |
|  | Actual Split [#] | Value | Logs the actual phase time (green + yellow + red) in coordination or free. Set at the end of phase red clearance (phase end). Events are logged for individual phases indicated by [#], 1–16. |
|  | Actual Cycle | Value | Logs the actual cycle time, each cycle at the end of the programmed sequence. This event is only logged when the controller is operating in coordination mode. |
|  | Actual Natural Cycle | Value | Logs the actual natural cycle time, each cycle at the end of the programmed sequence. This event is only logged when the controller is operating in free mode. |
|  | Actual Offset | Value | Actual cycle offset in seconds. |
|  | Sequence Change Request | Requested sequence ID | Logs the requested sequence number. |
|  | Master Cycle Zero |  | Indicates the time of the system zero in coordination. |
|  | Oversized Ped Transition |  | Logs an oversized pedestrian event that will cause transition when the event occurs (i.e. Start of phase with an oversized pedestrian call). |
|  | Transit Signal Priority (TSP) Delay Time | Amount of time | Tracks the time from TSP arrival to when TSP phase service begins. |
|  | Coord Pattern Select | Pattern number | Indicates the selection of coordination pattern during a TSP request. |
|  | TSP Hold Time | Phase hold time in deciseconds | The amount of time the TSP request held the active phase past the programmed force off or max green time. |
|  | TSP Early Force Off Time | Phase hold time in deciseconds | The amount time the TSP request terminated the active phase before the programmed force off or max green time. |
|  | TSP Delay Begin | Prioritor # (1–8) | Set when the TSP delay interval begins. |
|  | TSP Delay End | Prioritor # (1–8) | Set when the TSP delay interval ends. |
|  | Arrival Time Expired | Prioritor # (1–8) | Set when the TSP arrival time reaches 0 |
|  | Sequence Entry | Sequence # (1–20) | Set when a new sequence is started. |
|  | Pri/Pre Detector Low On | Pri/Pre Detector # (1–32) | Set when pri/pre detector low call is received. |
|  | Pri/Pre Detector Low Off | Pri/Pre Detector # (1–32) | Set when pri/pre detector low call is cleared. |
|  | Pri/Pre Detector High On | Pri/Pre Detector # (1–32) | Set when pri/pre detector high call is received. |
|  | Pri/Pre Detector High Off | Pri/Pre Detector # (1–32) | Set when pri/pre detector high call is cleared. |
|  | Pri/Pre Det Fault Open Loop | Pri/Pre Detector # (1–32) | Priority Detector Diagnostics: Set when "No Activity" on detector (See NTCIP for Veh Det Diagnostics). |
|  | Pri/Pre Det Fault Shorted | Pri/Pre Detector # (1–32) | Priority Detector Diagnostics: Set when "Max Presence" on detector (See NTCIP for Veh Det Diagnostics). |
|  | Pri/Pre Det Fault Excessive Charge | Pri/Pre Detector # (1–32) | Priority Detector Diagnostics: Set when "Erratic Count" on detector (See NTCIP for Veh Det Diagnostics). |
|  | Pri/Pre Det Restored | Pri/Pre Detector # (1–32) | Priority Detector Diagnostics: Set when detector fault on detector cleared (See NTCIP for Veh Det Diagnostics). |
|  | Pri/Pre Det Failed | Pri/Pre Detector # (1–32) | Set whenever a “No Activity,” “Max Presence,” or “Erratic Count” are true. |
|  | Priority Begin Service | Prioritor # (1–8) | Set when a prioritor service is started. |
|  | Priority End Service | Prioritor # (1–8) | Set when a prioritor service has ended. |

B.3. Econolite

| **ID** | **Name** | **Parameter** | **Description** |
| --- | --- | --- | --- |
|  | Reduced Gap Termination | To be determined | Value of a dynamic gap timer that expired. |
|  | Active Offset (Seconds) | Phase Number | Current value of an active (transition) offset, in seconds. Note that the value might be greater than 255. |
|  | Active Offset (Percent) | Phase Number | Current value of an active (transition) offset, in percent. |
|  | External/NTCIP Hold | Phase Number | Indicates a hold placed on a phase by external source. |
|  | External/NTCIP Force Off | Phase Number | Indicates a force off placed on a phase by external source. |
|  | External/NTCIP Omit | Phase Number | Indicates an omit placed on a phase by external source. |
|  | Signal Control and Prioritization (SCP) Phase Request | Phase Number | Indicates that phase is requested for a new SCP request (NTCIP 1211). |
|  | SCP Class Level | Level (1–10) | Provides the class level of an SCP request. |
|  | SCP Request ETA | To be determined | Provides the estimated time of arrival for a new request (not intended to be updated in real time). |
|  | SCP Vehicle Arrival | Phase Number | Indicates the arrival of a vehicle at an intersection. |
|  | SCP Phase Cancel | Phase Number | Indicates that an SCP request for a phase is canceled. |
|  | SCP Phase Begin Early Green | Phase Number | The controller is serving SCP with early green on the indicted phase. |
|  | SCP Phase Begin Green Extend | Phase Number | The controller is serving SCP with a green extension on the indicated phase. |
|  | Controller Database Change | Source of Change | Changes to the controller database have been made. Parameter might be used to identify either the type of change or source of change. |

B.4. McCain

| **ID** | **Name** | **Parameter** | **Description** |
| --- | --- | --- | --- |
| M1 | Controller Clock Update (Event 181) | Time correction | The time correction is entered as a number with possible values –127 to 127 instead of 0 to 255. Also, the event is timestamped with the clock time before the update rather than after the update. |
| M2 | Buffer Overflow | Event 185 / Param 0 | Indicates that a buffer overflow has occurred (it is stated that this “should never happen”). |
| M3 | Logging Enabled | Event 185 / Param 1 | Indicates that the data logger is enabled at the stated time. |
| M4 | Logging Disabled | Event 185 / Param 2 | Indicates that the data logger is disabled at the stated time. |
| M5 | Clock Time Update – New Time | Event 185 / Param 3 | Gives the timestamp after a clock update. Implemented as vendor-specific alarm. |

B.5. Additional Suggested Events

These are additional suggestions obtained from input from UDOT.

| **ID** | **Name** | **Parameter** | **Description** |
| --- | --- | --- | --- |
| U1 | Start Gap Reduction | Phase Number | Indicates that gap reduction logic has begun. |
| U2 | Minimum Gap | Phase Number | Indicates that gap reduction logic has ended, with the passage time now set to minimum. |
| U3 | Current Offset | Offset Value | Indicates a dynamic (adaptive or otherwise) offset change to a value other than the programmed (static) offset. |
| U4 | NTCIP Set Command | To be determined | Indicates that an NTCIP “set” command has been received by the controller. Parameter might be used to identify the type of command received. |
| U5 | Database Change | To be determined | Changes to the controller database have been made. Parameter might be used to identify either the type of change or source of change. |