Example Use of MIPI Battery Interface (BIF) Charging Data Object – A Rule Based Battery Charging Algorithm

White Paper

23 January 2013

Contents

Abstract

The next generation of battery packs will include some form of data containers which can be read by a portable device or a charging solution. In particular, the MIPI® BIF (Battery Interface) working group recently created the standard definition of charging data object ***[MIPI01]***. This charging data object offers, to the battery pack, the capability to clearly declare its most efficient and safest charging profile.

The variability of charging profile from one battery pack to another can be significant. This is problematic when a charging system must be capable of fulfilling precisely each battery requirements, taking into account that the charging profile is fetched from the battery pack data container (for example, using MIPI BIF). In fact, today's charging solutions typically use rigid charging algorithms that cannot be dynamically adapted for new and varying charging profiles.

This paper presents an alternative way of designing the charging algorithm using a rule based approach. This type of algorithm offers excellent adaptability while ensuring correct and more optimal charging system management.

Introduction

Battery charging requires particular attention to ensure safe operation for some battery types. This is true for Lithium based batteries widely used in portable devices today. While charging these batteries, appropriate constraints must be followed to control charging current and voltage precisely. The charging segment, defined as the charging voltage and current combination, depends generally on battery temperature and battery voltage. The charging segment is selected by a charging algorithm and applied on a precisely controllable CC/CV charger.

For a given battery, a charging profile, composed by a set of charging segments, can be established so that charging can be declared safe over temperature and battery voltage (an example is shown in ***Figure 1***). Disrespecting this charging profile may result in various defects such as accelerated aging of the battery, over-heating or even battery physical damages which can cause end-user injuries.

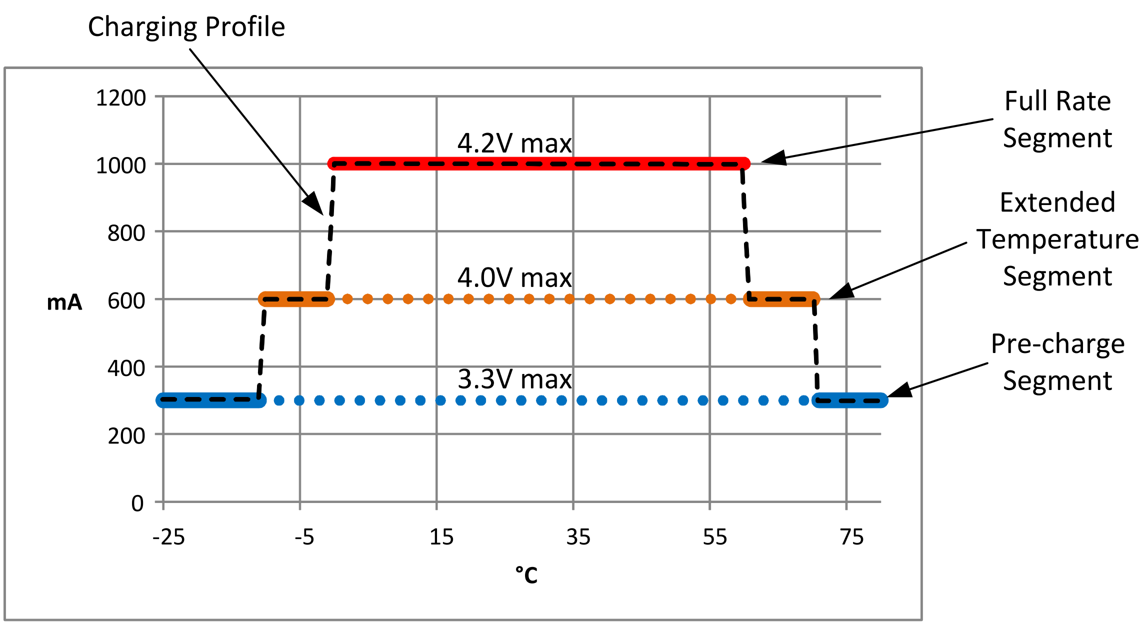

Figure 1 Charging Segments Composing the Charging Profile

A portable device charging subsystem is usually designed for a specific battery or family of batteries. It usually cannot guarantee safe operation or identical performance when used with batteries other than the design prototype. This inflexibility limits the choice of battery throughout the product life.

This strong link between a portable device and a battery type is a significant limitation. Consider the effect of this product design on the multi-sourcing of batteries. Logically, each source of the battery would provide its essential charging profile to the charging subsystem and the product charging subsystem would adjust appropriately. But with a fixed charging subsystem, the approach is quite opposite: the charging subsystem sets the charging profile requirement for the battery sources. Often that results in reduced charging performance and the batteries from different sources are not utilized optimally. It could also result in higher battery unit cost from a given battery supplier because they may need to modify and customize their battery design to support the established charging profile for the charging subsystem of each portable device.

The strong attachment of a battery type to a portable device charging subsystem may also limit the use of newer or improved battery technology which require a different charging profile. Once the portable device is widely available in the market to the end users, it usually cannot use the latest most advanced batteries or even adapt an updated charging profile (for example, improved safe charging rates). In the same way, the end-user may not be able to effectively use larger batteries in after-sales market (probably it would take longer time to charge than necessary for the new battery or perhaps operate with underutilized capacity).

These limitations can be resolved by storing all battery dependent charging profile information in the battery pack. This would allow the portable device to read the specific charging segments of the profile contained in each battery when connected and enable the device to adapt its charging subsystem optimally to the current battery. These key capabilities of the battery are already available on some proprietary smart battery systems but now are available in a public standardization level through MIPI BIF activities. In the future, retrieving charging segments from battery should become more and more popular with portable device manufacturers to ease overall system development effort and improve end-user experience during full product lifetime.

The effectiveness of such a system enhancement requires the portable device design to be capable of processing all the charging segments read from any kind of batteries, even future batteries which did not exist when the portable device was designed. The adaptability of charging algorithm is thus crucial for a robust portable device and the rule based processing proposed in this paper offers an elegant solution.

Limitation of Classical State Machines

Most charging algorithms embedded in portable devices rely on fixed state machines for their correct execution. Each state of the machine corresponds to a particular charging segment, either in direct link to the battery requirements or as needed for particular system use-cases. The transitions between each state are clearly defined to accomplish the desired runtime behavior.

Since the charging segments can now come from battery pack reads, the classical state machine solution has severe issues: the number of states to process must be dynamic as the number of charging segments is variable. Moreover, the valid transitions between states would be very difficult to establish. The simplicity of state machine operation would be significantly degraded by just adding the ability to embed dynamically new states with their transition conditions.

Rule Based Algorithm and Battery Charging

Some systems having complex behavior can be easily managed using the rule based concept. The rule based concept applies when it is possible to establish distinctive rules to change to a given system state.

A rule based system is essentially configured with a set of rules to model the required state transitions and new parameter values under changing conditions. Each rule should define the condition for which the rule is applicable. The rule should also define the system states or parameters to assign when the rule is applied.

The rule based algorithm runs with four consecutive steps. The first step, called acquisition, gets metrics and states of the system. The second step, called validation, validates the applicable rules with system metrics and states. The third step, called election, evaluates the validated rules so that only one is selected. The fourth and last step, called application, applies the states or parameters associated with the selected rule.

Rule based control systems, when fully implementing the concept, have key advantages compared to fixed state machine approaches. First, the transitions from/to any system states are implicit; these transitions are never explicitly coded. Secondly, the addition of rules (allowing for additional system states) does not require any change to the rule based control system but rather, the list of acquired rules to be validated would be longer. Thus the system is easily and dynamically updatable to meet different requirements.

The rule based concept applies well to a battery charging algorithm because only a small number of system metrics and states are sufficient to validate the rules. The validation metrics would be essentially based on battery voltage and battery temperature, and the rule application would mainly consist of CC/CV charger programming. In such applications, a rule represents a charging segment which is valid if temperature and voltage of battery are within the defined range.

A rule based algorithm applied for battery charging would be triggered by a regular charging clock tick (for example, in the range of 1s to 5s) but could also be triggered by asynchronous high priority events (CC/CV charger defaults for example).

At each tick the algorithm runs the four steps described before: acquisition, validation, election and application.

The acquisition phase consists of getting the key system states and metrics (***Figure 2***).

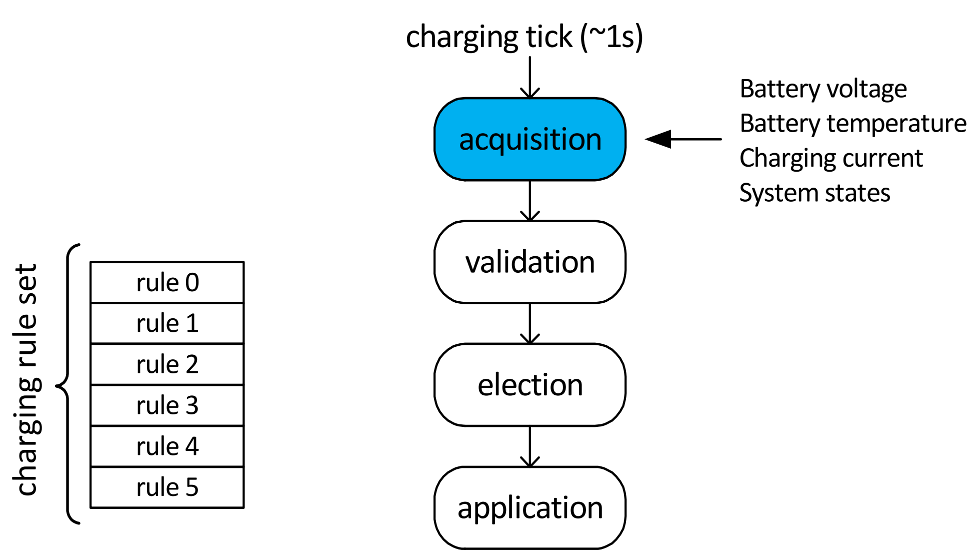

Figure 2 Acquisition Step

From these metrics and states, the algorithm processes the charging rule set and tags the ones which are valid. After this step, some rules would become invalidated (***Figure 3***).

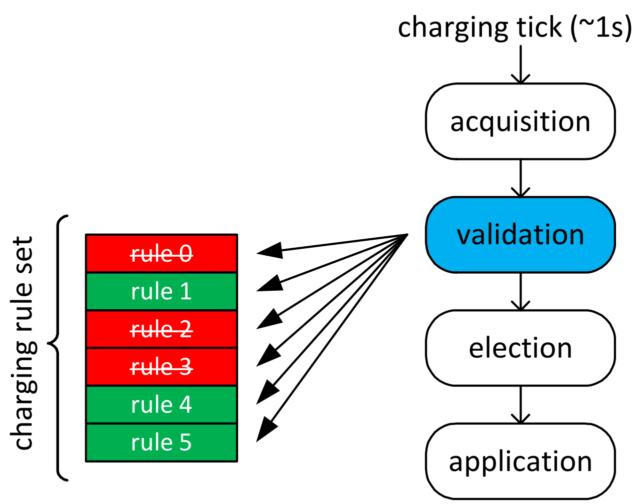

Figure 3 Validation Step

After validation, a single rule from the valid set should be elected as the active rule. The simplest way to make the election is to promote the first valid rule in the list (***Figure 4***). Thus, the ordering of the rules in the list is critical.

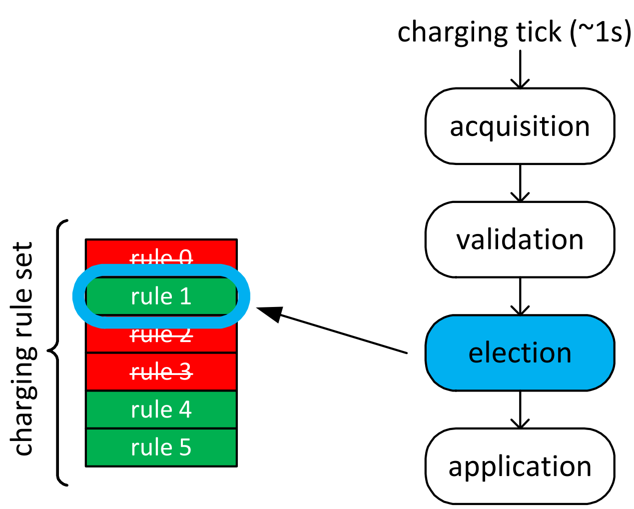

Figure 4 Election Step

Finally, the elected rule is applied on the CC/CV charger (***Figure 5***).

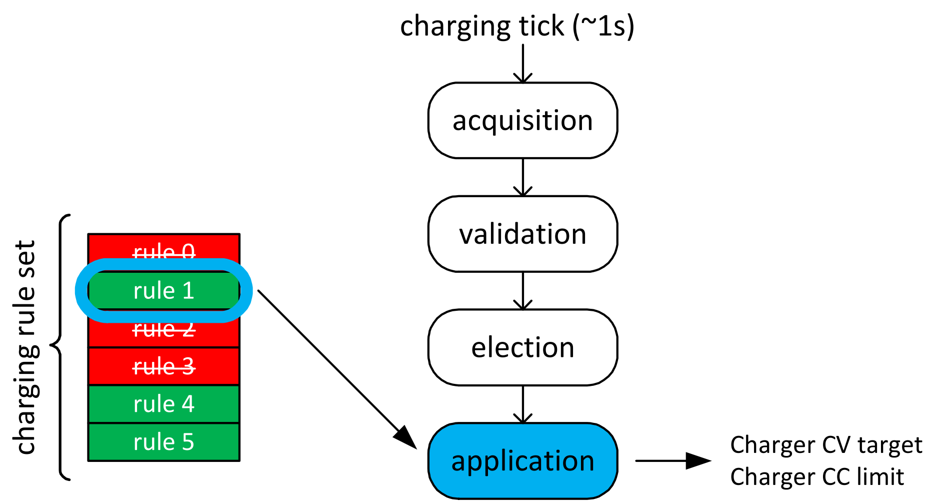

Figure 5 Application Step

The Charging Rules

Battery Specific Charging Rule

A battery related charging rule represents one charging segment of a battery charging profile (see ***Figure 1***).

A battery related charging rule should contain at least the battery temperature range and the battery voltage range for which it is valid.

The battery charging rule provides the charging voltage and charging current corresponding to the charging segment it models in the specific battery charging profile.

System Specific Charging Rule

In addition to battery specific rules, a charging rule can model system requirement with little or no association to actual battery characteristics. These rules are called system specific charging rules.

A typical example of pure system linked charging segment would be to safely run a firmware update on the portable device. In such use-case, losing power supply could be dramatic and the end-user would be prompted to plug in the wall adaptor before proceeding. Even with the wall adaptor plugged in, power supplied to the device is not guaranteed as the battery temperature might be outside the valid temperature range, limiting charging current. To secure this use-case, the device may impose a specific charging segment, not depending on temperature and only applicable if firmware update is on-going. This charging segment would ensure that charging current is at least equal to the current consumed by the device during firmware update.

The system related charging segment, like the firmware update example above, could be easily translated to a generic charging rule. In this situation, the rule must be enabled or disabled as per a particular system state such as a “firmware update running” Boolean state. Since the rule based algorithm is already capable of disabling rules at the validation step, it can also be used to manage the system state.

In addition to typical measures used to validate a rule, the rule may include two Boolean checks applied on system state, one allowing the rule to be validated (called “condition true”) and the other one forcing the rule to be invalid (called “condition false”). This enables the combination of battery specific and system specific rules in a generic way.

A typical implementation of such system state management would consist of arranging all the system states as bit fields and let each charging rule connecting its “condition true” or “condition false” metric to the designated bit in the field.

This could be represented as ***Figure 6***.

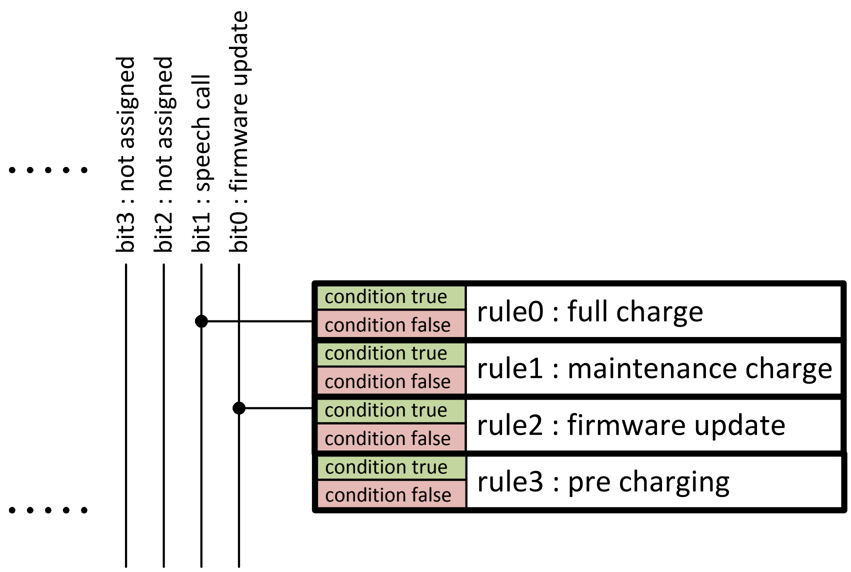

Figure 6 Condition True & False Bit Mapping

In ***Figure 6***, we can see that rule0 (the full charge rule) is forced to be invalid since the system is in "speech call". If "speech call" comes true, only rule1 to rule3 can be valid.

The rule2 (the firmware update rule) can be valid only if "firmware update" state is valid. That doesn't mean that this rule2 will be promoted since "firmware update" is true; it will be the case only if rule0 and rule1 are both invalid. On the other hand, if "firmware update" is false, the rule2 is invalid.

The coding of "condition true" or "condition false" can be a bit mask representing the connection to the system state bit field. For rule0, "condition true" = b'0000 & "condition false" = b'0010. For rule2, "condition true" = b'0001 & "condition false" = b'0000.

Generic Charging Rule Definition

A generic format of charging rule processed by the algorithm consists of the battery defined charging segments and charger host system related segments. As explained in ***"Battery Specific Charging Rule"***, for rule validation purposes, a battery related charging rule should contain at least battery temperature limits (T_{min} & T_{max} in ***Table 1***) and battery voltage limits (V_{min} & V_{max} in ***Table 1***).

The system related rules include the earlier described bit mask to code the "condition true" and "condition false" check-points for the system states, see ***"System Specific Charging Rule"***.

For system stability and additional features, the rule processed by algorithm usually includes a voltage hysteresis parameter (V_{hyst} in ***Table 1***). This hysteresis should be used during validation of the rule list when the battery voltage is compared to V_{max}. If the rule which is checked currently was the one applied at previous charger tick processing, the battery voltage is directly compared to V_{max} of the rule. If the rule was not applied at previous charger tick processing, the battery voltage is compared to (V_{max} - V_{hyst}).

To highlight the use of V_{hyst}, consider a battery voltage of 4.1 V at the time the wall adaptor is connected to the device. Also consider a single rule in the charging algorithm, called rule0, having a V_{max} at 4.2 V and a V_{hyst} at 0.2 V. At the time the wall adaptor is plugged, no rule was applied by the algorithm (charger cannot charge without a power supply). Because rule0 was not the previously applied rule, it will be compared with battery voltage including V_{hyst} in calculation. In this case, the comparison will be between 4.1 V and (4.2 V - 0.2 V) = 4.0 V. As battery voltage is higher than 4.0 V, the rule0 becomes invalid and no charging will happen. Later, because of device current consumption, the battery voltage will drop below 4.0 V. After this, rule0 will no more be invalid and it will be applied. Because rule0 is applied, the next validation of rules will not consider hysteresis for this rule0. The comparison will be done between the battery voltage and the V_{max} = 4.2 V of the rule without hysteresis. So, the rule0 will stay in force up to 4.2 V.

When the battery is considered to be fully charged, the charger should stop providing current (charging termination). In the classical CC/CV charging method, the termination happens when the battery voltage corresponds to the CV voltage and charging current drops below a given threshold. This termination method can be easily managed in a charging rule by including a minimum current (I_{min} in ***Table 1***). Since charging current drops below I_{min} of the applied rule, the rule is invalidated.

At the time that a rule is validated, elected and applied, the algorithm must furnish to the CC/CV charging subsystem the charging current (CC limitation) and the charging voltage (CV target). By this means, the battery voltage would never rise above the CV target. Hence, the CV target is the max voltage allowed from the battery and this max voltage is already coded in the charging rule (V_{max}). On the other hand, the CC current limitation does not exist yet in the rule. As the CV target, the CC limit is a max current which can flow to the battery during charge. So this max current should be coded in the rule (I_{max} in ***Table 1***) and would be applied to the charging subsystem when the rule applies.

Finally, it is highly recommended that the charging algorithm checks the time while actively charging the battery. As a safety feature, the charging algorithm needs a defined time-out period. If a charging phase is not completed within the time-out period, the battery is most probably damaged. To avoid any end-user risks or more damage to the battery, the charge operation should be stopped after this time-out period is reached.

An overall time-out can be processed, engaged at the plug in of wall adaptor and stopped when the charging termination is detected. As well, it is beneficial to include a time-out at the individual rule level which starts at the time a new rule is applied by the algorithm. In this case, each rule would contain its own time-out limit considering its use-case and/or its charging current.

All the parameters discussed above are needed for robust charging algorithm processing. However, the entire parameter set is not normally stored in the battery as it only is required to provide its charging segments profile to the system. For example, the "condition true" and "condition false" parameters are typically useful for system use-cases management but not to fulfill pure battery specifications.

On the other hand, some parameters are exclusively dependent on battery characteristics and must be fetched from it if available. For reference, the battery pack charging data object defined by MIPI BIF includes T_{min}, T_{max}, V_{min}, V_{max} and I_{max} parameters in the standardized data format.

Table 1 shows a summary of rule parameters and indicates which of the parameters are mandatory to be stored in a battery pack. The table explains also how optional parameters can be generated by the charging system when not present in the battery pack.

Table 1 Charging Rule Parameters

Parameter	Description	Mandatory in battery	How to generate parameter if not stored in battery pack
T_{min}	Minimum battery temperature. If temperature is lower than T_{min}, rule is invalid.	Yes	—
T_{max}	Maximum battery temperature. If temperature is above T_{max}, rule is invalid.	Yes	—
V_{min}	Minimum battery voltage. If voltage is lower than V_{min}, rule is invalid.	Yes	—

Table 1 Charging Rule Parameters (continued)

Parameter	Description	Mandatory in battery	How to generate parameter if not stored in battery pack
V_{max}	Maximum battery voltage. If voltage is higher than V_{max} (after hysteresis correction) rule is invalid. If rule applies, charger CV target is set to V_{max}.	Yes	—
V_{hyst}	Voltage hysteresis. If rule doesn't apply, maximum battery voltage is checked against (V_{max}-V_{hyst}). If rule applies, maximum battery voltage is checked with V_{max} directly.	No	When hysteresis voltage is configured as a direct factor of charging CC current, the algorithm behavior is stable. Hence, V_{hyst} can be established by charging device as K * I_{max}. K depends directly on the charging circuit impedance of the device (K between 0.1 and 0.2 typically). After calculation, V_{hyst} might be clamped to a given defined value (0.2 V max for example). Note that (V_{max}-V_{hyst}) must never be lower than V_{min} else the rule can never be validated.
I_{min}	Termination current. If rule applies and charging current drops below I_{min}, rule is invalidated.	No	As a first approach, termination current can be extrapolated from the charging CC current I_{max}. I_{min} could be 1/10th to 1/20th of I_{max}. The I_{min} could as well be bounded to a minimum value by the device such as 30 mA minimum.
I_{max}	CC limit. If rule applies, charger CC limit is set to I_{max}.	Yes	—
"condition true"	If system state bit field doesn't fit the "condition true" bit mask, rule is invalidated.	No	Purely system dependent
"condition false"	If system state bit field fits the "condition false" bit mask, rule is invalidated.	No	Purely system dependent
time-out	If the rule is applied for a longer time than time-out, rule is invalidated and charger algorithm goes in error condition.	No	Even if time-out information was stored in the battery, its effective value depends on the capacity of the wall adaptor to deliver the power to achieve I_{max} during CC phase. It may take 10 times more time to terminate a rule if a weak wall adaptor can only support 100 mA CC phase compared to a strong adaptor allowing a CC phase at 1 A. Because of this variability, it is preferable for the charging system to establish the time-out value for each rule. Easily determine the time-out value based on I_{max} parameter, the battery capacity and the (V_{max}-V_{min}) voltage step.

The List of Charging Rules

Rule List Ordering

As stated in previous sections, the rule base charging algorithm will operate on a set of charging rules which represents both the battery charging profile and the system specific charging segments. It is important to keep in mind that after validation, the election of rule will promote the first rule valid in the rule list order. Therefore it is critical to order the rules carefully to achieve correct functioning.

As a standard approach, the rules associated with battery specification (normally fetched from battery pack data container) should be ordered by decreasing magnitude of charging current. Hence the first rule would have the highest charging current. This ordering enables maximizing the charging performance because if more than one rule is valid at a time, the elected one will have the highest charging current so resulting in the fastest charging time.

If some different battery related rules have the same charging current, they are placed in the list in decreasing order of their maximum charging voltage value. This sorting would maximize the quantity of energy transmitted to the battery.

If some battery related rules have the same charging current and the same charging voltage, there is no particular order to respect. In fact, with a good design, these rules should be orthogonal in term of validity so that only one is valid at a time.

For reference, the battery pack charging data object defined by MIPI BIF respects the above mentioned rule order.

Rule to Rule Interaction

As a first approach, each rule in the rule list is independent from the others. However, the rule list design must ensure the continuity of charging profile. Typically, for a given battery temperature, there should be always a rule applicable up to the required end-of-charge voltage and current.

For example, if only two rules are applicable for a given battery temperature, the first rule having V_{min} = 3.0 V & V_{max} = 3.5 V and second rule having V_{min} = 3.7 V & V_{max} = 4.2 V, there will be a discontinuity of rules during the charging process. In fact, since battery voltage is between 3.5 V and 3.7 V, the first rule will not apply (V_{max} is 3.5 V) but the second will not apply as well (V_{min} is 3.7 V). As a consequence, there will be no charging operation for battery voltage between 3.5 V and 3.7 V.

Special attention is then required when designing the rule list to ensure continuity of charging profile over battery temperature and voltage. Temperature continuity can be checked over the charging profile by verifying that for every relevant battery voltage, the combined temperature ranges of all applicable rules is continuous. The example shown in ***Figure 7*** reveals that temperature continuity is failing for battery voltage at 3.7 V. A simple correction could be to make Rule1 T_{max} parameter equal to Rule2 T_{min} parameter.

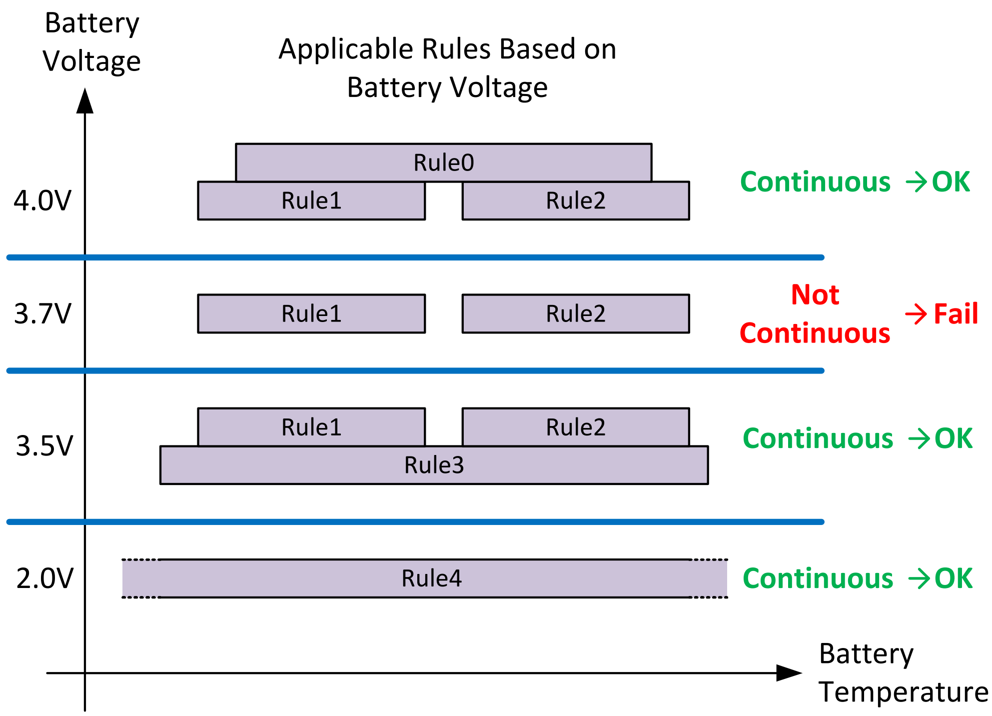

Figure 7 Example of Temperature Continuity Verification

A similar check for voltage continuity can be performed by drawing the applicable rules following battery voltage for every relevant battery temperature (x-axis being the battery voltage and y-axis being the battery temperature). This verification must take into account that V_{max} is checked including hysteresis voltage V_{hyst} since a rule is not applicable. The example, ***Figure 8*** reveals that voltage continuity is failing for battery temperature at 62°C. In fact, with counting hysteresis, Rule2 V_{min} is higher than Rule4 V_{max}-V_{hyst}. A simple correction could be to make Rule2 V_{min} parameter equal to Rule4 V_{max}-V_{hyst}.

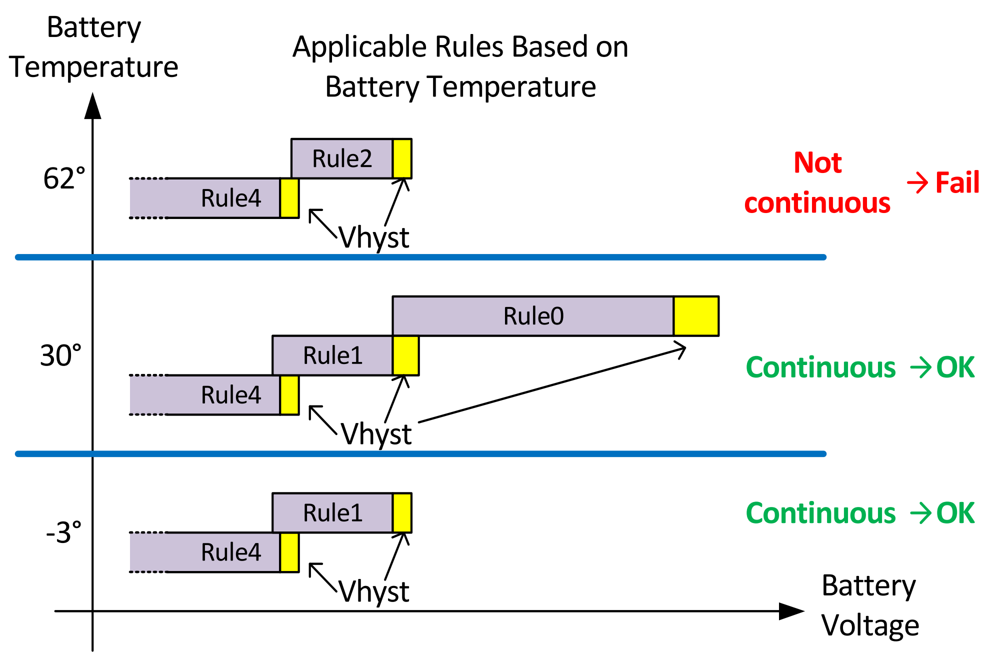

Figure 8 Example of Voltage Continuity Verification

Because of the voltage continuity requirement and the effect of hysteresis V_{hyst}, the rules in rule list overlap their V_{min} and V_{max}. As V_{hyst} can be generated by the charging system and is not necessarily defined by the battery itself, the rule list voltage overlapping should be large enough to ensure voltage continuity on all systems.

Note that rule based algorithm implementation may include a temperature and voltage continuity check. The algorithm may not use a non-continuous rule list. The algorithm can also check if its V_{hyst} generation is not creating voltage discontinuity. It may adjust the V_{hyst} for problematic cases if any.

Building a Computable Rule List

The full rule list used by the charging algorithm combines the battery specific charging rules and the system particular ones. The construction of a computable rule list would consist in three steps: read rules from battery data container, calculate any missing parameters from the battery rules and finally insert the system rules in the rule list.

Reading the battery rules allows building an initial list as shown in ***Figure 9***. The rules of this list are incomplete because some parameters are not stored in the battery pack container (V_{hyst}, time-out, etc.).

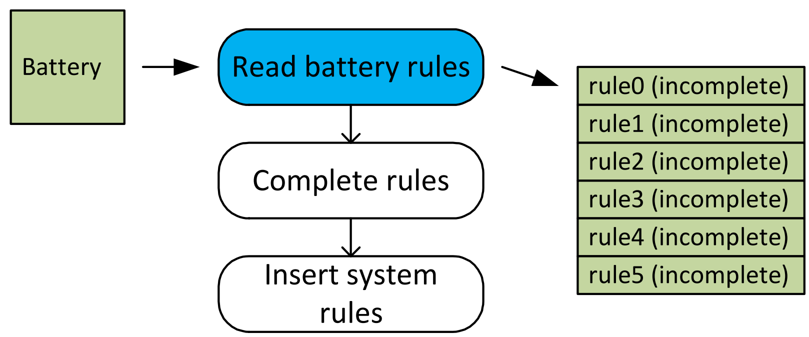

Figure 9 Battery Rules Reading Step

The charging device will then complete the initial rule list by filling the missing topics (***Figure 10***).

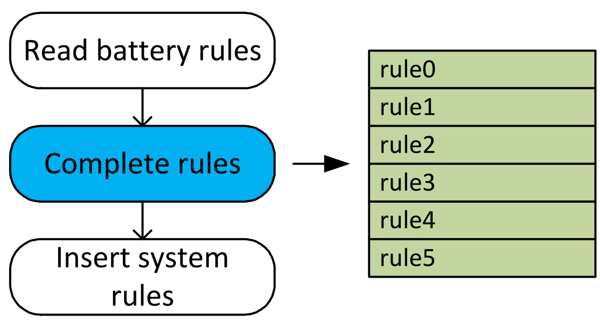

Figure 10 Battery Rules Completion Step

As explained in previous sections, the missing V_{hyst} topic can be computed by applying $V_{hyst} = K0*I_{max}$ (K0 between 0.1 and 0.2 typically). V_{hyst} might be clamped to a max value (0.2V for example) and (V_{max}-V_{hyst}) is verified to be higher than V_{min}. As a first approach, the missing I_{min} can be computed as $I_{min} = I_{max}$ / K1 (K1 between 10 and 20 typically) while I_{min} can't be lower than 30 mA typically. The missing rule time-out can be computed considering the (V_{max}-V_{min}) value, the battery capacity and the I_{max} current. For now, the missing "condition true" and "condition false" bit mask can be assigned to 0.

While completing the battery rule list, the system should locate any full charge rules at the first positions in the list. Full charge rules are defined as having the overall maximum V_{max} value, independent of I_{max} setting. In fact, battery voltage is directly linked to the quantity of energy stored in it. So charging a battery to its full capacity means raising its voltage to the maximum permitted value. Only the rules having the overall V_{max} value are capable of storing a full charge in the battery.

System specific rules should be located following the full charge rules or by adding "condition true" or "condition false" tags to the full charge rules to meet system requirements. Any requirements for fine adjustment of I_{min} can be done in a similar manner.

The battery rule list is primarily sorted by decreasing I_{max}. If the full charge rules (having the highest V_{max}) are not located in the first positions of the rule list, it will permit the battery to experience some very high charging current under some temperature and voltage range conditions. These high charging currents should not be allowed to reach full charge because the V_{max} is not the highest of the list to enforce battery limits. If such rule list is fetched from the battery, it would be advantageous to use a particular computation of I_{min} for rules placed before the first full charge rule. This I_{min} would be set slightly lower than the I_{max} of the full charge rule (typically 95%). This configuration of I_{min} would allow forcing the very high current charging rule to be invalid since the charging current passes below 0.95 times the full charge rule I_{max}. In such cases, there would be a smooth transition between the high current rule and the full charge rule.

When the battery specific rules are completed, the system will insert its own charging rules (for example, firmware update case, speech call case, charge maintenance rule, etc.). The list is going to grow (***Figure 11***).

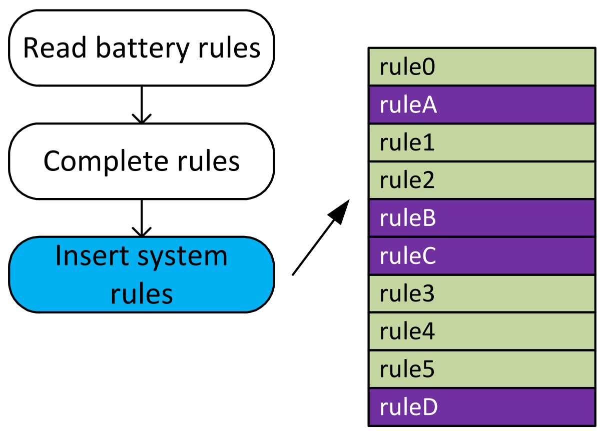

Figure 11 System Rules Insertion Step

As election of rules will be done by rule list ordering, it is important to insert the system rules at key positions. In most cases, the system charging segments are intend to ensure a minimal or maximal battery voltage. The charging current is not the critical sorting topic. As a standard procedure, the system rules should be inserted in the rule list as per decreasing V_{max} topic. For example, in the diagram above, ruleA should have a lower or equal V_{max} than rule0, ruleB and ruleC should have a lower or equal V_{max} than rule2, and so forth.

Examples

Examples of Typical System Rules

There are two main cases of system rules: the rules depending only on system specification and the rules generated from an existing rule fetched from the battery.

A system specific rule generally exists to ensure a minimal operating battery voltage in some particular situation. The firmware update use-case described earlier in this paper is typically a system specific rule. It probably would be created with a V_{min} at 0 V and a V_{max} slightly higher than the minimal battery voltage needed to run safely a firmware update (typically 3.4 V or 3.5 V). The I_{max} current would be just above the worst current consumed by the system during firmware update. This rule would have a "condition true" mask selecting "firmware update" bit of system state bit-field.

Generating a system rule derived from an existing battery rule would be a useful way to modulate battery charging profile as per system situations. This is typically done to decrease the charging rate when the current drawn on battery is known to be erratic (for example, 2G speech calls of mobile phone). This would also be common method to code a maintenance charging cycle.

An erratic current consumption on a battery could create some battery voltage overshoots during charge. In fact, when a high current is drawn by the system, part of it will come from charging path. If the system load current disappears suddenly, the charger subsystem may not have fast enough reaction so that the current it supplies would remain high for some time. This excess current provided to the battery would increase the battery voltage. This effect could be particularly problematic when battery voltage is at the CV target of charger (for example, 4.2 V on Li-ion battery). In fact, the overshoot resulting in this condition may stress and damage the battery.

A simple solution to avoid stress of battery linked to voltage overshoot would consist of reducing the CV target of the charger since the device load current is known to be erratic. For example, in 2G speech calls for a mobile phone. This case can be easily supported by creating specific rules based on the full charging rules fetched from battery.

As stated in previous sections, after the read of rules from battery and during the rule completion step, it is easy to locate the full charge rules as they will all have the overall maximum V_{max} voltage. Knowing these rules, the system would be able to assign their "condition false" mask to the "speech call" bit of system state bit-field ("speech call" or any other current erratic situation). By configuring this mask, the original full charge rules would be forced invalid since a speech call is running.

Right after these original full charge rules in the rule list, the system can create and insert a basic copy of them after two modifications: the V_{max} would be shifted down to ensure that overshoot would not stress the battery (for example, by 0.05 V~0.1 V) and "condition true" mask would be assigned to "speech call" bit.

The resulting rule list would then contain the original full charge rules fetched from battery but these rules can be valid only out of speech call use-case. The list would also contain similar full charge rules but having a CV target 50 mV or 100 mV lower. These rules can be valid only during speech call use-case. Hence, during speech call, the charger subsystem would prevent overshoot damages to the battery.

Another classical system created rule derived from battery rules is used to address the charge maintenance. When a battery is fully charged, the charging current would be stopped. As no charging current would flow (even if charger wall adaptor is still connected), the battery is going to be discharged based on the portable device current consumption. When the battery voltage drops below a predefined limit, the charge process should re-engage. The period between each charge activities could be small depending on device current consumption and battery capacity. If each charging activity was going up to the full charge point with maximum allowed charging current, the battery would age quickly because of too many strong charging cycles. Hence, after a full charge at full rate of the battery, it would be good practice to do subsequent charging cycles at a lower rate and at a lower CV target (50 mV lower than full charge point for example).

A way to implement this behavior would consist of creating and inserting a new rule derived from the full charge rule fetched from battery data container. This full charge rule is easily localizable as it has the maximum V_{max} value in the rule list. The created charge maintenance rule would be a copy of the full charge rule with some modulations: the V_{max} voltage will be shifted down to avoid the big cycling effect (by 50 mV to 100 mV typically) and the I_{max} will be lowered (divide by 2 typically with a bottom clamp corresponding to the typical device current consumption). The newly created rule would then ensure a lower CV charging target with a lower charging current.

Now, the full charge rule and the maintenance rule need to be adjusted so that full charge rule is promoted first up to termination but subsequent charging cycles would run on maintenance rule. This behavior is simple to implement by correct adjustment of V_{hyst} hysteresis voltage. The system has simply to ensure that V_{max}-V_{hyst} of maintenance rule is higher than V_{max}-V_{hyst} of full charge rule. As an example, let's assume V_{max} of full charge rule at 4.2 V with 0.2 V of V_{hyst}. The maintenance rule is created with a V_{max} at 4.15 V (50 mV shift down compare to full charge). A V_{hyst} of 0.1 V for this maintenance rule is enough to ensure the maintenance cycles.

In fact, after a full charge termination, as no rule applies, both full charge rule and maintenance rule will be affected by hysteresis voltage. Hence, full charge rule could become valid only if battery voltage goes below V_{max}-V_{hyst}, so 4.2 V - 0.2 V = 4.0 V. The maintenance rule would come valid only if battery voltage comes lower than 4.15 V - 0.1 V = 4.05 V. As battery voltage decreases, the first rule to be valid will be the maintenance rule so this one will be systematically promoted after a full charge phase.

Let's consider now that the battery was not fully charged at the time the wall adaptor is connected. Its voltage is 3.9 V. In this case, both full charge and maintenance rules are valid (3.9 V is below 4.0 V and below 4.05 V). As the maintenance rule is placed after the full charge rule in the rule list (its V_{max} is lower), at rule election step of algorithm, the full charge rule will be selected. Hence, a not fully charged battery will be initially charged using the full charge rule and not the maintenance one.

The natural adaptability of rule based algorithm is demonstrated by the following charge maintenance situation. Assume that a full charge of the battery was done and the charging algorithm performs a regular maintenance cycle based on the maintenance rule described above. This maintenance charge would be executed with a small charging current. Now, consider when the battery voltage is 4.1 V and the portable device starts consuming a larger current than the value programmed for charge maintenance. In this case, the battery would stop being charged and start discharging instead. Battery voltage will decrease and at some points it will pass below 4.0 V. Since the battery voltage passes below 4.0 V, the full charge rule is going to be valid while the maintenance rule remains valid. As the full charge rule is placed before the maintenance rule in the rule list, it will be elected and applied. Moving to full charge rule would allow coming back to a high charging current instead of a low maintenance one. Hence, a portable device with higher current consumption would be naturally compensated and the battery will stay highly charged.

Examples of Rule List and Simulation Results

Example 1

A battery specific rule list would be built as per comprehensive specification of battery. Let's consider the following specification example:

A safe charging situation is defined for a battery by requiring:

- A progressive charging current depending on battery voltage
- A limitation of charging current and charging voltage depending on temperature

For this battery, following charging rates have been defined:

- Wake-up rate: 100 mA
- Low rate: 300 mA
- Medium rate: 600 mA
- Full rate: 1000 mA

The safe usage of rates is defined as per comprehensive specification points:

1. The wake-up rate is basically used when battery is fully discharged. It can be engaged whatever is temperature but only if battery voltage is < 3.3 V.
2. In normal temperature range (from 0°C to 60°C):
   A. Charging voltage is limited to 4.2 V.
   B. If battery voltage is > 3.8 V, the full rate can be selected.
   C. If battery voltage is > 3.4 V, the medium rate can be selected.
   D. If battery voltage is > 3.1 V, the low rate can be selected.
3. In extended temperature range (from -5°C to 0°C and from 60°C to 65°C):
   A. The charge cannot be higher than medium rate.
   B. The charge should happen only if battery voltage is ≤ 3.9 V
4. In extreme temperature range (from -15°C to -5°C and from 65°C to 75°C):
   A. The charge cannot be higher than low rate.
   B. The charge should happen only if battery voltage is ≤ 3.7 V

This specification can be easily translated as a rule list processed by the rule based charging algorithm. It would come simply to four rules, each rule coding a given charging rate. The rules are ordered by decreasing rate.

Rule0 (full rate)

- T_{min} = 0°C, T_{max} = 60°C (battery spec 2).
- V_{min} = 3.8 V (battery spec 2.b)
- V_{max} = 4.2 V (battery spec 2.a)
- I_{max} = 1000 mA

Rule1 (medium rate)

- T_{min} = -5°C, T_{max} = 65°C (battery spec 2 & 3, medium rate is valid in normal temperature range and extended temperature range).
- V_{min} = 3.4 V (battery spec 2.c)
- V_{max} = 3.9 V (battery spec 3.b)
- I_{max} = 600 mA

Rule2 (low rate)

- T_{min} = -15°C, T_{max} = 75°C (battery spec 2 & 4, low rate is valid in normal temperature range and extreme temperature range)
- V_{min} = 3.1 V (battery spec 2.d)

- V_{max} = 3.7 V (battery spec 4.b)
- I_{max} = 300 mA

Rule3 (wake-up rate)

- T_{min} = -infinite, T_{max} = +infinite (battery spec 1, all temperatures are valid)
- V_{min} = 0 V (battery spec 1)
- V_{max} = 3.3 V (battery spec 1)
- I_{max} = 100 mA

After completion of battery fetched rules, the list processed by the algorithm is updated into the following list (green are completed topics).

Rule0 (full rate)

- T_{min} = 0°C, T_{max} = 60°C (battery spec 2).
- V_{min} = 3.8 V (battery spec 2.b), V_{max} = 4.2 V (battery spec 2.a)
- I_{max} = 1000 mA
- I_{min} = 50 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.2 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

Rule1 (medium rate)

- T_{min} = -5°C, T_{max} = 65°C (battery spec 2 & 3).
- V_{min} = 3.4 V (battery spec 2.c), V_{max} = 3.9 V (battery spec 3.b)
- I_{max} = 600 mA
- I_{min} = 30 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.12 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

Rule2 (low rate)

- T_{min} = -15°C, T_{max} = 75°C (battery spec 2 & 4).
- V_{min} = 3.1 V (battery spec 2.d), V_{max} = 3.7 V (battery spec 4.b)
- I_{max} = 300 mA
- I_{min} = 30 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.06 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

Rule3 (wake-up rate)

- T_{min} = -infinite, T_{max} = +infinite (battery spec 1, all temperatures are valid)
- V_{min} = 0 V (battery spec 1), V_{max} = 3.3 V (battery spec 1)
- I_{max} = 100 mA
- I_{min} = 30 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.02 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

With such rule list, plugging a wall adaptor while battery voltage is 2.9 V and battery temperature is 30°C would give the ***Figure 12*** charging sequence.

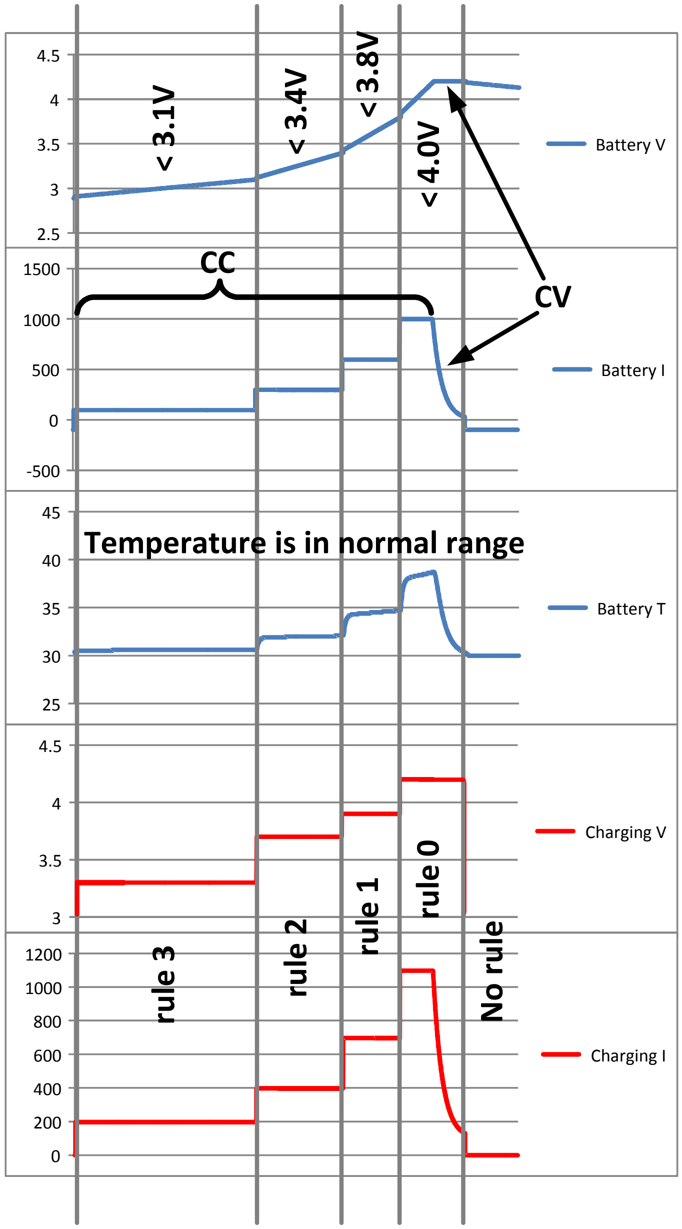

Figure 12 Charging in Normal Temperature Range

Notice the progression of rule application from rule3 to rule0. While battery voltage is rising up, more and more rules become valid. As per rule election procedure, the first rule valid in the rule list is selected and applied. This results as the expected progressive charging, from wake-up rate to full rate.

The rule list established as per battery specification would also fulfill the extended temperature range requirements. ***Figure 13*** shows a rule selection progression considering ambient temperature at -4.9°C.

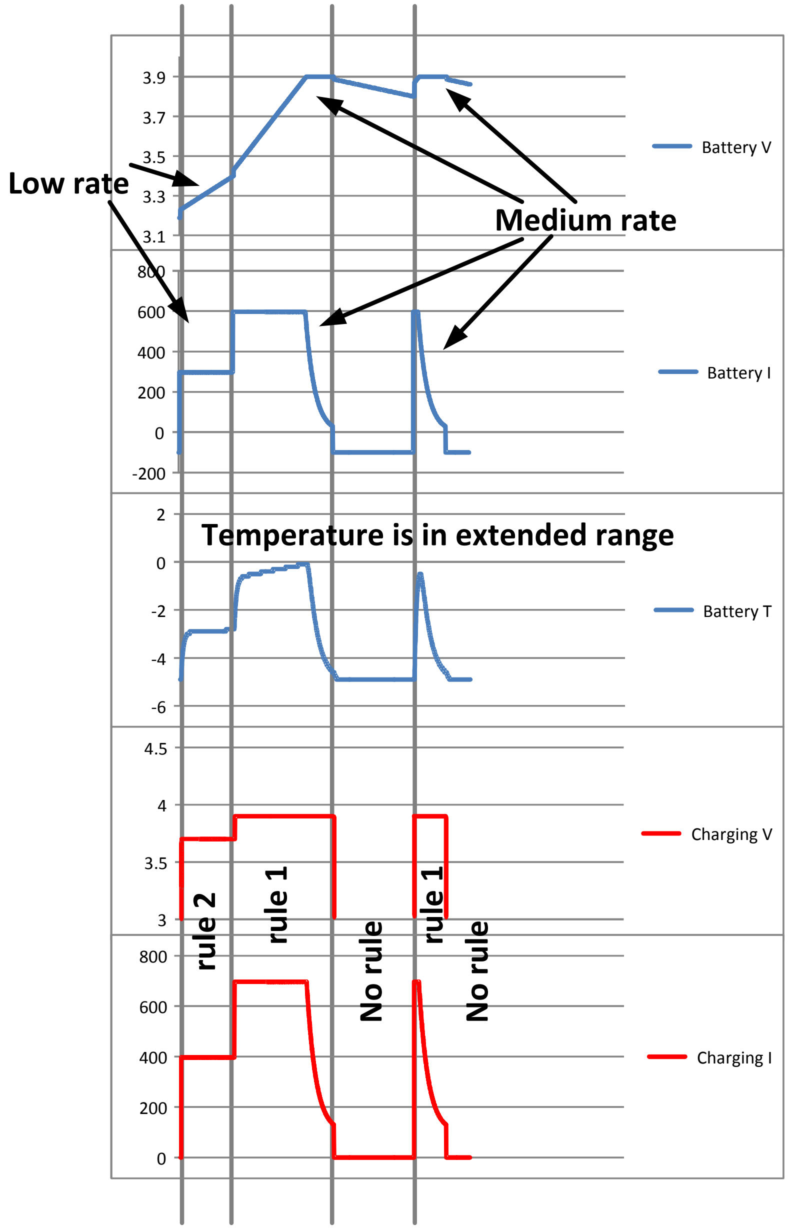

Figure 13 Charging in Extended Temperature Range

As required, full rate charge is never engaged and CV target is lower than 3.9 V. In fact, with such ambient temperature, the rule0 (coding the full rate charge) is never valid. It could be noticed that rule1 has been effectively completed with V_{hyst} at 0.1 V. Hence rule1 re-engages itself at 3.8 V.

A maintenance rule can also be generated by the system and inserted at rule1 position (the existing rule1 to rule3 would then be moved deeper in the list at rule2 to rule4 slots respectively). The rule0 full rate rule was completed with a V_{hyst} at 0.2 V. The created and inserted maintenance rule rule1 was essentially copying rule0 topics but with V_{max} shifted down by 50 mV (So V_{max} = 4.15 V) and I_{max} divided by 2 (reducing I_{max} = 500 mA). This maintenance rule V_{hyst} has to be set to 0.12 V. In such case, when all rules are invalid, rule0 can be validated again only if battery voltage drops below 4.0 V while maintenance rule would come valid if voltage drops below 4.03 V.

The full charge rule would be then followed by multiple charge maintenance cycles as ***Figure 14***.

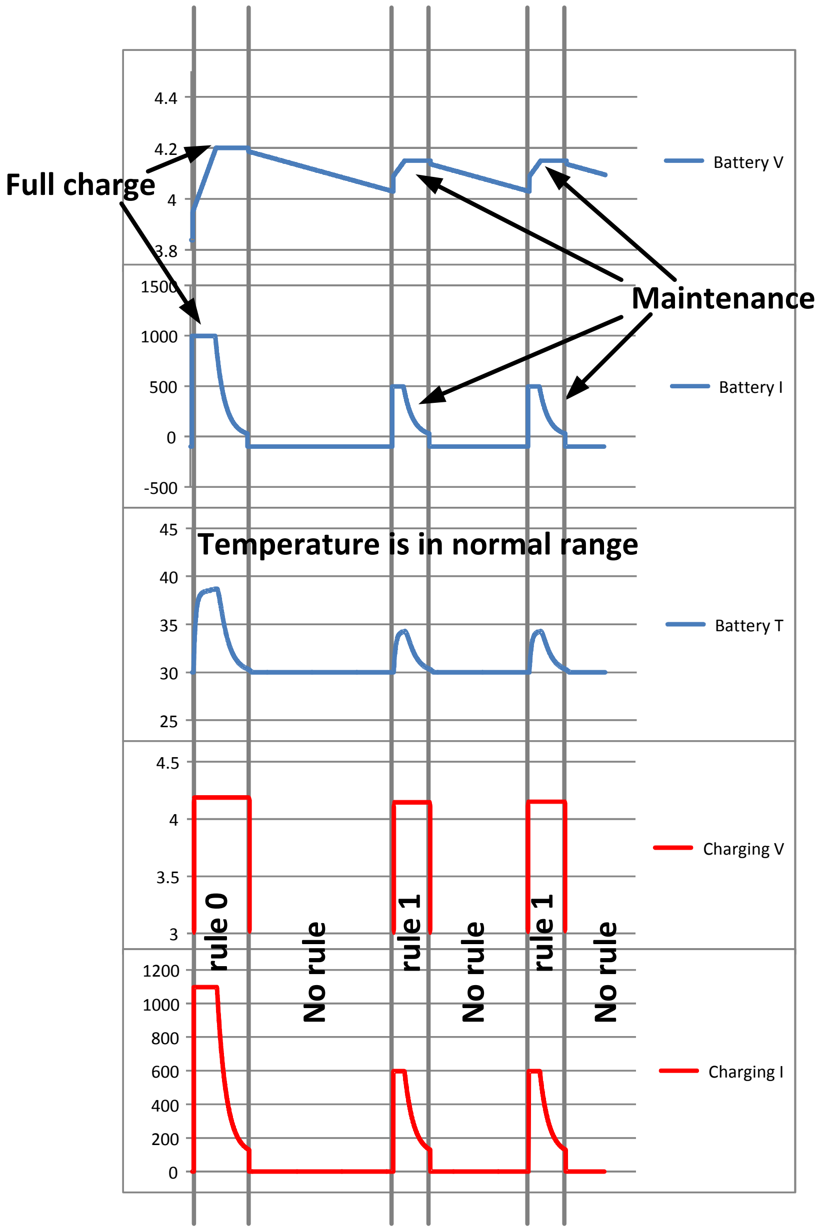

Figure 14 Maintenance Charging Rule

Example 2

Now consider another battery example connected to the charging system. This battery is specified as shown below. The major characteristic of this battery is its capability to sustain an extreme charging current in a reduced range of voltage and temperature (typically the case of charging above 1C).

For this battery, the following charging rates have been defined:

- Wake-up rate: 300 mA
- Medium rate: 500 mA
- Full rate: 800 mA
- Extreme rate: 1200 mA

The safe usage of rates is defined as per comprehensive specification points:

1. The wake-up rate is basically used when battery is fully discharged. It can be engaged whatever is temperature but only if battery voltage is < 3.3 V.
2. In reduced temperature range (from 10°C to 40°C), if battery voltage is between 3.6 V and 3.9 V, the extreme rate can be applied.
3. In normal temperature range (from 0°C to 60°C):
   A. Charging voltage is limited to 4.2 V.
   B. If battery voltage is > 3.5 V, the full rate can be selected.
   C. If battery voltage is > 3.2 V, the medium rate can be selected.
4. In extended temperature range (from -10°C to 0°C and from 60°C to 70°C):
   A. The charge cannot be higher than medium rate.
   B. If battery voltage is > 3.2 V, the medium rate can be selected.
   C. The charge should happen only if battery voltage is ≤ 4.0 V.

This specification can be easily translated as a rule list processed by a rule based charging algorithm. It would come simply to four rules, each rule coding a given charging rate. The rules are ordered by decreasing rate.

Rule0 (extreme rate)

- T_{min} = 10°C, T_{max} = 40°C (battery spec 2)
- V_{min} = 3.6 V (battery spec 2)
- V_{max} = 3.9 V (battery spec 2)
- I_{max} = 1200 mA

Rule1 (full rate)

- T_{min} = 0°C, T_{max} = 60°C (battery spec 3).
- V_{min} = 3.5 V (battery spec 3.b)
- V_{max} = 4.2 V (battery spec 3.a)
- I_{max} = 800 mA

Rule2 (medium rate)

- T_{min} = -10°C, T_{max} = 70°C (battery spec 3 & 4)
- V_{min} = 3.2 V (battery spec 3.c & 4.b)
- V_{max} = 4.0 V (battery spec 4.c)
- I_{max} = 500 mA

Rule3 (wake-up rate)

- T_{min} = -infinite, T_{max} = +infinite (battery spec 1, all temperatures are valid)
- V_{min} = 0 V (battery spec 1)
- V_{max} = 3.3 V (battery spec 1)
- I_{max} = 300 mA

After acquisition of rule list from the battery pack, as described in previous chapters, the charging system will have to complete the rules as it misses few topics (V_{hyst}, I_{min}, etc.). The completion step will primarily fill these topics with typical method. While doing the completion, the charging system locates the full charge rule being the one having the maximal V_{max}. This is rule1.

The full charge rule not being the first in the rule list, the completion procedure will adjust I_{min} of preceding rule (rule0 in this example). It will use the method described before in this paper. It will assign 95% of full charge rule I_{max} to the I_{min} of preceding rules. Thus in rule0, I_{min} will be set to 0.95*800 mA = 760 mA.

When rule completion is performed, charging system inserts its own rules in the list. In this example, the charging system algorithm chooses to only insert a charge maintenance rule. It will then create a rule A being a copy of full charge rule (which is rule1 in the list). The charging system will customize the ruleA by shifting down the V_{max} (50 mV for example) and by dividing the I_{max} (by 2 for example). This ruleA will then have V_{max} = 4.15 V and I_{max} = 400 mA. The V_{hyst} of this rule will be set so that (V_{max}-V_{hyst}) of ruleA is higher than (V_{max}-V_{hyst}) of rule1.

After completion of battery fetched rules and insertion of system rules, the list processed by the algorithm is updated into the following list (green are completed topics, and red are system created rules).

Rule0 (extreme rate)

- T_{min} = 10°C, T_{max} = 40°C (battery spec 2)
- V_{min} = 3.6 V (battery spec 2), V_{max} = 3.9 V (battery spec 2)
- I_{max} = 1200 mA
- I_{min} = 760 mA (completion based on 95% of full charge rule I_{max})
- V_{hyst} = 0.2 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

Rule1 (full rate)

- T_{min} = 0°C, T_{max} = 60°C (battery spec 3)
- V_{min} = 3.5 V (battery spec 3.b), V_{max} = 4.2 V (battery spec 3.a)
- I_{max} = 800 mA
- I_{min} = 40 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.16 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

RuleA (charge maintenance generated by charging system from rule 1)

- T_{min} = 0°C, T_{max} = 60°C (rule 1 copy)
- V_{min} = 3.5 V (rule 1 copy), V_{max} = 4.15 V (rule 1 copy shifted down by 50 mV)
- I_{max} = 400 mA (rule 1 copy divided by 2)
- I_{min} = 40 mA (rule 1 copy)
- V_{hyst} = 0.08 V (0.2*I_{max} with check that V_{max}-V_{hyst} of rule A is above V_{max}-V_{hyst} of rule 1)

Rule2 (medium rate)

- T_{min} = -10°C, T_{max} = 70°C (battery spec 3 & 4)
- V_{min} = 3.2 V (battery spec 3.c & 4.b), V_{max} = 4.0 V (battery spec 4.c)
- I_{max} = 500 mA
- I_{min} = 30 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.1 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

Rule3 (wake-up rate)

- T_{min} = -infinite, T_{max} = +infinite (battery spec 1, all temperatures are valid)
- V_{min} = 0 V (battery spec 1), V_{max} = 3.3 V (battery spec 1)
- I_{max} = 300 mA
- I_{min} = 30 mA (completion based on I_{max} / 20 with clamp at 30 mA)
- V_{hyst} = 0.06 V (completion based on 0.2*I_{max} with clamp at 0.2 V)

For this rule list, plugging in a wall adaptor while the battery voltage is 3.0 V and ambient temperature is 20°C would give the charging sequence shown in ***Figure 15***.

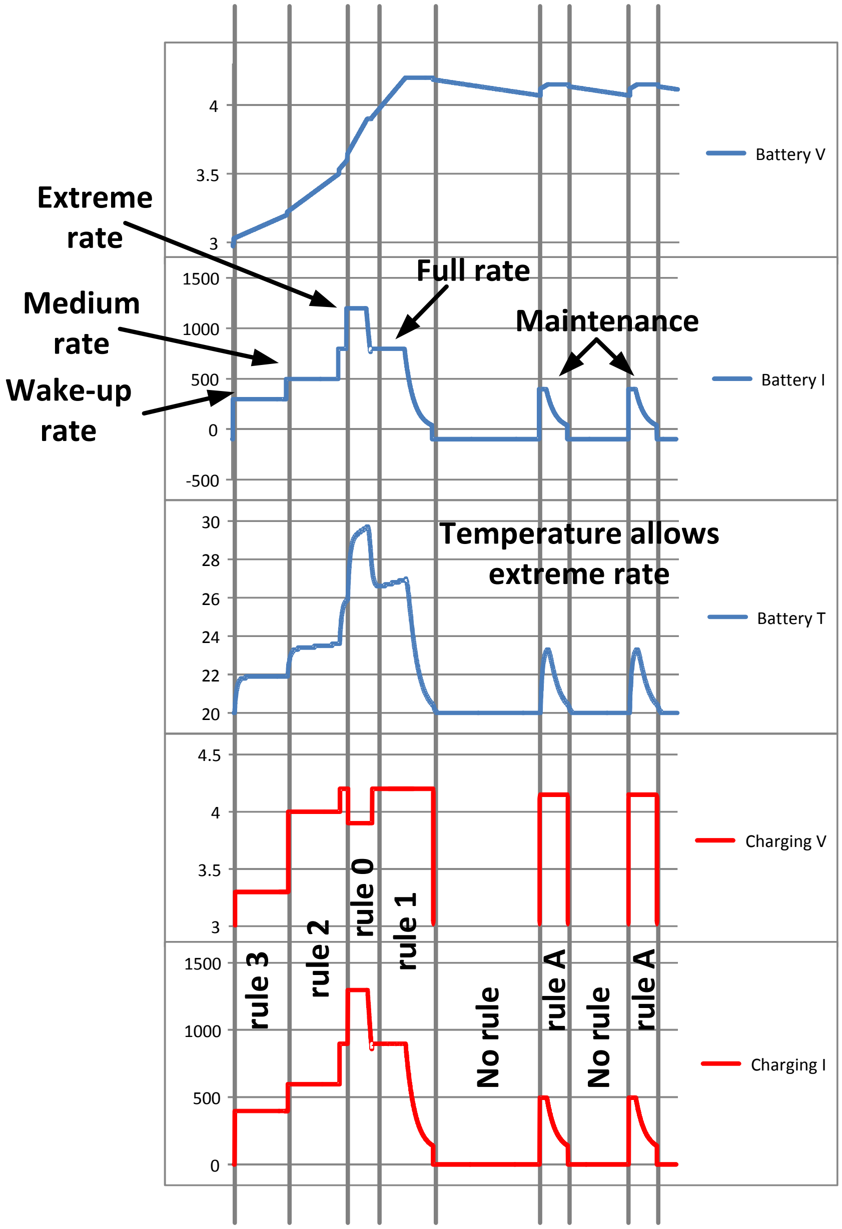

Figure 15 Extreme Rate Selectable

Notice that the extreme rate is selected because the battery voltage is in the correct range. In fact, the battery temperature allows this rate to be applied. Since extreme rate maximum voltage is reached (3.9 V) and battery charging current passes below 760 mA, extreme rate rule comes invalid. At this point, the full rate rule is the first valid rule in the list. It is then elected and applied.

After the full charge of battery, next charger activities are using the charge maintenance ruleA.

The same rule list will give different charging behavior if ambient temperature is 42°C. In fact, extreme rate would not be allowed (***Figure 16***).

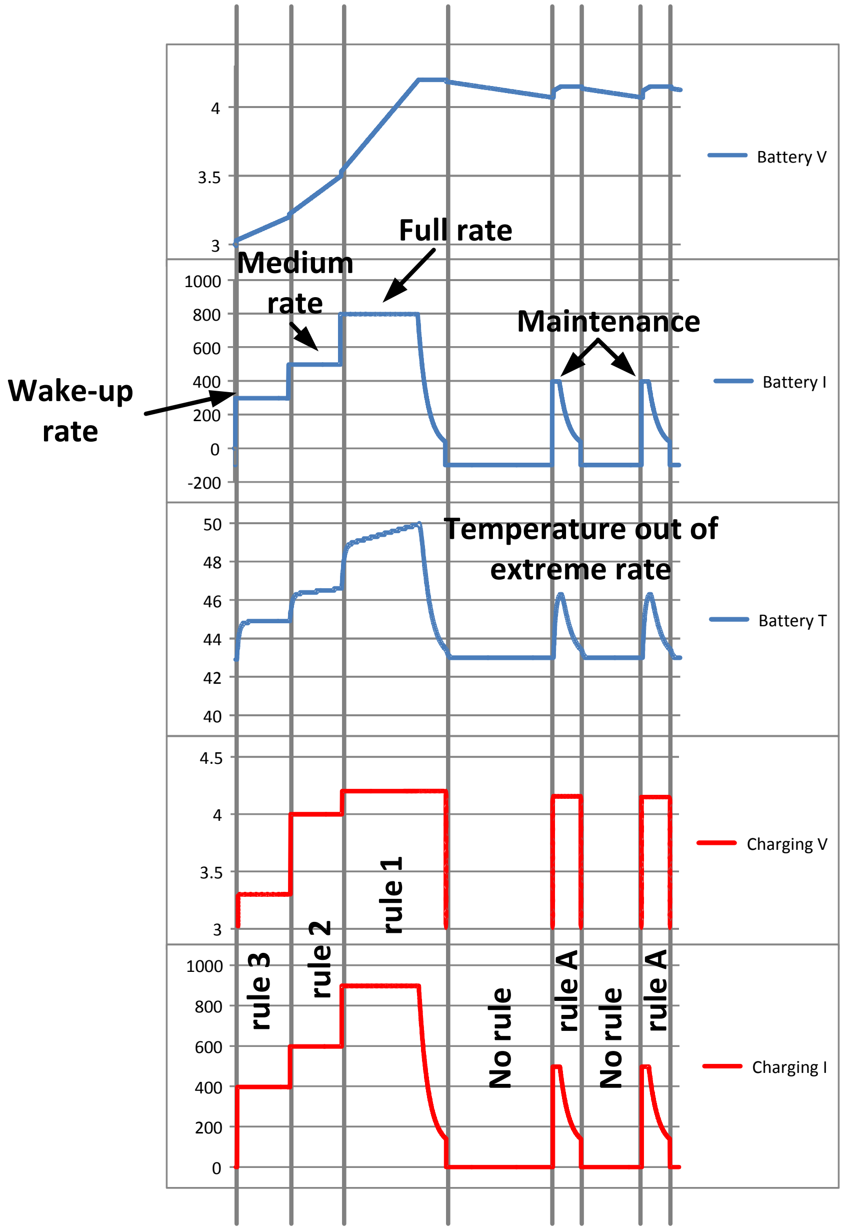

Figure 16 Extreme Rate Not Selectable

Figure 16 confirms the absence of extreme rate during the charging process. The maintenance rule is still regularly applied after the initial full charge cycle.

Obviously, charging in the extended temperature range fulfills the battery specification (***Figure 17***).

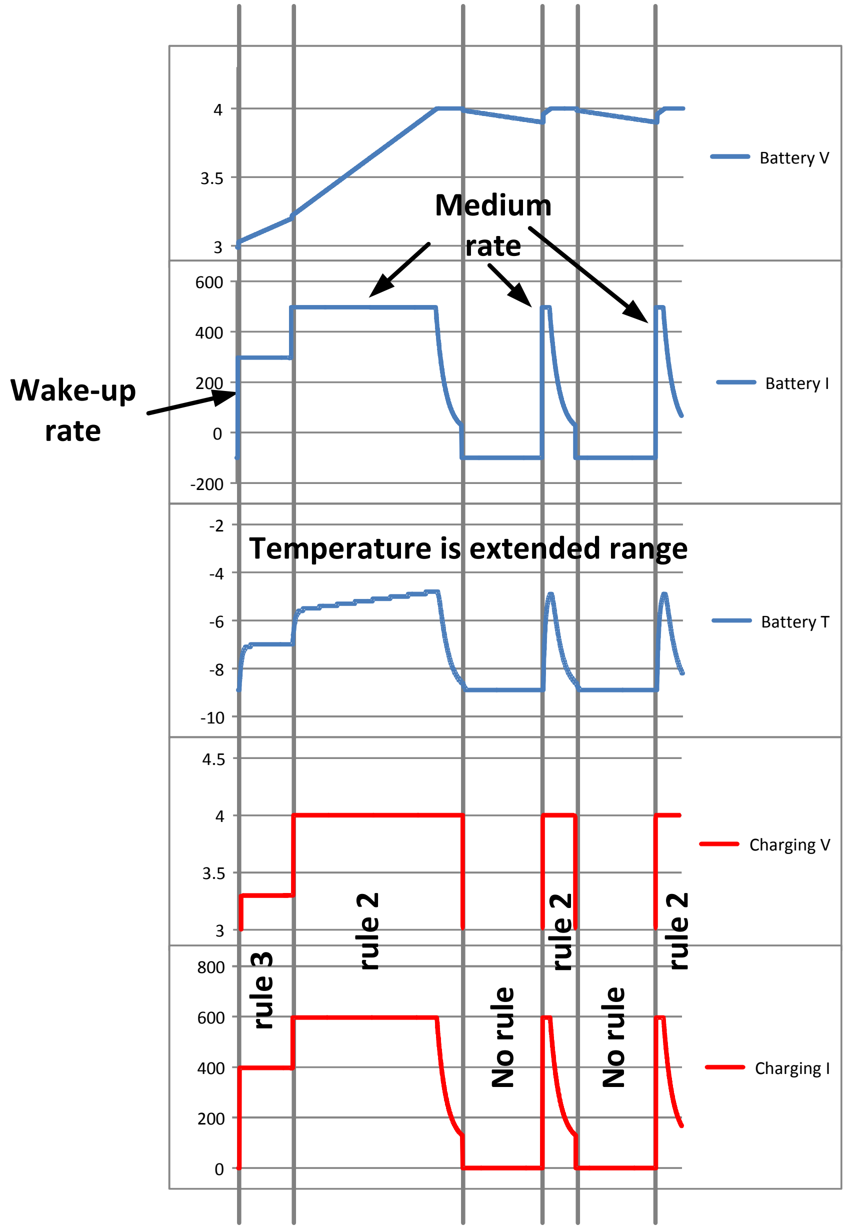

Figure 17 Charging in Extended Temperature Range

As required, only the wake-up or medium rates are used. The charge maintenance rule is not triggered. The re-charging cycles are executed using the medium rate rule.

Conclusion

A rule based charging algorithm offers a systematic method to treat variable charging characteristics of different batteries. These characteristics have simply to be arranged as comprehensive rule list for each battery. The processing machine treating these rule lists is fixed, generic and systematically applicable.

The algorithm is independent of the number of rules to process and inherently models all possible valid transitions from one charging segment to another. This independence of algorithm with regards to charging segment count solves the problem posed by having charging data included in battery pack.

The final rule list is not limited to charging segments in direct link with battery specification. The algorithm can also supports specific charger subsystem programming required by the system itself. The rule based charging algorithm is not only a solution to support battery variable charging characteristics, it is capable of managing the charger subsystem to fulfill the overall system requirements, beyond the battery as well.

Terminology

Definitions

Battery pack: A battery pack is an assembly of individual batteries or cells into one replaceable unit with specified electrical parameters, interconnection connector and physical dimensions.

Charging profile & charging segment: A battery charging profile is a series of charging rules representing the overall charging method for the battery. The profile is a function providing charging current dependant on battery voltage and battery temperature. When this function is represented as a graph with temperature in the x-axis and charging current in the y-axis, it is commonly composed of straight lines (see ***Figure 1***). Each line in the graph is a named charging segment described in this document.

CC charging: Constant Current charging is realized by maintaining the charging current constant independent of the battery voltage.

CV charging: Constant Voltage charging is realized by maintaining the battery voltage constant independent of the charging current.

CC/CV charger: A charger that maintains a Constant Voltage on the battery without providing more charging current than its programmed Constant Current. CC/CV charger can be modeled as a voltage source with current limitation.

Data container: Data container is a metafile format to establish a collection of data items into a shared object.

Full rate charging: Charging at full rate means that charging current can be as high as the maximum allowed current for the battery in the normal range of temperature. In some specific battery cases, higher charging current could be allowed in reduced operating areas (extreme rate for example).

Full charge rule: The battery voltage is a monotonic image of the energy stored in the battery. Hence a fully charged battery will have the highest possible voltage. By extension, a charging rule which allows for reaching the highest possible battery voltage is a full charge rule, independent of the charging current.

Wall adaptor: A physically separate part of a charging subsystem which plugs into a standard utility power receptacle to supply AC power to the mobile device. Typically consists of a power cord and large molded plug containing step-down transformer or other power conditioning circuits with the purpose to adapt the local utility power (country specific) to the supply voltage required by the mobile device charging subsystem.

Acronyms

BIF	Battery Interface MIPI standard
CC	Constant Current
CV	Constant Voltage
2G	2nd generation of digital mobile phones based on GSM standard
GSM	Global System for Mobile Communications is an international standard for digital cellular networks used by mobile phones.
MIPI	Mobile Industry Processor Interface

References

[MIPI01] *MIPI Alliance Specification for Battery Interface (BIF)*, version 1.0, MIPI Alliance, Inc., 23 December 2011.

[MIPI02] *Frequently Asked Questions for MIPI Alliance Specification for Battery Interface (BIF)*, version 1.0, MIPI Alliance, Inc., In Press.

Participants

The following list includes those persons who participated in the Working Group that developed this White Paper and who consented to appear on this list.

Lionel Cimaz – ST-Ericsson

Markus Littow – ST-Ericsson