dsm
Dynamic Structures & Materials, LLC

114 Southeast Parkway Ct.
Ste 160
Franklin, TN 37064
Tel: 615/595-6665
Fax: 615/595-6610

SUM-40

COMMUNICATION PROTOCOL

DYNAMIC STRUCTURES AND MATERIALS, LLC
REV. 180517

The SUM-40 is an all-in-one motor and controller system. This document provides a brief summary of the communication protocol used to send commands to a standard SUM-40.

Communication Setup

The chip used to manage USB communication on the SUM-40 is the Silicon Labs CP2102. The virtual COM port (VCP) version of the driver for this chip is used, so recent Windows operating systems should recognize the VCP when the USB cable for the SUM-40 is connected and should install drivers automatically. If there are communication problems, drivers can be downloaded using the following link:

http://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx

The operating system of the host PC will assign a COM port number to the VCP. If a terminal emulator program is used, it may be apparent what COM port numbers have been assigned by looking at the list of available COM ports, or the Device Manager can be used in recent versions of Windows to see a list of available COM ports. Although Microsoft stopped bundling Hyperterminal as a standard feature of Windows after the XP version, there is a wide range of open and proprietary terminal emulators available, including Termite, Tera Term, Putty, Clear Terminal, CoolTerm, etc. Either a dedicated communication program or one of these platforms can be used to send commands to the SUM-40, as described in the section “Serial Commands” below.

The communication settings that should be used with this port are:

- Baud rate: 921600
- Data bits: 8
- Parity: none
- Stop Bits: 1
- Flow control: none

In addition to communication via the virtual comport, the SUM-40 also has connections to support the RS-232 communication protocol. The same communication structure based on ASCII text strings is used for both the VCP and RS-232; this protocol is described below.

Serial Commands

Command Structure

Commands are sent to the SUM-40 via text strings terminated by the “carriage return” character (ASCII 13 or 0x0D in hexadecimal). The basic structure is a three character code to identify the command to be executed, followed in some cases by a numeric argument, then followed by the carriage return character signifying that the string is complete and should be executed. A space between the three character command and the numeric argument is acceptable but not required. Commands are not case sensitive. If a terminal emulator program such as Hyperterminal is used, a carriage return is transmitted by pressing the “enter” key. If using custom software, the command should be followed by a carriage return.

For example, one of the first commands likely to be sent to a motor is “HOM” which is used to begin a homing routine for the motor. There is no argument associated with this command. It can be sent as follows:

Text String	or	ASCII codes in Hexadecimal
HOM (followed by enter)		48 4F 4D 0D
also acceptable:	or	
hom (followed by enter)		68 6F 6D 0D

The backspace character is recognized and, when received, will cause the previously transmitted character to be discarded.

Responses

When a command is successfully received, the controller will send a brief acknowledgement; these responses are listed with the descriptions of each command below. Polling the current value of a setting is done by sending the three character op code without an argument. When this is done, the controller will respond with the setting for the parameter in question. Not all commands return values when polled. For example, the CLM command is used only to initiate moves, and will not return a value when called without an argument. Commands that return values when called without an argument are specified in the command list below. Commands that return values will terminate the response with a CR/LF (hex 0x0D 0x0A) at the end of the response.

Note that many commands will return a confirmation value upon receipt of the command, then will optionally return another value when the operation is completed. For example, the HOM command will respond that the Homing routine has begun immediately upon receipt of the HOM command. The command RME is used to enable or disable the notification that he move has been completed. By default, RME is set to 1, meaning the system will return a value once the move has been completed. To disable this feature, set RME to 0.

Command List

All commands consist, at minimum, of three characters. Some commands also accept an optional argument following the three character base command. The following commands are provided for customer use:

CCM – Set communication mode
Argument – 0 (USB) or 1 (RS232)
Typical Controller Response – CCM=*(mode)*
Use this command to set the controller to communicate via either the USB-C port or the RS-232 port. Use the SAV command after setting this command to store the value in nonvolatile memory so that it retains the new communications mode upon startup.

CLM – Make closed loop move, absolute
Argument – Decimal value
Units – Input value is in degrees
Controller Response – CLM = *(position in degrees)*
This command cannot be used until a homing routine has been executed (see: HOM command). Upon receipt of this command, the SUM-40 will drive to the specified target angle. A trapezoidal speed profile is used, with the target speed starting at 0 and increasing at a constant rate (set using TAC) until the target speed (SST) is reached. As the SUM-40 comes near to the target angle, the target speed will decrease at the rate specified by TDC command until it reaches the minimum speed which can be changed using STM.

How closely the output comes to the exact specified angle is dependent on a number of factors, including the position deadband (PDB) setting, minimum speed (STM) setting, and the inertia attached to the SUM-40. If continuous closed loop control is enabled (DAS=0) inappropriate settings combined with high inertia can lead to motor instability. The SUM-40 should not be operated in an unstable condition. Changes that can be made to avoid instability include disabling continuous closed loop control (DAS=1), increasing position deadband (PDB), decreasing minimum speed (STM), decreasing deceleration (TDC), and decreasing inertia and other external loads. DSM generally recommends a position deadband of at least 3 counts; if more precise positioning is needed, very low values for STM and TDC should be used.

The CLM command tracks angular position over multiple turns. For example, if the motor is located at position 0 (PSM=0) and the motor is directed to go to angle 720 (CLM 720) it will rotate in the positive direction two times and stop at the same motor orientation. CLM uses an absolute reference system, but an offset can be added using the POS command. The CRM (relative move) and CSM (shortest move) commands are similar to CLM and can be used in cases where an absolute angular move is not desired.

CLO – Closed loop offset
Argument – Decimal value
Units – Degrees
Typical Controller Response – CLO=*(offset value in degrees)*
The absolute position control moves CLM and CSM direct the SUM-40 to go to a specific angular orientation; this angle is defined by the location of the index on the incremental encoder that is part of the SUM-40. The index defines the location of "zero" and all moves are measured

relative to it. In some cases, it may be desirable to treat a different motor orientation as zero. The CLO command can be used to add an arbitrary offset to readings from the encoder, allowing a different zero datum to be set for the SUM-40.

CRM – Make closed loop move, relative
Argument – Decimal value
Units – Input value is in degrees
Controller Response – "Relative Closed Loop Move Started" ***(immediate)***
"Move stopped, status = 0" ***(upon completion. Set RME=0 to silence response)***

This command cannot be used until a homing routine has been executed. Upon receipt of this command, the SUM-40 will drive to the specified target angle. Considerations for issues such as acceleration and stability are the same those for the CLM move and are summarized in its description. The CRM command is different than CLM in that the specified move is relative, e.g. if "CRM 20" is executed it moves the motor 20 degrees in the positive direction from the current position, rather than moving it to the 20 degree absolute position.

DAD – Halt digital data capture
Argument – None
Typical Controller Response – Data acquisition halted
The SUM-40 has a capacity to log and report position data. This feature can be useful for troubleshooting and tuning performance of the motor. If a block of data has been logged by the SUM-40, it can be output using the DMO command. The SDM command is used to determine whether data is logged at the start of a new move. Unless action is taken to halt data acquisition, once it begins, it will continue until all of the available memory is filled. By sending the DAD command, data acquisition can be manually stopped, rather than waiting for the normal process to halt data acquisition when the buffer is full.

DAS – Disable after stop
Argument – 0 or 1
Units – None
Typical Controller Response – DAS=*(disable after stop status)*
When performing moves in closed loop angle control such as those initiated by the CLM command, the SUM-40 controller seeks to drive the output to a specified angle. If the DAS setting is set to 1, once the motor moves within the deadband around the position target (set with PDB command) the motor will stop and the closed loop control algorithm will also be disabled. If the DAS parameter is set to 0, the closed loop control setting will remain enabled even after the motor reaches the specified angle target and will stay active until disabled with STP or a similar command. If a 1 is sent as the argument for the DAS command, the parameter will be set to 1. Any other argument will cause the parameter to be set to 0. If no argument is set, it returns the current value for the DAS parameter.

The advantage of having DAS enabled (set to 1) is that the motor is less likely to have problems with stability. Instability is more likely if a large inertia is attached to the output or high speeds and decelerations are specified. The disadvantage is that with DAS enabled, the motor may overshoot or not finish a move as close to the angle target as desired. By setting DAS to 0, the closed loop algorithm can be left enabled to reduce the risk of unknowingly having an unacceptable error relative to the target angle.

DIS – Digital input status
Argument – None
Typical Controller Response – DIS=*(digital input status)*
The SUM-40 includes a digital input line that can be used for triggering operations or as a general-purpose input. This command can be used to read the status of this line. It returns a 1 if the line is "high," that is, if it has been pulled to an elevated voltage level. The switching level for this signal is generally around 3 Volts; a voltage above the switching level will cause the DIS command to return a 1, while lower voltages will result in a return value of 0. The DIS command can be useful to troubleshoot and insure that a digital input is operating correctly before using it as a trigger.

DMO – Output captured data
Argument – None
Typical Controller Response – *(output stream of position data)*
The SUM-40 has a capacity to log and report position data. This feature can be useful for troubleshooting and tuning performance of the motor. If a block of data has been logged by the SUM-40, it can be output using the DMO command. The SDM command is used to determine whether data is logged at the start of a new move. The data is sampled at 5 kHz.

DOS – Digital output status
Argument – 0 or 1
Units – None
Typical Controller Response – DOS=*(digital output status)*
The SUM-40 includes a general purpose digital output line. This command can be used to set or read the status of this line. If it is sent with an argument of 1, it will set the line high. Any other argument will cause the output line to be set low. If no argument is set, it returns the status of the output line (1 for high, 0 for low) without changing the value.

HOM – Execute homing routine

Argument – 0 or 1 ***(Optional- defaults to 1 if no argument is specified)***

Units – None

Typical Controller Response – Homing sequence started ***(immediately)*****/Homing sequence completed – status n** ***(where n = 1 for successful homing routine, or 0 for failed homing routine)***

The SUM-40 contains an incremental encoder that is used to measure angle and angular velocity. In order to ensure that the encoder operates properly, it should first be "homed" before making other moves. This accomplished with a homing routine, which is executed with the HOM command. When the HOM command is sent, the SUM-40 will rotate until it reaches the index point of the incremental encoder. After the move is finished, it is appropriate to execute CLM and similar closed loop angle control commands. The argument is used to specify the direction for the homing move. If the argument is 1, the motor will start in the positive direction during the homing move. Otherwise, it will rotate in the negative direction. If there is a travel limit that initially prevents the SUM-40 from reaching the encoder index, it will reverse rotation and try to reach index by traveling in the opposite sense.

JGB – Jog backward

Argument – None

Typical Controller Response – Jogged backward

Upon receipt of this command, the SUM-40 will rotate in the "backward" or clockwise sense. The move continues for a set duration rather than moving to a specified angle, so it will not move exactly the same amount every time. Because it is an open loop move, it is not necessary to execute a homing routine on the motor before making a jog move, but the controller will seek to rotate the SUM-40 at the speed target specified by the SST command. The rotational direction for a backward move corresponds to the motion that causes the encoder reading to become more negative. Direction conforms to the right hand rule and can be visualized by imagining your right hand grabbing the motor with the thumb pointed away from the base of the motor. Your fingers will curl in the direction of a "positive" or "forward" move. The duration of the jog move can be changed using the JGD command.

JGD – Jog duration

Argument – Decimal value

Units – Seconds

Controller Response – JGD = ***(duration in seconds)***

Jog moves executed by JGB or JGF move for a set duration that is set using the JGD command. The duration is specified in seconds. JGD can also be used to see the duration of jog moves without changing it by simply transmitting JGD without an argument.

JGF – Jog forward
Argument – None
Typical Controller Response – Jogged forward
Upon receipt of this command, the SUM-40 will rotate in the "forward" or counterclockwise sense. The move continues for a set duration rather than moving to a specified angle, so it will not move the same amount every time. Because it is an open loop move, it is not necessary to execute a homing routine on the motor before making a jog move, but the controller will seek to rotate the SUM-40 at the speed target specified by the SST command. The rotational direction for a forward move corresponds to the motion that causes the encoder reading to become more positive. Direction conforms to the right hand rule and can be visualized by imagining your right hand grabbing the motor with the thumb pointed away from the base of the motor. Your fingers will curl in the direction of a "positive" or "forward" move. The duration of the jog move can be changed using the JGD command.

MRS – Motor running status
Argument – None
Typical Controller Response – MRS=*(0 or 1)*
The MRS command returns a 1 if the SUM-40 is running, that is, if the motor is energized and trying to move. Otherwise, this command returns a 0. The MRS command can be useful in cases where an automated process with multiple operations is running and the controller needs to be polled by a supervisory program so that it will know when a move is complete.

PDB – Position deadband for closed loop position control
Argument – Positive decimal value
Units – Degrees
Typical Controller Response – PDB=*(deadband value in degrees)*
When the SUM-40 is moving to a specified position using CLM or a similar command, it does not have to close exactly to the specified angle. There is a "deadband" around the target angle, and any place within this deadband is acceptable. PDB is used to see the deadband value or to change it. PDB defines the acceptable variation from the position target in degrees. If the SUM-40 is within this band, the motor is stopped. A smaller deadband can cause the SUM-40 to close more precisely on the specified target angle, but presents more risk of instability.

POS – Report current position, single turn interpretation
Argument – None
Units – Degrees
Typical Controller Response – POS=*(rotary angle in degrees)*
This command reports the current SUM-40 angular position based on a reading from the integral encoder. Angle is reported in degrees. Note that the motor must first execute a successful homing routine in order for this reading to be meaningful; otherwise, the position is reported relative to the position of the system on power-up and may change unpredictably if the encoder index is passed. Angle is reported in the context of a single turn; if the motor has previously

rotated multiple times in a given direction, these revolutions are not included in the reported angle. If an offset has been previously added using the POS command, it will be included in the response.

PPV – Report user parameter values
Argument – None
Units – Various
Typical Controller Response – multiple lines with user values
Various control parameters in the SUM-40 can be configured to match the end user's specific application. Examples include target speed and rates of acceleration and deceleration. Many of these settings can be saved using the SAV command so that they are not lost when the SUM-40 is powered down. The PPV command prints these user values with a single command so that the user does not need to query each command individually. Values reported include the speed target, maximum and minimum allowable speed targets and position targets, jog duration, position offset, position deadband, acceleration rate, deceleration rate, and homing speed target.

PSM – Report current position, multiple turn interpretation
Argument – None
Units – Degrees
Typical Controller Response – PSM=*(rotary angle in degrees)*
This command reports the current SUM-40 angular position based on a reading from the integral encoder. Angle is reported in degrees. Note that the motor must first execute a successful homing routine in order for this reading to be meaningful; otherwise, the position is reported relative to the position of the system on power-up and may change unpredictably if the encoder index is passed. Angle is reported accounting for multiple turns; if the motor has previously rotated multiple times in a given direction, these revolutions are included in the reported angle. If an offset has been previously added using the POS command, it will be included in the response.

PTM - Set minimum position target
Argument – Decimal value
Units – Degrees
Typical Controller Response – PTM=*(minimum position value in degrees)*
In cases where CLM and similar commands are used to move the SUM-40 in closed loop position control, it is possible to set a minimum allowable position target. The PTM command sets the minimum target that will be accepted. Thus, PTM can offer an extra degree of protection to prevent an error if an unacceptable position target is accidentally commanded. This can be useful if the SUM-40 is used to drive a ball screw or a similar application in which travel is limited.

PTX - Set maximum position target
Argument – Decimal value
Units – Degrees
Typical Controller Response – PTX=*(maximum position value in degrees)*
In cases where CLM and similar commands are used to move the SUM-40 in closed loop position control, it is possible to set a maximum allowable position target. The PTX command sets the maximum target that will be accepted. Thus, PTX can offer an extra degree of protection to prevent an error if an unacceptable position target is accidentally commanded. This can be useful if the SUM-40 is used to drive a ball screw or a similar application in which travel is limited.

RME – Respond on move end
Argument – 0 or 1
Units – None
Typical Controller Response – RME=*(0 or 1)*
In cases where an automated process with multiple operations is running, a supervisory program may need to know when a move is complete. The RME command is used to control whether the SUM-40 transmits notification when a move is finished. If RME is set to 1, a statement is sent every time a move is finished. If RME is set to 0, no message is sent.

ROC – Rotate continuously
Argument – 0 or 1
Typical Controller Response – ROC=0
Upon receipt of this command, the controller will begin rotating the motor in a direction specified by the argument. If the argument is 1, the motor will rotate in the negative direction; for any other argument, the SUM-40 will rotate in the positive direction. No position target is associated with the move, so the SUM-40 will keep moving in this direction until a contravening command is received. Rotation will occur at a speed consistent with the existing speed target. The speed may be updated while the motor is moving (SST command) but the target will change instantaneously, rather than at a specified rate of acceleration or deceleration. The motor can be stopped with the STP command.

SAV – Save user values
Argument – None
Units – Various
Typical Controller Response – SAV=0
Various control parameters in the SUM-40 can be configured to match the end user's specific application. Examples include target speed and rates of acceleration and deceleration. Many of these settings can be saved using the SAV command so that they are not lost when the SUM-40 is powered down. The PCV command prints these user values with a single command so that the user does not need to query each command individually. Parameters saved using SAV include the speed target, maximum and minimum allowable speed targets and position targets, jog

duration, position offset, position deadband, acceleration rate, deceleration rate, and homing speed target. All of these parameters should be set to the desired values before the SAV command is executed, since all start-up values will be saved together.

SDM – Start data acquisition for new move
Argument – 0 or 1
Units – None
Typical Controller Response – SDM=*(start data acquisition status)*
The SUM-40 has a capacity to log and report position data. This feature can be useful for troubleshooting and tuning performance of the motor. If the SDM parameter is set to 1, when a new move is begun with position commands such as CLM or jog moves, the position of the SUM-40 will be logged. This information can be subsequently output using the DMO command. Note that any previous data will be lost when a new move is started with data acquisition enabled.

SSM – Motor soft start
Argument – 0 (disabled) or 1 (enabled)
Units – None
Typical Controller Response – SSM=*(start data acquisition status)*
When enabled, this sets the motor into "soft start" mode. In this mode, the motor will start each move very slowly. This is in addition to the standard acceleration ramp. Enabling this mode will cause the moves to be much slower, but may help with extremely small moves. For example, if you need to move only a few encoder counts, enabling this mode will allow the motor to start at a much lower speed than normal. Typically, the motor requires several counts to pass before being able to reliably determine the motor's speed; thus, when moving only a few encoder counts, the motor cannot adequately control its speed in such a short move. Enabling the SSM mode will allow the motor attempt to start at a very low speed, running without direct speed control for a few encoder counts.

SST - Set speed target
Argument – Positive decimal value
Units – Degrees per second
Typical Controller Response – SST=*(velocity value in degrees per second)*
The SUM-40 controller is able to vary the rotational speed of the motor. This command can be used to adjust speed target for the SUM-40. Many moves such as those executed by the CLM command use a planned trajectory in which the SUM-40 accelerates gradually at the beginning of a move and when a motor is near its final target, the controller reduces the speed in order to prevent overshoot. Consequently, if the target velocity is high, it will be overridden when close to the target and the motor may never achieve high speed targets for very short moves. The rates of acceleration and deceleration can be set using the TAC and TDC commands. Note that high speeds can result in overshoot and instability, especially if large inertia is connected to the output

of the SUM-40. More gradual acceleration and deceleration can help to reduce the risk of overshoot and instability.

STA – SUM-40 status
Argument – None
Typical Controller Response – short list of parameter values
Upon receipt of this command, the SUM-40 will send a response summarizing the current status of the SUM-40. These values are generally used for characterization at the factory and should not generally be needed by the end user. In some cases, this command may be useful for troubleshooting. One of the returned parameters is the current velocity, and is indicated in the string as Vel0=x.xxxxx, where x.xxxxx is the speed in degrees per second.

STH - Set speed target for homing move
Argument – Positive decimal value
Units – Degrees per second
Typical Controller Response – STH=*(velocity value in degrees per second)*
The SUM-40 controller is able to vary the rotational speed of the motor. This command can be used to adjust speed the target that the SUM-40 uses during a homing move. A homing move is generally executed immediately after the SUM-40 is first powered up. This is because an incremental encoder is used to measure the angle of the SUM-40, and the encoder will not provide consistent angular measurement until the homing routine is complete. It is a good practice to use a slow speed for the homing move.

STM - Set minimum speed
Argument – Positive decimal value
Units – Degrees per second
Typical Controller Response – STM=*(velocity value in degrees per second)*
Many moves such as those executed by the CLM command use a planned trajectory in which the SUM-40 accelerates gradually at the beginning of a move and when a motor is near its final target, the controller reduces the speed in order to prevent overshoot. However, the speed target is never reduced completely to zero during the move. This command sets the minimum speed target, which is used at the end of the move. Setting this value lower can help to reduce the risk of overshoot and instability. Setting the value higher allows the SUM-40 to finish a move slightly more rapidly. The lowest acceptable value for the minimum speed is 5 degrees per second.

STP – Stop all moves
Argument – None
Typical Controller Response – STP = 1
This command directs the SUM-40 to stop. All moves will be halted by this command. If the SUM-40 is operating in closed loop angular control, this setting will be cleared as well.

STX - Set maximum speed target
Argument – Positive decimal value
Units – Degrees per second
Typical Controller Response – STX=*(velocity value in degrees per second)*

The SUM-40 seeks to rotate at a speed set using the SST command. In cases where a high operating speed is not acceptable, the STX command can be used. STX sets the highest value that will be accepted when SST is used to set the speed target. Thus, STX can offer an extra degree of protection to prevent an error if an unacceptable speed target is accidentally commanded.

TAC – Rate of acceleration
Argument – Positive decimal value
Units – Degrees per second per second
Typical Controller Response – TAC=*(acceleration value in degrees per second per second)*

Many moves such as those executed by the CLM command use a planned trajectory in which the SUM-40 accelerates gradually at the beginning of a move and when a motor is near its final target, the controller reduces the speed in order to prevent overshoot. The target rate of acceleration can be set using the TAC command. The units for acceleration are deg/s^2. A higher rate of acceleration may allow the SUM-40 to complete moves more quickly, but may jar the payload more abruptly than lower acceleration. It should be noted that setting a high target acceleration rate does not guarantee that the SUM-40 will achieve this rate, particularly if a large payload is connected to the output.

TDC – Rate of deceleration
Argument – Positive decimal value
Units – Degrees per second per second
Typical Controller Response – TDC=*(deceleration value in degrees per second per second)*

Many moves such as those executed by the CLM command use a planned trajectory in which the SUM-40 accelerates gradually at the beginning of a move and when a motor is near its final target, the controller reduces the speed in order to prevent overshoot. The target rate of deceleration can be set using the TDC command. The units for deceleration are deg/s^2. A higher rate of deceleration may allow the SUM-40 to complete moves more quickly, but may jar the payload more abruptly than lower deceleration and run increased risk of motor instability. It should be noted that setting a high target deceleration rate does not guarantee that the SUM-40 will achieve this rate, particularly if a large payload is connected to the output.

TEM – Read internal temperature sensor
Argument – None
Units – Degrees Celsius
Typical Controller Response – TEM=*(temperature in degrees C)*

The SUM-40 includes an internal temperature sensor. It can be used to monitor system temperature. In some cases, such as running at high speed for extended time, temperature may become elevated and reach a level where performance or long-term reliability could be compromised. DSM does not generally recommend operating the SUM-40 at an indicated temperature greater than 80 degrees C; if the internal temperature reaches 80 C, it will stop moving until it has cooled down.

VER – Version information
Argument – None
The VER command will return a short string describing the SUM-40 version. This typically includes the firmware version and a compile date.