

# Exhibit 5



Valentine, Matthew E. Hocker, Ryan James Moran, Christopher L. Kelley, Howrey LLP, East Palo Alto, CA, Thomas John Ward, Jr., Ward & Smith Law Firm, Longview, TX, for Defendants.

Only the Westlaw citation is currently available.

CLAIM CONSTRUCTION ORDER

United States District Court,

DAVID FOLSOM, District Judge.

E.D. Texas,

Construing Terms in U.S. Patent Nos. 5,691,897,  
6,513,058, 6,516,236  
and 6,941,543

Marshall Division.

**ROY-G-BIV CORP., Plaintiff,**  
v.  
**FANUC LTD., et al., Defendants.**  
**Civil Action No. 2:07-CV-418 (DF).**

\*1 Before the Court are RGB's Opening Brief on Claim Construction (Dkt. No. 100), FANUC's Opening Claim Construction Brief (Dkt. No. 105), RGB's Reply Brief on Claim Construction (Dkt. No. 109), and FANUC's Surreply Brief (Dkt. No. 117). Also before the Court are the Local Patent Rule (LPR) 4-3 Joint Claim Construction and Prehearing Statement (Dkt. No. 93) and the LPR 4-5 Supplemental Joint Claim Construction Chart (Dkt. No. 119; Dkt. No. 119, Ex. B (Second Supplemental Exhibit B)). A claim-construction hearing, in accordance with *Markman v. Westview Instruments*, 52 F.3d 967 (Fed.Cir.1995) (en banc), aff'd, 517 U.S. 370, 116 S.Ct. 1384, 134 L.Ed.2d 577 (1996), was held in Texarkana on April 16, 2009. Dkt. No. 146 (hearing transcript). After hearing the arguments of counsel and reviewing the relevant pleadings, presentation materials, other papers, and case law, the Court finds the disputed terms of the patents-in-suit should be construed as set forth herein.

Aug. 25, 2009.

Lance Henry Lubel, Adam Quentin Voyles, John Milton Black, Heard Robins Cloud & Lubel, LLP, Russell Allen Chorush, Heim Payne & Chorush LLP, Houston, TX, Aaron J. Snow, D. Michael Underhill, Eric J. Maurer, Evan A. Parke, William A. Isaacson, Boies Schiller & Flexner LLP, Washington, DC, Harold Kip Glasscock, Jr., Kip Glasscock Attorney at Law, Beaumont, TX, for Plaintiff.

Henry Charles Bunsow, Howrey Simon Arnold & White, San Francisco, CA, Christina Finn, Anne M. Ortel, Daniel T. Shvodian, Gregory K. Storey, James F.

TABLE OF CONTENTS

I.	BACKGROUND .....	-1-
II.	LEGAL PRINCIPLES .....	-1-
III.	PATENTS-IN-SUIT .....	-2-
IV.	U.S. PATENT NO. 5,691,897 .....	-4-
	A. Overview .....	-4-
	B. Claim Construction .....	-5-
	1. "application program" .....	-5-
	2. "component function" .....	-7-
	3. "component code" .....	-11-
	4. "software driver(s)" / "drivers" .....	-12-
	5. "motion control operation(s)" .....	-15-
	6. "primitive operation(s)" .....	-18-
	7. "driver function(s)" .....	-19-
	8. "core driver functions" .....	-21-
	9. "driver code" .....	-22-
	10. "control command(s)" .....	-24-
	11. "developing a set of software drivers" .....	-26-
	12. "defining a [core/extended] set of [core/extended] driver functions" .....	-28-
	13. "defining a set of component functions" .....	-29-
	14. "defining a set of motion control operations" .....	-29-
	15. "providing component code for each of the component functions" .....	-30-
	16. "providing response stream code" .....	-31-

17.	"selecting at least one of the destinations" ("of control commands")	-32-
18.	"selecting from the set of software drivers the software driver developed for the selected motion control device"	-37-
19.	"selecting one motion control device"	-38-
V.	U.S. PATENT NO. 6,513,058	-38-
A.	Overview	-38-
B.	Claim Construction	-40-
1.	"network"	-40-
2.	"a control command generating module for generating control commands based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the software drivers"	-41-
VI.	U.S. PATENT NO. 6,516,236	-44-
A.	Overview	-44-
B.	Claim Construction	-46-
1.	"a selected destination of control commands"	-46-
2.	"a selected software driver"	-47-
3.	"motion control component"	-47-
4.	"a motion control component for generating the sequence of control commands for controlling the selected motion control device based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the selected software driver"	-48-
5.	"stream control means for communicating the control commands to the selected destination of control commands based on the transmit stream code contained by the stream associated with the selected destination of control commands"	-51-
6.	"the stream control means processes the response data based on the response stream code"	-54-
VII.	U.S. PATENT NO. 6,941,543	-55-
A.	Overview	-55-
B.	Claim Construction	-57-
1.	"selected from a plurality of software drivers"	-57-
2.	"selecting a software driver"	-58-
3.	"incremental motion step(s)"	-58-
4.	"identifies an incremental motion step"	-60-
VIII.	CONCLUSION	-61-

## I. BACKGROUND

\*2 In the present lawsuit, ROY-G-BIV Corp. ("RGB") contends certain software (and accompanying equipment) developed, sold, offered for sale, used or imported by FANUC Ltd., FANUC Robotics America, Inc., GE Fanuc Automation Americas, Inc., and GE Fanuc Intelligent Platforms, Inc. (collectively, "FANUC") infringe claims of U.S. Patent Nos. 5,691,897 ("the '897 Patent"), 6,513,058 ("the '058 Patent"), 6,516,236 ("the '236 Patent"), and 6,941,543 ("the '543 Patent"). Both the '897 and '236 Patents are entitled "Motion Control Systems," while the '058 Patent is entitled "Distribution of Motion Control Commands Over a Network," and the '543 Patent is entitled "Motion Control System and Method." All three later patents are continuations- in-part of the '897 Patent. '058 at [63]; '236 at [63]; '543 at [63].

## II. LEGAL PRINCIPLES

A determination of patent infringement involves two steps: first, the patent claims are construed, and second, the claims are compared to the allegedly infringing device. *Cybor Corp. v. FAS Techs., Inc.*, 138 F.3d 1448, 1455 (Fed.Cir.1998) (en banc). The legal principles of claim construction were reexamined by the Federal Circuit in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed.Cir.2005) (en banc). The Federal Circuit in *Phillips* expressly reaffirmed the principles of claim construction as set forth in *Markman v. Westview Instruments, Inc.*, 52 F.3d 967 (Fed.Cir.1995) (en banc), *aff'd*, 517 U.S. 370, 116 S.Ct. 1384, 134 L.Ed.2d 577 (1996), *Vitronics Corp. v. Conceptiontronic, Inc.*, 90 F.3d 1576 (Fed.Cir.1996), and *Innova/Pure Water, Inc. v. Safari Water Filtration Sys.*,

Inc., 381 F.3d 1111 (Fed.Cir.2004). Claim construction is a legal question for the courts. Markman, 52 F.3d at 979.

The Court, in accordance with the doctrines of claim construction that it has outlined in the past, will construe the claims of the RGB Patents below. See Pioneer v. Samsung, No. 2:07-CV-170, Dkt. No. 94, at 2-8 (E.D. Tex. filed Mar. 10, 2008) (claim-construction order).

### III. PATENTS-IN-SUIT

The patents-in-suit are directed to a particular software program development toolkit for controlling the motion of equipment and hardware, independent of the nature of the mechanical system that controls that motion. '897 Patent, 1:10-2:15. The '897 Patent consists of methods claims that issued on Nov. 25, 1997 from an application filed on May 30, 1995. Id. at 33:60-38:40, [45], [22]. The '897 Patent abstract reads:

A system for motion control in which an application is developed that is independent from the actual motion control hardware used to implement the system. The system comprises a software system that employs an application programming interface comprising component functions and a service provider interface comprising driver functions. A system programmer writes an application that calls the component functions. Code associated with the component functions relates these functions to the driver functions. A hardware designer writes driver code that implements the driver functions on a given motion control hardware product. The driver functions are separated into core and extended driver functions. All software drivers implement the core driver functions, while the software drivers need not contain code for implementing the extended driver functions. If the software driver does not contain code to implement an extended driver function, the functionality of the extended driver function is obtained through a combination of core driver functions. The system programmer may also select one or more streams that allow the control commands to be communicated to, and response data to be communicated from, motion control hardware. A system for allowing an application program to communicate with any one of a group of supported hardware devices comprising a software system operating on at least one workstation and a network communications protocol. The software system includes a control command generating module for generating control commands based on component functions of an application program, component code associated with the component functions,

and the driver code associated with software drivers associated with the hardware devices. The network communication protocol allows the control commands to be communicated from the control command generating module to at least one of the supported hardware devices over the network.

\*3 Id. at [57].

The '058 Patent issued January 28, 2003 from an application filed on February 27, 2001. '058 Patent at [45], [22]. The '058 Patent consists of system claims directed to the communication aspect of the software program development toolkit, which transmits commands to control the motion. Id. at 49:50-52:18. The '058 Patent abstract reads:

A system for allowing an application program to communicate with any one of a group of supported hardware devices comprising a software system operating on at least one workstation and a network communications protocol. The software system includes a control command generating module for generating control commands based on component functions of an application program, component code associated with the component functions, and the driver code associated with software drivers associated with the hardware devices. The network communication protocol allows the control commands to be communicated from the control command generating module to at least one of the supported hardware devices over the network.

Id. at [57].

The '236 Patent issued with system claims on February 4, 2003 from an application filed December 10, 2001. '236 Patent, 48:10-50:41, [45], [22]. The abstract from the '236 Patent mirrors that of the '897 Patent. Id. at [57].

The '543 Patent issued September 6, 2005 from an application filed on August 7, 2000. '543 Patent at [45], [22]. The '543 Patent consists of method and system claims directed to the drivers aspect of the software program development toolkit. Id. at 47:17-48:60. The '543 Patent abstract reads:

A system for motion control in which an application is developed that is independent from the actual motion control hardware used to implement the system. A software system employs an application programming interface comprising component functions and a service provider interface comprising driver functions. Code associated with the component functions relates these functions to the driver functions.

Driver functions are separated into core and extended driver functions. All software drivers implement the core driver functions, and optionally implement the extended driver functions. Extended driver functionality may be obtained through a combination of core driver functions.

Id. at [57].

#### IV. U.S. Patent No. 5,691,897

##### A. Overview

RGB has asserted claim 25 (dependent on claims 17 and 24) of the '897 Patent against FANUC in this lawsuit. Dkt. No. 119. For reference, claims 17, 24 and 25 are reproduced below (terms to be construed emphasized):

17. A method of generating a sequence of control commands for controlling a motion control device to perform a given series of motion steps based on an application program defining the given series of motion steps, the method comprising the steps of:

defining a set of motion control operations, where each motion control operation is either a primitive operation that is necessary to perform motion control and that cannot be simulated using other motion control operations or a non-primitive operation that does not meet the definition of a primitive operation;

\*4 defining a core set of core driver functions, where each core driver function identifies one of the primitive operations;

defining an extended set of extended driver functions, where each extended driver function identifies one of the non-primitive operations;

defining a set of component functions; providing component code for each of the component functions, where the component code cross-references at least some of the component function(s) with at least some of the driver functions;

developing a set of software drivers, where (i) each software driver is developed for a motion control device in a supported group of motion control devices and (ii) each software driver comprises driver code for implementing the motion control operations identified by at least

some of the driver functions;

selecting one motion control device from the group of supported motion control devices;

selecting from the set of software drivers the software driver developed for the selected motion control device; and

generating control commands based on the application program, the component code, and the driver code of the selected software driver.

24. A method as recited in claim 17, further comprising the steps of:

providing a plurality of destinations of control commands;

providing a plurality of streams, where each stream contains transmit stream code that determines how the control commands are to be transferred to at least one of the destinations of control commands;

selecting at least one of the destinations of control commands; and

transferring the control commands to the selected destination of control commands based on the transmit stream code contained by the stream associated with the selected destination of control commands.

25. A method as recited in claim 24, in which certain of the destinations of control commands generate response data, the method further comprising the steps of:

providing response stream code for the streams associated with the destinations of control commands that generate response data; and processing the response data based on the response stream code.

'897 Patent, 36:65-37:12, 38:17-39 (emphasis added).

##### B. Claim Construction

###### 1. "application program"

###### a. Parties' Positions

The parties offer the following constructions for the term "application program," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

-----  
"a software program designed to handle specific tasks"

-----  
"a hardware-independent program designed to move an object in a desired manner"

RGB contends the term is used generically in both the specification and the industry. Dkt. No. 100 at 39-40. To support its position, RGB cites examples in the specification that arguably use the term in a generic manner. Id. (citing '897 Patent, 8:6-8 (application program 26 "is any application that uses the system 22 by programming the motion control component 34" (emphasis added))). RGB also relies on a technical dictionary, which defines an application program as "a program that is specific to the solution of an application problem." Id. (citing IBM DICTIONARY OF COMPUTING 28 (10th ed., 1993)). RGB argues FANUC's definition imports limitations from the specification into the claims even though the specification uses the term broadly. Id.

\*5 In response, FANUC also relies on the specification to support its construction that the application program is for the specific task of "moving" and is "hardware-independent." Dkt. No. 105 at 12-14; Dkt. No. 117 at 4- 5. For example, FANUC argues that the "Objects of the Invention" section supports its construction when it states the invention "allow[s] the creation of high level motion control programs that are hardware independent." Dkt. No. 105 at 13 (citing '897 Patent, 3:25-31 (emphasis added)). In addition, the first sentence of the specification states that "[t]he present invention relates to motion control systems and, more particularly, to interface software that facilitates the creation of hardware independent motion control software." Id. at 12 (citing '897 Patent, 1:1-4 (emphasis added)). FANUC argues that its use of intrinsic evidence is superior to RGB's extrinsic evidence based on a dictionary meaning. Id. at 13. In sum, FANUC contends the application program provided by the patented invention is not a general purpose application. Dkt. No. 146 at 31. Rather the entire purpose of the patented software is to have a single application program control particular motions of any machine or hardware. Id.

RGB replies that it is improper to limit the term to the various features that FANUC desires because those features are already recited in Patent's dependent claims. Dkt. No. 109 at 21-22. Regarding the issue of hardware

independency, RGB argues that FANUC's descriptions are merely exemplary because the specification describes a preferred embodiment. RGB Hearing Slides at 47 (citing '897 Patent, 6:27-32).

#### b. Court's Construction

The Court is not persuaded to include FANUC's suggested limitations based upon a preferred embodiment. The Court finds no basis in the specification to narrow the ordinary meaning of the term "application program." Claim language is generally given its "ordinary and customary meaning" unless the term has no generally accepted meaning, the patentee has given the term a specific meaning, or the patentee has disavowed all or part of the scope otherwise encompassed by the ordinary meaning. Phillips, 415 F.3d at 1313, 1316-17. Based on Plaintiff's evidence, the Court finds that the term "application program" had an ordinary meaning within the field at the time of the '897 Patent's filing. The patentee is entitled to that ordinary meaning unless it has given the term a specific meaning or disavowed all or part of the scope otherwise encompassed by the ordinary meaning. Here the patentee has neither given the term a specific meaning or disavowed part of the ordinary meaning. Instead, the specification uses the term broadly and it should be given its full meaning. See '897 Patent, 8:6-8 ("[t]he application program 26 is any application that uses the system ...").

For the reasons stated above, this Court finds that "application program" means "a software program designed to handle specific tasks."

#### 2. "component function"

##### a. Parties' Positions

\*6 The parties offer the following constructions for the term "component function," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

"a controller independent function corresponding to a motion control operation and available on the motion control component"

"a hardware independent function that defines motion steps to be performed by a motion control device to move an object along a desired path and permits the application programmer to control the hardware in base incremental steps"

RGB considers this term a key phrase. Dkt. No. 100 at 11. Both parties agree the term is hardware independent. *Id.* at 19. Other than this point of agreement, RGB adopts a broad construction and argues that it encompasses FANUC's construction. *Id.* at 20. RGB contends there are different types of component functions, some defining "motion control operations" that also include "read" operations. *Id.* (citing '897 Patent, 8:2-6).

RGB also argues that its construction is the only one that takes into account the fact that component functions are "available on the motion control component." See *id.* RGB relies on a portion of the specification, which states that the motion control component "is the portion of the software system 22 that relates the component functions to the driver functions." *Id.* (citing '897 Patent, 9:3-5 (emphasis added)). Moreover, RGB alleges that FANUC's expert, Mr. Mercer admitted that "component functions" perform "motion control operations" in his declaration. Dkt. No. 146 at 18. Finally, RGB asserts FANUC is construing the term too narrowly by importing limitations from the specification or from dependent claims. Dkt. No. 100 at 20.

FANUC maintains there is no commonly understood definition for the term to those of skill in the art. Dkt. No. 105 at 23-24. FANUC therefore bases its construction on two parts of the specification. *Id.* First, FANUC relies on the statement "[a]n application program comprising a series of component functions defines a sequence of motion steps that must be performed by the motion control device to move the object along the desired path." *Id.* at 23 (citing '897 Patent, 3:51-53 (emphasis added)). Second, FANUC contends that RGB defined the invention as one meant to "control the hardware in base incremental steps" and to "allow the creation of high-level motion control programs that are hardware independent, but offer programmability of base motion operations." *Id.* at 24 (citing '897 Patent, 2:62-3:14, 3:25-27 (emphasis added)). According to FANUC, the base motion operations relate to base incremental steps. FANUC Hearing Slides at 70 (citing '897 Patent, 2:62-3:14). Finally, FANUC argues it is incorrect to state that the component functions are "available on the motion control component" because some "component functions" require a separate Motion Control Driver Stub Module. Dkt. No. 117 at 10-11.

\*7 In reply, RGB explains that it is wrong to incorporate FANUC's "requirement that the component function defines motion steps to be performed by a motion control device to move an object along a desired path" based on FANUC's own admission that "GET POSITION" is a component function but is not a motion step. Dkt. No. 100 at 18.

#### b. Court's Construction

Neither party submits that the term "component function" is a term that has an ordinary meaning in the art. Indeed, FANUC argues there is not commonly understood definition for the term. The Court concludes that "component function" is a coined term; thus, its potential meaning must be found in the specification or file history. See *Irdeto Access, Inc. v. EchoStar Satellite Corp.*, 338 F.3d 1295, 1303 (Fed.Cir.2004) (one must look to the specification to discern the meaning of "coined" terms).

The term "component function" appears nearly two-dozen times in the '897 Patent's specification. In nearly every use, the specification clearly distinguishes high-level component functions from more specific, hardware-dependent driver functions. *Id.* at 3:62-65 ("The use of component functions that are separate from driver functions isolates the programmer from the complexities of programming to a specific motion control device."). Thus, as both parties agree, component functions are hardware independent.

In addition, the specification defines the term based on its functionality. In one of the specification's first use of the term, the invention is summarized as "a high-level motion control application program comprising a sequence of component functions that describe a desired object path ...." '897 Patent, 3:38-40. The end goal of the high-level program is to merely move an "object along the desired object path." *Id.* at 3:46-47. Later the high-level application program is said to comprise "a sequence of component functions arranged to define the motion control operations necessary to control a motion control device to move an object in a desired manner." *Id.* at 8:3-6. Thus, component functions are nothing more than high-level instructions used to describe the path eventually taken by a downstream device.

Due to fact that component functions are high-level instructions, it is not necessary that each component function actually correspond to a motion step or a base incremental step. Instead, a component function might serve an administrative purpose. See *id.* at 7:29-38 ("The relationship between component function and driver functions need not be one to one: for example, certain component functions are provided for administrative purposes and do not have a corresponding driver function."). Such administrative purposes might include a "GET POSITION" operation, which merely queries the system for the current position of the downstream device. See *id.* at 16:22-24. The Court thus finds it inappropriate to include FANUC's requested "base incremental step" limitation in the term's construction.

\*8 Finally, the specification makes it clear that component functions are used by the "motion control component" to relate component functions to driver functions. Id. at 9:1-5 ("The motion control component module 35 is the portion of the software system 22 that relates the component functions to the driver functions."). As such, the component functions are available on the motion control component. They are, however, also used elsewhere--and thus available elsewhere--in the program hierarchy. Id. at 7:29-31 ("The software system designer next defines an application programming interface (API) comprising a set of component functions."). Thus, the Court finds that RGB's "available on the motion control component" limitation is too restrictive and could potentially be misleading to the jury.

RGB

-----  
"software code in the motion control component that associates component functions with driver functions"

RGB considers this term another key phrase. Dkt. No. 100 at 11. RGB argues the component software code is in the "motion control component" because the specification states the "motion control component module 35 thus contains the component code that makes the association between the component functions contained in the application program 26 and the driver functions." Id. at 21 (citing '897 Patent, 9:1-9).

FANUC argues that the word "many" should be in the definition of the term because the specification states the component code associates many--rather than all--component functions with driver functions. Dkt. No. 105 at 26 (citing '897 Patent, 8:11-12 ("As mentioned above, the component code associates many of the component functions with the driver functions ..."))).

b. Court's Construction

In the Patent's first use of the term, the specification states that "the software system designer writes component code that associates at least some of the component functions with at least some of the driver functions." Id. at 7:31-34. The specification later references this definition. Id. at 8:11-12 ("As mentioned above, the component code associates many of the component functions with the driver functions ...."). Thus,

For the reasons stated above, this Court finds that "component function" means "a hardware independent instruction that corresponds to an operation performed on or by a motion control device."

3. "component code"

a. Parties' Positions

The parties offer the following constructions for the term "component code," which appears in claim 17. Dkt. No. 119-3. There are only minor disagreements between the two parties regarding the phrases "many" and "motion control component" within the proposed constructions.

FANUC

-----  
"code that associates many of the component functions with the driver functions"

the specification's use of the word "many" is controlled by the Patent's previous definition. Accordingly, the Court finds it more appropriate to use the specification's initial definition of the term in its construction.

\*9 In addition, the Court finds that the specification locates the component code in the motion control component module. Id. at 9:1-9 ("motion control component module 35 thus contains the component code ...").

For the reasons stated above, this Court finds that "component code" means "software code in the motion control component that associates at least some of the component functions with at least some of the driver functions."

4. "software driver(s)" / "drivers"

a. Parties' Positions

The parties offer the following constructions for the term "software driver(s)" / "drivers," which appears in claim 17. Dkt. No. 119-3.



RGB

FANUC

-----  
"a controller dependent  
binary software module  
that controls and/or  
regulates a hardware  
device"

-----  
"driver(s) associated with a single motion  
control device that contain(s) the driver code  
for that device. All software drivers must  
support the core driver functions."

RGB argues its definition is based mostly on the ordinary, industry meaning of a "driver," whereas FANUC's construction imports limitations from the specification and is circular because the word "driver" is used as part of the definition. Dkt. No. 100 at 21; RGB Hearing Slides at 71. RGB also maintains the specification supports its construction, stating that "drivers" used in the patents are described as "binary modules." Id. at 22 (citing '897 Patent, 10:36-38, 9:41-55, Fig. 2). In addition, RGB uses a technical dictionary that refers to "software" as "binary code." Id. (citing MICROSOFT PRESS COMPUTER DICTIONARY); Dkt. No. 109 at 13). Also, RGB proposes the software is "controller dependent" based on what a driver does, namely, it translates signals and outputs them to a device. Dkt. No. 100 at 22. Finally, RGB argues that extrinsic evidence supports a conclusion that a driver "controls and/or regulates a hardware device." Id.

According to FANUC, this term and the term "driver code" (infra ) should be construed and considered together. Dkt. No. 105 at 27. Because the patent uses the term in a very specific and unique manner, FANUC argues that the claim construction should reflect the Patent's particular use of the term. Id. Relying on the specification, FANUC extracts three conditions related to software drivers:

The software drivers 30 are normally developed by a hardware designer and are [1] each associated with a single motion control device. The hardware designer writes [2] driver code that dictates how to generate the control commands for controlling the motion control device associated therewith to perform the motion control operations associated with at least some of the driver functions.

[3] All software drivers must support the core driver functions; the software drivers may also support one or more of the extended driver functions, although this is not required.

\*10 Id. (citing '897 Patent, 7:51-57, 4:2-5 (numbers and emphasis added)).

#### b. Court's Construction

As an initial matter, the Court notes that the term "driver" or "software driver" is common to all four patents

at issue. Because the patents are related, the Court finds that a consistent construction of these terms is appropriate and would aid the jury's understanding of the case.

As FANUC contends, the '236 Patent identifies motion control devices that can accept human readable (e.g. ASCII II) control commands. '236 Patent, 35:12-22 ("the language driver 44 will construct a command data string containing ASCII II characters"). Conversely, the map shown in Figure 2 of the '897 Patent, which includes a driver, purports to display "binary modules." '897 Patent, 10:36-38. There is, however, nothing in the '897 Patent that suggests the drivers must be in binary form. The Federal Circuit has made it clear that limitations from the specification may only be imported into the claims where it is clear that the patentee intended for the "claims and embodiments in the specification to be strictly coextensive." Phillips, 415 F.3d at 1323 (citing SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc., 242 F.3d 1337, 1341 (Fed.Cir.2001)). Because there is no evidence of a coextensive relationship between the drivers and binary code, the Court finds no reason to so limit the claims.

The Court also finds it inappropriate to limit each driver to a single motion control device. The '897 Patent suggest that during a driver's development it will usually be associated with a single motion control device. '897, 7:51-53 ("The software drivers 30 are normally developed by a hardware designer and are each associated with a single motion control device."). Once a driver is developed, however, it may later be used to control devices similar to the one to which it was initially associated. Id. at 32:30-37 ("Usually the control language used by a hardware vendor is supported by several different flavor of their motion control hardware. For example, some vendors have both PC Bus based and Serial I/O based motion control hardware that understand the same control language. In such a case, the same driver would be used for each hardware setup ...."); Dkt. No. 100, Ex. 8 ("Hooper Decl.") at i 16 ("one of ordinary skill would understand that 'the AT6400 driver,' which 'generates AT6400 command codes' for 'Compumotor 6400' motion control hardware devices, could be used to control all Compumotor 6400 motion control hardware devices"). Accordingly, the Court finds that a single driver may be associated with a group of related hardware devices run

by similar control language.

Finally, the '897 specification suggests that drivers must support some core driver functions. In its very first use of the term "software drivers," the Patent states that "[a]ll software drivers implement the core driver functions, while the software drivers need not contain code for implementing the extended driver functions." '897 Patent at [57]. The Patent later repeats this concept. *Id.* at 4:2-5 ("All software drivers must support the core driver functions; the software drivers may also support one or more of the extended driver functions ...."). FANUC's proposed construction correctly attempts to include this concept. RGB, however, argues that FANUC's proposal violates the principles of claim differentiation. RGB's reliance on claim differentiation is misplaced. Claim 2 of the '236 Patent, upon which RGB relies, merely requires that "the software drivers comprise driver code for implementing all of the core driver functions." '236 Patent, 48:11-13 (emphasis added). While the '897 Patent requires that the all drivers support core driver functions, it does not require all drivers to support all core driver

functions. The support for all core driver functions is limited to one dependent claim in the '236 Patent. All drivers, however, must support some core driver functions. A construction embodying this concept both comports with the patent specification and leaves intact the unique requirements of the '236 Patent's second claim.

\*11 For the reasons stated above, this Court finds that "software driver(s)" / "drivers" means "a controller dependent software module that supports some core driver functions and is used to control a hardware device or group of related hardware devices."

5. "motion control operation(s)"

a. Parties' Positions

The parties offer the following constructions for the term "motion control operation(s)," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

-----  
"read and/or  
write  
operation(s)  
used to perform  
motion control"

-----  
"operations used to perform motion control that are not  
specifically related to any particular motion control  
device but are instead abstract operations that all motion  
control device hardware configurations must perform in  
order to function"

RGB considers this yet another key term and believes both parties agree the term means "operations to perform motion control." Dkt. No. 100 at 11-12. The phrase, "motion control," is apparently agreed upon by the parties as "controlled movement" based on its plain meaning. Dkt. No. 100 at 14.

RGB maintains there are "numerous motion control operations used to operate motion control devices," including acquisition of position information, hardware-independent read/write operations, and so on. *Id.* at 8. RGB adds the additional phrase "read and/or write" for clarification purposes and argues it is within the understanding of one of ordinary skill in the art. *Id.* at 12. For those who are unskilled in the art, however, RGB's usage of the words "read" and "write" is best explained through examples of actual reads and actual writes. For instance, RGB points out "the RGB patents explicitly state that the motion control operations can include 'GET POSITION' operations (a 'read' operation) and 'MOVE RELATIVE' operations (a 'write' operation)." *Id.* (citing '897 Patent, 7:3-14, 16:24). RGB also explains motion

control operations may be either primitive or non-primitive. *Id.* at 12-13 (citing '897 Patent, 7:3-21).

In its reply brief, RGB contends FANUC's construction is wrong because not all motion control operations are performed by all motion control devices--only primitive operations are "necessary" for motion control. Dkt. No. 109 at 6.

FANUC maintains the term has no commonly-understood or agreed-upon definition to those of ordinary skill in the art. Dkt. No. 105 at 14-15. Thus, FANUC relies on the specification and argues that the motion control operations must "perform motion control." *Id.* (citing '897 Patent, 6:63-7:2). In addition, FANUC argues that motion control operations are not related to any particular hardware; instead, motion control operations are generic operations performed by motion control devices. *Id.* FANUC also relies on the language of Claim 17 to bolster its construction. '897 Patent, 36:47-53 ("defining a set of motion control operations ...").

\*12 FANUC is opposed to the "read and/or write" in RGB's construction. FANUC argues that read and write operations are associated with the system software, and such operations do not themselves perform motion control per se. Dkt. No. 105 at 15-16. Also, FANUC argues that a "read and/or write" construction imports a limitation without adequate supporting intrinsic evidence into the claims. Dkt. No. 117 at 7.

b. Court's Construction

Two full paragraphs in the '897 specification are devoted to the definition of "motion control operations." Those paragraphs read as follows:

The software system designer develops the software system 22. The software system designer initially defines a set of motion control operations that are used to perform motion control. The motion control operations are not specifically related to any particular motion control device hardware configuration, but are instead abstract operations that all motion control device hardware configurations must perform in order to function.

Motion control operations may either be primitive operations or non-primitive operations. Primitive operations are operations that are necessary for motion control and cannot be simulated using a combination of other motion control operations. Examples of primitive operations include GET POSITION and MOVE RELATIVE, which are necessary for motion control and cannot be emulated using other motion control operations. Non-primitive operations are motion control operations that do not meet the definition of a primitive operations [sic]. Examples of non-primitive operations include CONTOUR MOVE, which may be emulated using a combination of primitive

motion control operations.  
'897 Patent, 6:62-7:14.

FANUC's proposed construction quotes only from the first paragraph. As a result, FANUC's construction does not account the second paragraph at all. The second paragraph make clear that motion control operations may take the form of either read ("GET POSITION") or write ("MOVE RELATIVE") operations. The Court finds, however, that RGB's "read and/or write" proposal is too ambiguous and is potentially unhelpful to the jury. Instead, the Court finds that the inclusion of the specification's examples yields a more concrete construction.

In addition to GET POSITON and MOVE RELATIVE, motion control operations may also take the form of more complicated "non-primitive" operations. Because the Patent suggests that non-primitive operations can be emulated using a series of primitive operations, however, not all motion control devices need perform both primitive and non-primitive operation in order to function. Instead, the Patent suggests that all motion control devices need only utilize primitive operations.

For the reasons stated above, this Court finds that "motion control operation(s)" means "abstract operations (such as GET POSITION, MOVE RELATIVE, or COUNTOR MOVE) used to perform motion control."

6. "primitive operation(s)"

a. Parties' Positions

\*13 The parties offer the following constructions for the term "primitive operation(s)," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

"motion control operation(s), such as GET POSITION and MOVE RELATIVE, necessary for motion control and that cannot be simulated using a combination of other motion control operations"

"motion control operations that are necessary for motion control and cannot be simulated using a combination of other motion control operations"

The two parties propose almost the same construction for the term except that RGB includes examples in its construction in order to clarify what constitutes a "necessary" operation. Dkt. No. 100 at 13. RGB relies on the specification to provide the particular examples of

GET POSITION and MOVE RELATIVE. Id. (citing '897 Patent, 7:4-10, "Examples of primitive operations include GET POSITION and MOVE RELATIVE"). As for the rest of the words in the construction, both RGB and FANUC use the patent's definition verbatim. Dkt. No. 105

at 21 (citing '897 Patent, 7:3-7).

FANUC contends the examples in RGB's proposal are wrong because other embodiments could use operations other than GET POSITION and MOVE RELATIVE. FANUC Hearing Slides at 112 (citing Mercer Decl. i 35). FANUC gives an example that a software system designer could elect to use absolute moves instead of relative moves. Id.

b. Court's Construction

Both parties agree that primitive functions are necessary for motion control and cannot be simulated using a combination of other motion control operations. As discussed above, the Patent provides examples of "primitive operations" and this Court believes that the inclusion of these examples yields a more concrete construction. Although GET POSITION and MOVE RELATIVE are provided in the construction, the Court does not find that primitive operations are limited to the

stated examples. They are merely examples.

For the reasons stated above, this Court finds that "primitive operation(s)" means "motion control operation(s), such as GET POSITION and MOVE RELATIVE, necessary for motion control, which cannot be simulated using a combination of other motion control operations."

7. "driver function(s)"

a. Parties' Positions

The parties offer the following constructions for the term "driver function(s)," which appears in claim 17. Dkt. No. 119-3. The parties agree that "driver functions" relate to "motion control operations," but they dispute the level of detail to incorporate in the rest of the construction.

RGB	FANUC
<p>"a function corresponding to a motion control operation and available on a software driver"</p>	<p>"functions that are not related to a specific hardware configuration and that define parameters necessary to implement motion control operations in a generic sense but do not attach specific values to these parameters"</p>

\*14 RGB argues the specification and the claims both support its construction. Dkt. No. 100 at 25 (citing '897 Patent, 7:18-26, and Claim 17). Further, RGB complains that FANUC tries to limit the term by including a negative limitation rather than a positive one--defining what the term is not, as opposed to what it is. Id.

FANUC argues that the patentee acted as his own lexicographer and thereby restricted the scope of the term. Dkt. No. 105 at 30. Both RGB and FANUC use the same portion of the specification to define the term, but FANUC adopts the exact phrases in the specification. Id. (citing '897 Patent, 7:21-26). In reply, RGB maintains the sentences provided in the specification do not rise to the level of lexicography as alleged by FANUC. Dkt. No. 109 at 14. Also, RGB maintains FANUC'S construction is internally inconsistent with FANUC's definition of "incremental motion steps" and with Claim 2 of the '543 Patent, which recites "each driver function defining one or more incremental motion steps that may be performed by the motion control device." Id. at 13- 14.

b. Court's Construction

The parties cite to the same portion of the specification to support their arguments:

Driver functions may be either core driver functions or extended driver functions. Core driver functions are associated with primitive operations, while extended driver functions are associated with non- primitive operations. As with motion control operations, driver functions are not related to a specific hardware configuration; basically, the driver functions define parameters necessary to implement motion control operations in a generic sense, but do not attach specific values or the like to these parameters.

'897 Patent, 7:18-26.

Both driver functions and motion control operations are "not related to a specific hardware configuration." Id. Thus, in the parlance of the patent, both driver functions and motion control operations are abstractions. See id. at 6:62-7:14. In addition, the Patent clearly associates driver functions with motion control operations--either primitive

or non-primitive. *Id.* at 7:18-21. Finally, the use of the permissive term "basically" in the above-quoted portion of the specification suggests that driver function may define parameters necessary to implement motion control operation, but that they may also serve other non-specified functions.

For the reasons stated above, this Court finds that "driver function(s)" means "abstract functions that are associated with primitive or non-primitive motion control operations and may define parameters necessary to implement such operations."

RGB

FANUC

-----  
"driver functions  
corresponding to primitive  
operations"

-----  
"driver functions associated with primitive  
operations that all software drivers must  
support"

\*15 RGB contends the patent claims themselves make clear that the software driver need only support a subset of the core driver functions. *Dkt. No. 100* at 26. RGB relies on the claim language itself, which states that the drivers are only required to support "at least some of the driver functions." *Id.* (citing 36:65-37:3). FANUC asserts the "core driver functions" must be supported by "all software drivers." *Dkt. No. 105* at 31 (citing '897 Patent, 4:1-5 ("[a]ll software drivers must support the core driver functions; the software drivers may also support one or more of the extended driver function ....")). In reply, RGB argues that this statement is directed to only one preferred embodiment. *Dkt. No. 109* at 14.

b. Court's Construction

The parties seemingly agree that all software drivers must support some subset of the core driver functions, a position which is supported by the specification. See '897 Patent, 4:1-5. This position is also supported by this Court's construction of "software drivers." See *supra* § IV.B.4.b ("software drivers" means "a controller

RGB

FANUC

-----  
"controller dependent code that  
corresponds to one or more  
driver function(s) in a  
software driver and implements  
one or more motion control  
operation(s) "

-----  
"code that dictates how to generate control  
commands for controlling the motion control  
device associated with the driver in  
performing motion control operations  
associated with at least some of the driver  
functions"

RGB relies on its discussion for "software drivers," to

8. "core driver functions"

a. Parties' Positions

The parties offer the following constructions for the term "core driver functions," which appears in claim 25. *Dkt. No. 119-3*. The primary dispute between the parties is whether to include the phrase "that all software drivers must support" in the construction.

dependent software module that supports some core driver functions and is used to control a hardware device or group of related hardware devices"). Because this Court's construction of "software drivers" makes it clear that all software drivers must support some core driver functions, this Court finds it unnecessary to include similar language in the construction of "core driver functions."

For the reasons stated above, this Court finds that "core driver functions" means "driver functions associated with primitive motion control operations."

9. "driver code"

a. Parties' Positions

The parties offer the following constructions for the term "driver code," which appears in claim 17. *Dkt. No. 119-3*.

maintain that driver code is "controller dependent." *Dkt.*

No. 100 at 27. According to RGB, its proposal that driver code "corresponds to one or more driver function(s)" and "implements" one or more motion control operations is consistent with the specification. Id. at 27 (citing '897 Patent, 9:56-10:3 ("driver functions define the parameters necessary to carry out the motion control operations.")). RGB also points out the specification's repeated use of the word "implements" in relation to the driver code and its related functions. Id.

FANUC contends that because the patent uses the term in a very specific and unique manner, the claim construction should reflect this fact. Dkt. No. 105 at 27. The phrases FANUC uses come directly from the specification. Id. (citing '897 Patent, 7:54-57 ("The hardware designer writes driver code that dictates how to generate the control commands for controlling the motion control device associated therewith to perform motion control operations associated with at least some of the driver functions")). In addition, FANUC alleges RGB's construction is incorrect as a technical matter. Id. Among other things, FANUC argues the driver code need not be "controller dependent" because the driver code may be written in any computer language. Id. In reply, RGB argues that the Patent's abstract supports its position: "[a] hardware designer writes driver code that implements the driver functions on a given motion control hardware product." Dkt. No. 109 at 15 (citing '897 Patent at [57] ).

#### b. Court's Construction

\*16 In its first use of the term, the Patent states that a "hardware designer writes driver code that implements the driver functions on a given motion control hardware product." '897 Patent at [57]. Later uses echo the idea that driver code is used to implement motion steps in a motion control device. See id. at 3:56-58. While the term "implement" is relatively vague by itself, the specification provides clarification for the term; driver code aids in the

implementation of motion steps by generating, at least in part, the commands necessary for controlling a motion control device. Id. at 7:53-57. The specification, however, makes it clear that the driver code need not aid in the implementation of all types of motion (i.e. driver functions). Id. at 9:15-20 ("the hardware designer need not develop driver code to implement all of the driver functions ....").

Finally, the Patent does not suggest that there must be a one-to-one relationship between driver code and a motion control device; instead, as this Court has already held in connection with related terms, certain driver related code may be used to control a single hardware device or group of related hardware devices. See supra § IV.B.4.b. In addition, because the Court has already included the idea of "controller dependen[cy]" into the construction of "software driver" and because the two terms appear together in the claims, the Court finds it unnecessary and duplicative to include such a limitation in the construction of "driver code."

For the reasons stated above, this Court finds that "driver code" means "code associated with a hardware device or group of related hardware devices, which helps generate commands necessary to perform motion control operations associated with at least some driver functions."

#### 10. "control command(s)"

##### a. Parties' Positions

The parties offer the following constructions for the term "control command(s)," which appears in claim 17 and 24. Dkt. No. 119-3.

RGB

FANUC

"controller dependent command(s) used to implement motion control operation(s) "

"command codes in the hardware specific command language necessary to control a given motion control device to perform motion control operations"

FANUC contends RGB's construction is vague and overly broad. Dkt. No. 105 at 33. Therefore, FANUC selects phrases from the specification to provide the requisite characteristics of "control commands." Id. (citing '897 Patent, 1:38-41, 7:54-56, 6:19-22, and 12:63-67). In particular, FANUC relies on one portion of the

specification, which states:

[The software driver's] main purpose is to implement functionality that generates motion control commands for the specific hardware supported. For example, the AT6400 driver, used to control the Compumotor AT6400 motion

control hardware, generates AT6400 command codes.

Id. at 33 (citing '897 Patent, 12:63-67).

\*17 RGB concedes certain points in its reply, but argues FANUC's brief is incorrect regarding the statement that "control commands" must be in a "hardware specific command language necessary to control a given motion control device." Dkt. No. 109 at 20. RGB maintains the excerpt that FANUC relies on is merely a preferred embodiment. Id.

In its sur-reply, FANUC contends RGB is wrong to use "controller-dependent," a phrase that is not used in the patents, is of uncertain scope, and may cause jury confusion. Dkt. No. 117 at 12.

#### b. Court's Construction

The parties' central disagreement centers on whether control commands must be "hardware specific." The specification provides an example in which a AT6400 software driver is used to control a Compumotor AT6400 motion control device by generating AT6400 command codes. '897 Patent, 12:63-67. This Court has already

found, however, that once a software driver is developed it may later be used to control devices similar to the one to which it was initially associated. See supra § IV.B.4.b (citing '897 Patent at 32:30-37; Hooper Decl. at i 16). It follows that the control commands generated by a software driver may also be used to control a group of related devices. Thus, inclusion of FANUC's "hardware specific" limitation would be misleading.

For the reasons stated above, this Court finds that "control command(s)" means "command codes in hardware language, which instruct a motion control device to perform motion control operations."

#### 11. "developing a set of software drivers"

##### a. Parties' Positions

The parties offer the following constructions for the term "developing a set of software drivers," which appears in claim 17. Dkt. No. 119-3. The primary dispute between the parties regards the plain meaning of the word "developing."

RGB

FANUC

-----  
"developing a set of software drivers" The term "software drivers" is construed elsewhere, and there is no need to construe the additional language, which should be given its ordinary meaning.

-----  
"a person creates a set of software drivers"

RGB maintains this term does not need construction because it incorporates the phrase "software drivers," which is separately construed. Dkt. No. 100 at 24. According to RGB, because the remaining word in the term, "developing," is well-understood, it need not be construed and its ordinary meaning can apply. Id. In addition, RGB asserts FANUC's construction improperly limits the claim to "a person creates." Id. Although the specification references a person developing software drivers, the claims do not require it. Id. (citing *Resonate Inc. v. Alton Webssystems, Inc.* 338 F.3d 1360, 1364 (Fed.Cir.2003) ("the written description is not a substitute for, nor can it be used to rewrite, the chosen claim language")).

For the terms that require construction of the words "developing," "defining," "providing," and "selecting/selected," FANUC argues the need for an explicit reference to "a person" as opposed to a "computer" conducting the action. Dkt. Nos. 117 at 12;

105 at 38-41. The reason FANUC provides such a construction is that the patents allegedly describe only one embodiment, and the embodiment references a person performing the steps of "developing," "defining," "selecting/selected," or "providing." Dkt. No. 105 at 39. In addition, FANUC argues that the patents do not enable something other than a human user performing these steps. Id. FANUC cites to a part of the specification that emphasizes and explains the role of individuals: "the method described in this section will normally but not necessary involve the labors of at least two and perhaps three separate software programmers: a software system designer, a hardware designer ..." Id. (citing '897 Patent, 6:52-62).

##### b. Court's Construction

\*18 As already stated, limitations from the specification may only be imported into the claims where it is clear that the patentee intended for the "claims and embodiments in

the specification to be strictly coextensive." Phillips, 415 F.3d at 1323. Here, there is no evidence of a coextensive relationship between the claims and the preferred embodiment, which uses a number of persons to fulfill certain steps. The portion of the specification cited by FANUC to support its position that a person is required actually demonstrates that such a limitation is improper:

Initially, it should be noted that, in most situations, the method described in this section will normally but not necessary involve the labors of at least two and perhaps three separate software programmers: a software system designer, a hardware designer ... and a motion control system designer."

'897 Patent, 6:52-62 (emphasis added). As can be seen from the emphasized text, the specification suggests that system designers will only be used "in most situations." Thus, the Patent contemplates embodiments in which designers are not utilized. Accordingly, the Court finds it inappropriate to limit the claims to a preferred embodiment and require that "a person" perform the claimed action.

Furthermore, the Court finds that "developing" and "set" have plain and ordinary meanings. See Phillips, 415 F.3d at 1313, 1316-17 (claim language is given its "ordinary and customary meaning" unless the term has no generally

accepted meaning, the patentee has given the term is specific meaning, or the patentee has disavowed all or part of the scope otherwise encompassed by the ordinary meaning).

Because these findings resolve the dispute between the parties and because "software drivers" has already been construed, the Court finds that this term can be given its ordinary meaning and need not be separately construed. See *O2 Micro Int'l Ltd. v. Beyond Innovation Tech. Co. Ltd.*, 521 F.3d 1351, 1362 (Fed.Cir.2008). The Court therefore finds that the term "developing a set of software drivers" does not require separate construction.

12. "defining a [core/extended] set of [core/extended] driver functions"

a. Parties' Positions

The parties offer the following constructions for the term "defining a [core/extended] set of [core/extended] driver functions," which appears in claim 17. Dkt. No. 119-3. The primary dispute between the parties regards the plain meaning of the word "defining."

RGB

FANUC

-----  
"defining a set of driver functions  
corresponding to [primitive/non-primitive]  
operations"

-----  
"a person describes two or more  
[core/extended] driver  
functions"

The parties' arguments regarding the construction of the term "developing a set of software drivers" are repeated for the construction of the instant term: "defining a [core/extended] set of [core/extended] driver function."

b. Court's Construction

\*19 For the reasons stated above (supra § IV.B.11.b), the Court finds it inappropriate to limit the claims to a preferred embodiment and require that "a person" perform the claimed action. This finding resolves the dispute between the parties. Furthermore, the Court finds that "defining" and "set" have plain and ordinary meanings. Because "driver functions" and "core driver functions" have already been construed, the Court finds that this term can be given its ordinary meaning and need not be

separately construed.

The Court therefore finds that the term "defining a [core/extended] set of [core/extended] driver functions" does not require separate construction.

13. "defining a set of component functions"

a. Parties' Positions

The parties offer the following constructions for the term "defining a set of component functions," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC



-----  
"defining a set of component functions" The term "component functions" is construed elsewhere, and there is no need to construe the additional language, which should be given its ordinary meaning. "a person describes two or more component functions"

The parties' arguments regarding the construction of the term "developing a set of software drivers" are repeated for the construction of the instant term: "defining a set of component functions."

b. Court's Construction

For the reasons stated above (supra § IV.B.11.b), the Court finds it inappropriate to limit the claims to a preferred embodiment and require that "a person" perform the claimed action. This finding resolves the dispute between the parties. Furthermore, the Court finds that "defining" and "set" have plain and ordinary meanings. Because "component functions" has already been construed, the Court finds that this term can be given its

ordinary meaning and need not be separately construed.

The Court therefore finds that the term "defining a set of component functions" does not require separate construction.

14. "defining a set of motion control operations"

a. Parties' Positions

The parties offer the following constructions for the term "defining a set of motion control operations," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

-----  
"defining a set of motion control operations" The term "motion control operations" is construed elsewhere, and there is no need to construe the additional language, which should be given its ordinary meaning. "a person describes two or more motion control operations"

The parties' arguments regarding the construction of the term "developing a set of software drivers" are repeated for the construction of the instant term: "defining a set of motion control operations."

b. Court's Construction

For the reasons stated above (supra § IV.B.11.b), the Court finds it inappropriate to limit the claims to a preferred embodiment and require that "a person" perform the claimed action. This finding resolves the dispute between the parties. Furthermore, the Court finds that "defining" and "set" have plain and ordinary meanings. Because "motion control operations" has already been construed, the Court finds that this term can be given its ordinary meaning and need not be separately construed.

\*20 The Court therefore finds that the term "defining a set of motion control operations" does not require separate construction.

15. "providing component code for each of the component functions"

a. Parties' Positions

The parties offer the following constructions for the term "providing component code for each of the component functions," which appears in claim 17. Dkt. No. 119-3. The primary dispute between the parties regards the plain meaning of the word "providing."

RGB

FANUC

-----  
"making available component code for each of the component functions" "a person supplies component code for implementing each of the component functions"

The parties' arguments regarding the construction of the term "developing a set of software drivers" are repeated for the construction of the instant term: "providing component code for each of the component functions." RGB interprets "providing" as "making available" based on the ordinary meaning of the term. Dkt. No. 100 at 27. For the definition of this term, FANUC supports its position by using a statement in the specification:

The software system designer next defines an application programming interface (API) comprising a set of component functions. For these component functions, the software system designer writes component code that associates at least some of the component functions with at least some of the driver functions.

Dkt. No. 105 at 40 (citing '897, 7:29-34 (emphasis added)).

b. Court's Construction

For the reasons stated above (supra § IV.B.11.b), the Court finds it inappropriate to limit the claims to a

preferred embodiment and require that "a person" perform the claimed action. This finding resolves the dispute between the parties. Furthermore, the Court finds that "providing" has a plain and ordinary meaning. Because "component code" and "component functions" have already been construed, the Court finds that this term can be given its ordinary meaning and need not be separately construed.

The Court therefore finds that the term "providing component code for each of the component functions" does not require separate construction.

16. "providing response stream code"

a. Parties' Positions

The parties offer the following constructions for the term "providing response stream code," which appears in claim 25. Dkt. No. 119-3.

RGB

FANUC

-----  
"making available software code  
that defines the specific  
protocol for receiving response  
data"

-----  
"a person supplies response stream code for  
each of the control command destinations  
that generate response stream data"

The parties' arguments regarding the construction of the term "developing a set of software drivers" are repeated for the construction of the instant term: "providing response stream code." RGB interprets "providing" as "making available" based on the ordinary meaning of the term. Dkt. No. 100 at 27. For the definition of this term, FANUC supports its position by using a statement in the specification:

\*21 Data transmitted from a control command destination back to the system 22 will be referred to as response data. The software system designer thus further writes data response stream code for each of the streams 28a, 28b, and 28c that determines how response data is transmitted from the controllers 16 to the system 22. The system 22 thus processes the response data sent by the controllers 16 based on the data response stream code contained in the streams 28.

Dkt. No. 105 at 40 (citing '897, 8:59-67 (emphasis added)).

b. Court's Construction

For the reasons stated above (supra § IV.B.11.b), the Court finds it inappropriate to limit the claims to a preferred embodiment and require that "a person" perform the claimed action. This finding resolves the dispute between the parties. Furthermore, the Court finds that "providing" has a plain and ordinary meaning. Finally, the Court finds that RGB's construction of "response stream code" comports with the specification's use of the term. See '897 Patent, 8:61-65.

The Court therefore finds that the term "providing response stream code" means "providing software code that defines the specific protocol for receiving response data."

17. "selecting at least one of the destinations" ("of control commands")

a. Parties' Positions

The parties offer the following constructions for the term "selecting at least one of the destinations" ("of control

commands")," which appears in claim 24. Dkt. No. 119-3.

RGB

FANUC

-----  
"choosing at least one of the destinations of control commands at run time"      "a user selects a destination of control commands"

The two parties discuss the following terms collectively because they all share some form of the word "select":

- "a selected destination of control commands,"
- "a selected software driver,"
- "selected from a plurality of software drivers,"
- "selecting a software driver," and
- "selecting at least one of the destinations [of control commands]."

These limitations present three claim-construction issues: (1) whether a "user" must perform the selection; (2) whether the limitations should be construed to require choosing the driver, control command, or device at "run-time"; and (3) whether the word "a" in some of these limitations means "single."

Regarding whether to recite a "user" conducting the step of "selecting," RGB argues that none of the claims refer to a "user," thus leaving open the possibility of a machine doing the selecting. Dkt. No. 100 at 34-35. In addition, RGB argues the examples of a "user" in the patent specification are but examples and should not be imported to limit the claims. Id. at 35.

For the definition of the terms related to "select," FANUC supports its position by using a statement in the specification:

Once the application program 26 has been written and the software drivers 30 have been provided, the user 24 selects at least one motion control device from the group of supported motion control devices 20a, 20b, and 20e. Using a driver administrator module 32, the user 24 then selects the software driver associated with the selected motion control device.

\*22 Dkt. No. 105 at 36 (citing '897 Patent, 8:18-30). While FANUC lists only two examples from the specification, FANUC also relies on the declaration of Dr. Mercer, who states that the patents contain no embodiment in which the selection is made without a human user. Id. at 36 (citing Mercer Decl. i 89). Thus, FANUC argues the word "selecting/selected" should be construed to explicitly include a person. Dkt. No. 117 at 12 (citing *Ormco Corp. v. Align Tech., Inc.*, 498 F.3d

1307, 1312-16 (Fed.Cir.2007)).

RGB urges the Court to construe the "run-time" issue in the "selecting" terms consistent with the disclosure and goals of the patent. Dkt. No. 100 at 34. RGB contends the patents would make no sense if "run-time" were not included in the definition because one of the goals of the invention is to allow a single software system to apply to different motion control devices. Id. The patented system does not need to be recompiled each time a different motion control device is operated because the system uses a "motion control component" and driver modules to communicate with different motion control devices. Id.

FANUC argues that the specification makes clear that these selections can occur during the configuration period prior to run-time--the time the application program begins to execute. Dkt. No. 105 at 36. Relying on the specification and on its expert's declaration, FANUC provides examples where the selections can occur at two different times--(1) before the application program is executed during system configuration (by using the Driver Administrator CPL applet), or (2) after execution of the application program has begun. Id. (citing '897 Patent, 8:22-24; Mercer Decl. at ii 27, 89).

In reply, RGB points out a discrepancy between RGB's and FANUC's meaning of "at run-time." Dkt. No. 109 at 18-19. RGB defines "run-time" as a time period after the source code is compiled into object code. Dkt. No. 100 at 33. "Source code is a set of instructions that can be read by a person but not a machine; source code must be converted into object code ("compiled") before a computer can use it." Id. In contrast, FANUC's definition for "run-time" is narrower, restricting the word to mean a time period after the execution of the application program has begun. Dkt. No. 105 at 36. Due to the differences in definition, if the word "run-time" were included in the construction, this word itself requires construction. Dkt. No. 109 at 19.

RGB, however, offers to change the words "at run-time" to "after compilation time." Id. Although RGB does not state so explicitly, RGB's meaning of "after compilation

time" is presumably similar to the definition it initially provided for "at run-time." FANUC rejects this construction as well, stating that "compile" appears nowhere in the patents. Dkt. No. 117 at 13. FANUC asserts there is no temporal limitation as to when the selecting must occur either in the specification or the claims, and the Court should decline to import such a limitation into the claims. Id.

\*23 Finally, the parties disagree about whether the word "a" means "single" or "one." Only certain terms related to "selecting," however, expressly contain the word "a" and need construction. For such terms, RGB maintains that all the patent claims were drafted using the open form of the transition phrase "comprising." Dkt. No. 100 at 35. In such a situation, RGB cites case law that interprets "a" or "an" to mean "one or more." Id. (citing *Baldwin Graphic Sys., Inc. v. Siebert, Inc.*, 512 F.3d 1313, 1342 (Fed.Cir.2008)). FANUC counters that certain claims contain the word "one" expressly and claim construction should not ignore the language referencing "one" surrounding a particular phrase in the same claim. Dkt. No. 105 at 38 (citing '897 Patent at 37:4-5 ("Selecting one motion control device from the group of supported motion control devices" (emphasis added))).

#### b. Court's Construction

First, the Court finds that the claims do not require that a "user" perform the required selection. As already stated, limitations from the specification may only be imported into the claims where it is clear that the patentee intended for the "claims and embodiments in the specification to be strictly coextensive." *Phillips*, 415 F.3d at 1323. Although FANUC has provided citations to the specification in which a "user" performs the selection, FANUC provides and this Court finds no evidence that the patentee meant for this preferred embodiment and the claims to be coextensive. The Court finds that a "user" limitation from the preferred embodiment should not be imported into any claim language.

Second, the Court is not persuaded by RGB's arguments that the claimed selections must occur at "run-time" or after "compilation time." Instead, the specification suggests that, in an "exemplary system," the selection of one or more control command destinations may occur before the program is run. '897, 8:42-57 ("Using the driver administrator 32, the user 24 selects one or more of the control command destinations 16 and 34, and, later when run, the system 22 transfers the control commands

to the selected control command destination ...."). Having reviewed the entire specification, the Court finds no reason to include a temporal limitation into the selected claims. Such a limitation might, in fact, improperly read a preferred embodiment out of the claims. See *Vitronics*, 90 F.3d at 1583 (An interpretation of a claim, which would not include a preferred embodiment disclosed in the specification, is rarely, if ever, correct.).

Finally, the Court finds that none of the "selected/selecting" limitations to be construed should be to a "single" selected item. In those limitations employing the indefinite article "a," the Court finds that the claim could encompass "one or more." The Federal Circuit "has repeatedly emphasized that an indefinite article 'a' or 'an' in patent parlance carries the meaning of 'one or more' in open-ended claims containing the transitional phrase 'comprising.'" *Baldwin Graphic Systems, Inc. v. Siebert, Inc.*, 512 F.3d 1338, 1342 (Fed.Cir.2008).

\*24 For the reasons stated above, this Court finds that "selected/selecting" limitations should not be construed to include a "user" limitation or a temporal limitation. In addition, the Court finds that in the open-ended claims of the Patent, the "selected/selecting" limitation should be construed to allow for the selection of "one or more" items. Because the remaining limitations in these claims ("software driver" or "control command") have already been construed, the Court finds that these terms can generally be given their ordinary meaning and need not be separately construed. Where, however, a "selected/selecting" term uses the indefinite article, the Court finds that the article should be construed as "one or more." Accordingly, the Court finds that the phrase "selecting at least one of the destinations" should not be modified beyond its plain and ordinary meaning. In contrast, the phrase "selecting a software driver" means "selecting one or more software drivers."

18. "selecting from the set of software drivers the software driver developed for the selected motion control device"

#### a. Parties' Positions

The parties offer the following constructions for the term "selecting from the set of software drivers the software driver developed for the selected motion control device," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

"choosing from the set of software drivers the driver that corresponds to the selected motion control device at run time"

"a user selects from the set of software drivers the single software driver developed for the selected motion control device"

The parties' arguments regarding the construction of the term "selecting at least one of the destinations" are repeated for the construction of the instant term: "selecting from the set of software drivers the software driver developed for the selected motion control device."

be given its ordinary meaning and need not be separately construed. The Court therefore finds that the term "selecting from the set of software drivers the software driver developed for the selected motion control device" does not require separate construction.

b. Court's Construction

19. "selecting one motion control device"

For the reasons stated above (supra § IV.B.17.b), the Court finds that this limitation should not be construed to include a "user" limitation or a temporal limitation. This finding resolves the dispute between the parties. Because the remaining limitations in this claim ("software driver" and "motion control device") have already been construed or are not in contention, the Court finds that this term can

a. Parties' Positions

The parties offer the following constructions for the term "selecting one motion control device," which appears in claim 17. Dkt. No. 119-3.

RGB

FANUC

-----  
"choosing a motion control device at run time"

-----  
"a user selects one motion control device"

\*25 The parties' arguments regarding the construction of the term "selecting one motion control device" are repeated for the construction of the instant term: "selecting from the set of software drivers the software driver developed for the selected motion control device."

to communicate with any one of a group of supported hardware devices, the system comprising:

b. Court's Construction

For the reasons stated above (supra § IV.B.17.b), the Court finds that this limitation should not be construed to include a "user" limitation or a temporal limitation. Moreover, in the context of an open-ended comprising claim, the Court finds that "one" may also mean "one or more." Cf. *In re Crish*, 393 F.3d 1253, 1257 (Fed.Cir.2004). Accordingly, the Court finds that "selecting one motion control device" means "selecting one or more motion control devices."

a software system operating on at least one workstation, the software system comprising at least one application program comprising a set of component functions defining a desired motion sequence, the desired motion sequence being comprised of primitive operations that are necessary to define the desired motion sequence and non-primitive operations that may be simulated using a combination of primitive operations,

a core set of core driver functions, where each core driver function is associated with one of the primitive operations,

an extended set of extended driver functions, where each extended driver functions is associated with one of the non-primitive operations,

component code associated with each of the component functions, where the component code associates at least some of the component functions with at least some of the driver functions,

a set of software drivers, where each software driver is associated with one of the hardware

V. U.S. Patent No. 6,513,058

A. Overview

RGB has asserted claim 1 of the '058 Patent against FANUC in this lawsuit. Dkt. No. 119. For reference, asserted claim 1 is reproduced below (terms to be construed emphasized). Only the terms not already construed above are considered in this section.

1. A system for allowing an application program

devices and comprises driver code for implementing the driver functions, and a control command generating module for generating control commands based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the software drivers; and a network communication protocol that allows the control commands to be communicated from the control command generating module on the at least one workstation to at least one of the supported hardware devices over a network.

'058 Patent, 49:50-50:19 (emphasis added).

RGB

-----  
"interconnected computing devices with a protocol for addressing and communicating with each other"

\***26** FANUC asserts the word "network" appears only in the '058 patent claims and not in the rest of the specification. FANUC Hearing Slides at 122. FANUC therefore relies on extrinsic evidence, particularly expert testimony and dictionary definitions. *Id.* at 122-31. FANUC's dictionaries provide examples of three or more elements interconnected, but they do not expressly state the interconnection must be among three or more. *Id.* In contrast, FANUC's expert testimony provides that one of ordinary skill in the art in May 1995 would not consider that two hardware devices (e.g. computer and printer) connected together in a point to point configuration would constitute a network. *Id.* (citing Mercer Decl. i 29); Dkt. No. 105 at 42 (citing Mercer Decl. i 90).

RGB also relies on external evidence--dictionaries--to support its construction. Dkt. No. 109 at 20; RGB Hearing Slides at 164 (citing IBM DICTIONARY OF COMPUTING 454 (1994), BUSINESS DICTIONARY OF COMPUTERS 233 (1993), and JARGON 362 (1993)). RGB contends that the extrinsic evidence does not require three or more devices. Dkt. No. 100 at 37. As for intrinsic evidence, RGB argues FANUC's construction is contrary to the Patents' specifications, which discloses a computer connected to a motion control hardware device via a PC Bus or Serial port--implying only two devices. *Id.* (citing '897 Patent, 6:21-25, 17:56-18:12, 32:41-45, 32:52-53, 33:6-17).

#### b. Court's Construction

The Court finds no support for FANUC's contention that

#### B. Claim Construction

##### 1. "network"

###### a. Parties' Positions

The parties offer the following constructions for the term "network," which appears in claim 1. Dkt. No. 119-3. The primary dispute between the parties regards the number of elements needed to constitute a network.

FANUC

-----  
"an interconnection of three or more communicating entities"

a network must contain "three or more" entities. A person of ordinary skill in the art at the time of invention would have been familiar with "networks" connecting two computers (e.g. peer-to-peer networks). See Dkt. No. 100, Ex. 18 at 362. Moreover, the Patents itself uses the term network to refer to a connection between two entities. '058 Patent at [57] ("The network communication protocol allows the control commands to be communicated from the control command generating module to at least one of the supported hardware devices over the network."). FANUC's proposed construction would thus improperly exclude a preferred embodiment. Finally, because the patent separately references "network communication protocol," the Court finds it unnecessarily duplicative to refer to protocols in its construction of "network."

For the reasons stated above, this Court finds that "network" means "interconnected computing devices."

2. "a control command generating module for generating control commands based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the software drivers"

###### a. Parties' Positions

The parties offer the following constructions for the term "a control command generating module for generating control commands based on the component functions of the application program, the component code associated

with the component functions, and the driver code associated with the software drivers," which appears in claim 1. Dkt. No. 119-3. The primary dispute between the parties is whether the term should be construed under 35

U.S.C. § 112, i 6.

RGB

FANUC

-----  
This term is not governed by 35 U.S.C. § 112 i 6. Nevertheless, in compliance with PR 4-2, Plaintiff hereby identifies the following structure: "a software module within a software driver that contains the code responsible for generating control commands, and equivalents thereof"

-----  
"This limitation is a means plus function limitation. Its corresponding structure is the driver 30 as described at Cols. 13:37-14:6, the interfaces illustrated in Fig. 21, and the algorithm defined at 16:28-63 and illustrated by Fig. 19."

**\*27** The phrases "control command," "component function," "component code," "application program," "driver code," and "software drivers" are individually construed.

thereof, and such a claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof."

First, RGB maintains that the term does not include "means" language and a presumption that the term is not means-plus-function must be overcome. Dkt. No. 100 at 36 (citing *Lighting World, Inc. v. Birchwood Lighting, Inc.*, 382 F.3d 1354, 1359-60 (Fed.Cir.2004)). RGB also contends the claim discloses sufficient structure--a "module" or a "workstation" where the module resides--and is not governed by § 112, i 6. Id. Finally, RGB points to the fact that FANUC filed an Ex Parte Reexamination with no mention that the term should be considered a means-plus-function term--unlike some other terms that FANUC alleged constitute means-plus-function terms. RGB Hearing Slides at 170.

The Court must decide as a matter of law whether a particular term or phrase is governed by § 112, i 6. See *Phillips*, 415 F.3d at 1212; *Lighting World*, 382 F.3d at 1358 ("The task of determining whether the limitation in question should be regarded as a means-plus-function limitation, like all claim construction issues, is a question of law for the court, even though it is a question on which evidence from experts may be relevant."). When a limitation contains "means" language a presumption of § 112, i 6 applicability arises. *Enviroco Corp. v. Clestra Cleanroom, Inc.*, 209 F.3d 1360, 1365 (Fed.Cir.2000). Conversely, when "means" language is absent there is a presumption that § 112, i 6 does not apply. *Lighting World*, 382 F.3d at 1358. A limitation lacking "means" language may overcome the presumption against § 112, i 6 treatment if the claim term fails to recite sufficiently definite structure or else recites function without reciting sufficient structure for performing that function. Id.; *MIT and Elects. for Imaging, Inc. v. Abacus Software, Inc.*, 462 F.3d 1344, 1353 (Fed.Cir.2006).

In response, FANUC argues that the term lacks sufficiently defined structure even though it does not expressly use "means" language. Dkt. No. 105 at 43; FANUC Hearing Slides at 140 (citing *MIT v. Abacus Software*, 462 F.3d 1344, 1353 (Fed.Cir.2006); *Mas-Hamilton Group v. LaGard, Inc.*, 156 F.3d 1206, 1214 (Fed.Cir.1998)). To support its contention that there is a lack of structure, FANUC notes the limitation connotes nothing more than generic software; for example, both "component" and "module" were defined as "software," as are many of the other terms that require construction. Dkt. No. 105 at 43; FANUC Hearing Slides at 141.

**\*28** The Court finds the claim recites sufficient structure for performing the recited function. The Federal Circuit has made it clear that "the presumption flowing from the absence of the term 'means' is a strong one that is not readily overcome." *Lighting World*, 382 F.3d at 1358. Moreover, in determining whether the claim term recites sufficient structure "it is sufficient if the claim term is used in common parlance or by persons of skill in the pertinent art to designate structure, even if the term covers a broad class of structure and even if the term identifies structure by their function." Id. at 1359-60. The Court finds that, in the parlance of software design, a "command generating module" sufficiently designates structure. In

#### b. Court's Construction

35 U.S.C. § 112, i 6 (" § 112, i 6") provides: "An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support

addition, the recited function contains sufficient structure to identify structure by function. Therefore, the Court finds that § 112, i 6 treatment is inappropriate.

Because this finding resolves the dispute between the parties and because the remaining limitations in the term have already been construed or are not in contention, the Court finds that this term can be given its ordinary meaning and need not be separately construed. See *O2 Micro*, 521 F.3d at 1362. The Court therefore finds that the term "a control command generating module for generating control commands based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the software drivers" does not require separate construction.

## VI. U.S. Patent No. 6,516,236

### A. Overview

RGB has asserted claims 1, 3, 8, and of the '236 Patent against FANUC in this lawsuit. Dkt. No. 119. For reference, claims 1, 2, 3, 8, and 9 are reproduced below (terms to be construed emphasized). Only the terms not already construed above are considered in this section:

1. A system for generating a sequence of control commands for controlling a selected motion control device selected from a group of supported motion control devices, comprising:

a set of motion control operations, where each motion control operation is either a primitive operation the implementation of which is required to operate motion control devices and cannot be simulated using other motion control operations or a non-primitive operation that does not meet the definition of a primitive operation;

a core set of core driver functions, where each core driver function is associated with one of the primitive operations;

an extended set of extended driver functions, where each extended driver function is associated with one of the non-primitive operations;

a set of component functions;

component code associated with each of the component functions, where the component code associates at least some of the component functions with at least some of the driver functions;

a set of software drivers, where

each software driver is associated with one motion control device in the group of supported motion control devices,

\*29 each software driver comprises driver code for implementing the motion control operations

associated with at least some of the driver functions, and

one of the software drivers in the set of software drivers is a selected software driver, where the selected software driver is the software driver associated with the selected motion control device;

an application program comprising a series of component functions, where the application program defines the steps for operating motion control devices in a desired manner; and

a motion control component for generating the sequence of control commands for controlling the selected motion control device based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the selected software driver.

2. A system as recited in claim 1, in which the software drivers comprise driver code for implementing all of the core driver functions.

3. A system as recited in claim 2, in which the software drivers comprise driver code for implementing at least some of the extended driver functions.

8. A system as recited in claim 1, further comprising:

a plurality of destinations of control commands, with one of the plurality of destinations of control commands being a selected destination of control commands;

a plurality of streams, where each stream contains transmit stream code that determines how the control commands are to be transferred to at least one of the plurality of destinations of control commands; and

stream control means for communicating the control commands to the selected destination of control commands based on the transmit stream code contained by the stream associated with the selected destination of control commands.

9. A system as recited in claim 8, in which certain of the destinations of control commands generate response data, wherein:

the streams associated with the destinations of control commands that generate response data are each associated with response stream code; and

the stream control means processes the response data based on the response stream code.

'236 patent, 48:11-49:16, 50:8-29 (emphasis added).

### B. Claim Construction

1. "a selected destination of control commands"



a. Parties' Positions

"a selected destination of control commands," which appears in claim 8. Dkt. No. 119-3.

The parties offer the following constructions for the term

RGB

FANUC

-----  
"a recipient, chosen at run time, of control commands"

-----  
"the destination of control commands selected by a user"

The parties' arguments regarding the construction of the term "selecting at least one of the destinations" are repeated for the construction of the instant term: "a selected destination of control commands."

not be separately construed. The Court therefore finds that the term "a selected destination of control commands" does not require separate construction.

2. "a selected software driver"

b. Court's Construction

a. Parties' Positions

For the reasons stated above (supra § IV.B.17.b), the Court finds that this limitation should not be construed to include a "user" limitation or a temporal limitation. This finding resolves the dispute between the parties. Because the remaining limitation in this claim ("control commands") has already been construed, the Court finds that this term can be given its ordinary meaning and need

\*30 The parties offer the following constructions for the term "a selected software driver," which appears in claim 1. Dkt. No. 119-3.

RGB

FANUC

-----  
"a software driver chosen at run time"

-----  
"a single software driver selected by a user from a plurality of software drivers"

The parties' arguments regarding the construction of the term "selecting at least one of the destinations" are repeated for the construction of the instant term: "a selected software driver."

is whether the term is a means-plus-function limitation. The term is part of a longer claim limitation beginning with the words "motion control component for ...." Rather than construe this term separately, the Court will consider this term in the context of the larger phrase of which it is a part.

b. Court's Construction

For the reasons stated above (supra § IV.B.17.b), the Court finds that this limitation should not be construed to include a "user" limitation or a temporal limitation. In addition, the Court finds that the indefinite article should be construed as "one or more." See Baldwin, 512 F.3d at 1342. Accordingly, the Court finds that "a selected software driver" means "one or more selected software drivers."

4. "a motion control component for generating the sequence of control commands for controlling the selected motion control device based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the selected software driver"

a. Parties' Positions

3. "motion control component"

The parties offer competing constructions for the term "motion control component," which appears in claim 1. Dkt. No. 119-3. The primary dispute between the parties

The parties offer the following constructions for the term "a motion control component for generating the sequence of control commands for controlling the selected motion control device based on the component functions of the application program, the component code associated with

the component functions, and the driver code associated with the selected software driver," which appears in claim 1. Dkt. No. 119-3. The primary dispute between the parties is whether the term should be construed under 35

U.S.C. § 112, i 6.

RGB

FANUC

"motion control component associated with the production of a sequence of control commands using the component functions of the application program, the component code of the motion control component, and the driver code associated with the selected software driver" This term is not governed by 35 U.S.C. § 112 i 6. It is construed as described above. Nevertheless, if it is construed as governed by 35 U.S.C. § 112 i 6, then the corresponding structure is: "a binary software module comprising component code wherein the component code associates component functions with driver functions by calling the driver functions, and equivalents thereof"

"This limitation is a means plus function limitation. Its corresponding structure is the driver 30 as described at Cols. 13:37-14:6, the interfaces illustrated in Fig. 21, and the algorithm defined at 16:28-63 and illustrated by Fig. 19."

\*31 RGB argues against FANUC's construction by pointing out the construction is contrary to the specification, which references "motion control component 35" and not "driver 30." Dkt. No. 100 at 16 (citing '236 Patent, 9:25-33); RGB Hearing Slides at 18 (citing '897 Patent, 10:36-38, 13:4-6 ("When an application, using the system 22, is run, the component 35 communicates with the driver 30 directing it to carry out the appropriate motion control operations.")).

RGB contends the "motion control component" is structural; thus, the term is not a means-plus-function limitation. Dkt. No. 100 at 17. If the Court, however, holds the claim to be means-plus-function, then RGB cites "Fig. 11" and the associated text in the specification to further refute FANUC's proposal and to support its own proposed structure. Id. (citing '897 Patent, 13- 21-27 ("Referring now to Fig. 11, the module interaction-map displays all binary modules and their interactions with the driver 30. There are two modules that interact directly with the driver: the motion control component 35 and the driver administrator 32 ... the component 35 directs the driver to carry out motion control operations ....")).

FANUC argues the term not only lacks sufficient structure, but that RGB belied its own (RGB's) extrinsic evidence by defining "component" and "module" as "part of a software system having a function." Dkt. No. 105 at 43 (citing Finn Decl. at ii 12-15; Exs. I-L). FANUC maintains that its construction is consistent with the specification. Dkt. No. 105 at 43-44 (citing '897 Patent, 13:37-14:6, 16:28-63, Figs. 19 and 21). FANUC argues

that that "the only structure (algorithm) identified in the specification which generates control commands is the software driver module 30." Id. (citing Mercer Decl. at ii 102-103).

#### b. Court's Construction

The specification defines the motion control component with a large degree of specificity. '236 Patent, 10:39-13:15. The motion control component is "used by every application programming the system 22 to perform motion control operations." Id. 10:40-42. "When operating, the motion control component 35 interacts with the driver administrator 32, to get the current driver, and the driver 30 and driver stub 36, to carry out motion control operations." Id. at 10:44-47. More particularly, the motion control component is software--a binary module--that coordinates all interactions between the driver administrator 32, driver 30, and driver stub 36. Id. at 10:59-65.

Thus, the Court finds no basis for FANUC's proposed construction, which would structurally equate the motion control component 35 with driver 30. Instead, after considering the intrinsic evidence, the Court finds that RGB's proposed construction of "motion control component" as "a binary software module that associates component functions with driver functions by calling the driver functions" is consistent with the Patent specification. Once "motion control component" is so defined, the Court finds that the term sufficiently designates structure. Therefore, the Court finds that §

112, i 6 treatment of the instant term is inappropriate.

**\*32** For the reasons stated above, this Court finds that "motion control component" means "a binary software module that associates component functions with driver functions by calling the driver functions."

With regard to the larger term "a motion control component for generating the sequence of control commands for controlling the selected motion control device based on the component functions of the application program, the component code associated with the component functions, and the driver code associated with the selected software driver" the Court finds § 112, i 6 inapplicable. Because this finding resolves the dispute between the parties and because the remaining limitations in the term have already been construed or are not in contention, the Court finds that the term can be given its ordinary meaning and need not be separately construed.

5. "stream control means for communicating the control commands to the selected destination of control

commands based on the transmit stream code contained by the stream associated with the selected destination of control commands"

#### a. Parties' Positions

The parties agree that this term is subject to § 112, i 6 and further agree that the function for the term is "communicating the control commands to the selected destination of control commands based on the transmit stream code contained by the stream associated with the selected destination of control commands." Dkt. No. 109 at 15. In addition, the parties agree that in a computer-implemented means-plus-function term the corresponding structure is limited to the algorithm disclosed in the specification. RGB Hearing Slides at 172-73; Dkt. No. 105 at 42-43; Dkt. No. 109 at 15-16. The parties, however, disagree as to nature of the algorithm disclosed and make the following proposals.

RGB

FANUC

-----  
"the combination of transmit stream code, response stream code, and a computer, and equivalents thereof"; or in the alternative, "the combination of transmit stream code and response stream code responsible for storing data and controlling low level data transfer operations using, for example, the RS-232 or PC Bus protocol, and a computer, and equivalents thereof."

-----  
"[T]he objects, algorithms and interfaces described in the '897 Patent at Cols. 13:46-52; 14:64-16:15; 17:3-13; 17:39-20:45; 25:13-56; 26:6-31; 26:61-27:41; 32:12-33:49; and illustrated in figures 12, 14-29, 32, 40-41, 44, 46, 48-49, 51-52."

RGB argues that FANUC's proposal of 25 figures and over 500 lines in the specification is overly detailed and inconsistent with the law. Dkt. No. 109 at 15-16. RGB contends that the law does not require the inclusion of every single equation and step used in the algorithm. *Id.* (citing *Harris Corp. v. Ericsson Inc.*, 417 F.3d 1241, 1254 (Fed.Cir.2005)). In addition to the required algorithm, RGB argues that FANUC also improperly includes objects and interfaces in its proposed structure. *Id.*

**\*33** In response, FANUC argues that RGB's proposed structure does not identify an algorithm at all. Dkt. No. 105 at 42-43. Instead, RGB's proposed structure merely lists a combination of other elements already required by other limitations in claims 8 and 9. FANUC Hearing Slides at 135-37. FANUC contends its proposed structure includes all the required steps disclosed in the patent specification as legally required. Dkt. No. 105 at 42-43.

#### b. Court's Construction

In construing a § 112, i 6 claim, the recited function must first be identified. *ACTV, Inc. v. Walt Disney Co.*, 346 F.3d 1082, 1087 (Fed.Cir.2003). "The statute does not permit limitation of a means-plus- function claim by adopting a function different from that explicitly recited in the claim." *Micro Chem., Inc. v. Great Plains Chem. Co.*, 194 F.3d 1250, 1258 (Fed.Cir.1999)

Application of § 112, i 6 then requires "identification of the structure in the specification which performs the recited function." *Id.* at 1257. Federal Circuit precedent restricts computer-implemented means-plus-functions terms to the specific algorithm disclosed in the specification and the equivalents thereof. *Harris*, 417 F.3d at 1253 (citing *WMS Gaming, Inc. v. Int'l Game Tech.*,

184 F.3d 1339, 1348-49 (Fed.Cir.1999)). Section 112, i 6, however, does not permit "incorporation of structure from the written description beyond that necessary to perform the claimed function." *Micro Chem.*, 194 F.3d at 1257 (emphasis added).

The claim presently at issue deals with the process of communicating with a motion control device through a stream. Before such communication can take place, the stream should be registered and initialized, a process which the specification describes in detail. See '236 Patent, 19:9-49; 20:9-32. Once the stream has been initialized, it must then be opened. *Id.* at 20:33-49. Once the stream has been opened, "it is ready to perform data transport operations." *Id.* at 20:50-151. The communication involved in the instant term is a data transport operation and the specification details how such operations may be carried out:

After opening a stream, it is ready to perform data transport operations. There are two main data transport operations available: Reading data, and writing data. FIG. 30 describes the process of writing data to the stream. When writing to the stream, the following steps occur. First the driver directs the stream to write data to the target and passes the data to the stream. Next, the stream passes the data to the CStreamDisp object. The CStreamDisp object passes the block of data to the CIOMgr and directs it to write it to the target. The CIOMgr object either passes the complete block of data to the CIOHAL object, or stores the block in an internal buffer and then passes pieces of the buffer to the CIOHAL object until the complete buffer is sent. The CIOHAL object takes the data passed to it and either sends it directly to the target, passes it to a device driver, or calls COMM API to send the data to the Serial 10 port. The device driver or COMM API sends the data directly to the hardware controlled.

\*34 Certain streams, like the PC Bus and Serial 10 streams, return data after write operations occur on them. The data returned may be specific to a previous request for data, or status describing the success or failure of the previous write operation. FIG. 31 describes the process of reading data from the stream. It should be noted that not all streams are readable. Currently, the only readable streams are the PC Bus and Serial streams. During the operation of reading data from the target, the following steps occur.

First the driver directs the stream to read data from the target. The stream passes the call on to the CStreamDisp object. The CStreamDisp object directs the CIOMgr to perform the read.

Depending on how the stream is implemented, the CIOMgr may either make one call or multiple calls to the CIOHAL object. If multiple calls are made, all data read is stored in CIOMgr internal buffers. The CIOHAL object either directly communicates to the hardware, uses the COMM API, or a device driver to read the data. If a device driver or the COMM API are used, they directly communicate with the hardware to read the data.

*Id.* at 20:50-21-20.

The Court finds that these paragraphs outline one algorithm that may be used to perform the claimed communication, which is a data transport operation. Furthermore, the Court finds that the algorithm described above is merely a preferred embodiment and other equivalent algorithms may be used to accomplish the claimed function.

While the stream must first be registered, initialized, and opened, such actions are ancillary to the claimed function. The only algorithm necessary for performing the limited function of the instant term is the "CS treamDisp-->CIOMgr-->CIOHAL-->COMM API/Target/Device Driver" algorithm outlined above. Thus, the Court limits the claimed structure to this read/write algorithm and its equivalents.

For the reasons stated above, this Court construes the term "stream control means for communicating the control commands to the selected destination of control commands based on the transmit stream code contained by the stream associated with the selected destination of control commands" according to § 112, i 6 wherein the claimed function is "the control commands to the selected destination of control commands based on the transmit stream code contained by the stream associated with the selected destination of control commands" and the claimed structure is "software code responsible for sending and retrieving data to and from a specific destination--exemplified by the read and write algorithms at '236, 20:50-21-20, and their equivalents."

6. "the stream control means processes the response data based on the response stream code"

a. Parties' Positions

The parties offer the following constructions for the term "the stream control means processes the response data based on the response stream code," which appears in claim 9. Dkt. No. 119-3.

RGB

FANUC

"the stream control means [defined above] is capable of processing the response data using the response stream code"

"This limitation is a means plus function limitation. Its corresponding structures are the structures identified [above], and the objects, algorithms and interfaces described in the '897 Patent at Cols. 8:58-67; 13:65-14:4; 16:14-19; 16:36-40; 16:47-63; 20:46-21:19; and illustrated in figures 31 and 33."

**\*35** Both parties provide the same legal arguments as they did with the previous "stream control means" term.

#### b. Court's Construction

The Court finds that this term should be construed according to § 112, i 6. Furthermore, the Court finds that the claimed function is "processing the response data based on the response stream code." The Court finds that this function involves communicating with a motion control device through a stream (i.e. a data transport operation). For the reasons discussed above (supra § VI.B.5.b), the Court finds that the structure/algorithm necessary to perform the claimed function are the read/write algorithms outlined above.

Accordingly, this Court construes the term "the stream control means processes the response data based on the response stream code" according to § 112, i 6 wherein the claimed function is "processing the response data based on the response stream code" and the claimed structure is "software code responsible for sending and retrieving data to and from a specific destination--exemplified by the read and write algorithms at '236, 20:50-21-20, and their equivalents."

### VII. U.S. Patent No. 6,941,543

#### A. Overview

RGB has asserted claims 4, 5, 9, and 11 of the '543 Patent against FANUC in this lawsuit. Dkt. No. 119. For reference, claims 1-9, and 11 are reproduced below (terms to be construed emphasized). Only the terms not already construed above are considered in this section.

1. A method of moving an object in a desired manner using a motion control device from a group of supported motion control devices, comprising the steps of:

- (i) selecting a software driver from a plurality of software drivers, each of the plurality of software drivers comprising driver code to control one or more motion control devices;
- (ii) generating a control command based on an

application program and the driver code of the selected software driver; and

(iii) operating the selected motion control device in accordance with the control command to move the object.

2. The method of claim 1, wherein the step of generating a control command is further based on a set of driver functions, each driver function defining one or more incremental motion steps that may be performed by the motion control device.

3. The method of claim 2, wherein the application program comprises a sequence of component functions, and at least some of the component functions are associated with driver functions.

4. The method of claim 3, wherein the set of driver functions comprises a subset of first driver functions and a subset of second driver functions, and wherein each first driver function identifies an incremental motion step that may be performed by a motion control device and each second driver function identifies a plurality of incremental motion steps that may be performed by a motion control device.

5. A system for moving an object, comprising:

- \*36** (i) a motion control device;
- (ii) a memory storage device containing:
- (a) a software driver selected from a plurality of software drivers, each of the plurality of software drivers comprising driver code to control one or more motion control devices; and
  - (b) one or more control commands based on the driver code of the selected software driver; and
  - (iii) a computer processor in communication with the motion control device and the memory storage device to operate the motion control device by sending the control commands to the motion control device.

6. The system of claim 5, wherein the control commands are based on a set of driver functions, each driver function defining one or more incremental motion steps that may be performed

by the motion control device.

7. The system of claim 6, wherein the set of driver functions comprises a subset of first driver functions and a subset of second driver functions, and wherein each first driver function identifies an incremental motion step that may be performed by a motion control device and each second driver function identifies a plurality of incremental motion steps.

8. The system of claim 7, wherein the application program comprises a sequence of component functions, and wherein at least some of the component functions are associated with driver functions.

The system of claim 8, wherein the computer processor comprises a processor on a personal computer.

11. The system of claim 8, wherein the computer processor is installed on the motion control device.

'543, 47:18-48:16, 48:19-20 (emphasis added).

#### B. Claim Construction

1. "selected from a plurality of software drivers"

##### a. Parties' Positions

The parties offer the following constructions for the term "selected from a plurality of software drivers," which appears in claim 5. Dkt. No. 119-3.

RGB

FANUC

-----  
"chosen from two or more  
software drivers at run time"

-----  
"a single software driver selected by a user  
from a plurality of software drivers"

The parties' arguments regarding the construction of the term "selecting at least one of the destinations" are repeated for the construction of the instant term: "selected from a plurality of software drivers."

##### b. Court's Construction

For the reasons stated above (supra § IV.B.17.b), the Court finds that this limitation should not be construed to include a "user" limitation or a temporal limitation. In addition, the Court finds that no reason to limit the term to a single software driver. See supra § IV.B.17.b. Because the remaining limitation in this claim ("software drivers") has already been construed, the Court finds that this term

can be given its ordinary meaning and need not be separately construed. The Court therefore finds that the term "selected from a plurality of software drivers" does not require separate construction.

2. "selecting a software driver"

##### a. Parties' Positions

\*37 The parties offer the following constructions for the term "selecting a software driver," which appears in claim 1. Dkt. No. 119-3.

RGB

FANUC

-----  
"choosing a software driver  
at run time"

-----  
"a user selects a single software driver from a  
plurality of software drivers"

The parties' arguments regarding the construction of the term "selecting at least one of the destinations" are repeated for the construction of the instant term: "selecting a software driver."

##### b. Court's Construction

For the reasons stated above (supra § IV.B.17.b), the

Court finds that this limitation should not be construed to include a "user" limitation or a temporal limitation. In addition, the Court finds that the indefinite article should be construed as "one or more." See Baldwin, 512 F.3d at 1342. Accordingly, the Court finds that "selecting a software driver" means "selecting one or more software drivers."

3. "incremental motion step(s)"

"incremental motion step(s)," which appears in claims 4, 6, and 7. Dkt. No. 119-3.

a. Parties' Positions

The parties offer the following constructions for the term

RGB

FANUC

-----  
"primitive motion step(s) " "the smallest increment of movement of which the mechanical system in a motion control device is capable"

RGB contends "incremental" means primitive and corresponds to the term "primitive operation," which was previously construed. Dkt. No. 100 at 30- 31. To refute FANUC'S proposal, RGB maintains that there is only one reference to the word "incremental" in the specification and its use does not equate with "smallest increment of movement." Dkt. No. 109 at 16-17 (citing '543, 3:1-11). Instead, incremental "refers to the ability to control individual stepper motors according to their individual functions ...." Id. at 17. Finally, RGB argues that the construction of incremental motion steps must be reconciled with the construction of "driver functions" since the two are related. Id. at 16-17. Because driver functions are hardware independent, RGB argues that incremental motion steps must also be hardware independent. Id. In contrast, FANUC's proposed construction, which references the smallest movement of which a "device is capable," is necessarily hardware dependent. Id.

In response, FANUC argues that the Patent distinguishes itself over the prior art--such as a normal computer printer--by giving the programmer "the ability to control the hardware in base incremental steps." Dkt. No. 105 at 18 (citing '897 at 3:1-5). This ability means that the programmer of the invention would have be able to control each "individual stepper motor," an ability absent in the prior art. Id. FANUC argues that this ability to control individual motors offer the programmability of "base motion operations," which are the smallest increment of movement of which a motion control device is capable. Id. Finally, FANUC argues that RGB's proposal may cause juror confusion because "primitive motion steps" is not separately defined and has no plain meaning. Dkt. No. 117 at 9.

b. Court's Construction

\*38 The parties have already agreed on the construction of "motion steps" as meaning "steps performed by a motion control device to move an object along a desired path." Dkt. No. 119-2, Ex. A. Thus, the dispute revolves

around the meaning of the word "incremental."

Claim 4, in which the term "incremental motion step" first appears, claims a method "wherein the set of driver functions comprises a subset of first driver functions and a subset of second driver functions, and wherein each first driver function identifies an incremental motion step ... and each second driver identifies a plurality of incremental motion steps ...." '543, 47:39-45. This bipartite organization of driver functions to motion steps is discussed in detail within the specification. The Patent suggests that motion control operations may be either primitive or non-primitive. Id. at 5:62-63. Furthermore, primitive operations are associated with one set of driver functions (core driver functions) while non-primitive operations are associated with another set (extended driver functions). Id. at 7-14.

Although Claim 4 suggests that same basic organization, the driver functions in claim 4 identify individual steps to be performed rather than abstract operations. The steps to be performed, however, can be understood as being similar to primitive and non-primitive motion control operations. On the one hand, primitive motion control operations are those that "are necessary for motion control and cannot be simulated using a combination of other motion control operations." Id. at 5:63-66. Thus, primitive motion control operations instruct a device to perform basic steps, which cannot be emulated using a combination of other steps. On the other hand, non-primitive operations are more complicated and can be emulated suing a plurality of incremental motion steps. See id. at 6:2-6.

For the reasons stated above, this Court finds that "incremental motion step(s)" means "basic motion steps that cannot be simulated using a combination of other steps."

4. "identifies an incremental motion step"

a. Parties' Positions

claims 4 and 7. Dkt. No. 119-3.

The parties offer the following constructions for the term "identifies an incremental motion step," which appears in

RGB	FANUC
"associates with an incremental motion step"	"implements a single incremental motion step"

During the claim construction hearing, FANUC modified its proposed construction for the term. FANUC Hearing Slides at 169. FANUC now suggests that the term be construed as "identifies a single incremental motion step." *Id.* FANUC thus adopts the plain meaning of "identifies." Similarly, the plain meaning of "identifies" was adopted by both parties in the construction of the phrase: "identifies a plurality of incremental motion steps." *Id.*; RGB Hearing Slides at 133. The remaining disagreement between the parties--whether "an" means "single"--is already addressed under the construction of the "selecting" terms above. See *supra* § IV.B.17.a.

b. Court's Construction

\*39 For the reasons stated above (*supra* § IV.B.17.b), the Court finds that the indefinite article should be construed as "one or more." See *Baldwin*, 512 F.3d at 1342. In addition, the Court finds that the term "identifies" can be given its plain and ordinary meaning.

Accordingly, the Court that the term "identifies an incremental motion steps" means "identifies one or more incremental motion steps."

VIII. CONCLUSION

The Court hereby ORDERS the claim terms addressed herein be construed as indicated.

2009 WL 2971097 (E.D.Tex.), 2009 Markman 2971097