Actually Venus8 chip is used in NavSpark, not Venus6 chip.

Datasheet for Venus822 is currently available. What will be different, depending on success/failure of the NavSpark campaign is how much internals and peripheral control is to be disclosed. Right now the Venus822 datasheet has little detail on the internals, similar to datasheet from any major GPS/GNSS chipset vendor; it contains information mainly to successfully apply the chip to form a working GPS/GNSS receiver.

If the NavSpark campaign is successful, more information on internal processor, peripheral controls will be added to the Venus822 datasheet to allow user applying it as a microcontroller.

Yes, Venus822 can be purchased independently of the Indiegogo campaign. Right now we offer Venus822 SDK for $199, ships with the 4cm x 7cm development board you see on the Indiegogo campaign webpage. If NavSpark campaign is successful, effectively we'll be providing SDK for free, integrating it into the Arduino IDE and offering NavSpark at one tenths of the current SDK price.

Whether Venus822 chip or Venus838FLPx GPS module (or later Venus858F-GL GPS/GLONASS and Venus858F-BD GPS/Beidou module) is suitable for your application, please see the explanation in the press release:

http://www.skytraq.com.tw/PR20131221%20-%20NavSpark%20Press%20Release%20-%20E.pdf

With NavSpark Arduino IDE or SDK, user code reside in a function that the main() function will call. There is no RTOS, just the GPS/GNSS interrupt service routine and the background main() loop. The application code should be written in the form of do some action if some event is detected, so as not to block the background main() function from looping continuously. Since the GPS/GNSS kernel code is partition into an interrupt service routine and a background function called by the main() function, transparent to the user; so long as the user application doesn't block the main() function looping, it will appear having user code and GPS/GNSS running concurrently.

The user code is compiled and linked with the provided GPS/GNSS library to form the final binary file.