


## Descodificación del protocolo FIRMATA

En esta práctica aprenderá cómo configurar el osciloscopio **InfiniiVision MSO-X 2014A de Keysight** para decodificar la transmisión y recepción de tráfico del protocolo de comunicación FIRMATA que se envía por medio del protocolo serie UART. Además, aprenderá a utilizar la capacidad adquisición sencilla automática de búsqueda y navegación UART del osciloscopio, así como la familiarizarse con algunos de los comandos básicos del protocolo de comunicación FIRMATA.

Paso a realizar para activar la decodificación UART en el osciloscopio InfiniiVision MSO-X 2014A de Keysight:

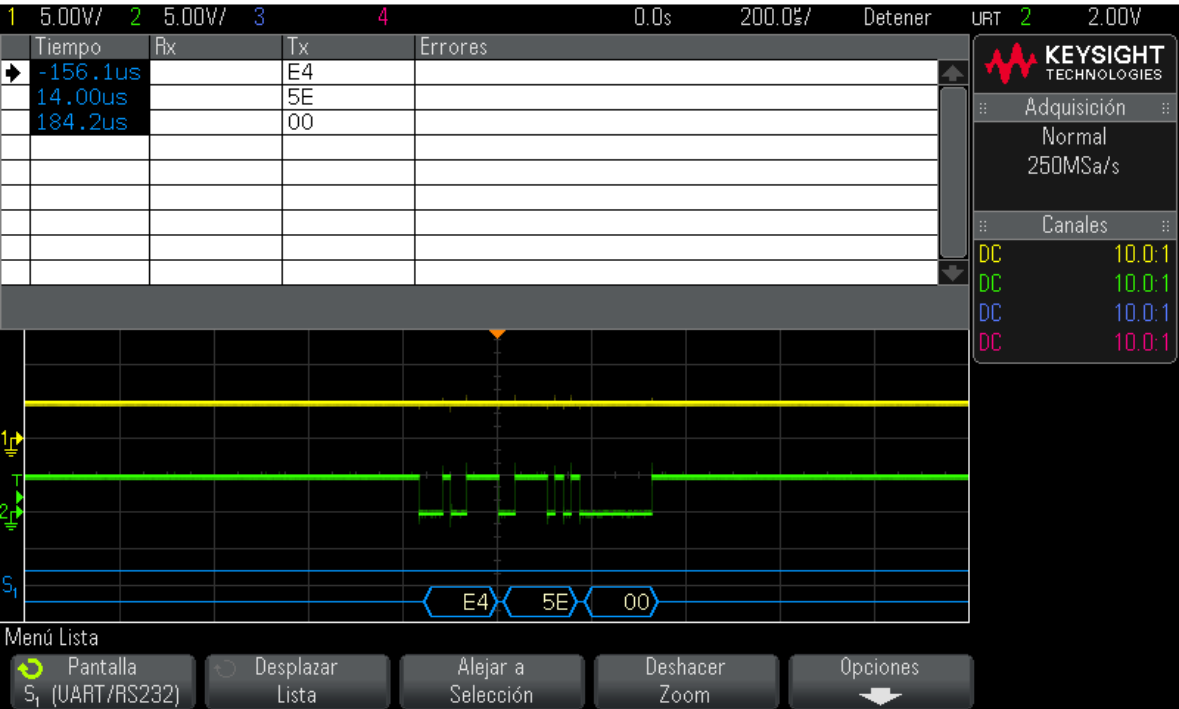
1. Conectar la punta de prueba del canal **UNO** al terminal **Rx** y tierra del Arduino.
2. Conectar la punta de prueba del canal **DOS** al terminal **Tx** y tierra del Arduino.
3. Presione el botón **[Default Setup]** en el panel frontal del osciloscopio.
4. Ajustar el canal UNO a **5.00 V/div**.
5. Ajuste el compensador del canal UNO a **-5.00 V**.
6. Presione el botón **[2]** del panel frontal para encender el canal DOS.
7. Ajustar el canal DOS a **5.00 V/div**.
8. Ajuste el compensador del canal DOS a **+5.00 V**.
9. Ajuste la base de tiempo del osciloscopio a **2.00 ms/div**.
10. Presionar el botón **[Serial]** del panel frontal.
11. Presione la tecla programable **Mode I<sup>2</sup>C**; luego seleccione **RS232/UART** usando la perilla de entrada.
12. Presione la tecla programable **Signals** y verifique que la señal **Rx** está definido como canal UNO, y que **Tx** está definido como canal Dos (condiciones por defecto).
13. Presione la tecla programable **Threshold** de Rx y ajustar a **2.00V** girando la perilla de entrada, luego repita el mismo procedimiento para Tx.
14. Presione el botón  (Atrás) del panel frontal para regresar al menú anterior.
15. Presione la tecla programable **Bus Config**.
16. Presione la tecla programable **Parity**; luego seleccione **None**.
17. Mientras el menú se encuentre activo, verificar también que el número de bits este ajustado en **8**, la tasa de Baudios este ajustada to **57.6 kb/s**, polaridad este configurada a **Idle High**, y el Orden de los bits este configurada a **LSB first**.

Ahora vamos a configurar el osciloscopio para disparar cuando la señal de transmisión de datos (TX) es igual a un Bit de detección de Tx, esto nos permitirá utilizar la modalidad de captura sencilla del osciloscopio para visualizar tramas de datos en el canal de transmisión del Arduino UNO.

18. Presionar el botón **[Trigger]** del panel de control; luego seleccionar **Seria 1 (RS232/UART)** utilizando la perilla de entrada.
19. Presione la tecla programable **Trigger Setup**; luego seleccionar **Tx detection bit** usando la perilla de entada.
20. Ajustar el tiempo base del osciloscopio a **2.00 ms/div**.

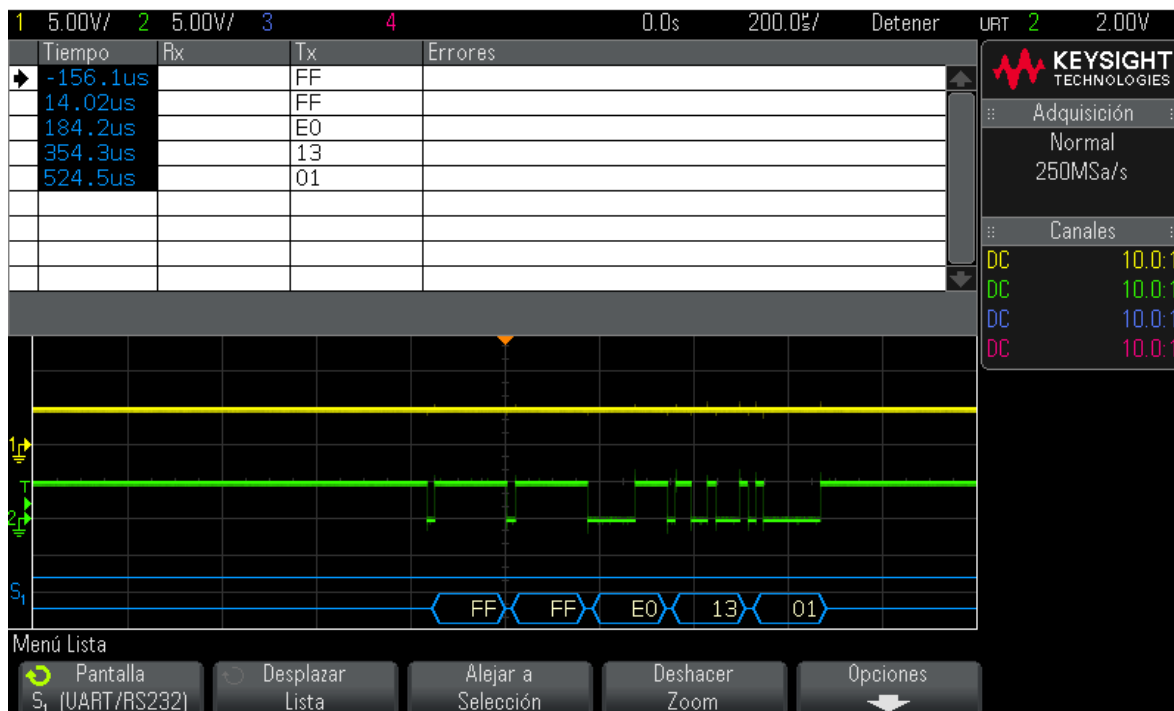
El osciloscopio ahora debe estar preparado para capturar una trama de datos serie UART utilizando el protocolo de comunicación FIRMATA bajo las condiciones de disparo.

- 21. Presione el botón **[Single]** del panel frontal para obtener una muestra de datos serie.
- 22. Ajustar el tiempo base del osciloscopio a **200.0  $\mu$ s/div**.
- 23. Presione el botón **[Serial]** del panel frontal.
- 24. Presione la tecla programable **Lister**; luego seleccione **Serial 1 (UART/RS232)** usando al perilla de entrada.



**Figura 1** - Captura de un comando de mensaje de lectura canal análogo del Arduino.

En la pantalla del osciloscopio debería de apreciarse alguna de las capturas de los reportes de lecturas análogas del Arduino UNO en el instante de tiempo en que se presionó el botón de captura sencilla, como se muestra en la imagen de la **Figura 1**. En la tabla generada se pueden enlistar los datos en hexadecimal capturados. Puede apreciarse el comando de mensaje de entrada análoga seguido de los datos de la señal análoga capturado en ese instante de tiempo.



**Figura 2** - Captura de un comando de mensaje de lectura canal análogo del Arduino con comando de reinicio.

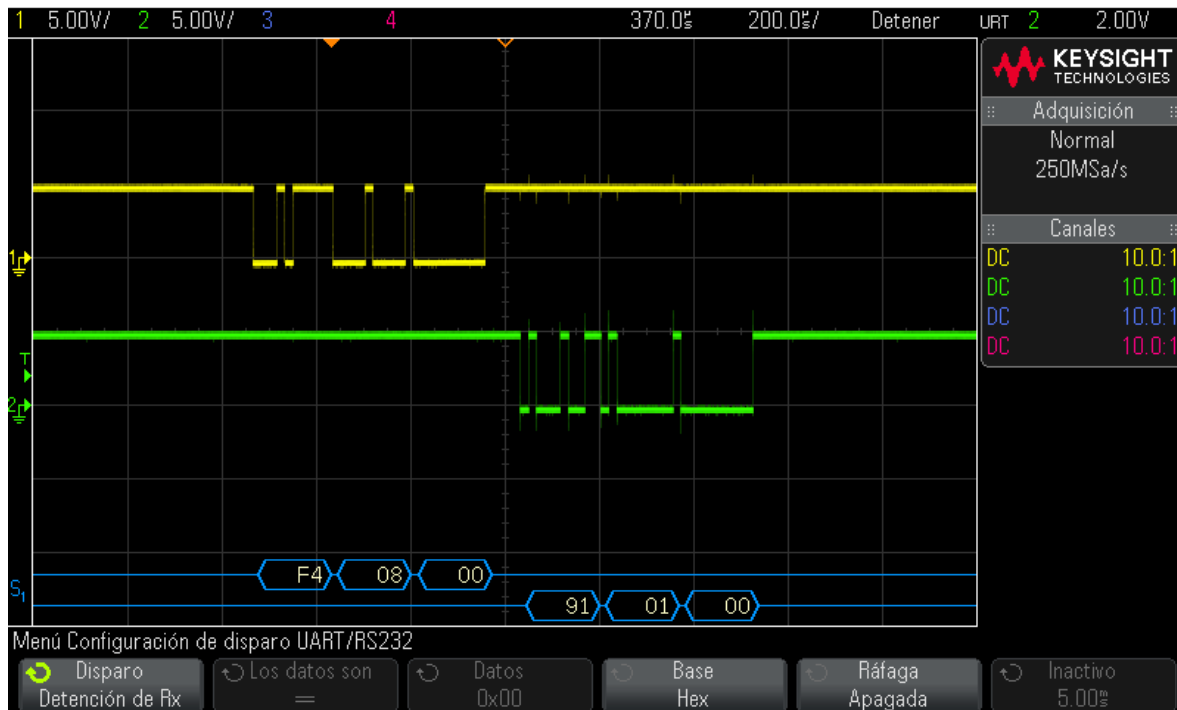
Puede presionar el botón Single del panel frontal de osciloscopio las veces que quiera para capturar varias lecturas de algunas de las entradas análogas del Arduino UNO. En un instante de tiempo podría observar dos datos en hexadecimal **FF** como se muestra en la **Figura 2**, que indica el inicio del lazo repetitivo con la lectura del primer canal análogo seguido de la lectura de la señal.

Ahora vamos a capturar los comandos de configuración de los pines para el Arduino UNO, pero primero debemos de reprogramar el Arduino UNO con un nuevo Sketch. Para esto nos vamos a Archivo, Ejemplos, Firmata y luego seleccionar el Skecth StandardFirmata. Con todo eso procedemos a abrir el programa de prueba del protocolo Firmata. Seleccionamos el puerto de comunicación en el que se encuentra conectado el Arduino UNO. Automáticamente el programa detecta el skecth programado en el Arduino UNO y se despliega la lista de todos los pines digitakes y análogos existentes en dicha tarjeta. Detenemos los reportes de mensajes análogos cambiando los pines análogos a entradas digitales.

Procedemos a configurar el osciloscopio para disparar en la recepción de datos del canal Rx del Arduino para poder apreciar los comandos de configuración del protocolo Firmata.

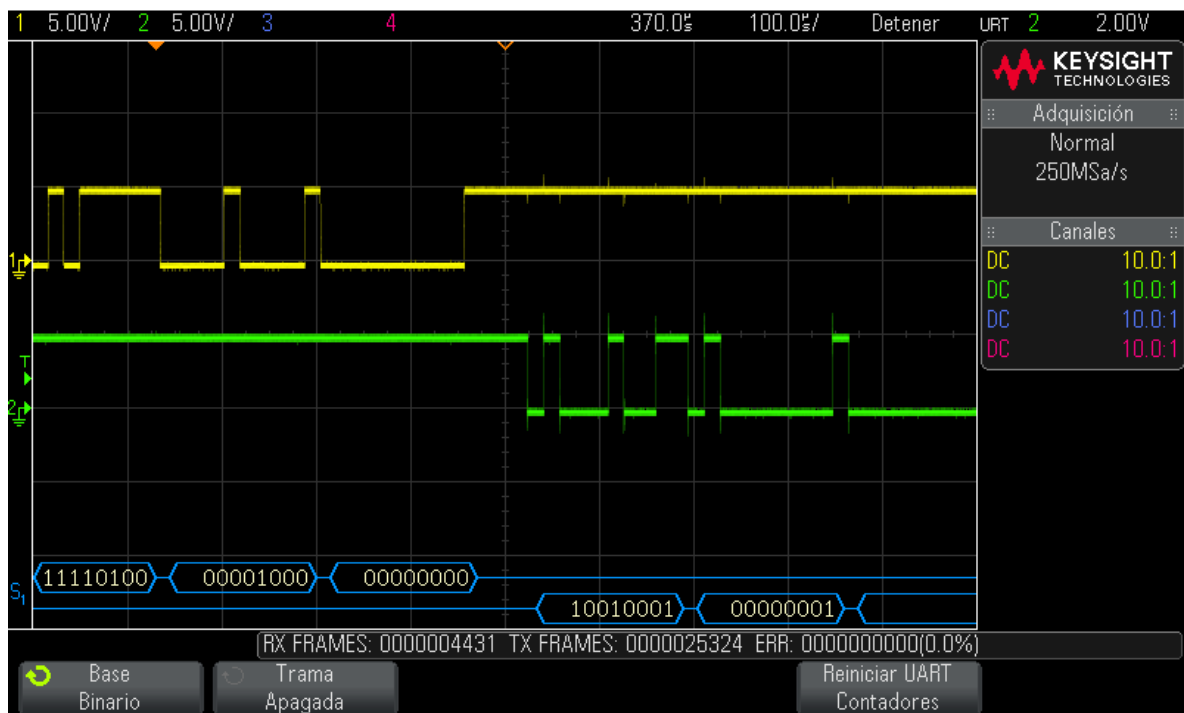
25. Gire la perilla de posición/retraso horizontal para reposicionar el punto de disparo a un retraso de **370 µs**.
26. Presionar el botón **[Trigger]** del panel de control del Osciloscopio.
27. Presione la tecla programable **Trigger Setup**; luego seleccionar **Rx detection bit** usando la perilla de entada.
28. Presione el botón **[Single]** del panel frontal para obtener una muestra de datos serie

El osciloscopio se encuentra lista para capturar una trama de datos del protocolo Firmata. Ahora en el programa de prueba procedemos a ajustar el pin 8 del Arduino como entrada para poder apreciar el comando de configuración del protocolo Firmata. Al ajustar el pin correspondiente observamos en el osciloscopio la captura del comando de configuración del pin como entrada digital seguido de una respuesta por parte del Arduino UNO.



**Figura 3** – Captura de comandos MIDI Standard del Arduino UNO.

En la **Figura 3**, la trama del canal UNO se puede interpretar de la siguiente manera: el primer Byte tiene el dato hexadecimal 0xF4 que indica un reporte de configuración, seguido de un segundo Byte con dato hexadecimal 0x08 que corresponde al pin número 8 del Arduino, seguido de un tercer Byte que indica la modalidad en que ese pin se encontrará configurado y la trama del canal DOS se puede interpretar de la siguiente manera: el primer Byte con datos hexadecimal 0x91 indica un mensaje digital de entrada y salida. En primer nibble indica el comando propiamente dicho, el segundo nibble indica el puerto de configuración del microcontrolador. El segundo Byte con datos hexadecimal 0x01 indica el estado del pin digital correspondiente a ese puerto.

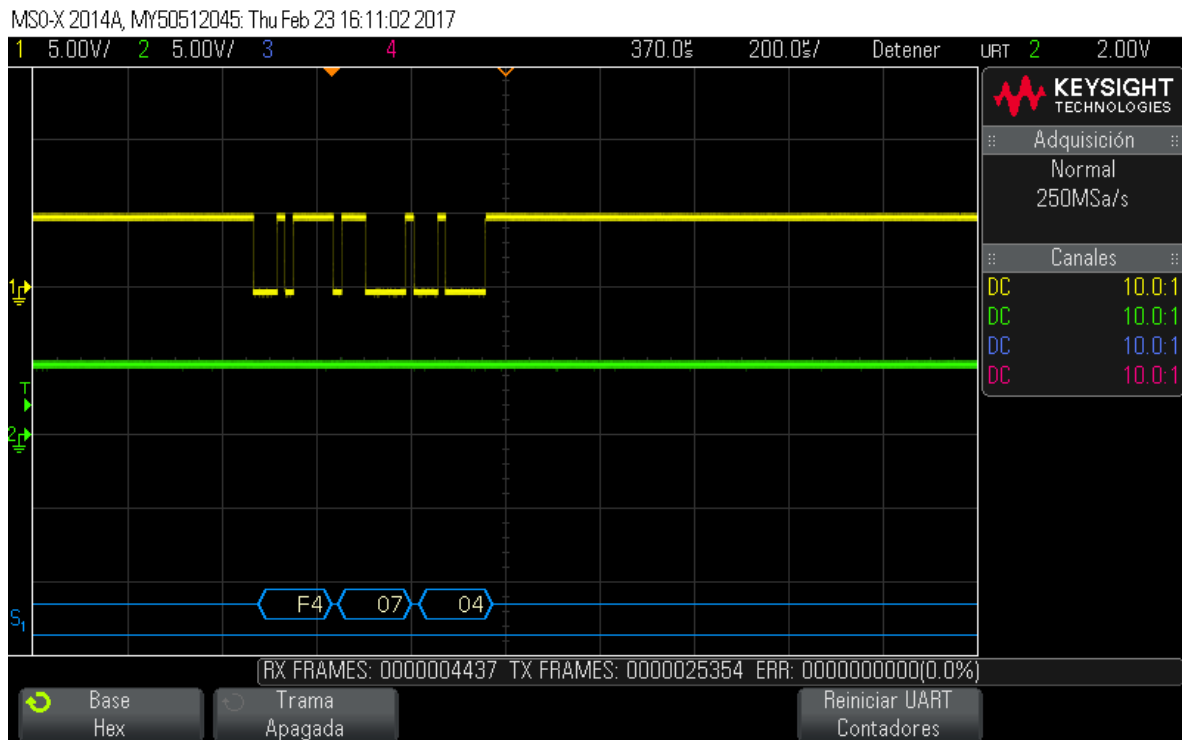


**Figura 4** – Conversión de base a binario de una captura de comandos MIDI Standard

Convirtiendo la base hexadecimal a binario como se muestra en la **Figura 4** podemos darnos cuenta que cada bit corresponde al estado de cada pin por puerto. Lo siguiente a realizar es capturar el dato del canal de Rx del Arduino para poder apreciar el comando MIDI Standard para la configuración del pin número 7 del Arduino UNO como servo.

29. Presione el botón **[Single]** del panel frontal para obtener una muestra de datos serie.

Ahora configuramos en el programa de prueba el pin número 7 del Arduino UNO como servo.

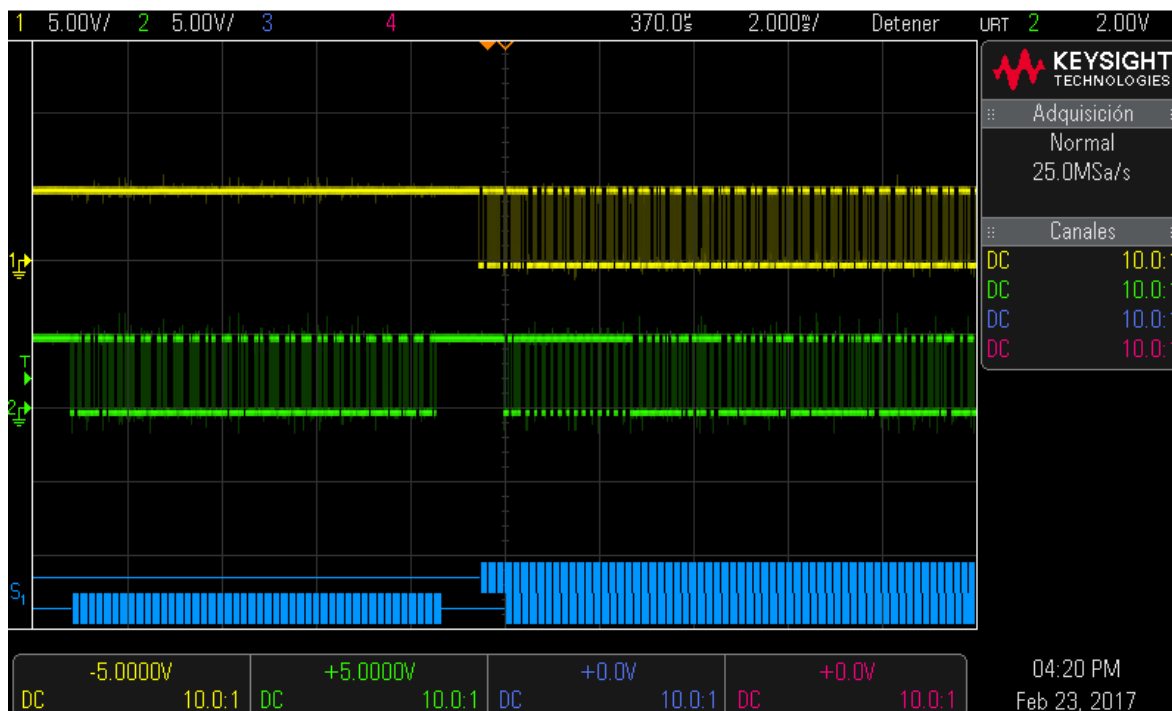


**Figura 5** – Captura de comandos MIDI Standard del Arduino UNO.

En el instante que se efectúa la configuración pertinente, el osciloscopio captura la trama de datos del canal Rx del Arduino UNO, como se muestra en la **Figura 5**. Los datos mostrados en el osciloscopio se pueden interpretar de la siguiente manera: el primer Byte – 0xF4. Correspondiente a un reporte de configuración, seguido de un Byte - 0x07. Correspondiente al número del pin del Arduino, seguido de un tercer Byte – 0x04. Correspondiente a la modalidad en el que el pin se encuentra ahora configurado, que en este caso es un Servo.

Ahora habiendo revisado algunos de los comandos MIDI Standard más esenciales en el protocolo de comunicación Firmata, procedemos a capturar los comandos con estructura SysEx. Este tipo de datos se llevan a cabo en la inicialización del Arduino, por lo cual haremos uso de condiciones de captura en recepción de datos para el canal Rx del Arduino UNO así como también poner en Reset el Arduino para inicializar la comunicación.

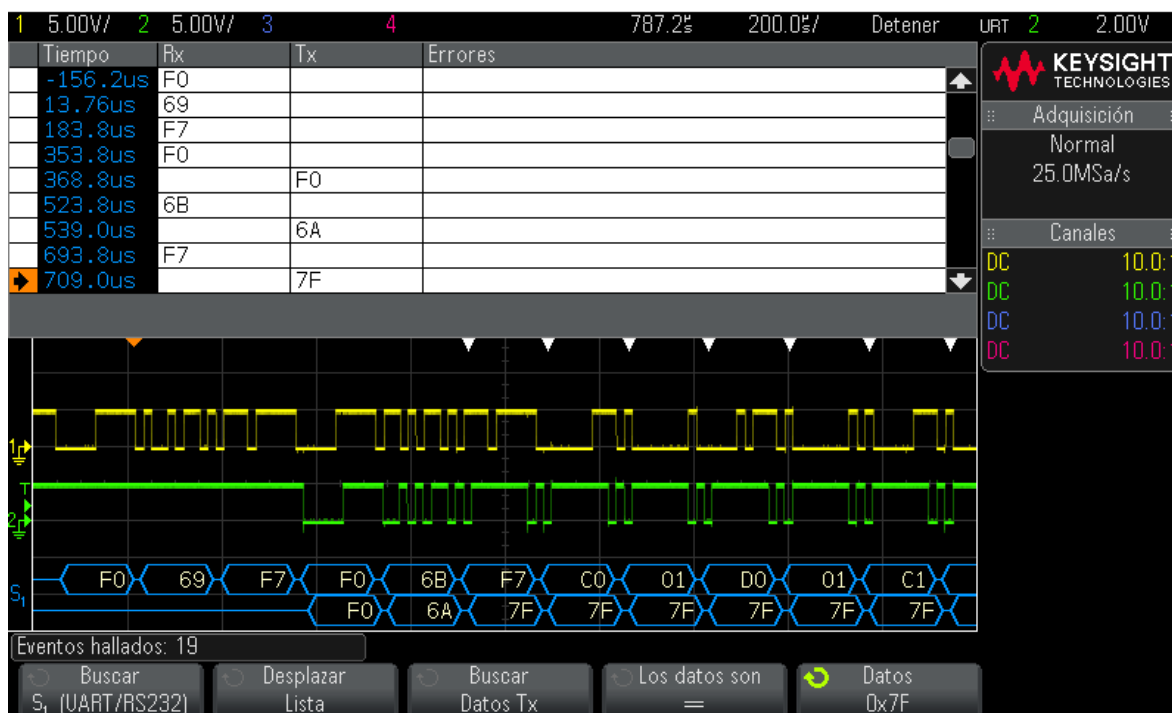
30. Ajustar el tiempo de base del osciloscopio a **2.00 ms/div**.
31. Presione el botón **[Single]** del panel frontal para obtener una muestra de datos serie.
32. Presione el botón **[Reset]** del Arduino.



**Figura 5** – Captura de una trama de inicialización del protocolo de comunicación Firmata

Los datos de inicialización como se muestra en la **Figura 5** que genera el programa de prueba permiten configurar el Arduino a las condiciones por defecto según la programación, para que luego sean configuradas a voluntad del usuario. Ahora procedemos a revisar una trama de mensajes SysEx para una solicitud de mapeo de todos los pines del Arduino, efectuaremos una búsqueda programada para concretar el número de pines tanto digitales como analógicos de ese mapeo.

33. Ajustar el tiempo de base del osciloscopio a **200.0  $\mu$ s/div**.
34. Presionar el botón **[Search]** del panel frontal de osciloscopio.
35. Presionar la tecla programable **Search for**; luego seleccionar **Tx Data** usando la perilla de entrada.
36. Presionar la tecla programable **Data** e ingresar el valor **0x7F** moviendo la perilla de entrada.



**Figura 6** – Ocurrencias encontradas en base a la condición de búsqueda establecida.

En el inicio de la trama SysEx del canal UNO del osciloscopio podemos apreciar el comando de inicio 0xF0, seguido de una solicitud de mapeo de los pines – 0x69, finalizando con un comando de fin de la trama SysEx – F7. Automáticamente cuando esa trama finaliza se puede apreciar en el canal DOS la respuesta del mapeo. Comenzando por el Byte de inicio – 0xF0. Siguiéndole un byte de comando 0x6A, que indica una respuesta de la información de mapeo.

Los triángulos blancos cerca de la parte superior del área de forma de onda marcan la ubicación de tiempo de cada ocurrencia "encontrada" de nuestra operación de búsqueda como se muestra en la **Figura 6**. Estos cuadros también están marcados en naranja en la primera columna/puntero de la tabla de registro. Su osciloscopio debería haber encontrado y marcado 19 ocurrencias de esta operación de búsqueda. Tenga en cuenta que el número de eventos "encontrados" se indica cerca de la parte inferior de la pantalla. Contamos las primeras catorce ocurrencias encontradas. Estas indican la cantidad de pines digitales del Arduino configurados para enviar mensajes en tiempo real. Luego de estos catorce Bytes sigue lo que parece ser un conteo en hexadecimal, estos seis Bytes indican los pines analógicos del Arduino UNO listos para un envío de mensajes en tiempo real, luego la trama de mensajes SysEx concluye con un Byte de fin SysEx como se muestra en la **Figura 7**.



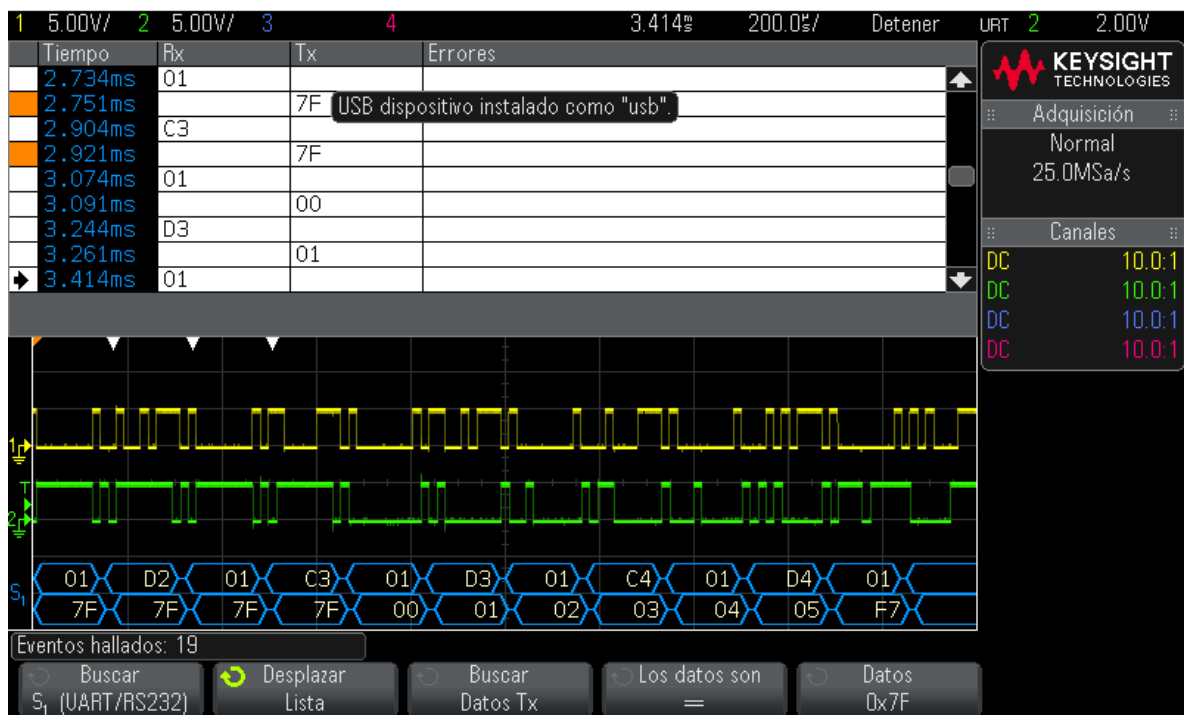


Figura 7 – Finalización de una trama de datos SysEx (Byte 0xF7).