

|  |
| --- |
| Guía rápida y traspaso de información de la tarjeta fuelsensor v1.6 |
| Enfoque técnico |
| **Realizada el 02/05/2018** |

|  |
| --- |
| Mayo 2018  Preparado por: Pedro Durán C. |



|  |  |  |
| --- | --- | --- |
| Fecha | Cambio | Responsable |
| 02/mayo/2018 | Creación del documento inicial | PDC |

**INDICE**

[1. Introducción 2](#_Toc513205762)

[2. Ubicación de información de interés. 3](#_Toc513205763)

[3. Procedimiento de instalación de firmware. 4](#_Toc513205764)

[4. Procedimiento de Instalación del sensor en el estanque. 6](#_Toc513205765)

[4.1 Scripts de instalación. 6](#_Toc513205766)

[4.2 En terreno. 8](#_Toc513205767)

[4.3 Conclusiones 9](#_Toc513205768)

[5. Información adicional relevante 9](#_Toc513205769)

# Introducción

Este documento tiene como objetivo hacer una entrega de información relevante y de interés para la empresa. Se abordaran los siguientes tópicos:

1. Se indica la documentación con información importante en las instalaciones de Collahuasi y DGM; y donde encontrarla.
2. Procedimiento de instalación de firmware.
3. El paso a paso de las en el proceso de instalación del sensor debajo del estanque. Poniendo énfasis en las tareas que realizaba Pedro Duran.
4. Varios tips técnicos/prácticos surgidos de conocimiento adquiridos en las diferentes instalaciones del sensor.
5. Recomendaciones o sugerencias para las siguientes versiones posteriores a la v1.6.
6. Etc.

# Ubicación de información de interés.

En el repositorio del proyecto “REPOSITORY\_ROOT\_PATH/source/source/fuelsensor\_v1\_5“

1. Este documento y el archivo Excel con la información detallada de los sensores instalados en Collahuasi y DGM tal como fecha de instalación, firmware usado, etc. puede ser encontrada en “REPOSITORY\_ROOT\_PATH/documentation”
2. Los datos obtenidos en las pruebas realizadas debajo del estanque en las diferentes faenas mineras pueden ser encontrados en “REPOSITORY\_ROOT\_PATH/source/matlab/test\_en\_terreno/”
3. Los script usados en la instalación se encuentran en:   
   “REPOSITORY\_ROOT\_PATH /source/matlab/fuelsensor\_scripts/En\_terreno/”
4. Los script utilizados para debugging se encuentran en:   
   “C:\fuel-sensor-fpga\source\matlab\fuelsensor\_scripts\Debug\_Online”
5. Los scripts modbus que permiten leer y setear parámetros del sistema pueden ser encontrados en:  
   “REPOSITORY\_ROOT\_PATH /source/modbus\_poll”
6. Toda la información respecto a los registros del sistema almacenados por los servidores en la respectiva faena, tanto para DGM como Collahuasi, puede ser encontrado en:   
   “REPOSITORY\_ROOT\_PATH /source/matlab/registros\_almacenados\_proveidos\_por\_servidores\_de\_las\_faenas/”

Los script de instalación son explicados más adelante. A continuación se explica brevemente los script de debugging remoto:

1. “fuelsensorDataRealtime\_on\_tcpip\_conditions.”: Es usado tanto para el algoritmo “filter” o sdft. Captura los 65536 datos del sensor. Puede ser usado para filtrar límites en muestras y threshold. También se puede usar downsamplig para reducir el número de muestras (el downsampling puede ir de 1 a 6). Se debe tener cuidado con este script ya que detiene la operación de envió de pulsos, y una vez finalizado vuelve a restablecerlo.
2. “get\_vector\_heigh\_and\_maxvalue\_on\_tcpip.m”: Es usado solo para el algoritmo “filter”. Obtiene el vector de 100 muestras [altura del combustible, nivel de señal].
3. “getting\_sdft\_debug\_registers\_on\_tcpip.m”: Es usado solo para el algoritmo “SDFT”. Captura los vectores
   1. Max\_value
   2. Max\_pos
   3. Data.
   4. Dataf2.
4. “sdft\_norealtime\_debug\_registers\_on\_tcpip.m”: Es usado solo para el algoritmo “SDFT”. Captura en tiempo real las variables:
   1. Sdft\_arg\_max
   2. Sdft\_max
   3. Pos
   4. Eco\_mean
   5. Eco\_var.

# Procedimiento de instalación de firmware.

1. Realizar un checkout en el repositorio “fuel-sensor-fpga\_from-v1\_5” en la versión de firmware que se quiere utilizar.
2. Abrir el programa “Xilinx SDK” de Xilinx
3. Setear el workspace en “REPOSITORY\_ROOT\_PATH/source/source/fuelsensor\_v1\_5“
4. Ir al Proyecto xilkernel\_platform\_v1\_5 y revisar que el archivo “main\_loading\_SP\_to\_flash.c” este incluido en la compilación. Ademas “main.c” y “main\_test\_flashMemory.c” no deben estar incluidos en la compilación. Para verificar esto, se debe hacer click derecho del mouse sobre el archivo > Resource Configuration > Exclude from Build > y luego clickear “select all” o “Deselect All” según corresponda.
5. Además es importante verificar en el archivo “modbus/modbus\_thread.c” la línea:
   1. “unsigned char modbus\_rtu\_enable = 1” para modbus rtu.
   2. “unsigned char modbus\_rtu\_enable = 0” para modbus ascii.
6. **Cargar los registros del sistema en la memoria flash:** En “main\_loading\_SP\_to\_flash.c” asegurarse de que la línea   
   “#define LOAD\_PARAMETERS” este des comentada. Luego del cambio el proyecto será compilado automáticamente.
7. En la barra de herramientas clickear “Program FPGA”. Setear todo tal como aparece en la figura 1 y clickear “program”. Ahora los registros del sistema están alcenados en la memoria flash.
8. **Cargar el fimware en la memoria flash:** En “main\_loading\_SP\_to\_flash.c” comentar la línea   
   “#define LOAD\_PARAMETERS”. Luego del cambio el proyecto será compilado automáticamente.
9. En la barra de herramientas clickear “Program FPGA”. Setear todo tal como aparece en la figura 2 y clickear “program”.
10. En la barra de herramientas clickear “Program FLASH”, setear los parámetros tal como aparecen en la figura 3, y clickear “program”.
11. Nuevamente ir a la barra de herramientas clickear “Program FLASH”, setear los parámetros tal como aparecen en la figura 4, y clickear “program”. Ahora el firmware estará cargado en la memoria flash de la tarjeta.
12. Desenergizar y energizar la tarjeta para que operé normalmente.
13. Nota: Una referencia de este procedimiento puede ser encontrada en: https://reference.digilentinc.com/learn/programmable-logic/tutorials/htsspisf/start

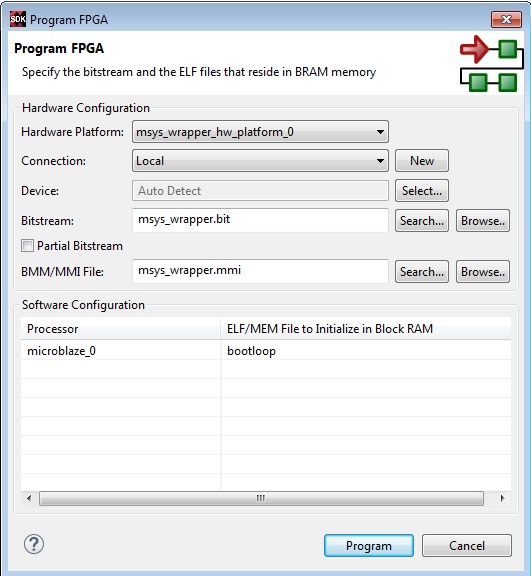


Fig. 1. Cargando registros del sistema en memoria flash: Paso 1.

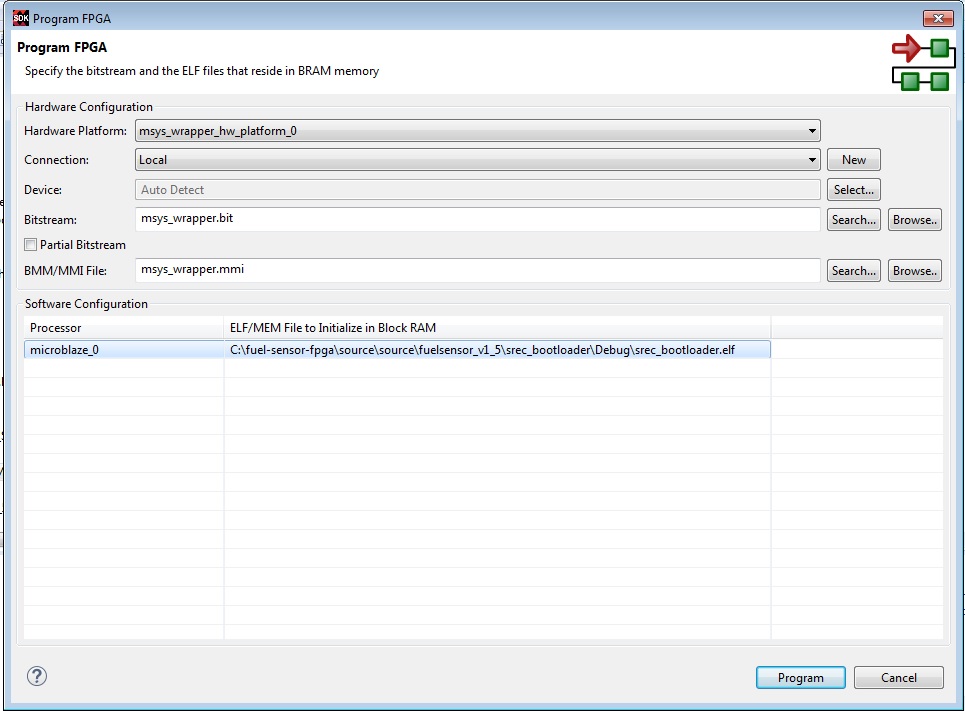


Fig. 2. Cargando firmware en memoria flash: Paso 2.

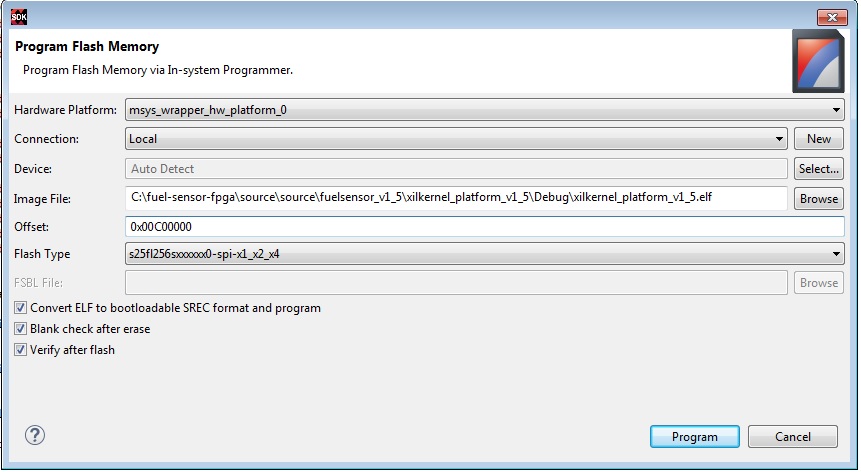


Fig.3. Cargando firmware en memoria flash: Paso 3.

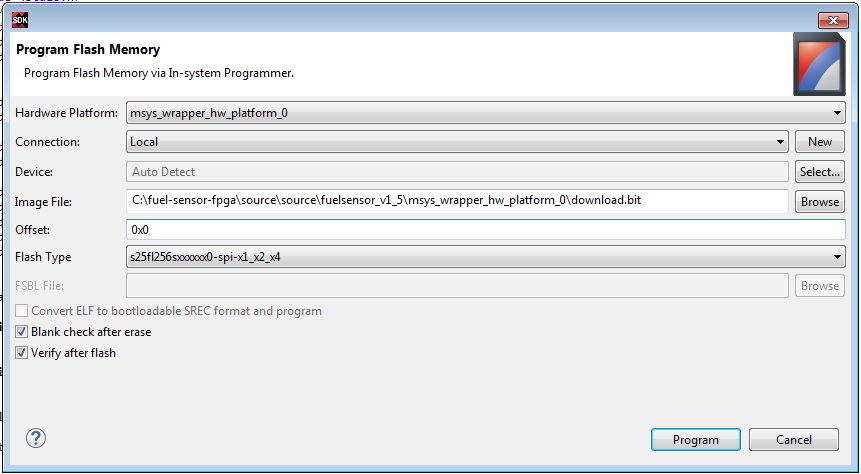


Fig. 4. Cargando firmware en memoria flash: Paso 4.

# Procedimiento de Instalación del sensor en el estanque.

Se presenta los pasos necesarios para la instalación del sensor de combustible debajo del estanque. La idea es presentar como se usan los script para encontrar un buen punto de señal y encontrar los parámetros óptimos de operación del sensor. No se detallará los procedimientos que son bien conocidos por Alberto Parra.

## Scripts de instalación.

Se tiene 2 scripts de matlab en el procedimiento de instalación: “fuelsensorDataRealtime\_finding\_optimal\_parameters.m” y “fuelsensorDataRealtime\_finding\_optimal\_parameters\_single.m”

1. “fuelsensorDataRealtime\_finding\_optimal\_parameters.m”: El fin de este script es encontrar los parámetros óptimos de periodo, ancho (width), y numero de pulsos de manera de maximizar el nivel de señal del rebote ultrasónico.
2. “fuelsensorDataRealtime\_finding\_optimal\_parameters\_single.m”: Script que permite obtener la gráfica de la señal de rebote utilizando solo un pulso a la vez; o puede ser usado en captura rápida (sin grafica de la señal) para entregar la altura del combustible y nivel de señal cada 1 segundo. Para usar este script es necesario ingresar los parámetros óptimos encontrados.

Respecto al periodo óptimo se puede decir lo siguiente: En las inmediaciones del periodo T= 1000[ns] (frecuencia de 1[MHz]) se tiene uno o dos peak óptimos, como se puede ver en la 5. Un caso práctico por ejemplo es tener un peak de señal en T1 = 880[ns] y otro en T2 = 1310[ns]. Lo que se ha visto hasta ahora es que si el sensor es cambiado de posición debajo del estanque, estos peaks no varían su frecuencia (periodo) pero si pueden variar su magnitud. Por ejemplo en la zona 1 puede tener un mejor nivel de señal usando el periodo T1 en lugar de T2; pero podría pasar que en la zona 2 la mejor señal ocurre cuando se utiliza el periodo de señal T2.

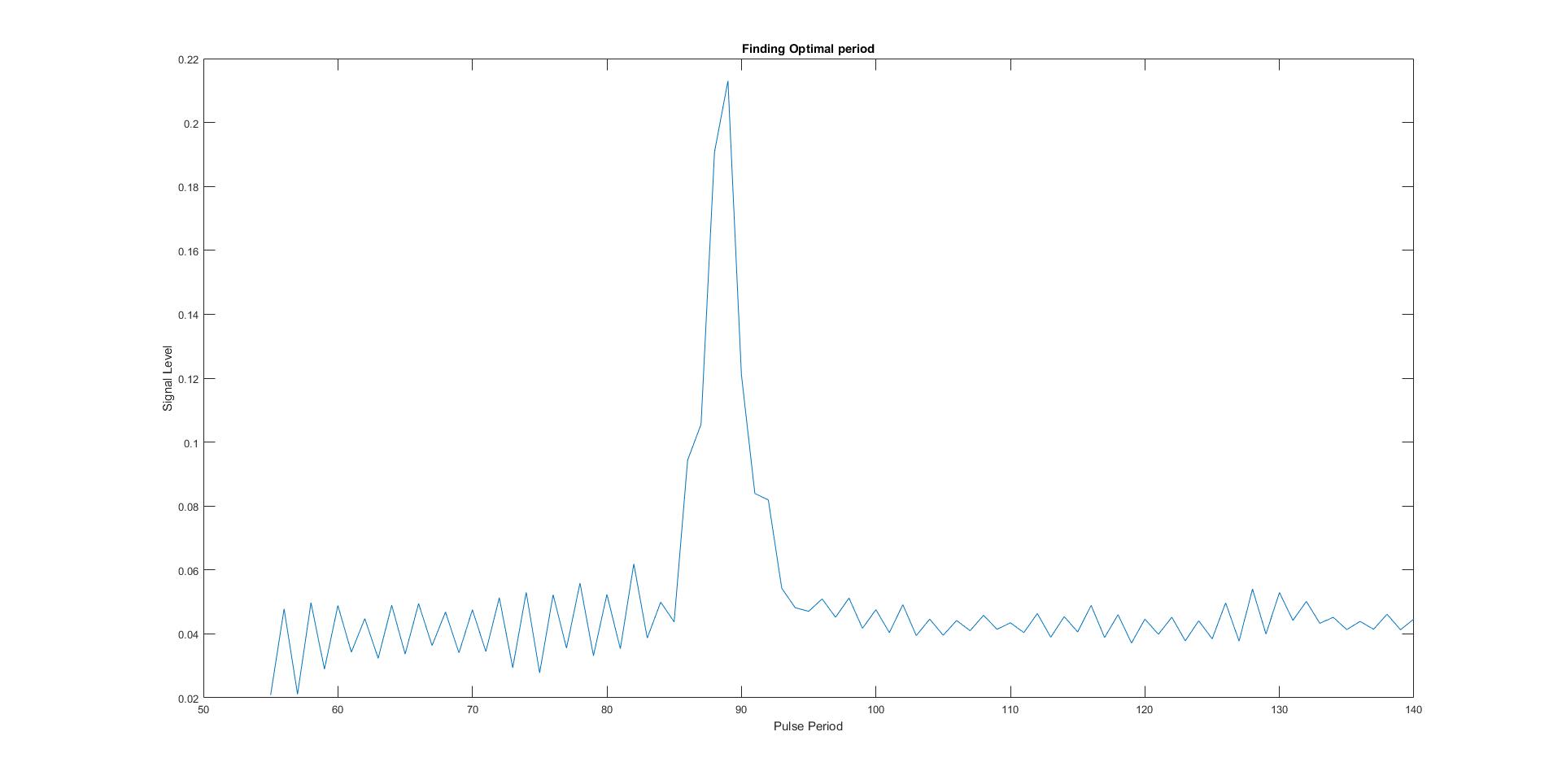


Fig. 5. Ejemplo de barrido de “periodo del pulso” con pga = 0.

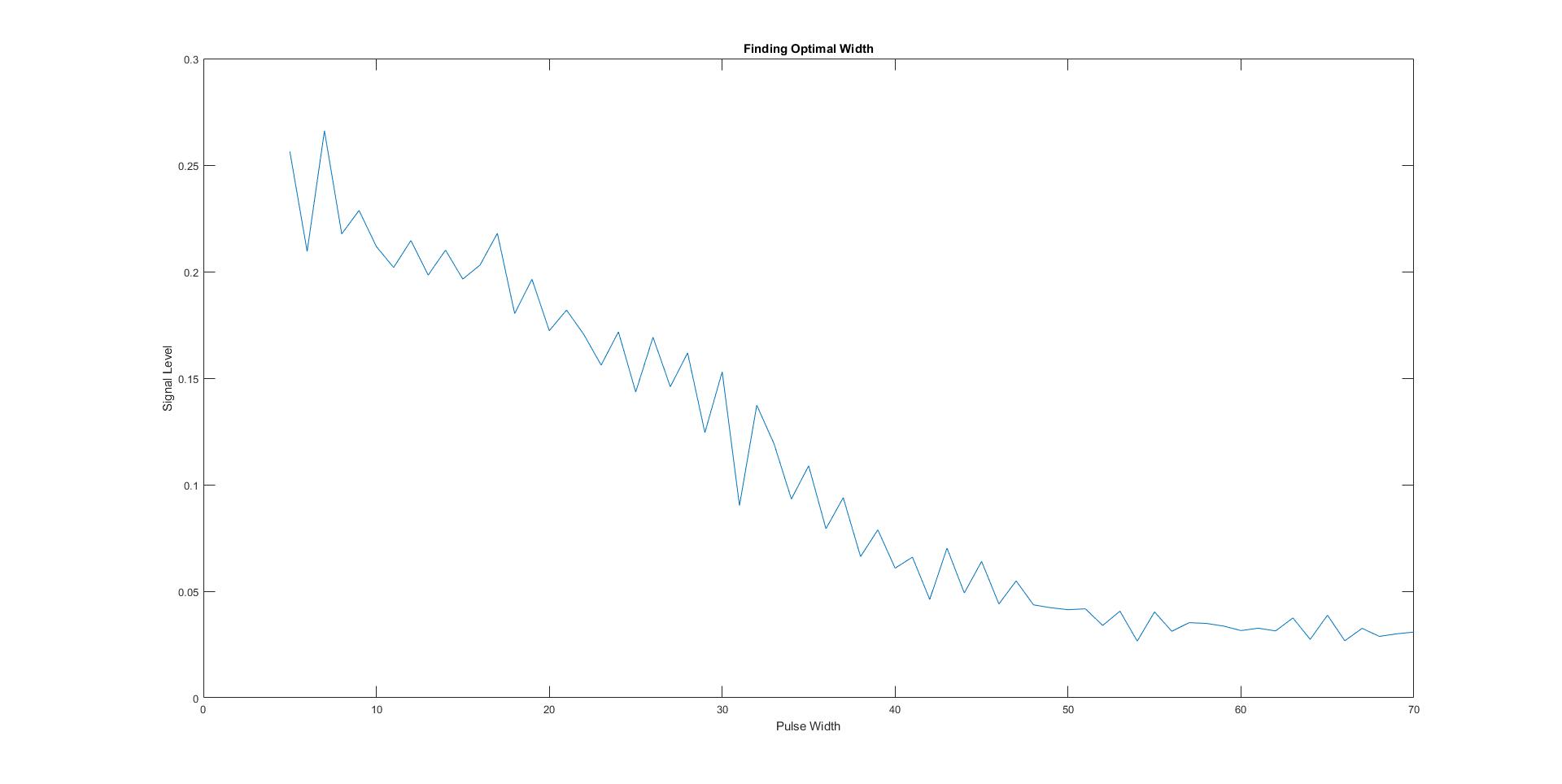


Fig. 6. Ejemplo de barrido de “ancho de pulso” con pga = 0.

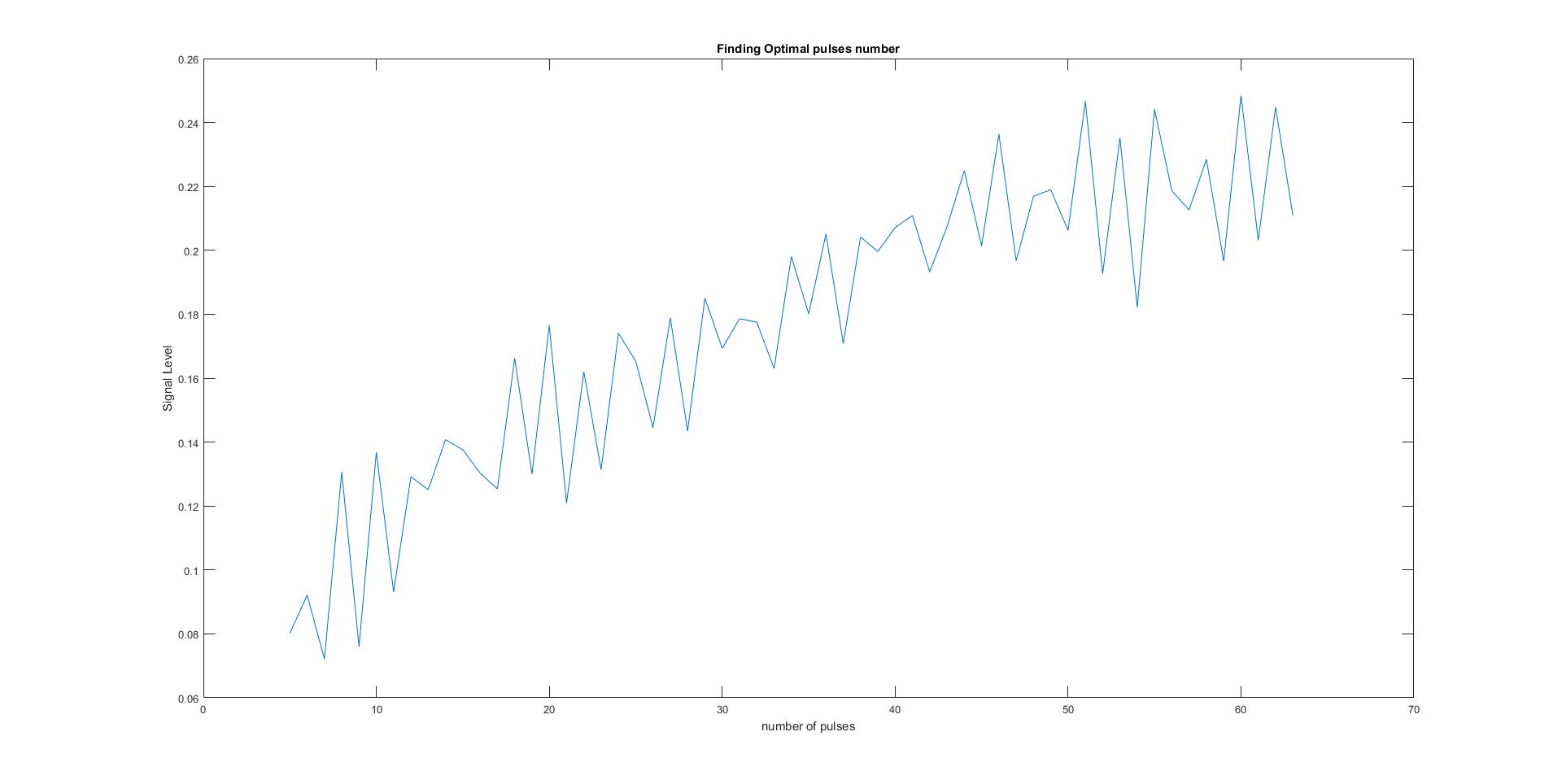


Fig. 7. Ejemplo de barrido de “numero de pulsos” con pga = 0.

## En terreno.

En general de la experiencia en las instalaciones se sabe que existen dos zonas en las que se puede encontrar un buen nivel de señal: Estas son en los alrededores cerca del eje de la válvula de despiche del estanque, este sector se llamará “zona 1”, y en un sector lateral al estanque que se denominará “zona 2”. Ambas áreas son bien conocidas por Alberto.

Estos son los pasos generales de instalación del equipo de medición de combustible:

1. Se debe limpiar la zona debajo del estanque alrededor de la zona 1 y 2. Este procedimiento lo conoce bien Alberto.
2. Para efectos de las pruebas debajo del estanque, es recomendable previo a las pruebas de señal, tener una noción del porcentaje de combustible en el estanque. Las fuentes de información son:
   1. Revisar el visor lateral del estanque.
   2. Medidor analógico en cabina.
   3. Averiguar con personal de la faena el porcentaje de combustible aproximado en el estanque.
3. Con ayuda de una gata y mesa hidráulica se debe instalar se debe presentar el sensor para encontrar un punto de buena señal, tal como aparece en la fig. 8. Se debe instalar el sensor en el eje del válvula de despiche, a unos 20 o 25cm de este, y la prueba debe ser realizada con gel ultrasónico. Este procedimiento lo conoce bien Alberto.
4. Se ejecuta el script “fuelsensorDataRealtime\_finding\_optimal\_parameters” para obtener los 3 parámetros óptimos mencionados anteriormente. En algunos casos sucede que este sector del estanque esta obstruido, y esto puede verse en que se tiene una señal de rebote (o varias) a baja altura. Si el rebote se encuentra a menos de 50cm, se debe tomar como un punto con alta probabilidad de obstrucción. Por eso es siempre importante tener una referencia del nivel de combustible. Se debe tomar registro de los parámetros óptimos encontrados y guardar la data recopilada. Pensando en que con el movimiento del CAEX, un buen nivel de señal debiera al menos de 30% con un pga multiplicado por 10 (pga = 1).
5. Si el nivel de señal es buena en la zona 1, se debe continuar en el punto 6. Si el nivel de señal no es buena se debe seguir buscando en las mediaciones de la zona 1 o directamente buscar en la zona 2. Siempre es recomendable en primer lugar correr el script “fuelsensorDataRealtime\_finding\_optimal\_parameters” para obtener el periodo óptimo. Una vez teniendo este valor, si el nivel de la señal no es tan bueno o se tiene el tiempo, se puede realizar una búsqueda fina con el script “fuelsensorDataRealtime\_finding\_optimal\_parameters\_single”, utilizando el valor optimo encontrado previamente. Este script realiza actualizaciones del vector (altura del combustible, nivel de señal) cada 1 segundo, lo que permite una búsqueda de manera más precisa.
6. Una vez se consigue un buen nivel de señal es una buena práctica correr el script “fuelsensorDataRealtime\_finding\_optimal\_parameters\_single.m” en modo captura de señal completa y guardar la señal encontrada. Se debe marcar el sector encontrado ya que la marca podría desaparecer con el esmerilado.
7. Una vez marcado la zona de buena señal, de debe desmontar la gata hidráulica y retirar el sensor.
8. A continuación se procede a pegar el sensor a la estructura del estanque en el punto encontrado. Este procedimiento es bien conocido por Alberto.
   1. Se esmerilar la zona marcada donde se pegara el sensor.
   2. Usando la gata y mesa hidráulica se hace presión para pegar el sensor.
   3. Pegado del sensor utilizando loctite 330 y su respectivo activador.
9. Mientras el pegamento está fraguando, lo cual tarda unos 15 min, se puede aprovechar de volver a repetir las pruebas realizadas anteriormente con los scripts “fuelsensorDataRealtime\_finding\_optimal\_parameters.m” y “fuelsensorDataRealtime\_finding\_optimal\_parameters\_single.m”.



Fig. 8. Posicionamiento para prueba de señal.

## Conclusiones

De acuerdo a la experiencia se tiene haciendo pruebas los CAEX’s en diferentes faenas se puede concluir que: En la práctica de los 3 parámetros de interés el más relevante es el del periodo, ya que pequeños cambios genera grandes diferencias en el nivel de señal del rebote recibido. En general el periodo optimo va entre 860[ns] hasta 930[ns]. En la mayoría de los casos se encuentra entre 870[ns] y 890[ns]. El ancho (width) del pulso casi siempre está entre 70[ns] y 200[ns], el aporte de este no es tan relevante, variaciones pequeñas en este parámetro no producen grandes variaciones en el nivel de señal recibida. En el caso del número de pulsos, el aumento de este crece de manera casi lineal con el nivel de señal (voltaje) recibida como se puede ver en la fig. 7. Por lo tanto lo mejor es utilizar un número alto de número de pulsos, en las instalaciones de DGM y Collahuasi se utilizó 63 pulsos.

# Información adicional relevante

1. Si se quiere realizar de manera automática el seteo de los parámetros óptimos se debe considerar de que el camión en movimiento podría alterar las mediciones de señal
2. Pensando en la versión 1.7, para búsqueda más precisas del punto donde será instalado el sensor, es importante tener un programa que diga que vaya actualizando rápidamente el vector [altura del combustible, nivel de señal].
3. En la gran mayoría de los casos el nivel de la señal es mejor cuando con el gel ultrasónico que cuando está pegado. La señal podría llegar a disminuir hasta en un 30% respecto a la señal obtenida con gel.
4. En fallida primera instalación en DGM se tiene la sospecha de que ocurrieron dos eventos que hicieron que no se pudiera ver la señal de rebote.
   1. El sensor al estar afirmado con imán imposibilito una buena señal
   2. y/o había una obstrucción superior entre 90 y 160cm, ya que en la primera ida se tuvo muy buena señal a los 90cm, pero en la segunda ida cuando el nivel estaba en 160cm se tuvo nula señal en el mismo punto y se tuvo que instalar en un punto totalmente diferente. Esta es solo una hipótesis y sería la primera vez que pasa algo así, por lo tanto si fuera así esto puede considerarse como un caso muy puntual.
5. Las actuales instalaciones se puede ver que existe ruido en la señal recibida como se puede ver en las figuras 9, 10, y 11. Este ruido es aleatorio, va y viene. Los datos guardados pueden ser vistos en “REPOSITORY\_ROOT\_PATH source/matlab/fuelsensor\_scripts/Debug\_Online/fulldata\_obtenida\_con\_script\_debug\_online/collahuasi”. Lo curioso de los peaks de ruidos son dos cosas:
   1. Tienen componentes cercanas (por no decir iguales), a la componente de la frecuencia del pulso enviado por la tarjeta electrónica. Por lo que debe haber un loop en el sistema que este causando este problema.
   2. Como se puede ver en la figura 11, este ruido sigue apareciendo a pesar de que se deja de transmitir pulsos por la tarjeta. Por lo tanto existe una autoexcitación del piezo, seguramente producto del mismo ruido circundante.
6. Una recomendación para discriminar el pulso de los peaks de ruido seria aplicar la SDFT y luego filtrar por ancho de pulso. Ya que estos ruidos siempre son de considerablemente menor espacio de tiempo que el pulso enviado (con número de pulsos igual a 63).
7. Hace muy poco, justo antes de la instalación en Gabriela Mistral se encontró que el piso de ruido aumenta o disminuye considerablemente en casos particulares. Las conclusiones rápidas que se obtuvieron con muy poco tiempo de análisis (un día domingo en la noche a horas de tomar el avión al norte) fueron:
   1. Cuando la tarjeta se cambió la entrada de energía de “regulated” a “unregulated”, el piso de ruido disminuyo de manera significativa.
   2. Usando la misma tarjeta pero diferentes cases el piso de ruido también cambia significativamente. Cuando se saca la tarjeta del case y se hace funcionar todo normalmente, el piso de ruido disminuye significativamente. Una opción puede ser que el largo del cable que conecta el piezoeléctrico con la tarjeta puede estar jugando un papel importante, ya que en el case con menor largo de cable, el ruido fue considerablemente menor.

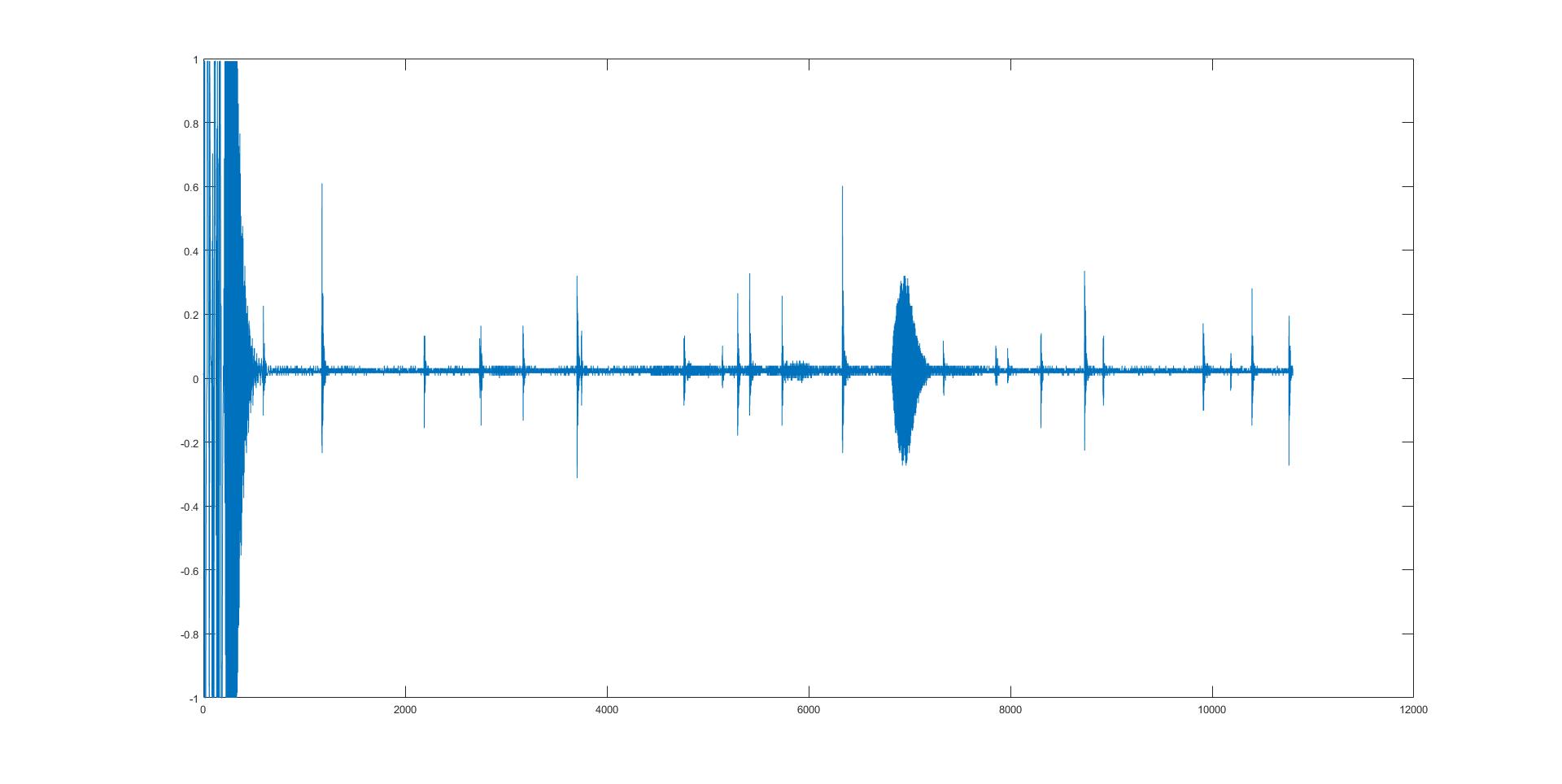


Fig. 9. Señal ruidosa en CAEX 100 de Collahuasi.

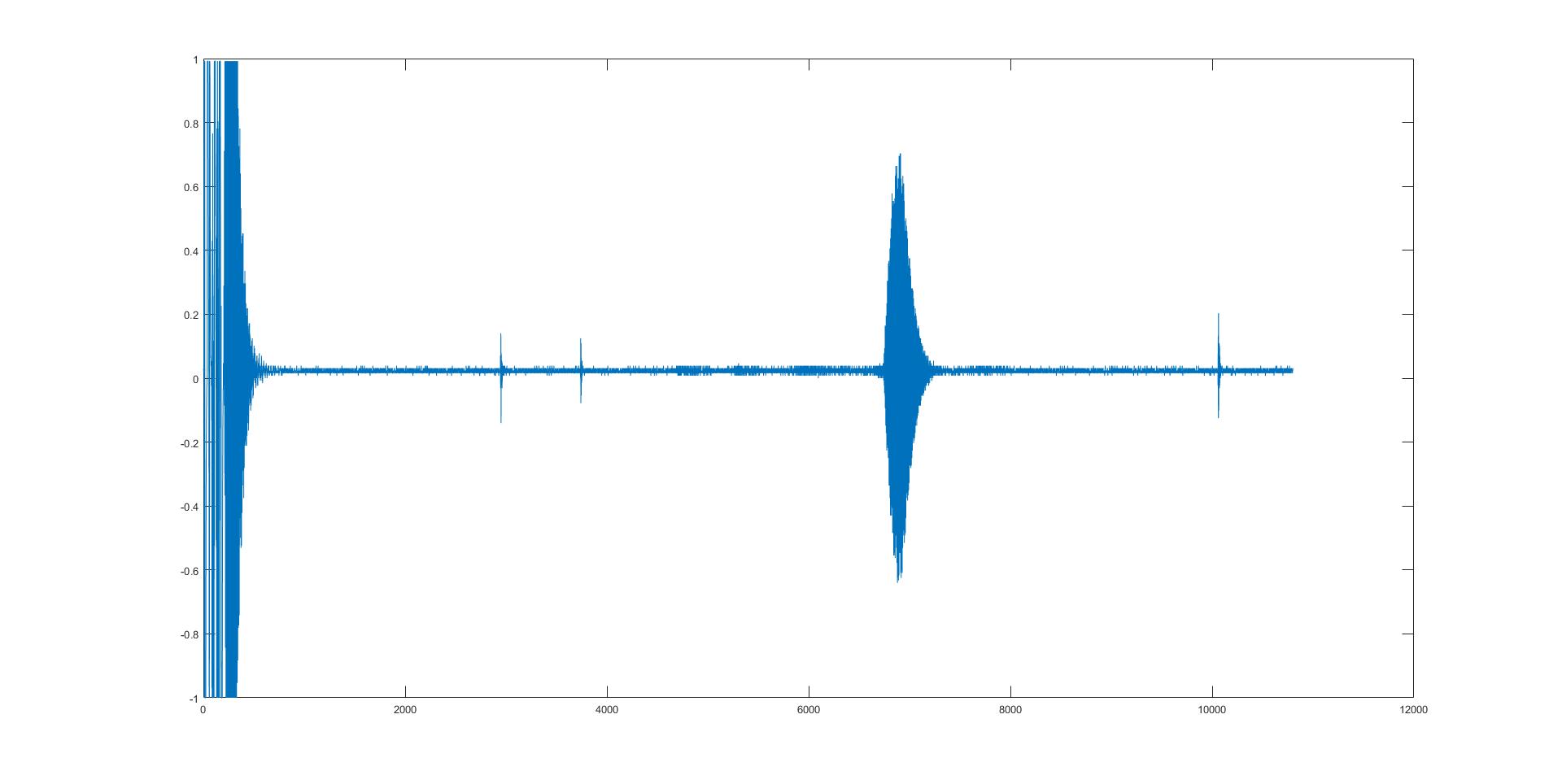


Fig. 10. Señal con muy poco ruido en CAEX 100 de Collahuasi.

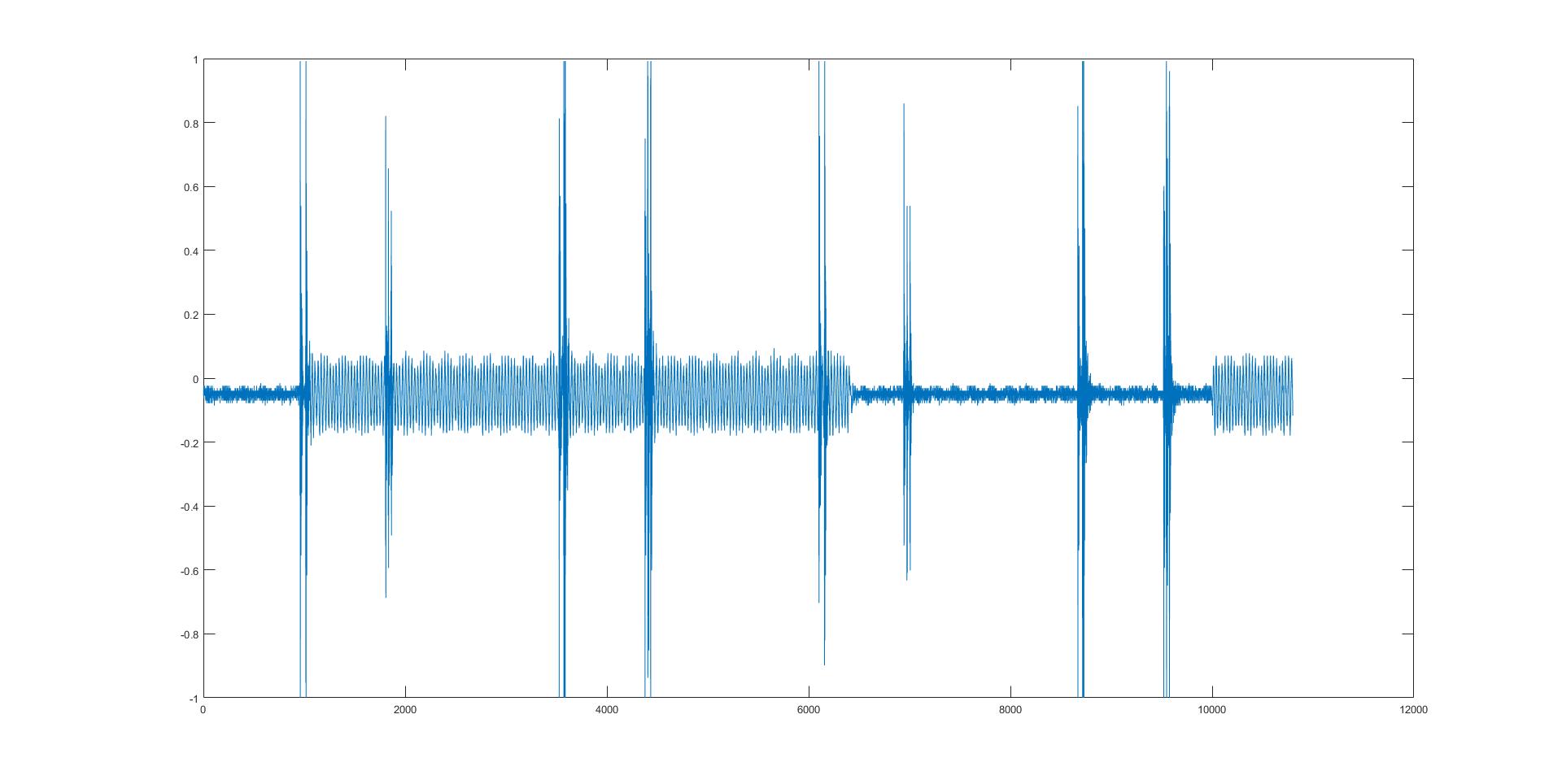


Fig. 11. Señal con mucho peaks de ruido en CAEX 79 de Collahuasi. El pulso no fue transmitido.