

OB1203

Algoritmo de pulsioximetría para SpO₂, frecuencia cardíaca y frecuencia respiratoria

Esta nota de aplicación presenta el algoritmo de SpO₂ (saturación de oxígeno en sangre periférica) y frecuencia cardíaca (FC) del OB1203, incluida la detección de la frecuencia respiratoria (FR). Se centra en la teoría de funcionamiento y los métodos utilizados.

Dispositivo de destino

El hardware es el OB1203SD-RL-EVK o OB1203SD-RL2-EVK Heart Rate, SpO₂, and Respiration Rate Evaluation Kit con el microcontrolador de 16 bits RL78/G13 (R5F100BG) y pantalla OLED. El código está en C. El entorno de compilación es E2 Studio, con un compilador CC-RL.

Nota: El algoritmo se proporciona como referencia, demostración y/o evaluación. No se garantiza ni se asegura su rendimiento, ni su idoneidad para ninguna aplicación de dispositivos médicos. Renesas no indemnizará a los clientes que utilicen cualquier parte del código de referencia proporcionado. El OEM es responsable del rendimiento de los productos fabricados con el código de referencia proporcionado.

Contenido

1.	Algoritmo	2
1.1	Algoritmos destacados	2
1.2	Contenido relacionado	2
1.3	Bucle de funcionamiento principal	2
1.4	Cronometraje	3
1.5	Programa principal Configuración	4
1.6	Flujo de algoritmos	4
2.	Descripción del algoritmo	5
2.1	Preprocesamiento	5
2.2	Algoritmo de frecuencia cardíaca	5
2.2.1.	Método del umbral de frecuencia cardíaca	5
2.2.2.	Método de la ventana para la frecuencia cardíaca	6
2.2.3.	Método de fusión	7
2.3	Algoritmo SpO ₂	8
2.3.1.	Mecanismo SpO ₂	8
2.3.2.	Corrección de la presión de los dedos	9
2.3.3.	Compensación del ruido	10
2.4	Algoritmo de frecuencia respiratoria	11
3.	OB1203 Recorrido del algoritmo	13
3.1	Filtros Kalman	15
3.2	Comprobación de la calidad de los datos	15
4.	Código de ejemplo	15
5.	Historial de revisiones	15

1. Algoritmo

1.1 Algoritmos destacados

El algoritmo de ejemplo OB1203 para frecuencia cardíaca (FC), saturación de oxígeno periférico (SpO2) y frecuencia respiratoria (FR) presenta varios métodos de procesamiento de señales únicos y potentes implementados para su uso en pequeños microcontroladores (MCU) como los MCU Renesas RL78 de 16 bits. También está disponible código de ejemplo para MCUs Renesas RA y dispositivos Bluetooth Renesas Dialog. El algoritmo se ejecuta en aproximadamente 20 ms en una MCU de 16 bits.

Las características incluyen:

- Detección eficaz (recursiva) de la frecuencia cardíaca (FC) basada en ventanas
- Métodos robustos y sencillos de cruce de umbrales para el seguimiento de la FC
- Detección de la presión dactilar y corrección de la saturación de oxígeno (SpO2)
- Medición de los cambios dinámicos de la respiración (frecuencia respiratoria)
- Filtro Savitsky-Golay "pobre" para eliminar el ruido
- Filtros Kalman para eliminar valores atípicos
- Control automático de la corriente del LED del sensor
- Acceso del usuario a todos los métodos y parámetros de ajuste del algoritmo

1.2 Contenido relacionado

- Para obtener información detallada sobre la configuración de las opciones del algoritmo, consulte *la Nota de aplicación del código de ejemplo del OB1203*.
- En la página del producto [OB1203](#) encontrará kits de evaluación, guías de diseño, muestras y código de ejemplo.

1.3 Bucle de funcionamiento principal

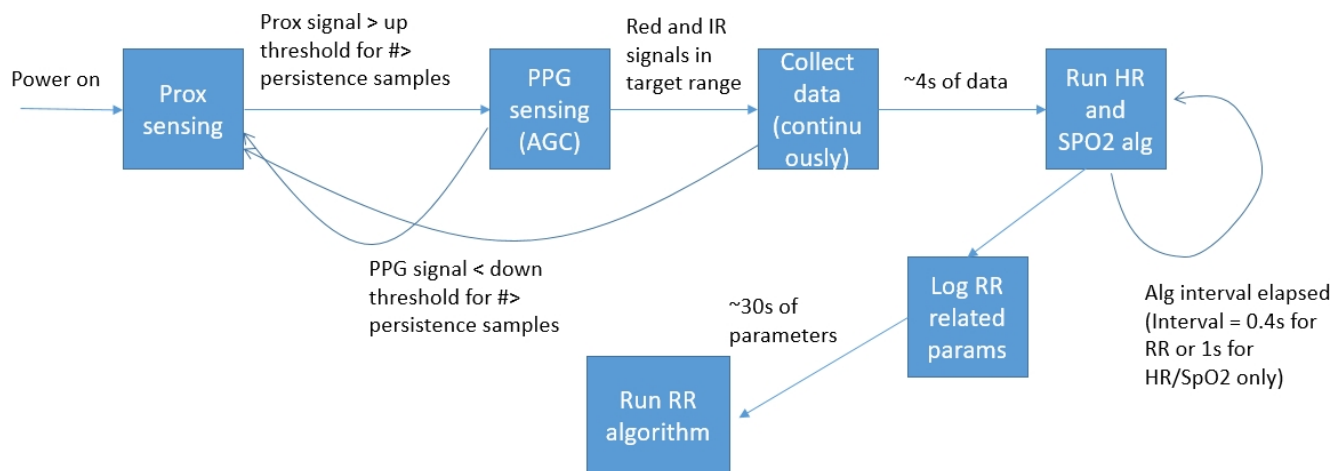


Figura 1. Funcionamiento de la aplicación principal

Tras pulsar el botón de inicio, la aplicación arranca en un modo de detección de proximidad de bajo consumo. Cuando el OB1203 detecta un dedo, activa el pin de interrupción (activo bajo). A continuación, el firmware de la aplicación cambia el OB1203 al modo PPG (PPG = fotopletismografía, la detección de sangre en el tejido con absorción de luz). Un bucle de control de ganancia automática (AGC) ajusta las intensidades de los LED para situarse dentro de un rango objetivo especificado de un punto de ajuste. Si es necesario para la piel oscura, se aumenta la ganancia del sensor. A continuación, la aplicación empieza a almacenar datos de la memoria RAM FIFO del sensor. Se almacenan unos segundos de datos antes de que se ejecute el algoritmo completo. A partir de entonces, el algoritmo se ejecuta cada número INTERVALO de muestras.

El algoritmo se ejecuta nominalmente a una velocidad de datos de salida de 100 muestras por segundo, pero también incluye una opción para ejecutarse a la mitad de la velocidad de muestreo (50 muestras por segundo). Para obtener más información sobre las opciones de configuración del algoritmo, consulte la *Nota de aplicación del código de ejemplo del kit de evaluación OB1203* en el sitio web de Renesas.

Para las mediciones de la frecuencia respiratoria (FR), el `INTERVALO` es de 40 (0,4seg). Para la frecuencia cardíaca (FC) y la saturación periférica de oxígeno (SpO2) solamente, el `INTERVALO` puede ser 100 (1seg). A partir de los datos medidos, se estiman la FC y la SpO2 y se miden las características de la forma de onda para calcular la frecuencia respiratoria (FR). El algoritmo de frecuencia respiratoria funciona con unos 30 segundos de datos de FC almacenados en la memoria intermedia. Es muy sensible al movimiento y sólo está indicado para sujetos inmóviles con buena estabilidad de los dedos y/o con una pinza en el dedo y sin movimiento.

Para mejorar la estabilidad y la calidad de los datos, el sensor suele cubrirse con un cubreobjetos o rodearse de un soporte para los dedos. Si la calidad de la señal se sale de un límite más amplio, la recogida de datos se detiene y la aplicación vuelve al modo de control automático de ganancia (AGC) para reajustar los niveles de corriente de los LED. Si se alcanza un nivel bajo persistente con los LED al máximo y la ganancia del sensor al máximo, la aplicación vuelve al modo de detección de proximidad.

En el código de demostración, se proporcionan las siguientes carpetas/bibliotecas/bloques de código:

- SPO2 - Este código contiene los algoritmos de frecuencia cardíaca, SpO2 y frecuencia respiratoria. El usuario llamará `spo2_init()` antes de comenzar las mediciones y luego llamará `add_sample()` después de obtener los datos del OB1203. Cada intervalo la aplicación principal debe llamar a `do_algorithm()`.
- KALMAN - Este código contiene una rutina de filtro de valores atípicos, que hace referencia a una matriz de estructuras donde cada estructura contiene los parámetros de un filtro diferente. Hay cinco filtros en la matriz `struct`: `umbral`, frecuencia cardíaca, `spo2`, frecuencia respiratoria y amplitud pico.
- OB1203 - Este código contiene las rutinas necesarias para configurar el OB1203 y leer los datos.
- SAVGOL - Este código contiene una variante de alta eficacia del filtro Savitsky-Golay para reducir el ruido sin distorsionar la forma de onda.
- OXIMETER - `Oximeter.c` es el programa principal con dos archivos `.h` de soporte: `oximeter.h` que maneja UART y cosas relacionadas con la pantalla y `oximstruct.h` que es donde se configuran todas las opciones de alto nivel y se ubican los parámetros para ejecutar el algoritmo, el estado del sensor y el control AGC.

1.4 Cronometraje

En el modo de biodetección, un bucle de control automático de ganancia (AGC) ajusta la intensidad del LED para alcanzar el nivel de señal deseado, aproximadamente el 80% de la escala completa. Esto es necesario para obtener una alta relación señal-ruido (SNR) de los controladores LED y del convertidor analógico digital. Los datos se leen después de cada resultado de conversión ADC (interrupción PPG). En este modo la corriente del LED se ajusta después de cada nuevo dato. Cuando los niveles de señal infrarroja y roja están ambos dentro del rango, el dispositivo cambia a "modo FIFO" donde 10 muestras cada una de rojo e infrarrojo (IR) son leídas de la RAM FIFO de datos del OB1203 cada 100ms. Los pasos del algoritmo y las actualizaciones de la pantalla se realizan entre las mediciones de las muestras.

La aplicación principal se ejecuta en `oximeter.c`. Configura el OB1203 en modo proximidad y luego espera una interrupción indicando que un dedo presente. Luego llama a `switch_mode()` para cambiar al modo PPG de detección de HR/SpO2. Durante el AGC, se utiliza una interrupción PPG para alertar al microcontrolador (MCU) después de adquirir cada muestra. Cuando los canales IR y Rojo están ambos dentro del rango, la operación cambia a "modo FIFO" para leer diez muestras del FIFO cada vez que se afirma la interrupción `FIFO_A_FULL`. Se sigue la siguiente secuencia, representada esquemáticamente en la Figura 2.

1. Espere la interrupción. Leer 10 muestras cada una de IR y rojo.
2. Llama a `do_algorithm()`.
3. Espere la interrupción. Leer 10 muestras cada una de IR y rojo.
4. Actualiza la pantalla.
5. Hasta que expire el `INTERVALO`: Espere la interrupción. Lea 10 muestras cada una de IR y rojo (es decir, para `INTERVALO` 40, lea muestras una vez más).

Si la visualización de la onda PPG está activada en `oximeter.c`, la parte de la onda PPG de la pantalla se actualiza después de la tercera medición de muestra.

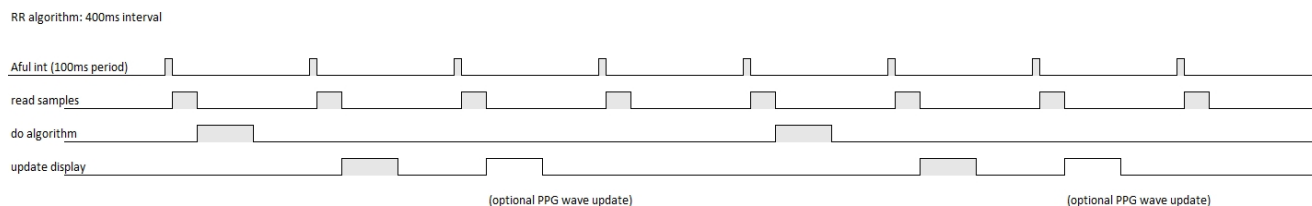


Figura 2. Diagrama de recogida de muestras y temporización del algoritmo

Los parámetros de funcionamiento recomendados para obtener la mejor SNR son una frecuencia de muestreo de 1600Hz tanto para IR como para Rojo, con 16 promedios en chip para una frecuencia de salida de datos de 100Hz, un valor de consigna cercano a 200.000 recuentos ADC y una lectura de la FIFO basada en interrupciones FIFO_A_FULL. Para más información, consulte Código de ejemplo.

1.5 Programa principal Configuración

En `oximstruct.h` hay varios ajustes configurables por el usuario que se establecen con líneas `#define` en el código C. Para obtener más información, consulte la *Nota de aplicación del código de ejemplo OB1203*.

1.6 Flujo de algoritmos

El algoritmo de demostración OB1203 ejecuta dos algoritmos complementarios en paralelo, como se muestra en la siguiente figura.

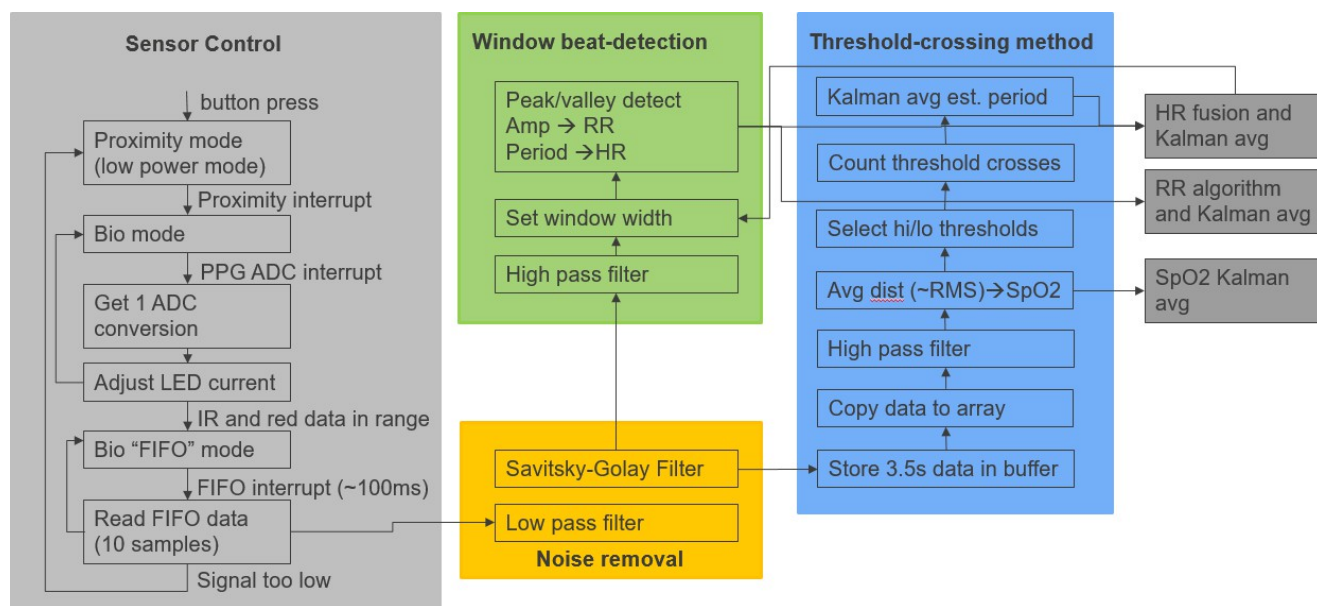


Figura 3. Flujo del algoritmo de funcionamiento de alto nivel de la demostración del OB1203

El preprocesamiento incluye un filtro de paso bajo de media móvil y un filtro de ruido Savitsky-Golay modificado.

A continuación, los datos filtrados se pasan a dos algoritmos, un filtro de paso alto, un algoritmo de detección de frecuencia cardíaca gruesa de cruce de umbral y un método de detección de latidos basado en ventanas.

El algoritmo latido a latido tiene la mayor precisión, pero es susceptible a los segundos armónicos procedentes de la detección de muescas dicroticas y requiere una SNR ligeramente mejor (perfusión). Para verificar la precisión de los métodos respectivos, se obtiene una estimación aproximada de la frecuencia cardíaca utilizando un filtro de paso alto y contando los cruces de umbral. Mediante la fusión de las dos mediciones independientes de la frecuencia cardíaca se obtiene un resultado coherente de la frecuencia cardíaca.

Nota: los resultados de SpO2 y frecuencia cardíaca utilizan un filtro Kalman de orden cero para eliminar los valores atípicos y generar una media de ocho segundos biológicamente relevante.

El código de demostración, incluidas las funciones de visualización, requiere unos 8-9kB de RAM y ~77kB de flash.

2. Descripción del algoritmo

2.1 Preprocesamiento

En el preprocesamiento de los datos, se utiliza una media móvil como filtro de paso bajo para limitar el ancho de banda del ruido a las frecuencias biológicamente relevantes para la PPG. Se utiliza una media móvil de cuatro muestras para datos de 50 muestras por segundo (sps) o una media móvil de ocho muestras para datos de 100sps. En frecuencias cardíacas altas (>120bpm) la longitud del filtro se reduce a la mitad y la longitud del filtro se restablece de nuevo por debajo de 110bpm.

Un filtro Savitsky-Golay (SG) modificado elimina el ruido de la señal. Un filtro SG realiza un ajuste de curva móvil a los datos, eliminando el ruido *sin distorsionar la amplitud de la señal*. Para señales que varían suavemente, como la PPG, cualquier sección de los datos puede ajustarse bien a un filtro cuadrático (filtro SG de orden 2nd). Sin embargo, el ajuste de curvas implica una pseudoinversión matricial que puede requerir operaciones de multiplicación y división de orden N^3 , donde N es el número de puntos que deben ajustarse a la curva en cada paso. Esto puede llevar demasiado tiempo para las MCU pequeñas. Sin embargo, si se utiliza exclusivamente el orden cuadrático, el ajuste de curvas de orden N^3 puede simplificarse a una solución exacta que requiera sólo unas pocas multiplicaciones mediante un inteligente cambio de operaciones. En lugar de resolver la curva que mejor se ajusta a todos los puntos N, calculamos un ajuste exacto a tres puntos espaciados utilizando sólo dos operaciones de multiplicación. A continuación, incrementamos esos tres puntos espaciados a través de los puntos a filtrar y promediamos los N ajustes de curva resultantes en el punto central. A continuación, añadimos un nuevo punto de datos, eliminamos el más antiguo y repetimos los N ajustes de curva y el promedio para el nuevo punto de datos central. Si se eligen intervalos de muestreo (número de puntos que deben filtrarse) que sean potencias de dos (por ejemplo, 4, 8, 16), el promediado sólo implica una suma y un desplazamiento de bits. Este filtro SG "de pobre" se ejecuta en tiempo de orden N, lo que supone una aceleración cuadrática respecto a los métodos publicados. El algoritmo ejecuta el filtro SG en hasta 16 muestras a 100sps u 8 muestras a 50sps. A frecuencias cardíacas altas, la longitud del filtro se reduce a la mitad para admitir el ancho de banda de señal necesario.

2.2 Algoritmo de frecuencia cardíaca

2.2.1. Método del umbral de frecuencia cardíaca

El método de cruce del umbral para la frecuencia cardíaca (como el recuento de cruces por cero) cuenta el número de veces que la señal cruza un valor umbral. Dado que la señal PPG es asimétrica, movemos el umbral ligeramente lejos de cero para evitar contar los cruces por cero de los picos secundarios cerca de las muescas dicróicas.

A partir de una memoria intermedia de 3,5 segundos con datos de 100 muestras por segundo, calculamos una media móvil centrada de 64 muestras y la restamos de los datos para obtener un resultado filtrado de paso alto. La figura 4 muestra el efecto del filtro de paso alto en los datos de frecuencia cardíaca baja y alta. El efecto es similar al de una derivada y desambigua la forma de onda pulsátil de la deriva o el movimiento de la línea de base. En particular, este método evita detectar oscilaciones en la señal PPG que siguen a la sístole (muescas dicróicas) como latidos cardíacos. Obsérvese que los datos están sin procesar, no invertidos como suele hacerse con las pantallas de los sensores PPG, por lo que la sístole aparece como un valle en estos datos.

El umbral se elige como 1/3-1/2 (ajustable por el usuario) de la señal PPG máxima en el búfer de datos PPG.

Con los datos filtrados de paso alto, medimos la distancia media de los datos desde cero. Se trata de una estimación eficaz, de tipo RMS, de la amplitud de CA de la señal, que se utiliza en el cálculo de la SpO2.

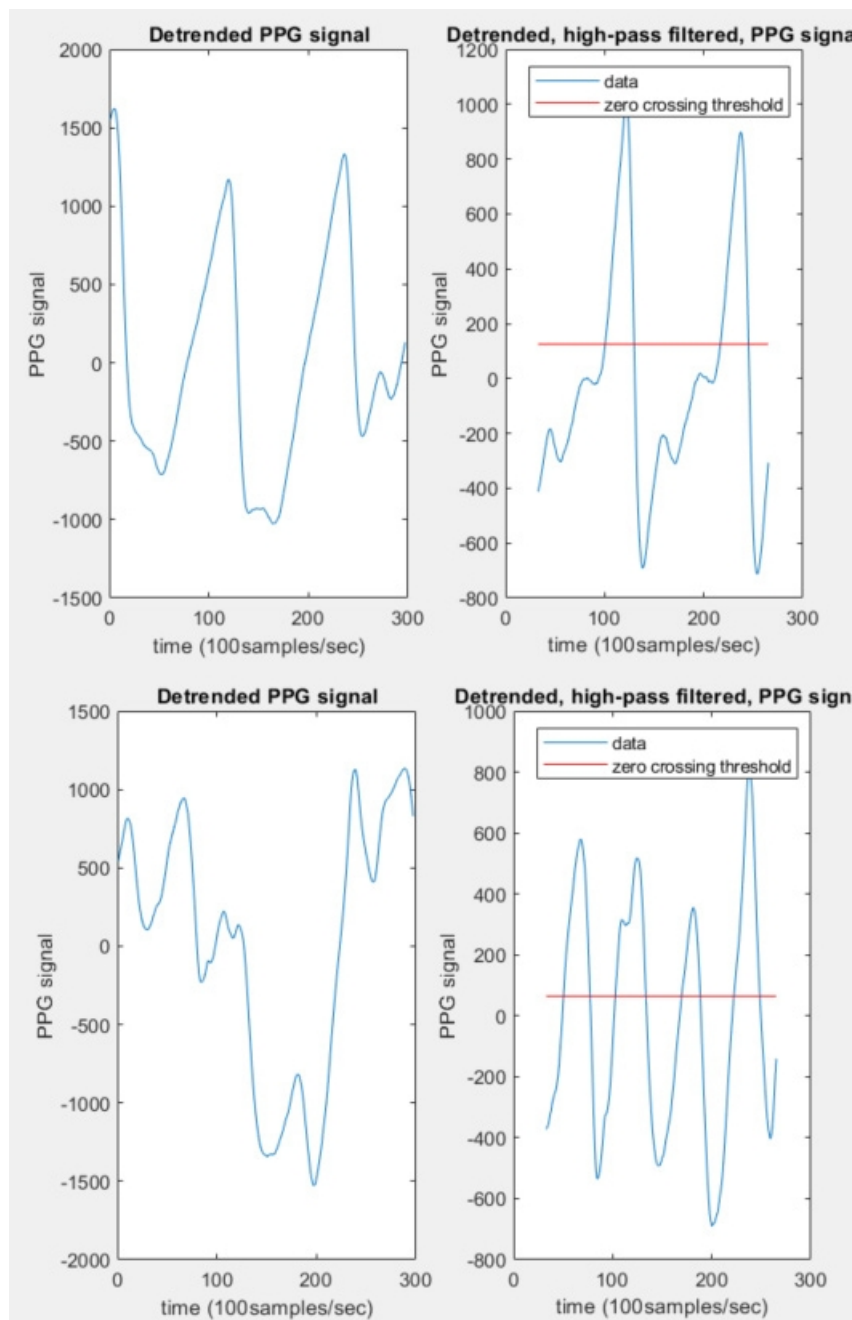


Figura 4. Método de cruce del umbral

En la práctica, una presión excesiva de los dedos puede distorsionar la forma de onda y la muesca dicrótica puede aparecer elevada en la señal y cruzar el umbral, lo que da lugar a una sobreestimación de la frecuencia cardíaca. Para evitarlo, utilizamos dos umbrales: uno positivo y otro negativo, y elegimos el más bajo de los dos recuentos de cruce de umbral. Además, para frecuencias cardíacas por debajo de 140 lpm, utilizamos una función de "no rebote", en la que no se cuenta ningún cruce del umbral que se encuentre a menos de la mitad de la frecuencia cardíaca estimada actual del cruce anterior en la misma dirección.

2.2.2. Método de la ventana para la frecuencia cardíaca

La detección del latido real puede ser un reto para las formas de onda PPG debido a un pico secundario que sigue a la característica de muesca dicrótica. Para intentar encontrar los picos correctos, se puede utilizar una lógica compleja que compare las alturas de los picos, el tiempo transcurrido desde el último pico, etc., o bien los filtros de seguimiento pueden intentar suavizar los picos secundarios.

Sin embargo, nuestro método utiliza una "ventana móvil" extremadamente sencilla, rápida y robusta para detectar los verdaderos picos (diástoles) y valles (sístole) en los datos entrantes del sensor. (Tenga en cuenta que, en los equipos hospitalarios, la forma de onda PPG se invierte, de modo que las sístoles aparecen como picos, ya que esto se correlaciona

con la presión en el

arterias). El método de ventana se ejecuta cada intervalo de medición (por ejemplo, 40 muestras cada 400 ms). Los pasos son los siguientes:

1. Ajuste la anchura de la ventana a 2/3 del periodo estimado actual del latido. Inicialmente, el periodo estimado se determina basándose en el método de cruce del umbral. Posteriormente, el periodo del latido cardíaco se determina mediante una fusión de las estimaciones de la ventana y del cruce del umbral. La selección correcta del ancho de la ventana es *fundamental* para detectar picos verdaderos y no falsos con el método de la ventana.
2. Para cada nueva muestra, compruebe si es mayor que el pico anterior o menor que el valle anterior. Si una muestra sigue siendo un pico o un valle durante todo el ancho de la ventana, se trata de un pico o un valle verdadero.
3. Manejar adecuadamente casos especiales como cambios en el ancho de la ventana.

Tenga en cuenta que este método no requiere una memoria intermedia de las muestras de la , ni comparar la nueva muestra con todas las muestras de la ventana. Basta con registrar las posiciones y los valores del pico y el valle candidatos y compararlos con la última muestra. Como resultado, este método es extremadamente rápido y consume muy pocos recursos del sistema.

Visualmente, si mostramos el máximo y el mínimo de la ventana, el número parecerá "congelarse" cuando alcancemos un pico o un valle y luego se liberará cuando la ventana se mueva fuera del pico o del valle. Si el valor se congela durante todo el ancho de la ventana, se trata de un verdadero pico o valle.

La razón por la que este método funciona es que la muesca dicrótica se encuentra dentro del ancho de una ventana de la característica principal de la sístole (valle en los datos brutos, pico en las pantallas de los equipos hospitalarios). Por lo tanto, el pico secundario no es el valor extremo para todo el paso de la ventana a través de la característica, y no califica como un verdadero extremo (es decir, no se "congela" el tiempo suficiente). La diferencia pico-valle, a veces llamada índice de variabilidad de la pleth (PVI), se registra para su uso en la detección de la frecuencia respiratoria.

Obsérvese que la eliminación del ruido mediante el filtro de paso bajo de media móvil y el filtro SG es beneficiosa para la precisión de los picos y valles.

2.2.3. Método de fusión

Este método de ventana es un ejemplo de filtro de seguimiento. En la práctica, cuando está "activado", el método de la ventana es siempre más preciso que el método del cruce del umbral. Sin embargo, por sí mismo, el método de la ventana no ofrece ninguna garantía de que la ventana se haya fijado correctamente. Si se fija inicialmente basándose en una estimación o suposición incorrecta del periodo del latido, este método también puede bloquearse en un subarmónico, o segundo armónico. Por lo tanto, el algoritmo necesita una forma de "salir" cuando esto ocurre.

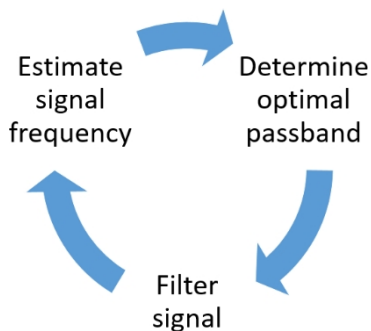


Figura 5. Principio del filtro de seguimiento

La solución aplicada en la demostración consiste en fusionar varios algoritmos con ventajas complementarias. El método de cruce del umbral, aunque menos preciso, no hace suposiciones sobre la periodicidad de la señal. Por lo tanto, absorbemos una pequeña disminución de la precisión en los raros casos en los que el cruce del umbral se ve engañado por una respiración fuerte o un movimiento, a cambio de la capacidad de "salirse" y evitar errores graves.

Además, un filtro Kalman de orden cero (valores atípicos) reiniciará el algoritmo si se producen demasiados valores atípicos , lo que permitirá al filtro readquirir y seguir más rápidamente la frecuencia correcta.

El método de fusión de frecuencia cardiaca de la función `calc_hr()` combina las estimaciones de frecuencia cardiaca del método de cruce de umbral y del método de ventana.

La lógica de la fusión es la siguiente:

1. Si el método de la ventana no produce un latido, utilice la estimación del cruce del umbral.
2. Si el método de la ventana está a más del 25% de la estimación del cruce del umbral, ignore el resultado del método de la ventana. Tenga en cuenta que los errores tienden a ser la omisión de un latido, como un latido ectópico (latido omitido) o el recuento de una muesca dicrótica durante el movimiento, por lo que si la medición es errónea, el error suele ser del orden del 100%. Cuando se eliminan esas detecciones extrañas, la varianza de cualquiera de los dos métodos desciende a menos del 25%.

2.3 Algoritmo SpO2

La PPG reflectante se mide contra la piel con el emisor y el receptor en el mismo lado, de forma similar a un sensor de proximidad infrarrojo (IR) activo. Parte de la señal PPG se deriva del cambio del volumen sanguíneo en el tejido, aumentando la absorción a medida que los vasos situados aguas arriba de los capilares se llenan tras el latido del corazón. Otra parte de la señal deriva de la aglomeración de células a velocidades de flujo sanguíneo bajas y de la desagregación a velocidades de flujo más altas.

La medición de la PPG reflectante, aunque cómoda en dispositivos móviles, no es óptima para personas con dedos muy blandos, incluidos niños pequeños y algunas personas mayores.

La señal observada tiene un gran componente estático o de CC sobre el que aparece la pequeña señal cardiovascular de CA. La relación CA/CC se denomina índice de perfusión (IP).

$$PI = \frac{IR_{AC}}{IR_{DC}} * 100\%$$

El índice de perfusión aumenta cuando hay más sangre "perfundida" en el tejido. El número real utilizado para el PI depende de varios detalles, como la longitud de onda y la configuración del LED/receptor. PI es importante porque los instrumentos y algoritmos tienen un límite inferior para el índice de perfusión por debajo del cual el cálculo de SpO2 pierde precisión.

En los sistemas transmisivos que emiten luz a través del dedo, el índice de perfusión de transmisión (tPI) es del orden de varios puntos porcentuales, lo que significa que la ondulación de CA es entre 10 y 100 veces menor que la señal de CC.

Los sistemas reflectantes miden la PPG con la fuente de luz y el detector desde el mismo lado del dedo. Con una región de muestra más pequeña y una longitud efectiva del trayecto de absorción más corta, se observa una menor modulación de la PPG a partir de la sangre pulsante. Por lo tanto, el índice de perfusión reflectante (rPI) es necesariamente menor que el tPI.

Obsérvese que el IP reflexivo y transmisivo no son físicamente equivalentes, ya que se miden diferentes regiones tisulares. Aun así, las mediciones de perfusión tPI y rPI para dispositivos transmisivos y reflectivos son útiles para evaluar la validez de las respectivas mediciones de SpO2.

Tenga en cuenta que, en el algoritmo de ejemplo, el rPI se estima a partir de la desviación media o "RMS" de la señal infrarroja y el nivel medio de recuento de CC. El valor de rPI mostrado (`avg_PI`) se da en porcentaje equivalente de pico a pico del nivel de CC e incluye una media móvil de ~5 segundos.

2.3.1. Mecanismo SpO2

La medición de la saturación de la sangre por el oxígeno se realiza con mediciones PPG en dos longitudes de onda. Las longitudes de onda más comunes son el rojo y el IR, debido a la fuerte absorción diferencial, disponibilidad de fuentes de luz, espectros relativamente planos (para la tolerancia al cambio de longitud de onda). Esencialmente, la absorbancia del canal rojo cambia con la oxigenación, pero la del IR no, por lo que podemos utilizar el IR como nivel de referencia estable.

El nivel de saturación de oxígeno en sangre o nivel de SpO2 se calcula a partir de una curva de calibración que traza la SpO2 frente a la relación de señales rojas e IR. Para ello, se calibra con un instrumento conocido o con una referencia de análisis de sangre en un laboratorio de hipoxia mediante una prueba de respiración descendente. Este trazado de calibración, también conocido como "curva R", depende de los detalles mecánicos de un sistema PPG.

La relación de proporciones "R" viene dada por:

$$R = \frac{R_{AC} R_{DC}}{IR_{AC} IR_{DC}}$$

La SpO2 se calcula con coeficientes positivos (en una aproximación lineal) como:

$$SpO_2 = (c_{(0)} - c_{(1)}R) * 100\%$$

Pueden utilizarse términos de orden superior o una tabla de consulta. Fisiológicamente, menos oxígeno significa una mayor señal roja de CA.

El valor R o "relación de relaciones" es eficaz para anular las variaciones de CC debidas a la colocación de los dedos, el tono de la piel, etc., al tiempo que preserva los cambios específicos de la química sanguínea en la relación de las señales pulsátiles (CA).

Si cambia algo en el diseño mecánico u óptico (por ejemplo, diferente color, forma o materiales de la carcasa), o incluso la presión de la pinza de dedo, debe rehacerse el procedimiento de calibración de la prueba de hipoxia para obtener una nueva curva R.

La SpO2 requiere una SNR mucho mayor que la frecuencia cardíaca porque no sólo estamos intentando detectar el latido cardíaco, que es una pequeña ondulación sobre un gran fondo, sino que estamos intentando cuantificar el tamaño de esa ondulación y dividirlo por otra ondulación muy pequeña. Es importante un control estricto de la longitud de onda del LED rojo, o una calibración específica de la longitud de onda. Además, los artefactos debidos al movimiento de los dedos deben identificarse y eliminarse de los datos o compensarse.

La señal de CA puede estimarse a partir de las distancias pico-valle; sin embargo, esto requiere datos con poco ruido. Un método alternativo que es robusto a niveles moderados de ruido, pero susceptible a cambios rápidos de CC, es calcular la desviación cuadrática media (RMS), también conocida como desviación estándar, de la señal. También tiene la ventaja de ser numéricamente robusta al ruido de digitalización en señales de menor amplitud. En nuestro caso, calculamos la distancia media desde cero, ya que es más eficiente numéricamente, aunque conceptualmente similar a la RMS.

Esta estimación de la amplitud de la señal de CA puede obtenerse a partir de los datos filtrados de paso alto en el método de ventana recursiva, o a partir de los datos filtrados de paso alto en el método de cruce de umbral. En teoría son idénticos; sin embargo, un método totalmente recursivo es más susceptible a los errores de datos que continúan propagándose, mientras que la amplitud del método de cruce de umbral se calcula ab-initio a partir del buffer cada vez, por lo que cualquier error inesperado no se propaga. Por lo tanto, se prefiere la amplitud de señal del método de cruce de umbral.

Para el algoritmo de SpO2, se utiliza un filtro de Kalman asimétrico especial que permite saltos más altos hacia arriba que hacia abajo. Esto se debe a que la oxigenación del cuerpo humano puede volver a la normalidad en una sola respiración tras una hipoxia, mientras que la disminución de la SpO2 requiere un tiempo mucho mayor.

Para un índice de perfusión muy bajo, los valores de SpO2 estimados serán ligeramente inferiores a los esperados, debido al efecto del ruido o de la deriva que afecta más a la señal roja más pequeña que a la señal infrarroja, siendo por tanto el valor R superior al esperado y el valor de SpO2 resultante inferior.

2.3.2. Corrección de la presión de los dedos

La calibración del dispositivo de SpO2 depende de la presión ejercida por el dedo. Este efecto es mayor en los dispositivos PPG reflectantes, que tienen una región de detección más pequeña. Cuando el diseño mecánico de la pinza para el dedo no es óptimo, un dedo grande ejerce más presión sobre el sensor que un dedo pequeño. Otros casos de uso pueden depender de que el usuario proporcione la presión correcta.

Normalmente, una presión más alta hace que la estimación de SpO2 sea más baja de lo esperado. Una presión más alta es visible en los datos de la forma de onda PPG sin procesar como una "cuadratura" de la forma de onda en la región de diástole. Empieza a tener un aspecto más trapezoidal. Si se puede cuantificar este efecto, se puede estimar y corregir la sobrepresión del dedo.

Ostensiblemente, caracterizar la forma de la onda de esta manera requeriría algún tipo de análisis de tiempo-frecuencia para caracterizar el contenido armónico o el aprendizaje automático, sin embargo, hay una manera más sencilla. Cuando se elimina una línea de base de media móvil de la forma de onda PPG, especialmente si la anchura de la media móvil es menor que el periodo del latido, el efecto de la presión de los dedos se hace evidente en la relación pico-valle.

El algoritmo OB1203 incluye un método opcional de corrección de la sobrepresión de los dedos basado en la relación pico-valle.

Cuando está activada, la corrección de la presión de los dedos aplica una compensación lineal al valor de SpO2 notificado para relaciones de amplitud valle-pico superiores a un mínimo.

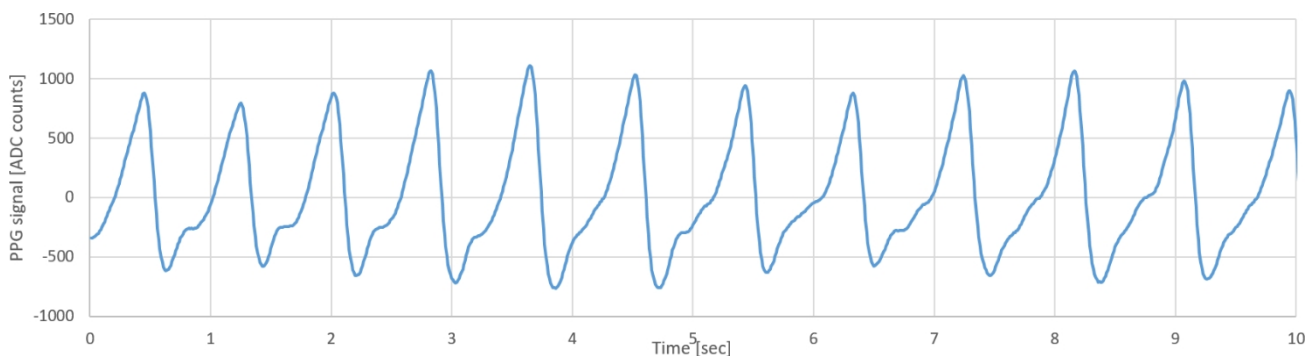


Figura 6. Forma de onda de PPG de presión dactilar baja tras filtrado de paso alto (eliminación de línea de base) que muestra picos de diástole con mayor amplitud que los valles de sístole.

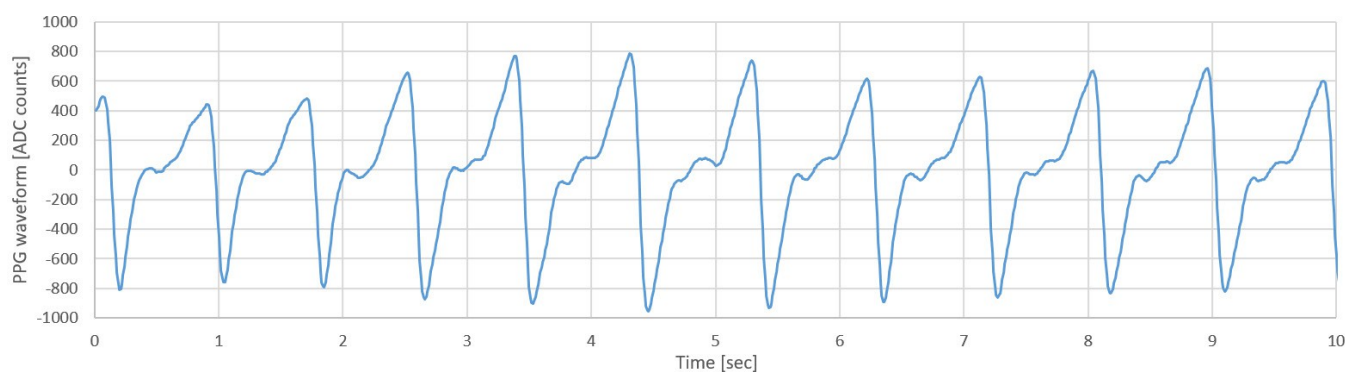


Figura 7. Forma de onda PPG de alta presión dactilar después del filtrado de paso alto (eliminación de la línea de base) que muestra aumento de las amplitudes del valle de sístole relativamente mayores.

2.3.3. Compensación del ruido

El ruido de medición afecta a la precisión de la SpO2 en niveles bajos de perfusión. El ruido se añade a la señal en cuadratura.

$$Meas^2 = Signal^2 + Noise^2$$

Cuando la señal es mucho mayor que el ruido, el efecto del ruido en la medición es despreciable. Sin embargo, a medida que la señal se aproxima al ruido de fondo, el ruido domina la amplitud de la medición o RMS. En las mediciones de SpO2, la señal del canal rojo suele ser aproximadamente la mitad que la del canal infrarrojo. En consecuencia, el canal rojo pasa antes a estar dominado por el ruido. Su valor ya no disminuye en proporción a la señal.

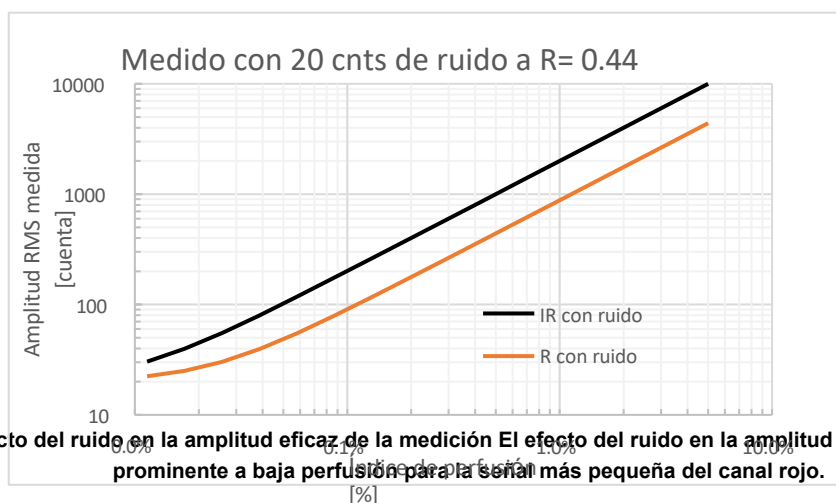


Figura 8. Efecto del ruido en la amplitud eficaz de la medición. El efecto del ruido en la amplitud RMS de la medición es prominente a baja perfusión para la señal más pequeña del canal rojo.

La SpO2 es inversamente proporcional a la entre el rojo y el infrarrojo, por lo que, en presencia de ruido, el valor de SpO2 estimado es inferior al esperado.

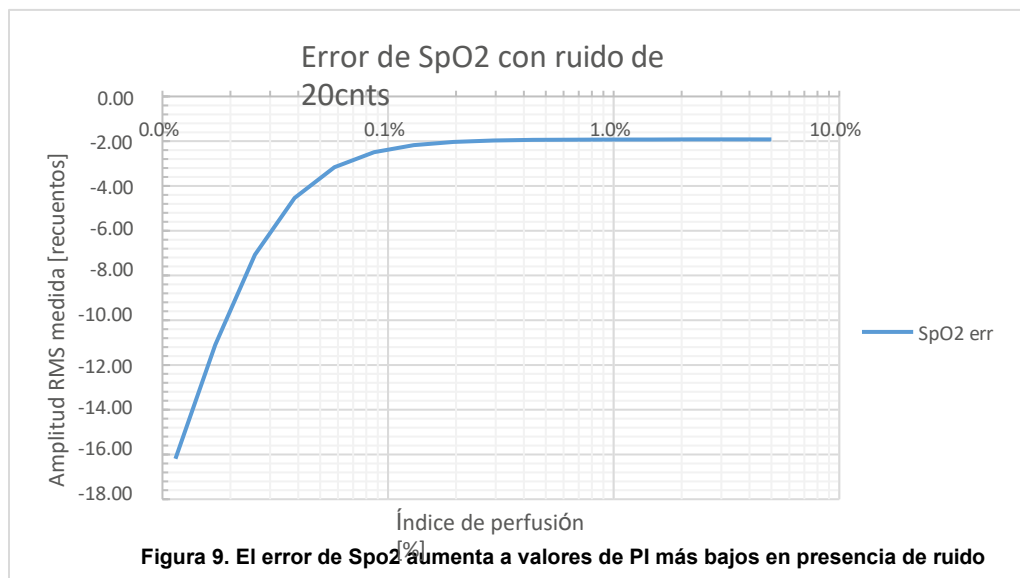


Figura 9. El error de SpO2 aumenta a valores de PI más bajos en presencia de ruido

El ruido de medición se escala principalmente con la intensidad del LED cuando la fuente limitante de ruido es el ruido de transmisión (Tx). Este es el caso para corrientes de LED superiores a unos 50 mA y valores de recuento superiores a unos 60.000 (con ganancia 1 del ADC). Por lo tanto, el ruido de medición para cada canal será proporcional a la corriente del LED.

El algoritmo puede compensar parcialmente el efecto del ruido de la medición en la SpO2 restando el ruido estimado de la medición.

$$\text{Estimated signal} = \sqrt{\text{Meas}^2 - \text{estimated noise}^2}$$

Esto puede reducir el efecto del ruido en las mediciones de SpO2 a baja perfusión.

Nos aseguramos de que las mediciones y el ruido sean equivalentes; por ejemplo, tanto la señal como el ruido deben estimarse del mismo modo (por ejemplo, RMS, desviación media o pico a pico). En el algoritmo se utiliza la desviación media.

Para medir el ruido de la señal, se coloca una tarjeta de objetivo sobre el sensor y se exportan los datos sin procesar. Para medir el ruido en el ancho de banda correspondiente, aplicamos una media móvil de ocho muestras y eliminamos una línea de base móvil centrada de 64 muestras, de forma idéntica a como se procesa la señal en el algoritmo de SpO2. A continuación, calculamos la desviación media. El ruido medido variará de un dispositivo a otro en función del ajuste del LED, por lo que para cualquier diseño de hardware la estimación del ruido no será perfecta y la varianza aumentará (el ruido es aleatorio). Para las aplicaciones del cliente, debe establecerse un límite inferior en el índice de perfusión por debajo del cual no se informarán los valores de SpO2, normalmente en el rango de 0,05% a 0,1%, dependiendo del rendimiento de ruido alcanzado (consulte la opción "CHECK_PI" de la *Nota de aplicación del código de ejemplo OB1203 RL78*).

2.4 Algoritmo de frecuencia respiratoria

La frecuencia respiratoria puede medirse mediante la modulación de los latidos del corazón en función de la cantidad de aire en los pulmones, lo que crea el "ritmo sinusal" en el ritmo cardiaco. Cuando los pulmones están llenos, los barorreceptores del corazón hacen que éste emita latidos más frecuentes con un volumen sistólico menor. La reducción del volumen sistólico aumenta el nivel de CC de la señal PPG. Esto se denomina "oscilación de la línea de base" o variación de intensidad (IV) inducida por la respiración. Asimismo, la variación del volumen sistólico modula la amplitud de la PPG (AM), mientras que la variación de la frecuencia cardiaca produce una modulación de la frecuencia (FM), como se muestra en la Figura 10.

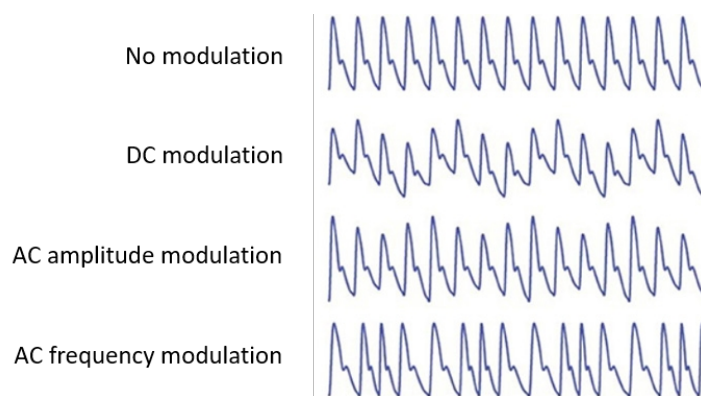


Figura 10. Modulación de la PPG por la respiración

El seguimiento del valor pico de la sístole (valle en los datos brutos) o de la amplitud pico-valle codifica la modulación de la amplitud y es un método robusto de seguimiento de la respiración. Esta modulación de la amplitud se denomina variabilidad de la plejía y el tamaño de la modulación se denomina índice de variabilidad de la plejía o IVP.

Es similar a la demodulación de audio a partir de una señal de radio AM. Los latidos del corazón son la onda portadora. La envolvente que varía lentamente (PVI) se debe a la respiración.

Para la detección de la respiración, la estimación de la amplitud de la envolvente procede de las diferencias pico-valle medidas en el método de la ventana. Una vez que tenemos un búfer de valores de amplitud de latido, buscamos cambios periódicos. Esencialmente, estamos detectando una pequeña ondulación lenta (respiración) sobre una pequeña ondulación (latido) en la señal PPG. También nos enfrentamos al reto de que la respiración es irregular, además de una deriva más aleatoria y efectos autónomos (biológicos subconscientes) extraños. En resumen, es mucho más difícil que detectar los latidos del corazón.

Por ejemplo, si intentamos utilizar la variación de frecuencia, más fácil de detectar, nos encontramos con que la modulación FM también está sujeta a variaciones autonómicas no relacionadas con la respiración. También puede tener dificultades para seguir una respiración rápida. Por este motivo, preferimos la PVI (amplitud del pulso). Los requisitos de precisión del PVI son exigentes, por lo que esta medición no tiene en cuenta el movimiento del usuario.

Además, debido a las variaciones autonómicas aleatorias, normalmente no es posible distinguir la "ausencia de respiración" de la "respiración lenta", por lo que las mediciones de la respiración basadas en PPG tendrán un límite inferior de la frecuencia respiratoria en torno a las 5rpm (respiraciones por minuto, que no deben confundirse con las revoluciones por minuto de los motores de automóvil).

Dado que registramos los valores de IVP a una velocidad fija (INTERVALO/RATE_MUESTRA) y la frecuencia cardiaca no es conmensurable, obtenemos una "fluctuación" significativa además del ruido de desplazamiento de la señal debido a los cambios físicos. Por lo tanto, los métodos de recuento de picos que dependen de una señal periódica son algo problemáticos para la determinación de la RR. Por ejemplo, una transformada rápida de Fourier (FFT) suele generar múltiples picos de respiración posibles y la elección del pico correcto requerirá cierto conocimiento de la historia de la señal. Entre los métodos más avanzados se incluyen las transformadas wavelet, la compresión sincronizada y la autorregresión.

Sin embargo, para nuestros propósitos, un método de recuento de cruces por cero conveniente para procesadores de 16 bits resulta ser robusto, *siempre que se utilice un filtro de eliminación de línea de base adecuado*. Esto plantea el problema del "huevo y la gallina", en el que la falta de información sobre RR significa que no podemos seleccionar el filtro adecuado para detectar RR. Algunos de los métodos clave en el algoritmo de detección de RR demostrado se refieren a cómo se ajustan los filtros con el fin de (1) adquirir rápidamente, (2) ajustarse a los cambios, y (3) proporcionar una estabilidad razonable de la señal. Tenga en cuenta que no todos los algoritmos de dispositivos comerciales pueden ajustarse a cambios repentinos en RR.

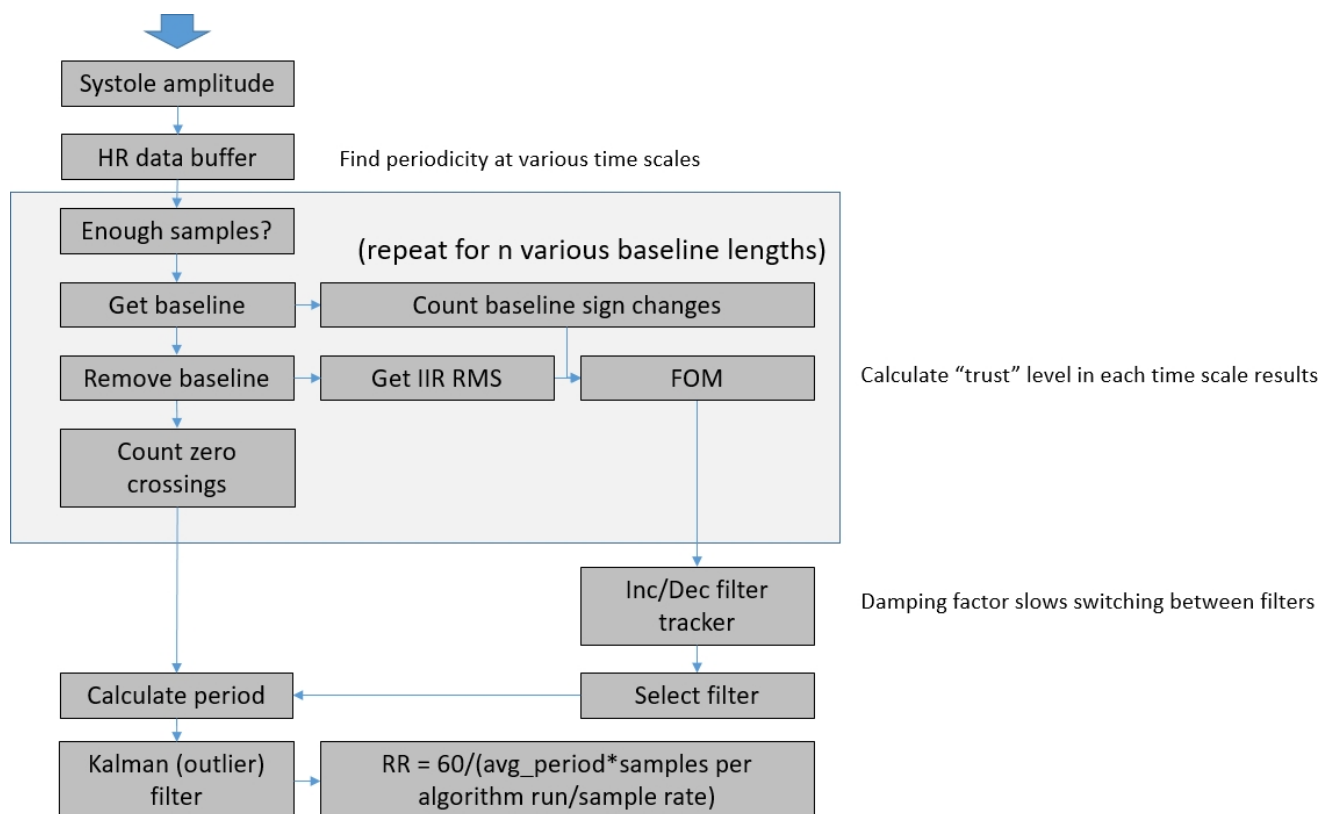


Figura 11. Flujo del algoritmo de frecuencia respiratoria

El parámetro crítico en un método de paso por cero es la longitud de la línea de base utilizada para nivelar o eliminar la tendencia de los datos. Como la tasa de respiración no se conoce a priori, no se puede elegir un filtro de longitud adecuado.

Dado que la continuación del cruce por cero es tan rápida y sencilla, podemos calcular los cruces por cero para una variedad de longitudes posibles de la línea de base que van desde valores cortos, ideales para detectar respiraciones rápidas, hasta valores largos de la línea de base, ideales para detectar respiraciones lentas. Para elegir el filtro adecuado, calculamos una figura de mérito basada en la relación entre la estabilidad de las estimaciones de RR resultantes (tiende a favorecer frecuencias respiratorias más lentas) y el número de cambios de signo en la línea de base (tiende a favorecer frecuencias respiratorias más altas). Para la estabilidad, un "factor de amortiguación" limita la velocidad con la que permitimos cambios en la longitud del filtro de la línea de base.

El software de aplicación ejecuta una media móvil de longitud variable en la frecuencia respiratoria calculada para mejorar la estabilidad de la salida. Puede ajustarse en mayor o menor medida para obtener más estabilidad o menos retraso.

Debido a la naturaleza del muestreo de la frecuencia respiratoria a partir de los valores máximos, no podemos detectar con precisión respiraciones más rápidas que el doble de la frecuencia cardiaca. Esto es similar al criterio de Nyquist. Basándonos en nuestra limitada frecuencia de ejecución del algoritmo, las respiraciones superiores a 30rpm pueden subestimarse, aunque en el caso de frecuencias cardiacas muy elevadas pueden observarse respiraciones de hasta 40rpm. La frecuencia cardiaca mínima es de 5rpm.

Importante: El algoritmo RR requiere muy poco movimiento. Asimismo, los cambios bruscos en la amplitud o frecuencia de la respiración reducirán temporalmente la precisión.

Los datos de RR se muestran unos 30 segundos después de los datos de frecuencia cardiaca y alcanzan la precisión nominal después de unos 40 a 50 segundos cuando las estadísticas de la señal están disponibles.

3. OB1203 Recorrido del algoritmo

El programa principal obtiene datos del OB1203 a 100Hz tanto para rojo como para IR en la función `get_sensor_data()`. Ordena los bytes de la FIFO y llama `add_sample()`. La función `add_sample()` mantiene un promedio de ocho muestras (12Hz) y carga los datos filtrados en un buffer circular. El búfer es lo suficientemente largo para dos latidos a la frecuencia cardiaca más lenta de ~40Hz, más una muestra extra para hacer

la suma de cuadrados fácil, más la longitud máxima del filtro de media móvil (muesca dicrótica). Estos valores se definen en `spo2.h`.

El algoritmo se ejecuta una vez cada intervalo de medición (por ejemplo, 1 segundo para FC/SpO2 o 0,4 segundos para RR).

El algoritmo de FC y SpO2 en `do_algorithm()` se divide en dos partes: `do_algorithm_part1()` y `do_algorithm_part2()` (cruce de umbral). Se leen nuevas muestras del FIFO entre la ejecución de las dos partes del algoritmo. `do_algorithm_part3()` y `do_algorithm_part4()` hacen el algoritmo de frecuencia respiratoria (RR) y se ejecutan inmediatamente después de la parte 2.

`do_algorithm_part1()` copia los datos usando `copy_data()` desde el buffer circular a los arrays lineales `AC1f`, obtiene el valor DC para los cálculos PI y R.

`do_algorithm_part2()` realiza el filtrado de paso alto de los datos (`high_pass_filter()`), detección de ventana o frecuencia cardiaca y estimación gruesa de la frecuencia cardiaca utilizando el de cruce de umbral. También calcula PI, y SpO2 usando ambos métodos, realiza el promedio biológico (8 segundos) vía filtros Kalman respectivos, y arbitra cual resultado del método usar para la visualización.

`calc_hr()`, `calc_R()`, `calc_R_p2p()` y `calc_spo2()` se explican por sí mismas. El valor R calculado en `calc_R()` se basa en la distancia media desde cero (cuasi-RMS). El valor R de `calc_R_p2p()` es la diferencia pico-valor.

El cálculo de `spo2` en `calc_spo2()` utiliza un polinomio en R para calcular el nivel de SpO2 a partir del valor de R más estable.

En `do_algorithm_part2()`, se realizan los siguientes pasos:

1. Ejecuta el método recursivo de ventana para la detección de latidos `hpf_window_peak_detect()` y el método opcional de detección de presión `get_pressure()`.
2. Filtro de paso alto de los datos utilizando `high_pass_filter()`. Una media centrada y móvil de 64 muestras (0,64 segundos) se sustrae de los datos de la memoria intermedia. Esto es necesario para aplanar las oscilaciones del movimiento y la respiración.
3. Calcule el umbral como 1/3 a 1/2 de la señal máxima en los datos filtrados y el umbral negativo como 1/3 a 1/2 de la señal mínima.
4. Cuente los cruces de umbral y conviértalos en un período de FC estimado utilizando $60 \text{ seg/min} * 100 \text{ muestras/seg} / (X/2)$, donde X es el número de cruces de umbral (es decir, dos por latido). Esto se mantiene en precisión de punto fijo.
5. Ejecuta un filtro Kalman para encontrar y eliminar valores atípicos en los cruces de umbral.
6. Negociar el mejor valor de HR en la función `calc_hr()`.
7. Calcular la relación de proporciones `calc_R()` con corrección de presión opcional.
8. Calcular `spo2` `calc_spo2()`.
9. Ejecuta un filtro Kalman para encontrar y eliminar valores atípicos tanto en SpO2 como en FC y realiza una media móvil de 8 segundos (clínica).

A continuación, el algoritmo realiza `do_algorithm_part3()`. Aquí es donde el algoritmo almacena la última amplitud del latido del corazón y calcula la tasa de respiración aproximada (RR). La RR se calcula construyendo una serie de medias móviles o líneas de base. El algoritmo calcula el número de cruces por cero relativos a cada línea de base sobre una longitud de muestras `MAX_BREATH_FILTER_SPAN` es igual a 1,6 veces la media móvil más larga. Esto está pensado para la precisión en respiraciones lentas. `MAX_BREATH_FILTER_SPAN` puede optimizarse para una mayor precisión o un inicio más rápido.

Caracterizamos las líneas de base contando el número de cambios de signo de la línea de base. Cuantos más cambios de signo, mejor es la línea de base. Se cree que cuanto más plana es una línea de base (es decir, abarca un número entero de respiraciones) más cambios de signo se producen, ya que pequeñas fluctuaciones pueden causar un cambio de signo en una señal plana. Este método no se describe en la literatura y puede ser exclusivo de nuestro algoritmo.

En `do_algorithm_part4()` intentamos elegir la mejor estimación de la frecuencia respiratoria. Caracterizamos los cálculos recientes de paso por cero para cada longitud de línea de base utilizando una aproximación basada en un filtro IIR de la raíz cuadrática media (RMS) que es eficiente para procesadores de 16 bits.

Calculamos una cifra de mérito (badness) basada en la relación entre los cruces por cero RMS y los cambios de signo de la línea de base. Encontramos el filtro con la mejor figura de mérito (menor maldad) e incrementamos el rastreador de filtro hacia ese filtro. Se utiliza un factor de amortiguación de filtro en el rastreador de filtro para que el cambio de filtro se produzca lentamente. Esto mejora la estabilidad. Aumentar la amortiguación del filtro significa que puede llevar más tiempo alcanzar el valor correcto o adaptarse a un cambio, pero hay menos desviación de un valor correcto una vez que lo tenemos.

El valor RR se calcula a partir de los cruces por cero como respiraciones por , luego el valor se filtra por Kalman y se muestra. Una media móvil de duración configurable adicional que puede aumentarse para la estabilidad de la señal o disminuirse para mejorar el tiempo de respuesta.

3.1 Filtros Kalman

Se utilizan filtros de Kalman de orden cero para el resultado del filtro de estimación gruesa, la media de 8 segundos de la frecuencia cardiaca, la media de 8 segundos de SpO2, la frecuencia respiratoria, el periodo latido a latido y la amplitud latido a latido.

El código del filtro Kalman está en `Kalman.c`.

El filtro Kalman de orden cero comprueba la variación RMS en el búfer de datos recientes y elimina los valores atípicos antes de añadir muestras válidas al búfer Kalman. El búfer Kalman se utiliza para calcular la media.

Los parámetros del filtro Kalman se definen en `Kalman.h`. Los filtros Kalman se reinician después de un número definido de entradas nulas (el algoritmo falla) o demasiados valores atípicos seguidos para permitir que el filtro recupere más rápidamente el valor correcto.

3.2 Comprobación de la calidad de los datos

Para los dispositivos relacionados con la salud, a menudo es preferible no mostrar ningún valor en lugar de mostrar un valor potencialmente incorrecto. Las opciones de configuración en `oximstruct.h` permiten comprobaciones adicionales de la calidad de los datos habilitadas por los ajustes en el paso 6 definiendo `CHECK_PI`. Las siguientes condiciones resultan en un "congelamiento" de los datos y un eventual reinicio del algoritmo:

- PI por debajo del umbral mínimo
- PI por debajo del umbral bajo y variación RMS en PI mayor que el umbral bajo PI RMS
- PI por encima del umbral bajo y variación RMS en PI superior al umbral normal PI RMS

4. Código de ejemplo

Descargue el código fuente de demostración de la página del producto [OB1203SD-RL-EVK](#). Póngase en contacto con su representante de ventas local para obtener ayuda.

5. Historial de revisiones

Revisión	Fecha	Descripción
3.01	25 de abril de 2022	Actualizada la primera frase en "Configuración de los ajustes del programa principal".
3.00	14 de enero de 2022	Importante actualización del algoritmo que incluye el método de la ventana y la corrección de la presión. Eliminado el método de filtro FIR triangular.
2.00	21 de octubre de 2021	Importante actualización del algoritmo que incluye el filtro Savitsky-Golay y los métodos de filtro de paso alto. Eliminado el método de correlación.
1.00	23 de julio de 2021	Lanzamiento inicial.

AVISO IMPORTANTE Y DESCARGO DE RESPONSABILIDAD

RENESAS ELECTRONICS CORPORATION Y SUS FILIALES ("RENESAS") PROPORCIONAN ESPECIFICACIONES TÉCNICAS Y DATOS DE FIABILIDAD (INCLUIDAS LAS HOJAS DE DATOS), RECURSOS DE DISEÑO (INCLUIDOS LOS DISEÑOS DE REFERENCIA), ASESORAMIENTO SOBRE APLICACIONES U OTROS DISEÑOS, HERRAMIENTAS WEB, INFORMACIÓN DE SEGURIDAD Y OTROS RECURSOS "TAL CUAL" Y CON TODOS LOS FALLOS, Y RECHAZAN TODAS LAS GARANTÍAS, EXPRESAS O IMPLÍCITAS, INCLUIDAS, ENTRE OTRAS, LAS GARANTÍAS IMPLÍCITAS DE COMERCIABILIDAD, IDONEIDAD PARA UN FIN DETERMINADO O NO INFRACCIÓN DE LOS DERECHOS DE PROPIEDAD INTELECTUAL DE TERCEROS.

Estos recursos están destinados a desarrolladores que diseñan con productos Renesas. Usted es el único responsable de (1) seleccionar los productos adecuados para su aplicación, (2) diseñar, validar y probar su aplicación, y (3) garantizar que su aplicación cumple las normas aplicables y cualquier otro requisito de seguridad o de otro tipo. Estos recursos están sujetos a cambios sin previo aviso. Renesas le autoriza a utilizar estos recursos únicamente para desarrollar una aplicación que utilice productos Renesas. Queda terminantemente prohibida cualquier otra reproducción o uso de estos recursos. No se concede licencia alguna sobre ninguna otra propiedad intelectual de Renesas ni terceros. Renesas declina toda responsabilidad, y usted indemnizará plenamente a Renesas y a sus representantes por cualquier reclamación, daño, coste, pérdida o responsabilidad derivada del uso que usted haga de estos recursos. Los productos de Renesas se suministran únicamente con sujeción a los Términos y Condiciones de Venta de Renesas u otros términos aplicables acordados por escrito. El uso de cualquiera de los recursos de Renesas no amplía ni altera de ningún modo las garantías aplicables ni las exenciones de garantía de estos productos.

(Cláusula de exención de responsabilidad Rev.1.01)

Sede social

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokio 135-0061, Japón www.renesas.com

Marcas

Renesas y el logotipo de Renesas son marcas comerciales de Renesas Electronics Corporation. Todas las marcas comerciales y marcas registradas pertenecen a sus respectivos propietarios.

Información de contacto

Para obtener más información sobre un producto, una tecnología, la versión más actualizada de un documento o la oficina de ventas más cercana, visite www.renesas.com/contact-us/.