



Universidad de Oviedo



Título del Proyecto:

**DIAGNÓSTICO EN TIEMPO REAL DE FALLOS
ASIMÉTRICOS EN LOS BOBINADOS ROTÓRICOS Y
ESTATÓRICOS EN GENERADORES DE INDUCCIÓN
(VERSIÓN TRADUCIDA)**

Tutor Pedagógico:

Shahin Hedayati Kia

Presentado por:

Fernández Fernández, Javier

AMIENS – Francia

JUNIO 2016



Agradecimientos

Antes de comenzar, me gustaría agradecer a la Universidad de Picardie Jules Verne y al departamento de Ingeniería Eléctrica por acogerme durante este año de estancia en Francia para el fin de mis estudios. Y especialmente al coordinador de mi intercambio internacional, Humberto Henao, por el apoyo ofrecido durante mi estancia en esta universidad, y a mi tutor y responsable durante la realización de este trabajo, Shahin Hedayati Kia, por la oportunidad de trabajar juntos y su colaboración a mi desarrollo personal como futuro ingeniero.



ÍNDICE

0. Lista de figuras	p. 3
1. Introducción.....	p. 4
2. Método utilizado	p. 6
3. Desarrollo del programa	p. 7
4. Programa definitivo para la detección de fallos rotóricos y estáticos	p. 26
5. Test y resultados observados.....	p. 28
6. Conclusión.....	p. 32



LISTA DE FIGURAS

Figura 1: Interfaz FPGA 1.....	p. 7
Figura 2: Interfaz FPGA 2.....	p. 8
Figura 3: Recepción de datos.....	p. 9
Figura 4: Lectura de datos.....	p. 10
Figura 5 : Calculo de la amplitud instantánea	p. 11
Figura 6: Obtención de envolventes 1.....	p. 13
Figura 7: Obtención de envolventes 2.....	p. 14
Figura 8: Análisis DWT.....	p. 16
Figura 9: Nivel de descomposición.....	p. 17
Figura 10: Evaluación de la energía.....	p. 17
Figura 11: Indicador de fallo	p. 18
Figura 12: Cálculo de defectos estatóricos 1.....	p. 18
Figura 13: Cálculo de defectos estatóricos 2.....	p. 19
Figura 14: Obtención del vector de corriente.....	p. 20
Figura 15: Corrientes de fase y vector de entrada	p. 21
Figura 16: Cálculo del tiempo de ejecución.....	p. 22
Figura 17: Registro de resultados.....	p. 23
Figura 18: Interfaz Gráfica 1.....	p. 24
Figura 19: Interfaz Gráfica 2.....	p. 24
Figura 20: Interfaz Gráfica 3.....	p. 25
Figura 21: Programa final 1.....	p. 26
Figura 22: Programa final 2	p. 27
Figura 23: VI de lectura de resultados	p. 28
Figura 24: Test de fallo rotórico	p. 29
Figura 25: Test de fallo estatórico	p. 30



1. INTRODUCCIÓN

Las máquinas de inducción de rotor bobinado son ampliamente utilizadas en numerosas aplicaciones industriales hoy en día. Como todo equipo o mecanismo electromecánico, su utilización continuada así como el paso del tiempo causan variaciones en su estructura interna a causa del desgaste y esto puede variar su funcionamiento. Esto acabará causando un fallo en la máquina antes o después.

La aparición de un defecto o fallo de cierta importancia tiene como consecuencia directa la parada forzada de la máquina y por lo tanto, del proceso ligado y dependiente de esta, y corre el riesgo de dañar a otras instalaciones afectadas por la parada y a los utilizadores de las mismas que pueden encontrarse trabajando en ellas y no esperar la parada. Todo ello supone normalmente un coste económico importante asociado y un interés general de evitar este tipo de situaciones para asegurar la seguridad de los trabajadores y la estabilidad y el buen funcionamiento de los procesos industriales más críticos que pueden depender de estas máquinas.

Poder prever los defectos, y reaccionar antes de que un fallo tenga lugar, es especialmente importante en aplicaciones como la producción de energía eólica, donde este tipo de máquina está muy presente en su uso como rotor eólico gracias a sus posibilidades de regulación y se encuentran normalmente en lugares de difícil acceso. Este sistema nos permite programar los mantenimientos con mayor eficiencia para evitar la necesidad de reemplazo de maquinaria costoso y al mismo tiempo largas paradas no previstas, evitando así la pérdida económica asociada a la falta de producción así como otros efectos nocivos que la variación de la producción de energía eléctrica estimada pudiera causar en la red eléctrica.



Por tanto, en este proyecto tendremos como objetivo el desarrollo de una aplicación en de diagnóstico en tiempo real, utilizando el programa LabView y el sistema embarcado CompartRIO (cRIO 9068) de “National Instruments” capaz de detetar los fallos y desequilibrios presentes en una máquina de inducción de rotor bobinado utilizando el método desarrollado en el artículo “Windings Monitoring of Wound Rotor Induction Machines” realizado por el departamento de Ingeniería Eléctrica de la “Université de Picardie Jules Verne”, el cual nos servirá de fundamento teórico para nuestro trabajo.

Esta aplicación debe permitirnos detectar en tiempo real la presencia de fallos rotóricos o estáticos utilizando el tratamiento de las señales de corriente estáticas y rotóricas respectivamente, representarlos de forma fácilmente reconocible por un operador et también registrar el histórico de funcionamiento para su análisis posterior.

La aplicación deberá ser validada con una máquina de inducción de rotor bobinado de 90W y 1430 rpm sobre la cual simularemos fallos causando desequilibrios a través d la manipulación de las fases de la máquina y la introducción de resistencias en serie con sus bobinados. Sin embargo, la máquina utilizada reveló durante los ensayos la presencia de desequilibrios inherentes por lo que el programa no ha podido ser validado completamente para un estado de la máquina saludable (aunque hay datos obtenidos durante las etapas de desarrollo) y la detección de fallos estáticos ha debido de ser validad utilizando un generador de señal para simular el fallo.



2. MÉTODO UTILIZADO

Una vez la problemática está presentada, hace falta ver que es lo que debemos analizar y utilizar como indicado de la presencia de un fallo y poder realizar así una aplicación que trate los datos para poder al final obtener y representar claramente este indicador y mostrar la presencia de un defecto en la máquina.

Los estudios que han tenido lugar en esta materia, y especialmente el artículo que utilizamos como referencia para el estudio de los fallos y el desarrollo de esta aplicación, establecen que la presencia de un desequilibrio se muestra por el efecto que tiene en una determinada banda de frecuencia de las corrientes estáticas y rotóricas de la máquina a analizar.

Para el fallo rotórico, debemos observar la corriente estática en la banda:

$$[0 \text{ Hz}, 2k_r s_r f_s \text{ Hz}]$$

Y para el fallo estático observaremos sobre la corriente rotórica:

$$[(2 - s_r) f_s \text{ Hz}, 2f_s \text{ Hz}]$$

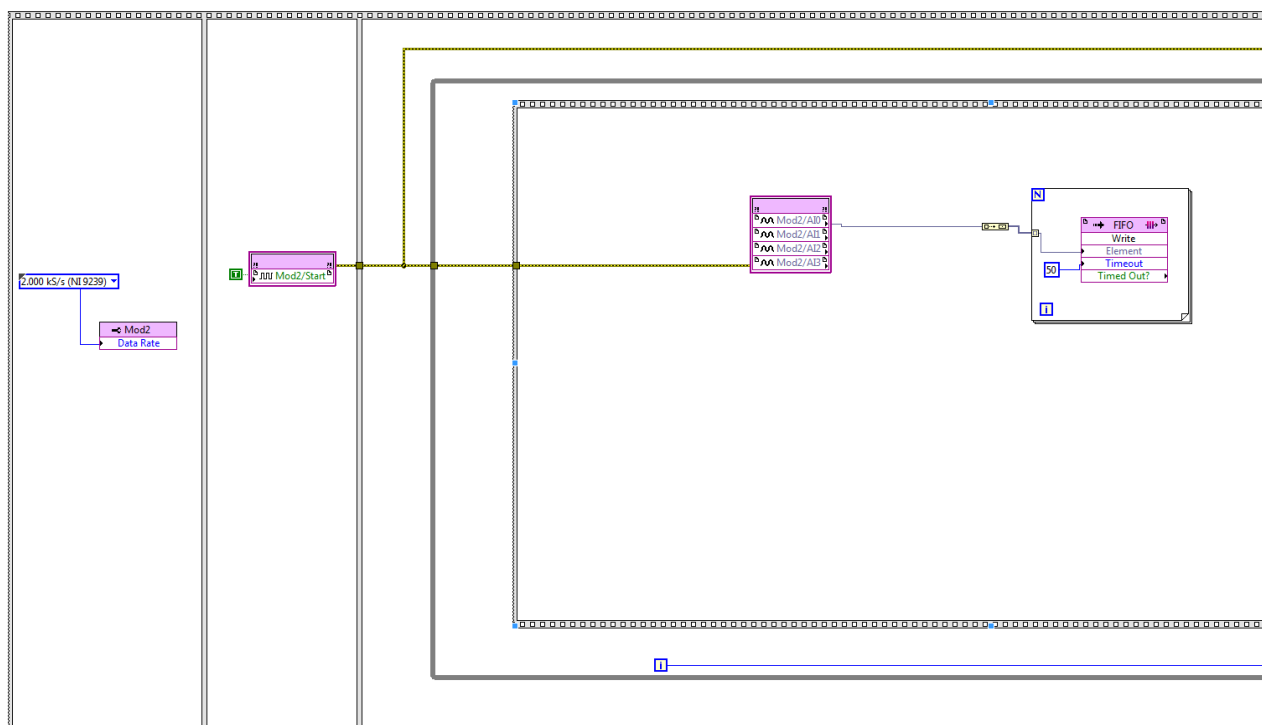
Siendo $k_r = \text{Integer} \left[\frac{1-s_r}{2p s_r} \right]$, « sr » el deslizamiento nominal, « fs » la frecuencia fundamental y « p » el número de pares de polos.

Por tanto en nuestro programa vamos a realizar la adquisición de las corrientes del motor para que seguidamente sean tratadas, normalizadas y efectuar un análisis DWT para obtener las bandas de frecuencia asociadas a los fallos y evaluar el nivel de energía continuamente en ellas para detectar la presencia de un fallo o desequilibrio.

3. DESARROLLO DEL PROGRAMA

Antes de comenzar con el programa de análisis propiamente dicho, ha sido necesario realizar una parte encargada de la adquisición de datos a la entrada de este. Hemos utilizado una interfaz FPGA para enviar las corrientes recibidas en el sistema embarcado (cRIO 9068) a nuestro programa para poder utilizarlas como datos de entrada, usando una frecuencia de muestreo de 2000 Hz.

Figura 1. Interfaz FPGA 1



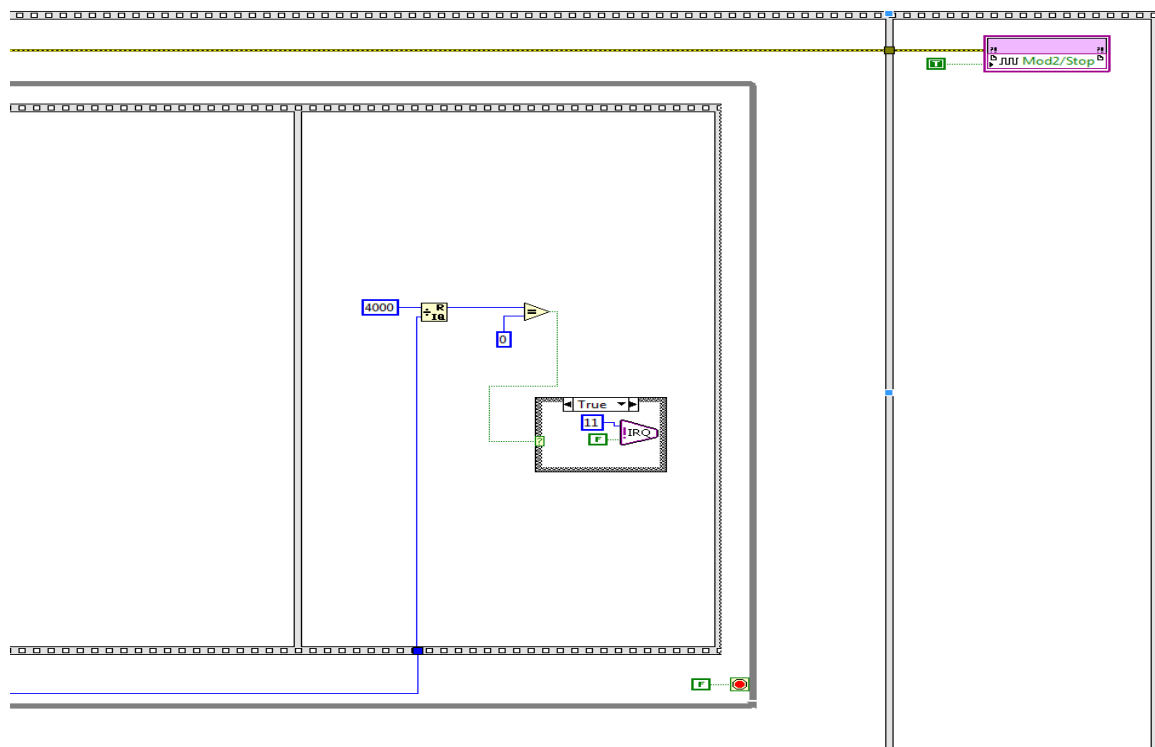
Almacenamos la información en un “FIFO” que después podemos utilizar para recuperar los datos enviados por nuestra interfaz FPGA en diferentes instancias de nuestro proyecto en LabView.



Hemos decidido que cada ciclo de funcionamiento esté comprendido de 4000 puntos de datos, lo cual hace que con nuestra frecuencia de muestreo obtengamos 2 segundos de datos analizados y representados a la salida por cada ciclo de funcionamiento del bucle de análisis.

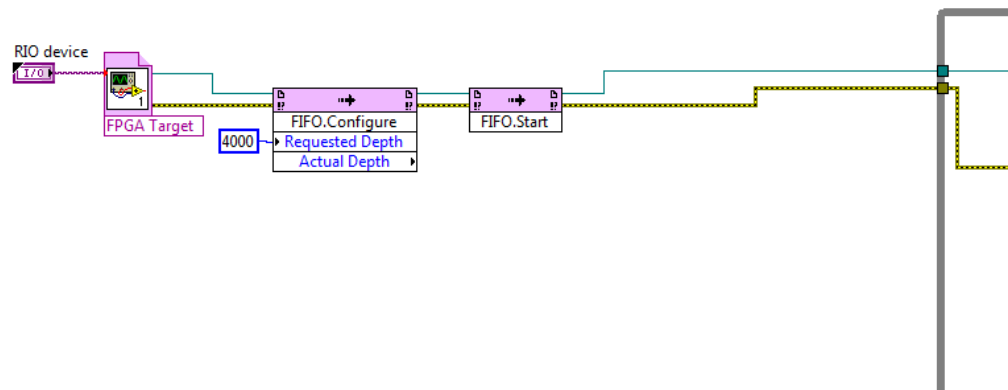
Un periodo de tiempo suficiente para tener una cantidad de datos lo suficientemente importante para poder realizar los análisis y cálculos correctamente y que se actualizan a un ritmo lo suficientemente lento para que un seguimiento en tiempo real por un operario sea plausible y sencillo.

Figure 2. **Interfaz FPGA 2**



Una vez en la instancia donde nuestro programa de análisis va encontrarse, es necesario hacer una llamada al dispositivo que utilizamos para la recepción de datos y recuperar los datos almacenados en el FIFO por nuestra interfaz FPGA. Por lo que la realizamos e inicializamos el FIFO para su utilización.

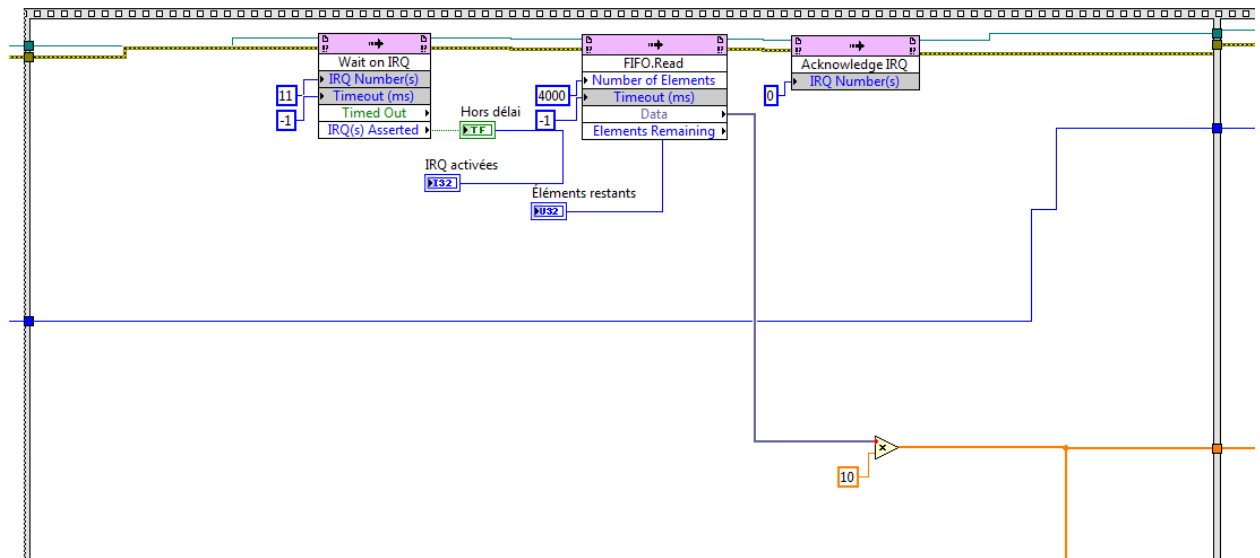
Figura 3. **Recepción de datos**



Solicitamos los 4000 puntos de datos para representar todos los datos recibidos por nuestro dispositivo y poder contar con una adquisición continua e ininterrumpida.

Después, entramos ya en la primera etapa del bucle de programación, donde leemos el FIFO ya inicializado y finalizamos la configuración de los parámetros de adquisición de datos.

Figura 4. Lectura de datos



Adjuntamos una amplificación antes de pasar a la segunda etapa del bucle, donde se encuentra el programa de tratamiento de la señal, para escalar los datos y trabajar con los valores reales, compensando la atenuación introducida por las sondas de corriente.

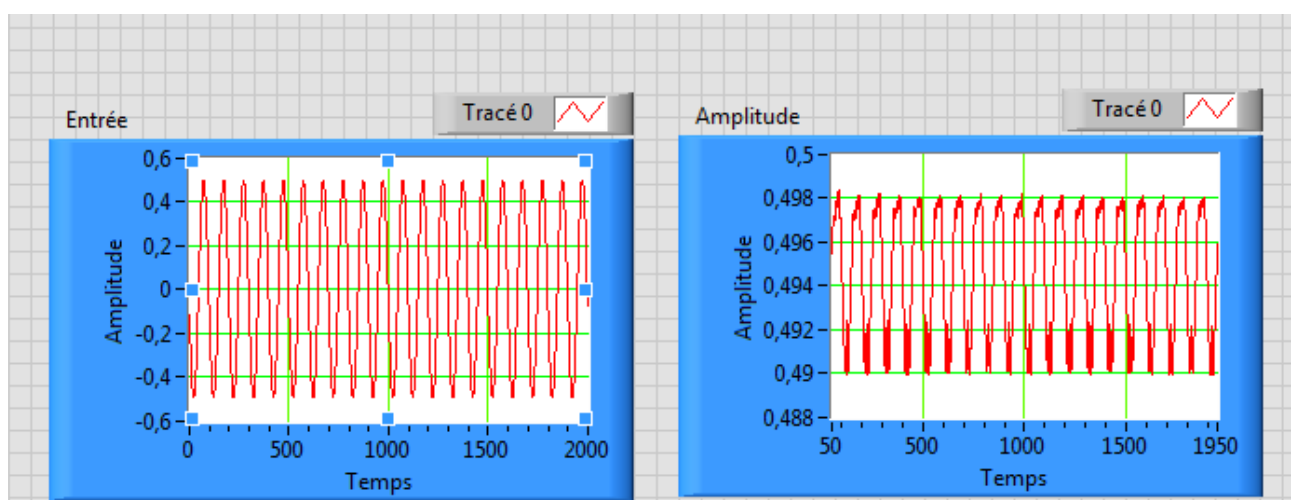
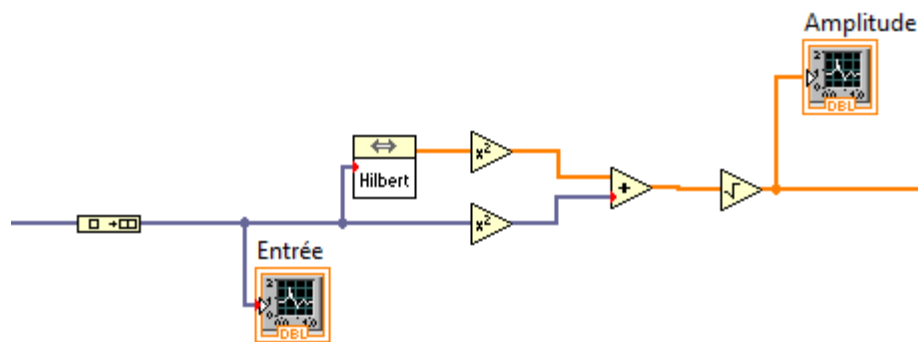
Hemos escogido este tipo de bucle en dos etapas para asegurar que la adquisición y la lectura de los datos de entrada tienen lugar correctamente antes de la ejecución del programa de análisis.

Ahora, vamos a comenzar a revisar las diferentes etapas de desarrollo del programa de detección de fallos propiamente dicho. Hacemos notar que al principio se utilizó un generador de señal para validar las diferentes etapas durante su programación antes del paso al trabajo con la máquina de test.

Comenzamos con la parte del código orientada al análisis de la corriente estatórica para la detección de fallos rotóricos.

El primer objetivo era obtener la amplitud instantánea de la corriente de entrada. Para ello utilizamos la transformada de Hilbert así como otros bloques con cálculos aritméticos simples para implementar el cálculo completo.

Figura 5. Cálculo de la amplitud instantánea



Una vez tenemos la amplitud instantánea de la señal de entrada, es necesario normalizarla para la evaluación de la energía final. Sin embargo, en el caso de la



corriente estatística encontramos un problema. La presencia de una componente de continua que va variando en el tiempo cuando trabajamos en condiciones de carga variable. Por ello, para poder tener un buen análisis en estas condiciones, necesitamos poder determinar esta componente de continua dinámicamente en lugar de utilizar un método de media estática.

Para ello, y como aconseja el artículo en el que nos basamos, hemos decidido encontrar las envolventes superior e inferior de esta señal para eliminar esta componente de continua. Para construir estas envolventes tendremos que utilizar diversas funciones presentes en el programa.

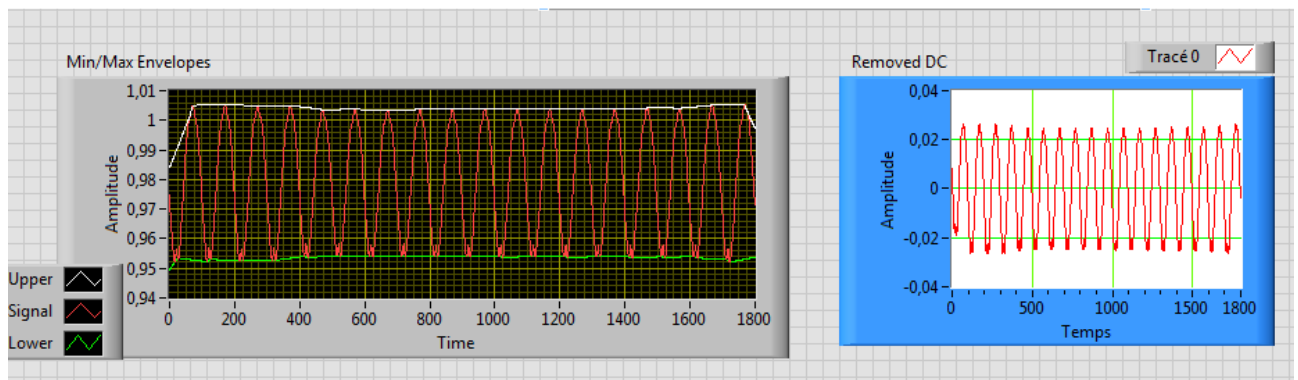
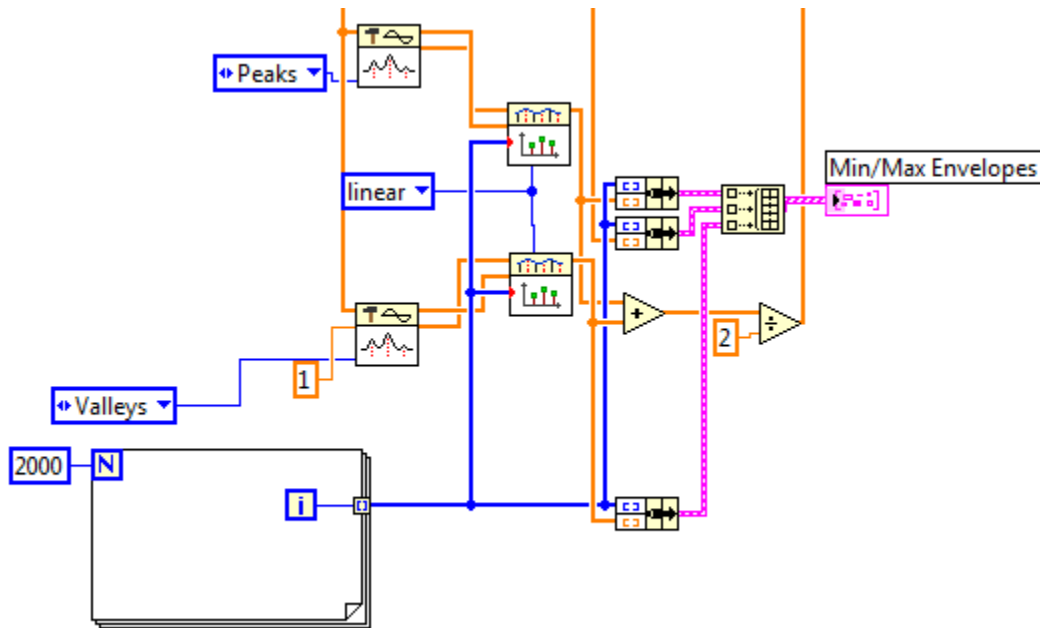
Primero, tenemos un bloque que podemos configurar para detectar las crestas y valles. Utiliza un criterio de detección de mínimos y máximos que se puede ajustar para detectar las crestas y valles a partir de un cierto límite que establecemos, para así sólo detectar crestas por encima de él o valles por debajo.

Podemos configurar también un mínimo de longitud de datos para que una cresta o valle sea detectado y evitar las detecciones erróneas de pequeñas crestas o valles causadas por el ruido o la presencia de componentes de diferente frecuencia.

Este bloque nos proporciona a la salida dos arrays con los valores de las crestas y valles y sus posiciones respectivamente. Para después construir la envolvente es necesario adjuntar un bucle que nos permita crear un array de la longitud de datos que pensamos utilizar y utilizar otro bloque para interpolar los valores obtenidos y asociar los valores de las crestas y valles con sus posiciones y unir los diferentes puntos para formar la envolvente.

Finalmente, juntamos las tres señales, La analizada y las dos envolventes construidas, para representarlas en un mismo diagrama y verificar el funcionamiento.

Figura 6. Obtención envolventes 1



Sin embargo, esta etapa ha demostrado ser la más compleja y problemática y ha dado multitud de problemas una vez hemos comenzado a trabajar con la máquina real.

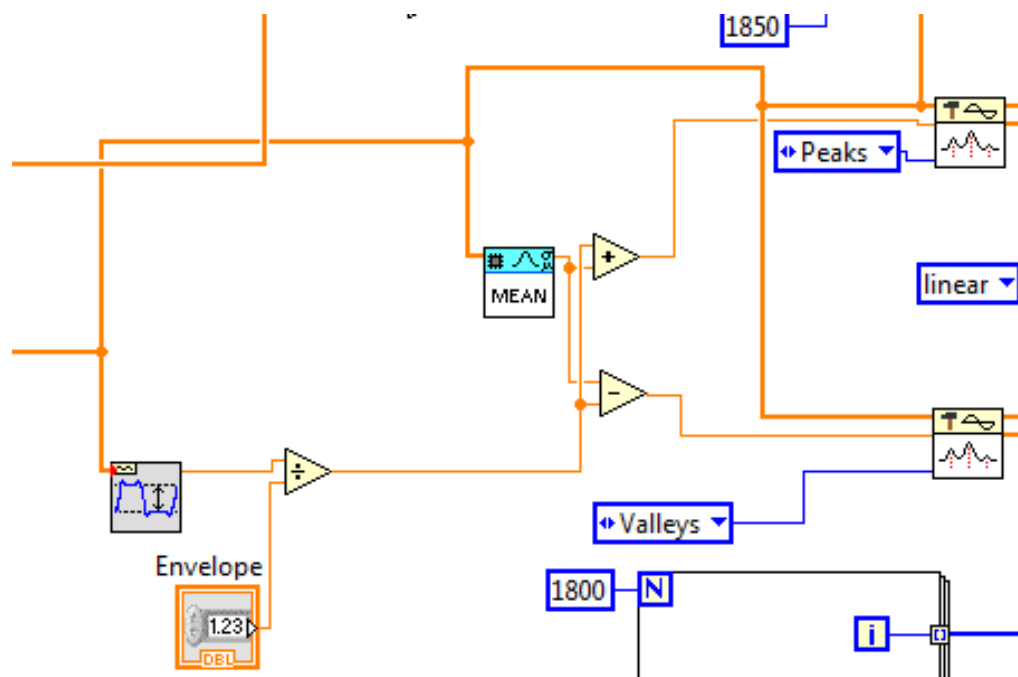
La detección de crestas y valles era errónea y múltiples modificaciones y pruebas tuvieron lugar hasta alcanzar la configuración final.

Ha sido necesario determinar el límite de detección de forma dinámica, utilizando la media aproximada de la señal analizada y una calibración aritmética que ha debido ser probado y modificada en muchas ocasiones para poder adaptarse a las diferentes señales y variaciones.

Hemos intentado también limitar la longitud de los datos analizados y la envolvente construida al percatarnos que los errores de detección se producían sobre todo en los extremos de la envolvente.

Pero esto creaba tantos problemas como solucionaba, pues observamos que utilizar este método introducía desfases entre la envolvente y la señal a analizar por lo que no fue utilizado en la versión final del programa.

Figura 7. Obtención envolventes 2



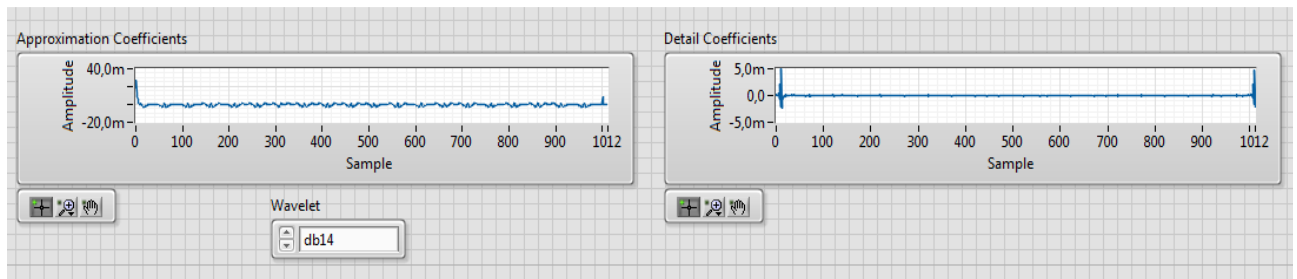
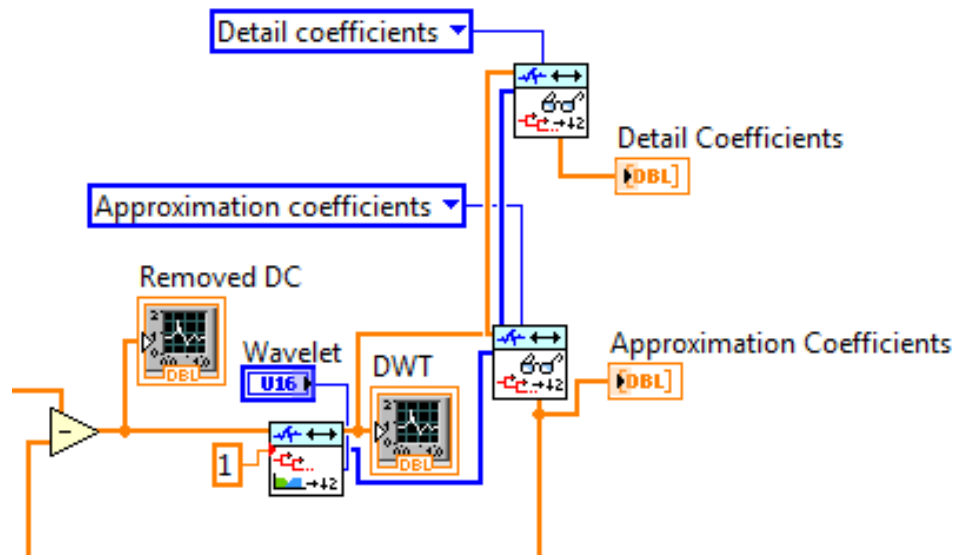


Otra cosa que intentamos fue tener una señal más limpia para que la detección fuera más sencilla y hallar así el verdadero valor de continua. Consideramos añadir una segunda etapa calculando las envolventes del resultado de la primera media hallada pero al final lo que se mostró más efectivo fue cambiar el orden original de las etapas que teníamos en el programa, realizando el análisis DWT antes de esta etapa para construir las envolventes sobre una señal ya filtrada, más limpia y sin las altas frecuencias que causaban los fallos en la detección.

La siguiente etapa realizada fue la implementación de dicho análisis DWT. En el programa que utilizamos encontramos esta función incorporada mediante la utilización de dos bloques. Uno en el cual indicamos el nivel de descomposición requerido y el método utilizado y es el que realiza la DWT propiamente, y luego un segundo bloque que permite leer y representar los resultados, escogiendo bien los coeficientes de detalle o aproximación.

Hemos utilizado el método Daubechies 14 porque es un método que ha dado buenos resultados en este tipo de aplicaciones, de acuerdo a las referencias y artículos consultados, y el orden 14 es el más preciso que el programa LabView nos permite utilizar.

Figura 8. Análisis DWT

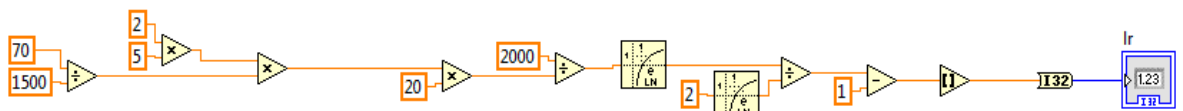


En este caso, para detectar los fallos rotóricos nos interesan los coeficientes de aproximación para encontrar la banda de frecuencia donde el desequilibrio se hará visible, y es la que posteriormente utilizaremos para la evaluación de la energía.

El nivel de descomposición deseado va a depender de las características de la máquina que será diagnosticada, y de la frecuencia fundamental y de muestreo utilizadas. Por tanto, para encontrarlo de manera más sencilla, una sección de código ha

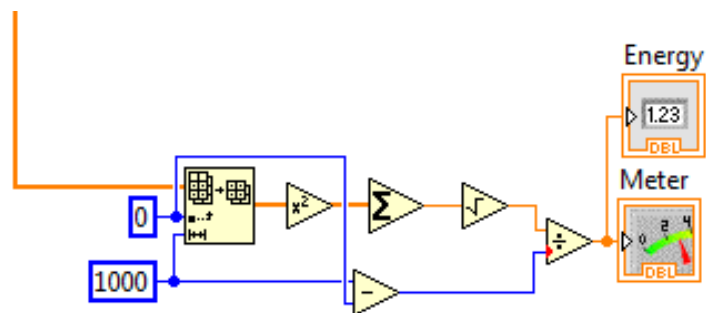
sido añadida para calcularlo automáticamente a partir de los valores de partido mencionados y que son conocidos.

Figura 9. Nivel de descomposición



Para finalizar esta parte del programa, adjuntamos el cálculo de evaluación de la energía de la señal obtenida tras el tratamiento.

Figura 10. Evaluación de la energía



Adjuntamos también un indicador visual para tener una referencia más directa y visible del nivel de energía y poder identificar de un vistazo la presencia de un fallo o desequilibrio.

Figura 11. Indicador de defecto



Una vez esta parte ha sido finalizada, seguimos un proceso similar para implementar el cálculo de defectos estáticos a partir de la corriente rotórica. En este caso el proceso era bastante más simple y rápido dado la falta de necesidad de retirar la componente de continua del análisis dado que el propio filtro del análisis lo hace.

La principal diferencia es la utilización de los coeficientes de detalle en lugar de los de aproximación en el análisis DWT. Es por ello que no adjuntamos una explicación más detallada.

Figura 12. Cálculo defectos estáticos 1

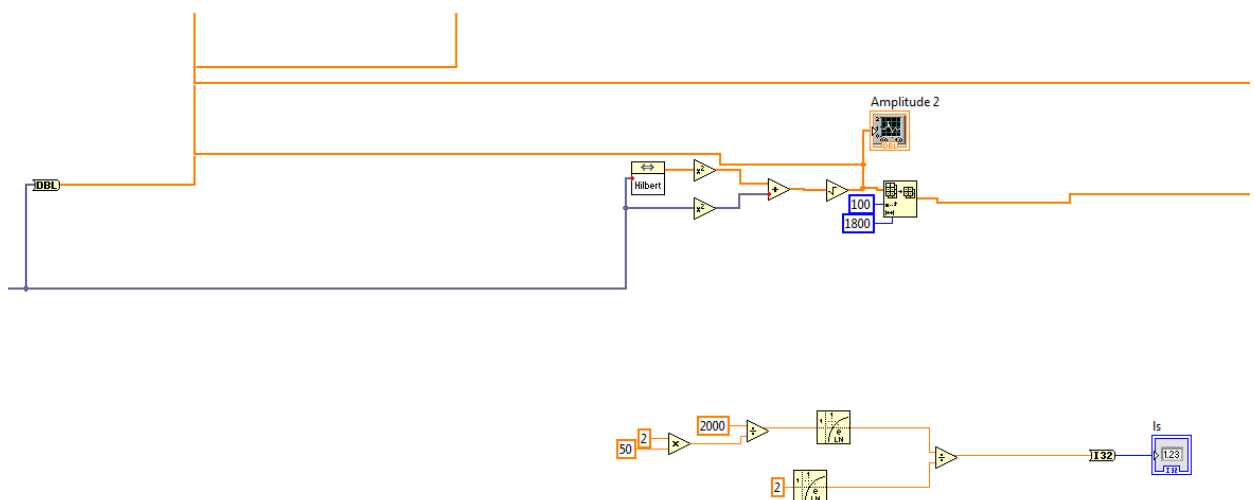
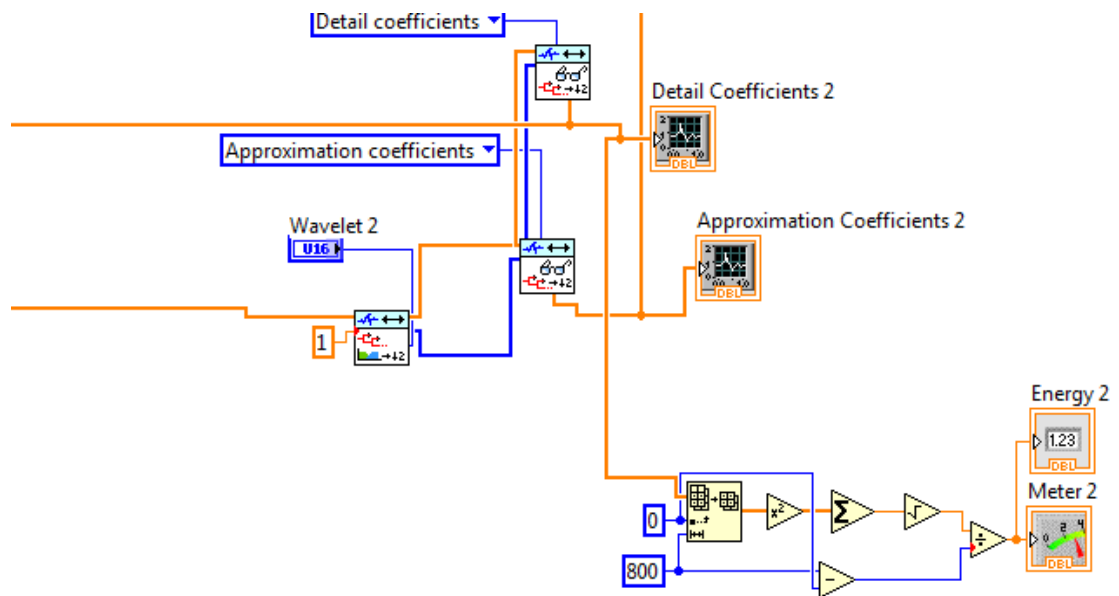
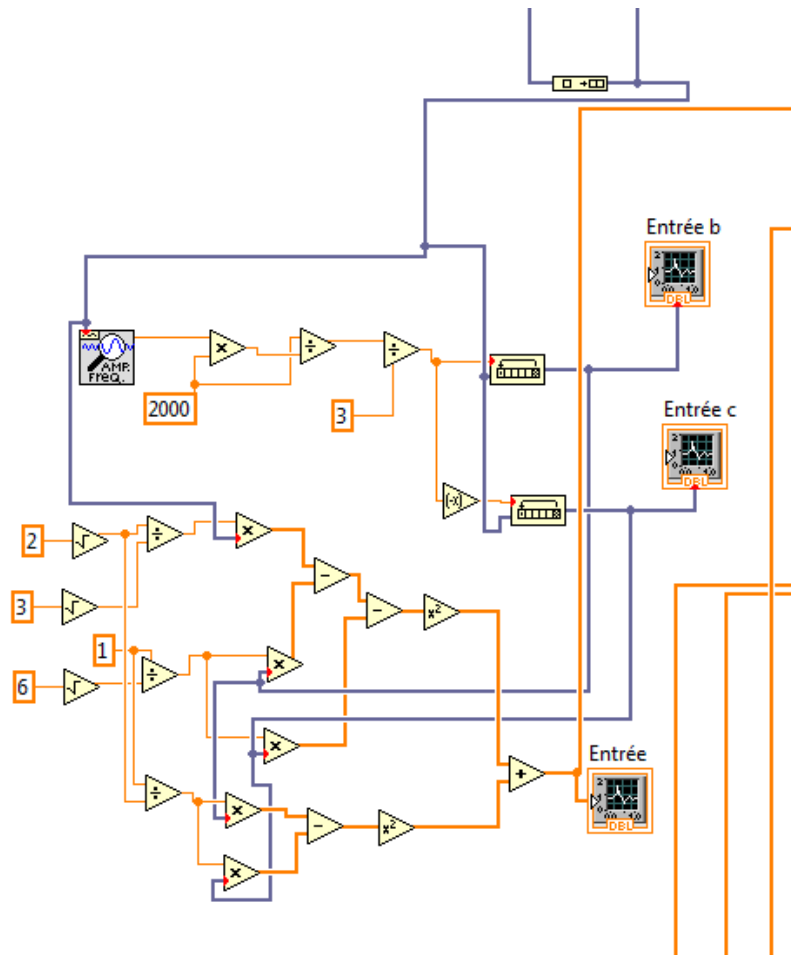


Figura 13. Cálculo defectos estáticos 2



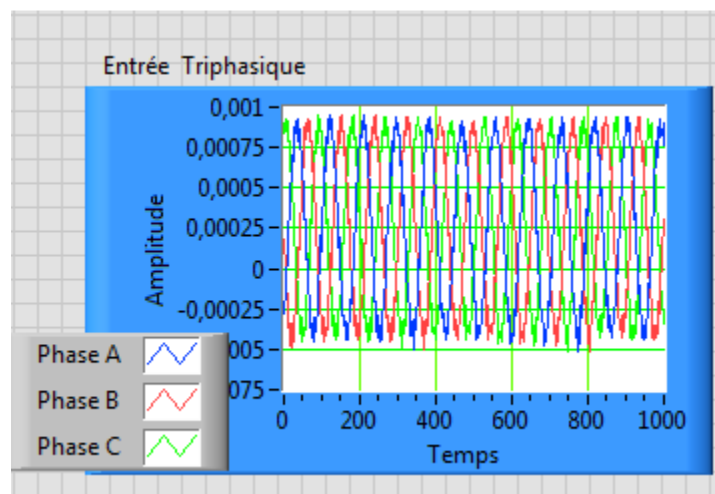
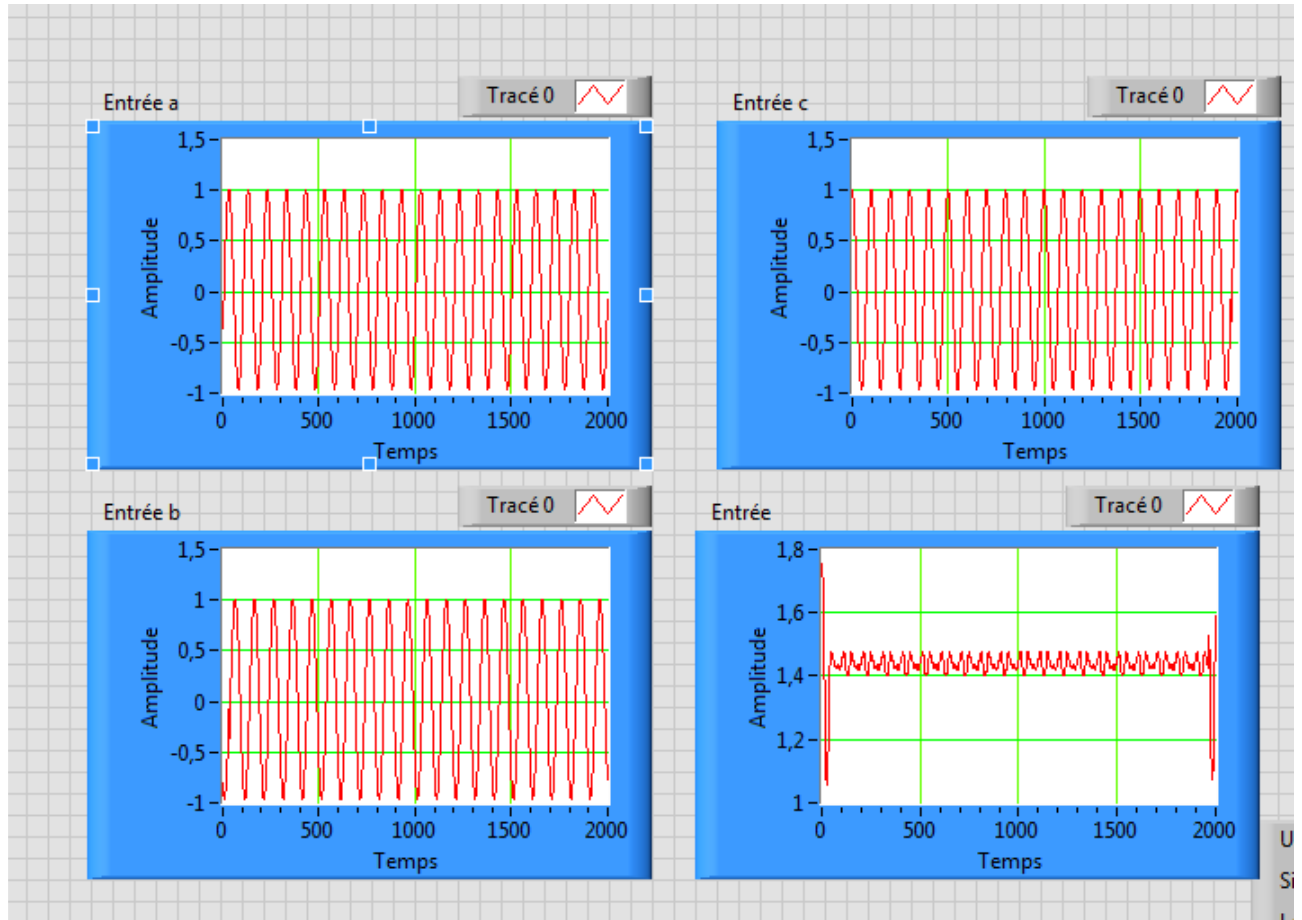
Una vez el programa estuvo realizado, trabajamos sobre la entrada al bucle de análisis para prepararla para una entrada trifásica. Adjuntamos código para poder simular esta entrada trifásica con el generador de seña y para calcular el vector de corriente de entrada a partir de la corriente trifásica en vez de usar la corriente de fase directamente, cómo estábamos haciendo hasta entonces.

Figura 14. Obtención vector de corriente



Utilizamos principalmente bloques aritméticos para el cálculo de dicho vector de entrada junto con bloques para detectar el valor instantáneo de la señal y desplazarlo 120 grados para simular las otras fases.

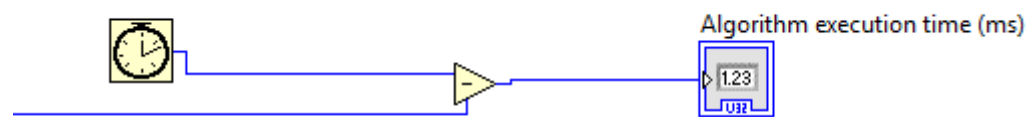
Figura 15. Corrientes de fase y vector de entrada



Dos funciones complementarias fueron introducidas en el programa. La primera con el objetivo de poder calcular el tiempo de ejecución del mismo y detectar si hay un mal funcionamiento debido a que la ejecución sea demasiado lenta para poder mantener el ritmo de análisis en tiempo real.

Para ello utilizamos el denominado bloque « Tick count » que nos permite ver el tiempo transcurrido desde el inicio de una instancia del programa hasta el fin del procesado del bucle :

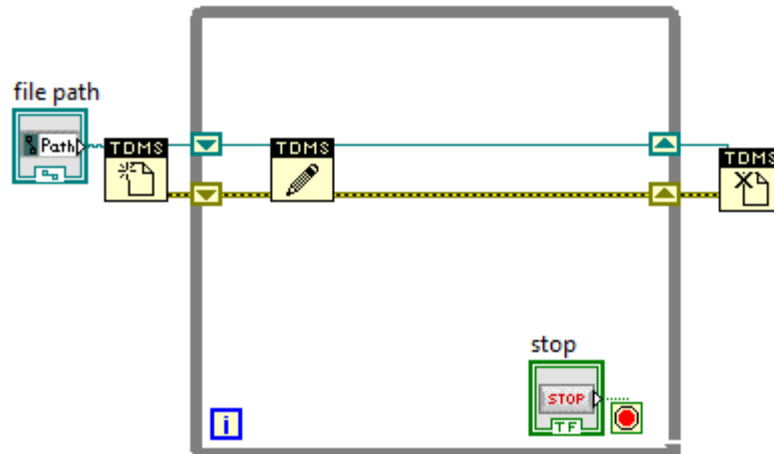
Figura 16. **Cálculo tiempo de ejecución**



La otra función tiene como objetivo adjuntar un sistema de registro de los datos interesantes obtenidos por el programa de análisis para poder representarlos y consultarlos más tarde aparte del funcionamiento y visualización en tiempo real.

Para ello utilizamos los bloques TDMS que permiten crear un archivo en este formato (o abrir uno ya existente) y escribir en él los diferentes datos obtenidos durante el procesamiento mientras el bucle “while” de ejecución del programa está activo, y cerrar el archivo una vez el bucle se detiene.

Figura 17. Registro de resultados



Para finalizar, hemos trabajado sobre la interfaz visual para que las informaciones relevantes sean limpia, ordenada y correctamente representadas para poder discernir fácilmente que es cada cosa y diferenciar las partes referentes al control del roto y estator.

Aquí encontramos también un campo para indicar el directorio donde se encontrará el archivo donde se registran los datos así como un menú deslizante donde indicamos el dispositivo que vamos a utilizar para la adquisición de datos en ese momento. También podemos modificar el método utilizados para el análisis DWT si encontramos que otro se ajusta mejor a nuestras necesidades o condiciones de trabajo. Finalmente encontramos el botón de parada, que detiene tanto el bucle de análisis como la adquisición de datos.



Figura 18. Interfaz gráfica 1

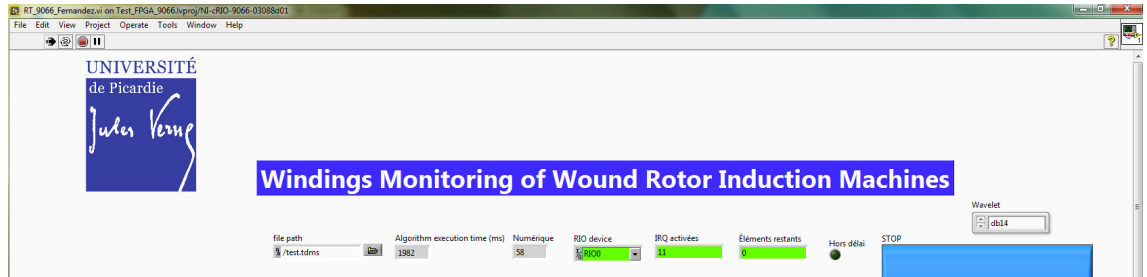


Figura 19. Interfaz gráfica 2

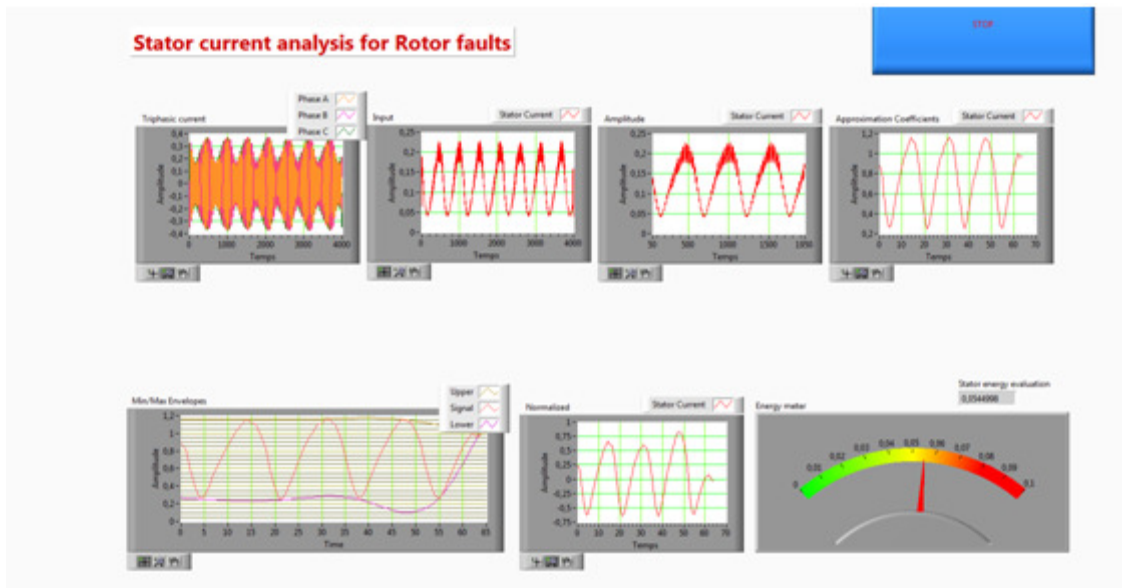
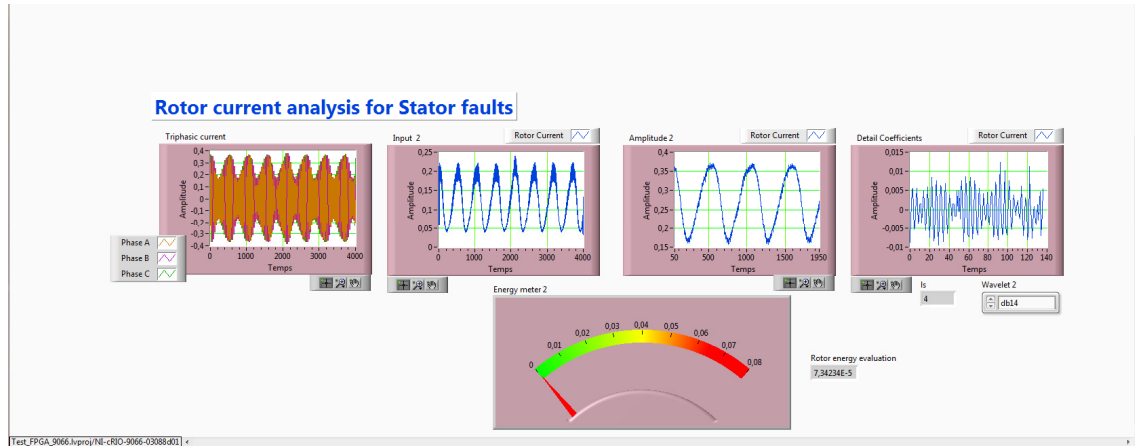




Figura 20. Interfaz gráfica 3



4. PROGRAMA DEFINITIVO PARA LA DETECCIÓN DE FALLOS ROTÓRICOS Y ESTATÓRICOS

Durante los test con la máquina de inducción, varias modificaciones tuvieron lugar. Especialmente para la configuración del cálculo y construcción de las envolventes, como ya comentamos anteriormente, pero otras pequeñas modificaciones también se sucedieron, como la utilización de un bloque que permite obtener a su salida las dimensiones de un array de entrada. Gracias a esto si modificamos la longitud del array de datos utilizado para centrarnos en una parte concreta más interesante del total de los datos recibidos a la entrada del programa, en caso de ser necesario eliminar las partes deformadas de estos para el tratamiento, esta modificación será transmitida a lo largo de todo el programa manteniendo la concordancia en la longitud de la series de datos en lugar de tener que modificarlo manualmente en varias instancias y correr riesgo de inconsistencias.

Figura 21. Programa final 1

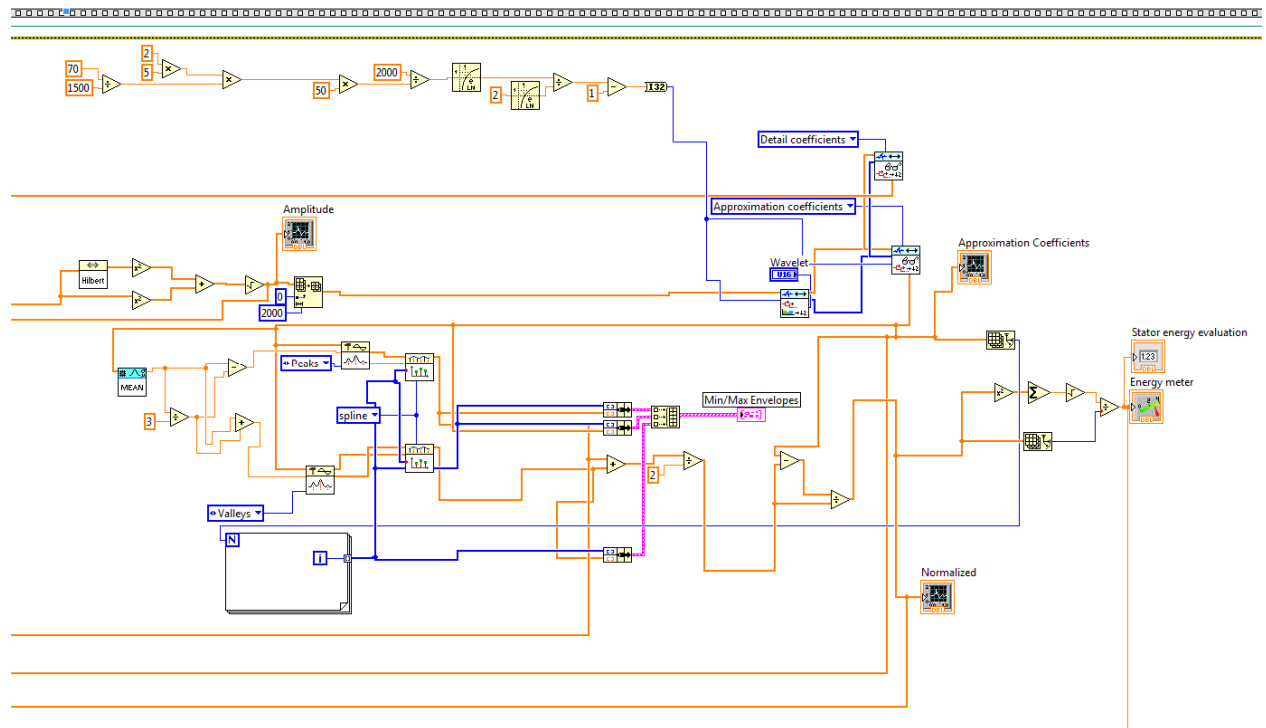
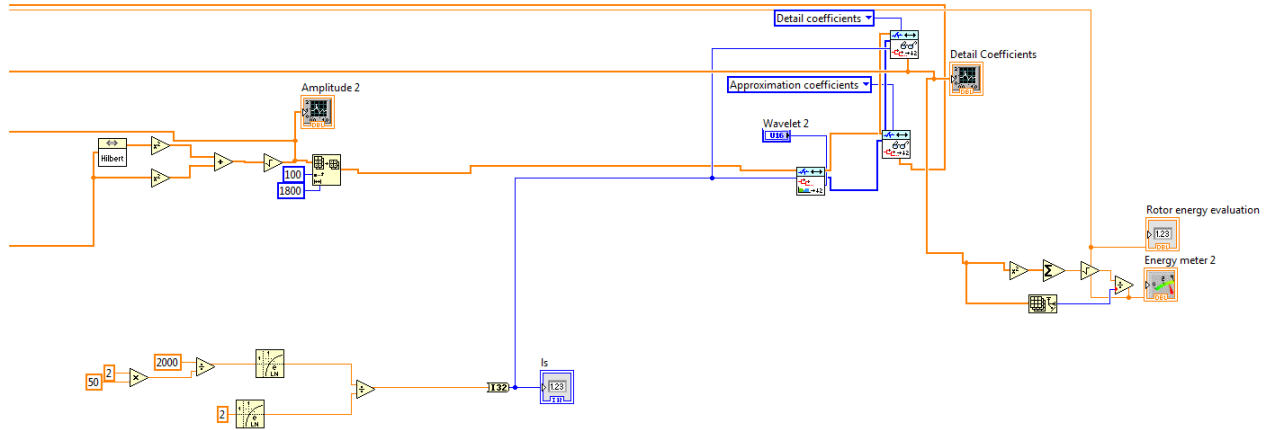




Figura 22. Programa final 2



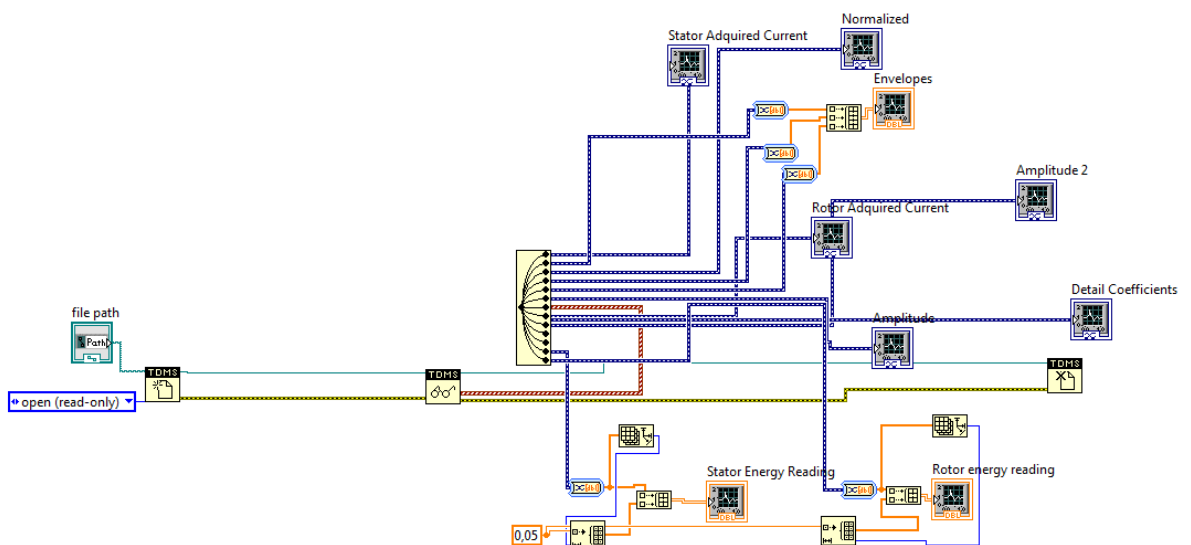
5. TEST Y RESULTADOS OBSERVADOS

Numerosos test han sido realizados con la máquina de inducción de rotor bobinado indicada al comienzo de este documento, en los cuales hemos variado la carga de la máquina alrededor de la velocidad de funcionamiento nominal, utilizando un sistema de frenado digitalmente controlado.

Arrancamos nuestro programa en un punto de funcionamiento sin carga o con una carga muy débil y aumentamos dicha carga lentamente hasta sobrepasar ligeramente el punto de funcionamiento nominal para acto seguido volver a la situación inicial progresivamente, en periodos totales de entre 10 y 15 segundos de test.

Para poder analizar los resultados y observar la evolución de los diferentes índices en el tiempo, creamos una VI (instancias que usa el programa LabView con interfaz propia cada una) complementaria para leer el archivo creado por nuestro programa de análisis y representar los diferentes gráficos de interés.

Figura 23. VI de lectura de resultados



La situación de desequilibrio de la máquina utilizada para los test nos ha impedido de obtener datos precisos de la situación sin fallo o desequilibrio, pero conocemos, debido a los valores observados durante el desarrollo de la aplicación, que la energía normalizada evaluada en la banda de frecuencia oscila entre 0,01 y 0,02 cuando no se observa defecto alguno.

Sin embargo, en los test finales observamos que el desequilibrio presente en la máquina hace los valores oscilar entre 0,05 y 0,06. Esto indica que un desequilibrio no demasiado grave está presente en la máquina, está aún puede continuar funcionando un cierto tiempo aunque no en condiciones perfectas. Sabemos gracias a la documentación consultada que un defecto grave puede causar valores que superen fácilmente el 0,1 en la evaluación de la energía.

Vemos en el test adjuntado como la máquina supera de forma casi permanente el valor de 0,5 marcado como límite de presencia de un fallo.

Figura 24. Test fallo rotórico

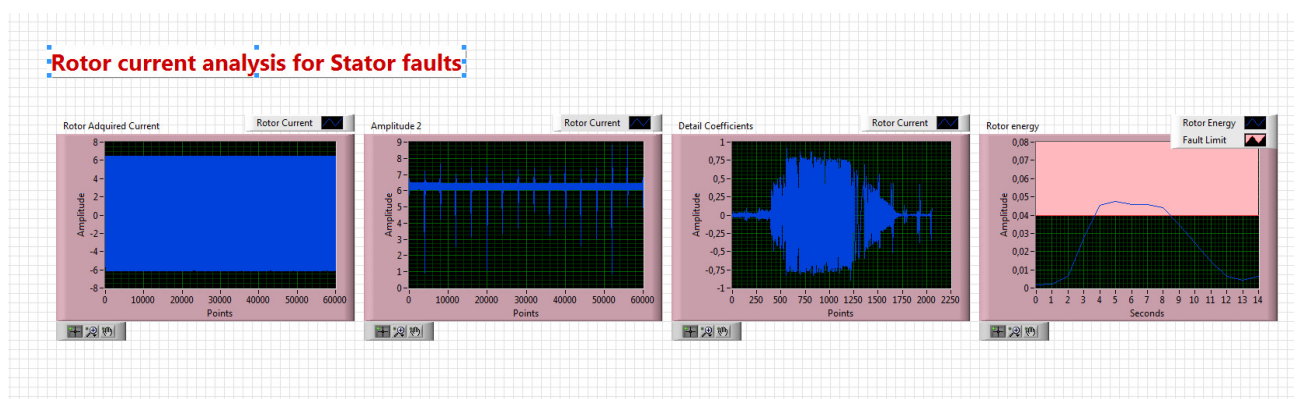


Para el fallo estático, como indicamos al principio, hemos debido utilizar un generador de señal para los test. Arrancamos el análisis con una frecuencia baja, de entorno a 5 Hz y la aumentamos lentamente hasta alcanzar la banda de frecuencia donde el fallo se manifiesta, y después seguimos aumentándola aún unos segundos más.

Observamos que cuando estamos lejos de la banda estipulada de energía que de acorde a la teoría es donde el fallo se presentará, la energía evaluada permanece entre 0 y 0,01 y después empieza a aumentar en cuanto nos acercamos a dicha banda.

Una vez en la banda el valor se estabiliza en torno a 0,05 y comienza a descender de nuevo cuando la frecuencia del generador sigue aumentando.

Figura 25. Test fallo estático



Por lo tanto podemos confirmar que la aplicación desarrollada parece funcionar correctamente como esperamos, a falta de tener una máquina disponible en condiciones perfectas y sin desequilibrios para poder realizar test completos y más detallados.



Vemos que la teoría aplicada parece ser correcta y los fallos se hacen efectivamente visibles en las bandas de frecuencia estipuladas.

El límite de 0,05 en la energía evaluada parece adecuado basándonos en los ensayos para determinar el inicio de la existencia de problemas en la máquina analizada, aunque según nuestro objetivo personal y el nivel de anticipación que deseemos obtener puede que queramos modificar este límite ligeramente, especialmente para diferenciar los fallos estáticos y rotóricos ya que los primeros presentan valores en general más bajos.



7. CONCLUSIÓN

El trabajo aquí realizado nos permite ver la viabilidad de una aplicación que aplique métodos de detección de fallos en el dominio industrial y su interés práctico. Un tema sobre el que aún quedan progresos importantes por realizarse con el futuro desarrollo de sistemas más flexibles capaces de hacer detecciones más precisas y en condiciones más extremas de funcionamiento, así como adaptarse a una mayor variedad de máquinas, condiciones de trabajo y equipos utilizados.

Personalmente, durante estos tres meses he tenido la oportunidad de descubrir y aprender sobre esta problemática, de detección de fallos en las máquinas de inducción durante condiciones de funcionamiento normales con carga variable, que está presente aún hoy en día.

Esta ha sido una oportunidad importante para mí, ya que me ha permitido por primera vez trabajar sobre una aplicación práctica con interés industrial, y descubrir y hacer frente a los problemas que se presentan en el desarrollo de una aplicación práctica, cuando queremos aplicar la teoría a la resolución de un problema real presente en las empresas del sector eléctrico.

Ha sido también una experiencia muy educativa, para mí personalmente, en el dominio de la programación dada mi falta de experiencia en el desarrollo de programas y aplicaciones informáticas, y también como una primera toma de contacto con el tema de la detección de fallos en las máquinas eléctricas, específicamente en este caso las máquinas de inducción de rotor bobinado, el cual nunca había tratado.



Titre du Project:

**DIAGNOSTIC EN TEMPS REEL DES DEFAUTS
ASYMETRIES DANS LES ENROULEMENTS DU STATOR
ET DU ROTOR DES GENERATEURS D'INDUCTION
(VERSION ORIGINAL)**

Tuteur Pédagogique:

Shahin Hedayati Kia

Présenté par :

Fernández Fernández, Javier

Amiens – France

Juin 2016



Remerciements

Avant commencer, je voudrais remercier à l'Université de Picardie Jules Verne et à le département d'Ingénierie Electrique pour m'accueillir cette année pendant mon séjour en France pour le fin de mes études. Et en spécial à le coordinateur de mon échange international, Senior Member IEEE Humberto Henao, pour le support offris pendant ma stance à cette université, et à mon encadrant et responsable pendant la réalisation de ce travail, Senior Member IEEE Shahin Hedayati Kia, pour l'opportunité de travailler ensemble et son collaboration à me développent personnel comme futur Ingénieur.

TABLE DES MATIERES

0. Liste des figures	p. 3
1. Introduction	p. 4
2. Méthode utilisé	p. 6
3. Développement du programme	p. 7
4. Programme définitif pour détection des défauts rotorique et statorique	p. 26
5. Test et résultats observés	p. 28
6. Conclusion	p. 32

LISTE DES FIGURES

Figure 1: Interface FPGA 1	p. 7
Figure 2: Interface FPGA 2	p. 8
Figure 3: Réception des données	p. 9
Figure 4: Lecture des données	p. 10
Figure 5 : Calcul de l'amplitude instantanée	p. 11
Figure 6: Obtention enveloppes 1	p. 13
Figure 7: Obtention enveloppes 2	p. 14
Figure 8: Analyse DWT	p. 16
Figure 9: Niveau de décomposition	p. 17
Figure 10: Evaluation de l'énergie	p. 17
Figure 11: Indicateur du défaut	p. 18
Figure 12: Calcul défauts statoriques 1	p. 18
Figure 13: Calcul défauts statoriques 2	p. 19
Figure 14: Obtention vecteur de courant	p. 20
Figure 15: Courants de phase et vecteur d'entrée	p. 21
Figure 16: Calcul temps d'exécution	p. 22
Figure 17: Enregistrement des résultats.....	p. 23
Figure 18: Interface Graphique 1	p. 24
Figure 19: VI de lecture de résultats	p. 24
Figure 20: Interface Graphique 3	p. 25
Figure 21: Programme final 1.....	p. 26
Figure 22: Programme final 2	p. 27
Figure 23: VI de lecture de résultats	p. 28
Figure 24: Test default rotorique	p. 29
Figure 25: Test défaut statorique	p. 30

1. INTRODUCTION

Les machines d'induction de rotor bobiné sont extensivement utilisées dans plusieurs applications industrielles. Comme tout équipement électromécanique, son utilisation continuée et le temps, risque de provoquer des variations dans son structure interne et son fonctionnement, lesquelles finiront pour causer une faille dans la machine.

L'apparition d'une faille a comme conséquence directe l'arrêt forcé de la machine et du processus lié à celui-ci, et risque d'endommager des autres installations et utilisateurs qui peuvent se rencontrer proches. Tout ça entraîne d'habitude un cout économique important et un intérêt général d'éviter cette situation pour rassurer la sécurité des travailleurs et la stabilité et bon fonctionnement des processus industriels critiques qui peuvent dépendre de ces machines.

Pouvoir prévoir des failles et agir avant qu'elles ont lieu deviens spécialement dans quelques applications, comme la production d'énergie éolique, ou cette type de machine est très présente dans les routeurs éoliques et ceux sont d'habitudes placés dans des locations de accès difficile. Ca nous permet de mieux programmer les maintenances pour éviter le remplacement des machines coûteuses et en même temps éviter des arrêts longs que n'étaient pas anticipés, en évitant la perte d'argent associé ainsi que des autres effets nocives comme la variation dans la production estimée d'énergie avec les répercussions que ça cause dans un réseau électrique.

Donc, dans le cadre de ce projet, on a comme objectif développer une application en temps réel, à l'aide de le logiciel LabView et de le system embarquée CompactRIO (cRIO 9068) de « National Instruments », capable de détecter les failles et déséquilibres présents dans un machine d'induction de rotor bobinée en utilisant la méthode développé dans le papier « Windings Monitoring of Wound Rotor Induction Machines

under Fluctuating Load Conditions » par le Département d'Ingénierie Electrique de l'Université de Picardie Jules Verne.

Cette application doit nous permettre de détecter en temps réel la présence d'un défaut bien statorique ou rotorique à l'utilisation et traitement des courants rotoriques et statoriques respectivement, l'afficher de façon facilement reconnaissable par un operateur et aussi de enregistrer l'historique de fonctionnement pour son analyse postérieur.

L'application aurait du être validé avec une machine d'induction de routeur bobiné de 90W et 1430 rpm sur le quelle on simule des défauts en causant des déséquilibres avec manipulation des phases et introduction des résistances. Pourtant la machine utilisé présenté des déséquilibres donc la situation sans défaut n'a pas été complètement testé (mais il y a des donnés obtenus pendant le développent de l'application) et les défauts statoriques ont du être validés à l'utilisation d'un générateur de signal pour valider le défaut.

2. METHODE UTILISE

Une fois la problématique est présenté, il faut voir qu'est qu'on doit analyser et utiliser comme indicateur de la présence d'un défaut et pouvoir réaliser un application qui traite les donnés pour pouvoir à la fin obtenir et afficher clairement cet indicateur et montrer la présence d'un défaut.

Les études avant lieu sur ce métier, et spécialement le papier qu'on utilise comme référence pour l'étude des défauts et le développent de cette application, ont établi que la présence d'un déséquilibre est révélé par l'effet dans un certain bande de fréquence dans les courants statoriques et rotoriques de la machine à analyser.

Pour le défaut rotorique, on doit observer sur le courant statorique la bande :

$$[0 \text{ Hz}, 2k_r s_r f_s \text{ Hz}]$$

Et pour le défaut statorique sur le courant rotorique :

$$[(2 - s_r) f_s \text{ Hz}, 2f_s \text{ Hz}]$$

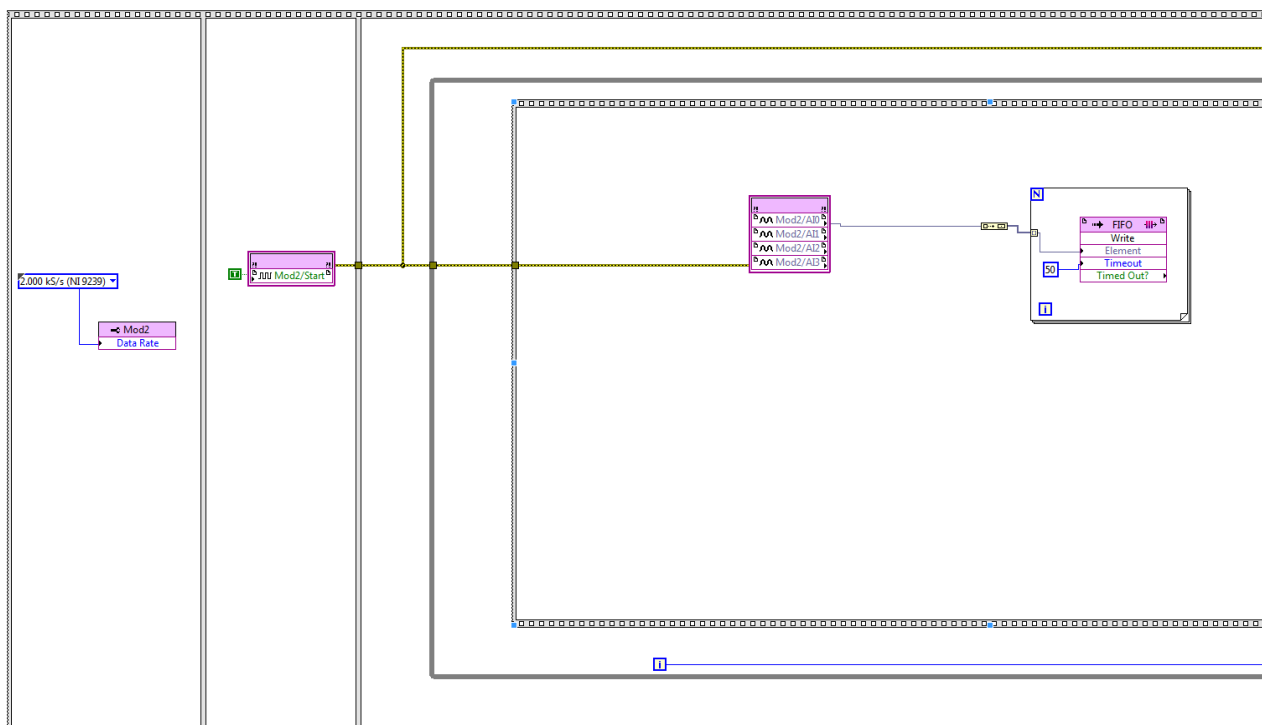
Avec $k_r = \text{Integer} \left[\frac{1-s_r}{2ps_r} \right]$, « sr » le glissement nominal, « fs » la fréquence fondamentale et « p » le nombre de paires de pôles.

Donc dans notre programme on va réaliser une acquisition des courants du moteur pour après le traiter, normaliser et effectuer une analyse DWT pour obtenir les bandes de fréquences associées au défaut et évaluer le niveau d'énergie continument dans celles-ci pour détecter la présence d'un défaut ou déséquilibre.

3. DEVELOPPEMENT DU PROGRAMME

Avant commencer avec le programme de traitement, une partie d'acquisition des données a été réalisé. On a utilisé une interface FPGA pour pouvoir renvoyer les courants reçus par la cible (cRIO 9068) dans le logiciel pour après l'utiliser comme données d'entrée pour notre programme avec une fréquence d'échantillonne de 2000 Hz.

Figure 1. Interface FPGA 1

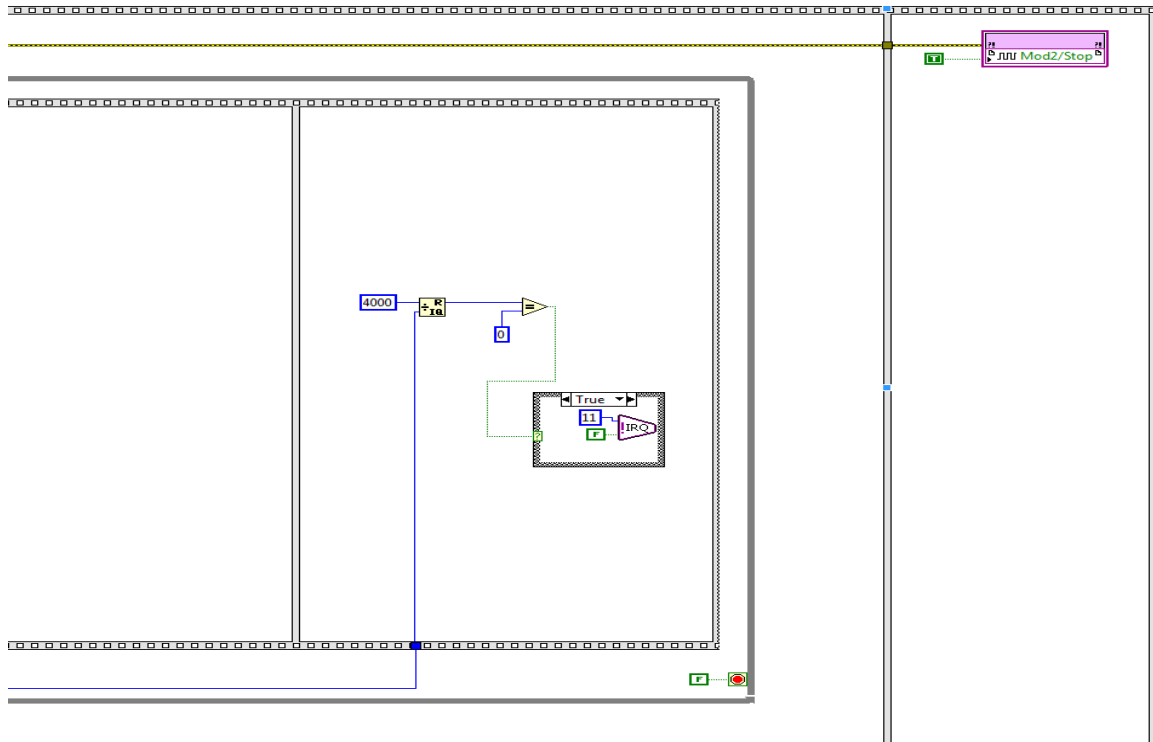


On store l'information dans un FIFO qu'après on peut utiliser pour récupérer les données dans des autres instances de notre projet sur LabView.

On a choisi de faire chaque cycle comprendre de 4000 points des données, avec notre fréquence d'échantillonnage ca nous donne 2 secondes de données affichés par boucle

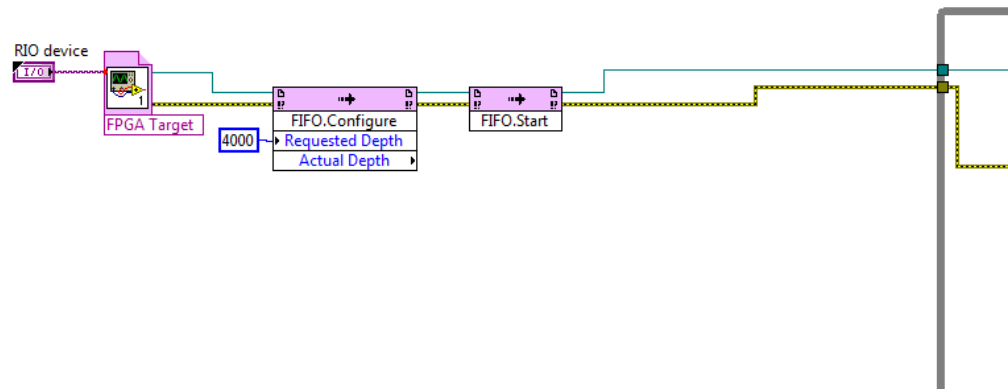
de fonctionnement. Une période de temps suffisant pour avoir des relèves importants et aussi faciles à suivre s'il y a un opérateur qui doit surveiller le fonctionnement.

Figure 2. **Interface FPGA 2**



Une fois dans le fichier où notre programme d'analyse va se trouver il faut faire un appel à la cible et récupérer les données du FIFO avant stockés merci à l'interface FPGA. Donc on fasse l'appel à la cible et on initialise le FIFO.

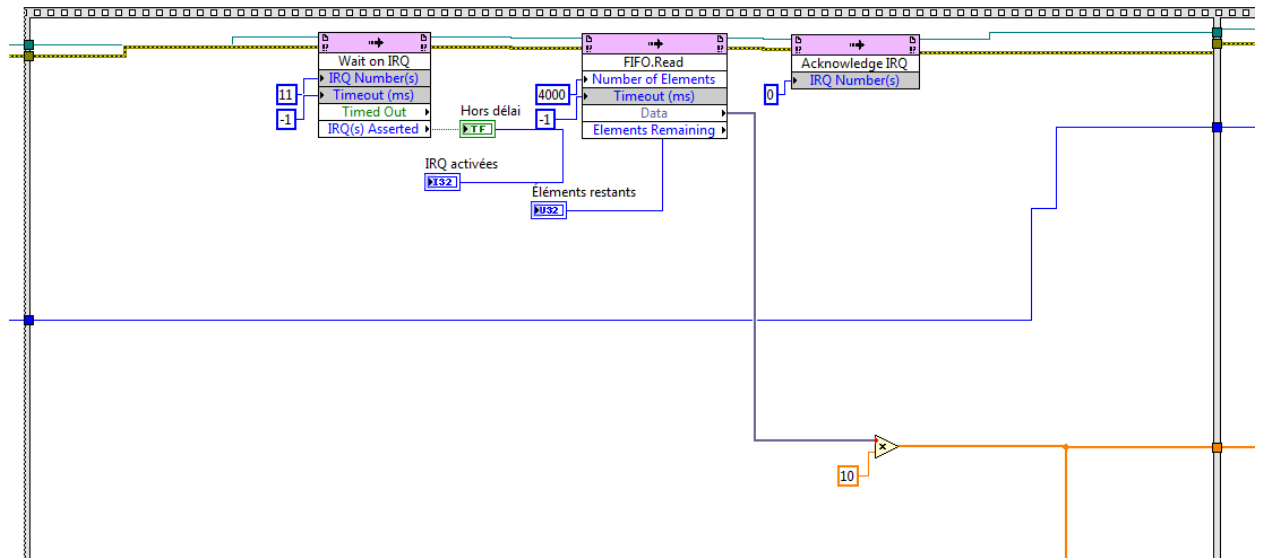
Figure 3. Réception des données



On demande les 4000 point des données pour afficher tout ce qui est reçu par la cible et pour avoir une acquisition continue et ininterrompue.

Après on rentre dans la première étape dans le boucle de programmation où on lit le FIFO avant initialisé et on finit la configuration de l'acquisition.

Figure 4. Lecture des données



On ajoute une amplification avant de passer à la deuxième étape du boucle ou la partie de traitement se trouve, pour escaler les données et travailler avec les valeurs réelles en compensant l'échelle introduite par les sondes de courant.

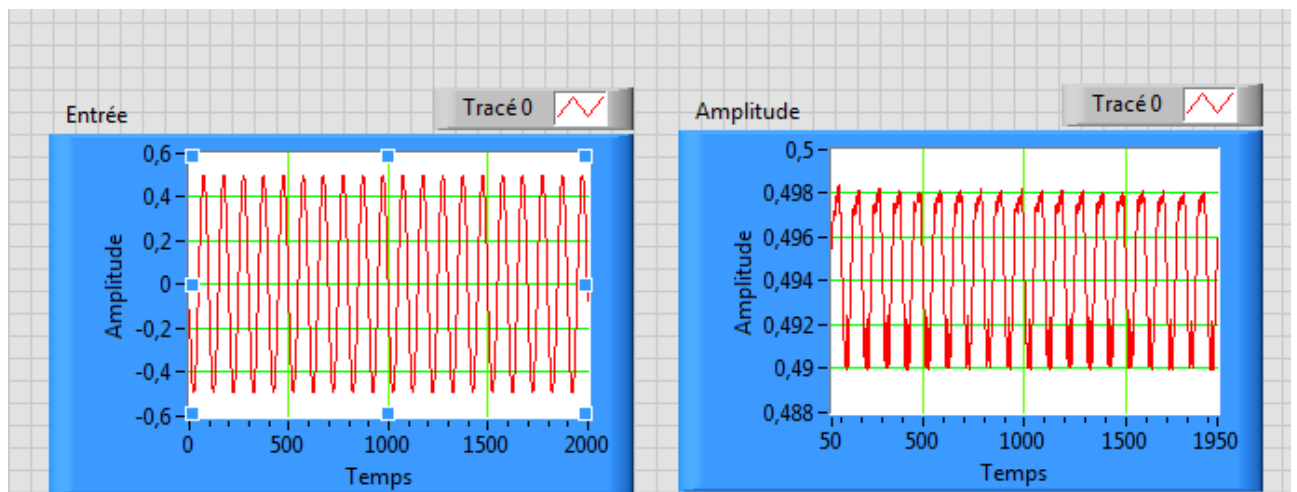
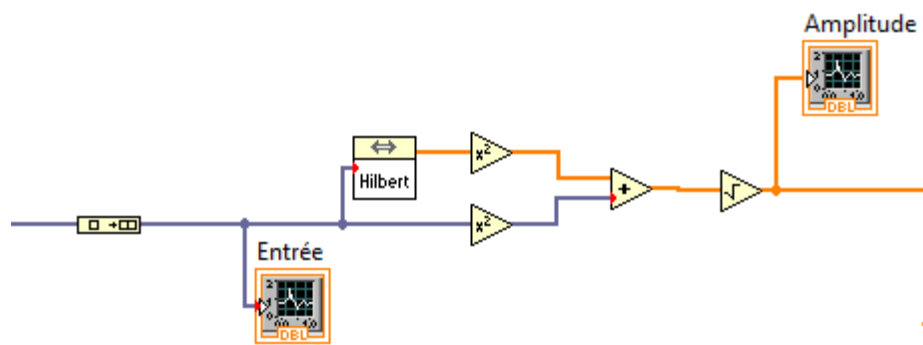
On a choisi ce type de boucle en deux étapes pour assurer que l'acquisition et lecture des données d'entrée a lieu correctement avant l'exécution de le programme d'analyse.

Maintenant, on va commencer à revoir les plusieurs étapes de développement du programme de détection des défauts propre. Pourtant, il faut noter qu'un générateur de signal était utilisé au début pour vérifier le fonctionnement de chaque étape.

On démarre avec la partie du code orienté à l'analyse du courant statorique pour la détection des défauts rotoriques.

Le premier objectif c'était de retrouver l'amplitude instantanée du courant d'entrée. Pour ça on a utilisé le transformé de Hilbert et des autres bloques arithmétiques pour implémenter le calcul.

Figure 5. Calcul de l'amplitude instantanée



Une fois on a bien l'amplitude instantanée du signal d'entrée il faut normaliser le signal pour l'évaluation de l'énergie.

Par contre, dans le cas du courant statorique, on trouve une problématique. La présence d'un composant continue du courant que va varier dans le temps dans des conditions de charge variable. Donc, pour pouvoir avoir un bon analyse dans ces conditions on a besoin de pouvoir déterminer et enlever cette composant continue dynamiquement au lieu de utiliser une méthode de moyennage statique.

Pour ça, et comme c'est conseillé dans le papier, on a décidé de trouver les enveloppes supérieur et inférieur de cette signal et après trouver la moyenne pour enlever cette composant continue.

Pour construire ces enveloppes on a besoin de combiner plusieurs fonctions présentes dans le logiciel.

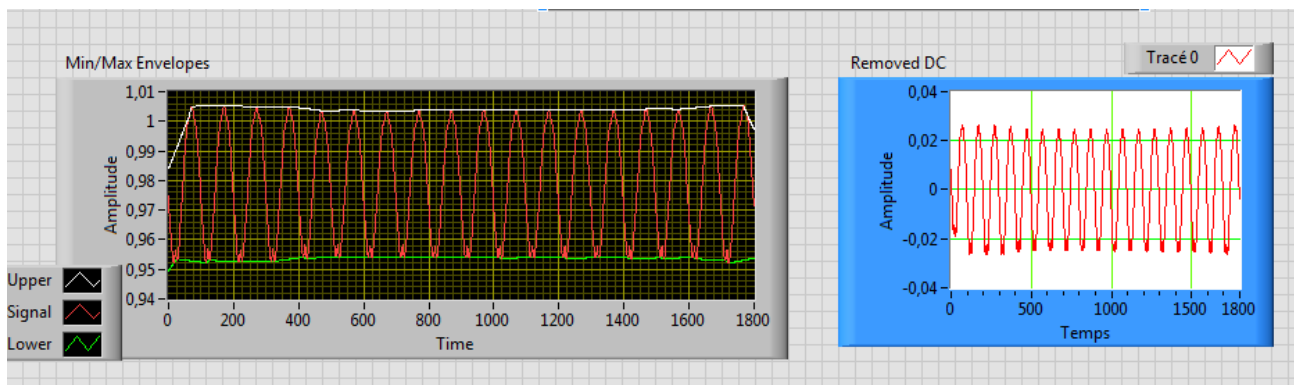
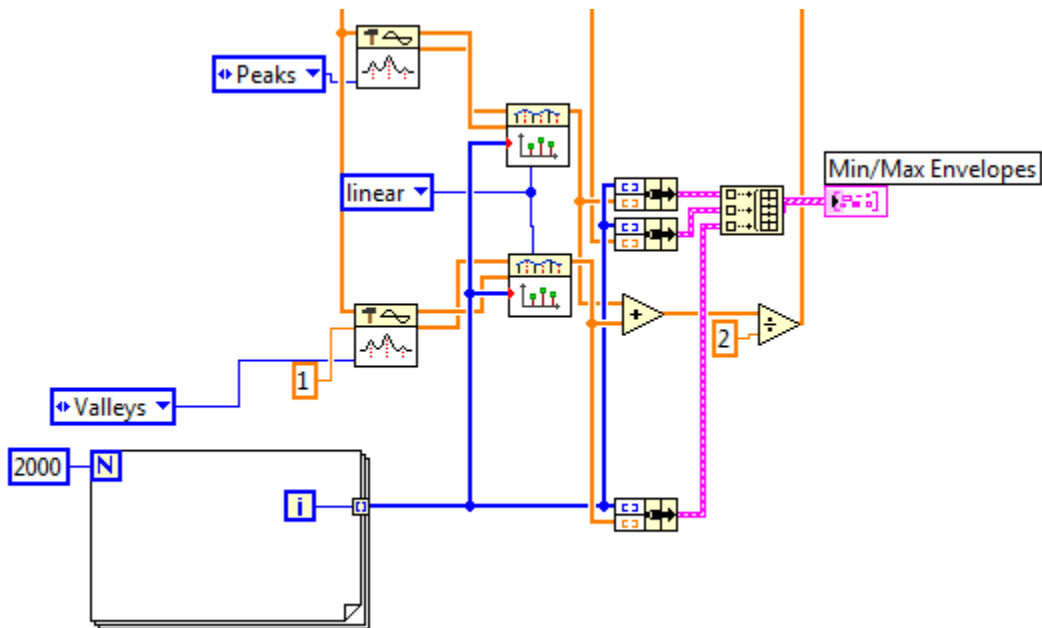
On a un premier bloc qui est configuré pour détecter les crêtes ou les vallées. Il utilisé un critère de détection max/min qui est configuré pour détecter les crêtes au dessus d'un limite ou les vallées dessous ce limite.

On peut aussi configurer un minimum de longueur pour qu'une crête ou vallée soit détecter et éviter d'avoir faux détection de petites crêtes causes par le bruit ou des fréquences plus élevés.

Ce bloque nous donne comme sortie deux arrays avec les valeurs de les crêtes ou vallées et ses positions respectivement. Pour après construire les enveloppes il faut ajouter un boucle for pour créer un array de la longueur des donnés qu'on va utiliser et utiliser des blocs des interpolations pour relier les valeurs et position des ces crêtes et vallées et finalement compléter chaque enveloppe.

Après on rajoute les trois signaux, le signal à analyser et les deux enveloppes construites pour les afficher sur un même diagramme et vérifier le fonctionnement.

Figure 6. **Obtention enveloppes 1**



Pourtant, cette étape a démontré être la plus complexe et a donné beaucoup des complications une fois on a commencé à travailler sur la machine réelle.

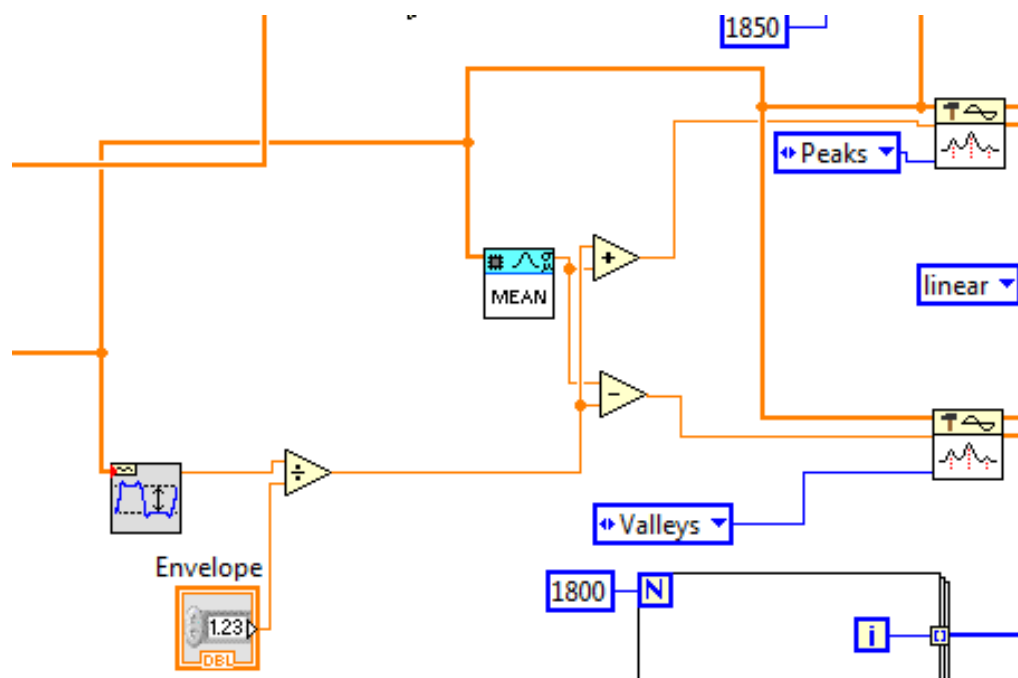
La détection des crêtes et vallées est devenue faillante et beaucoup des modifications on était lieu jusqu'à la configuration finale.

Il était nécessaire de déterminer le limite de détection de façon dinarique en utilisant la moyenne de la signal analysé et un calibration arithmétique que a du être testé en modifié en plein d'occasions, pour pouvoir se adapter à des différents signaux et variations.

On avait essayé aussi de limiter la longueur de l'analyse et de l'enveloppe construit quand on s'a rendu compte que les erreurs c'étaient concentrés sur tout aux extrêmes de l'enveloppe.

Mais ca cause autant des problèmes comme des solutions parce qu'on a vu que ca introduisait aussi des décalages entre les enveloppes et la signale donc la plus part était enlevé à la fin.

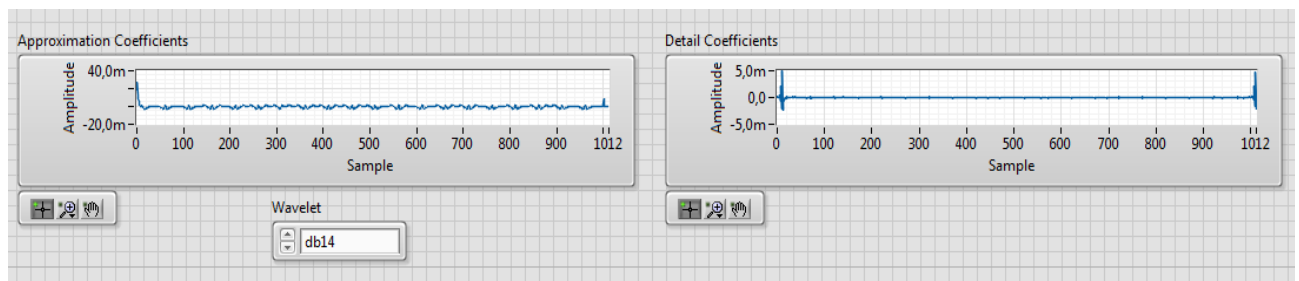
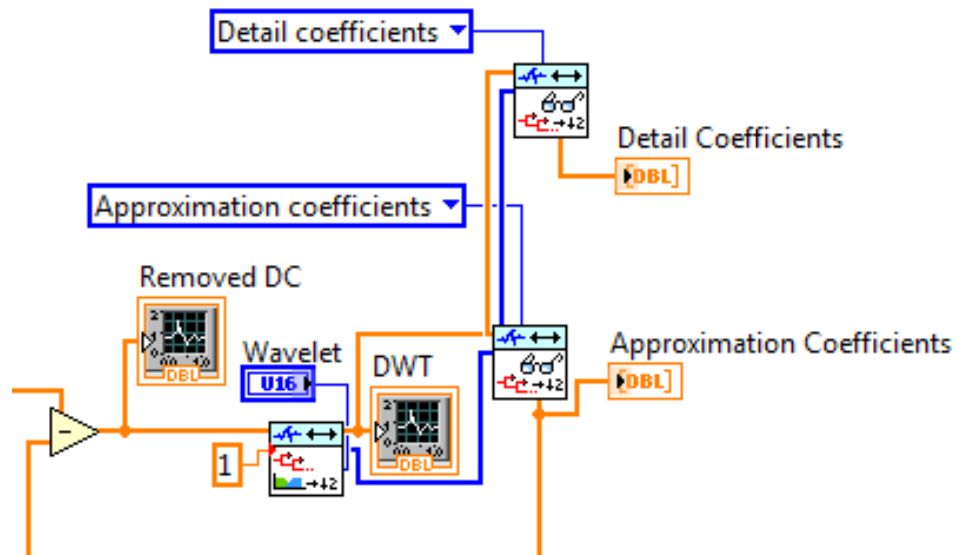
Figure 7. **Obtention enveloppes 2**



Une autre chose essayé c'était d'avoir une signale plus propre pour la meilleure détection et calcul de la vraie valeur continue. On a considéré utiliser le faire en deux étapes avec quatre enveloppes mais à la fin ce qui a démontré être plus effectif c'était changer l'ordre du traitement des donnés dans le programme en faisant le analyse DWT avant et après obtenir les enveloppes du signal déjà filtros, plus propre et sans les hautes fréquences qui causent détection des crêtes et vallées erronés.

Le suivant étape c'était la réalisation de l'analyse DWT. Dans le logiciel on trouve cette fonction incorporée en utilisant deux bloques. Un bloque sur lequel on indique le niveau et procédure utilisée dans la décomposition et applique le DWT et une autre qu'on utilisé pour lire et pouvoir afficher les résultats de ce premier bloque. On a utilisée le Daubechies 14 parce que le Daubechies c'est une méthode que donne des bonnes résultats dans cette type d'applications d'accord à les références qu'on utilise et le 14 c'est le plus précis que le logiciel LabView est capable de travailler avec.

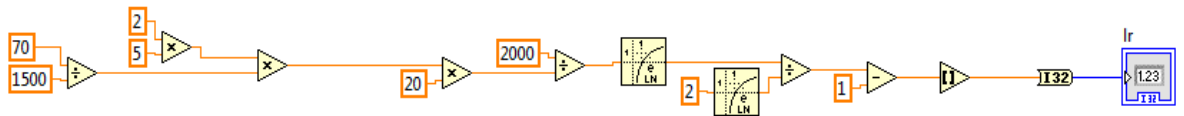
Figure 8. Analyse DWT



Dans c'est cas pour détecter les fautes rotoriques il nous intéresse les coefficients d'approximation pour trouver la bande de fréquence où le déséquilibre serait visible, et c'est ca ce qu'on va utiliser après pour l'évaluation de l'énergie.

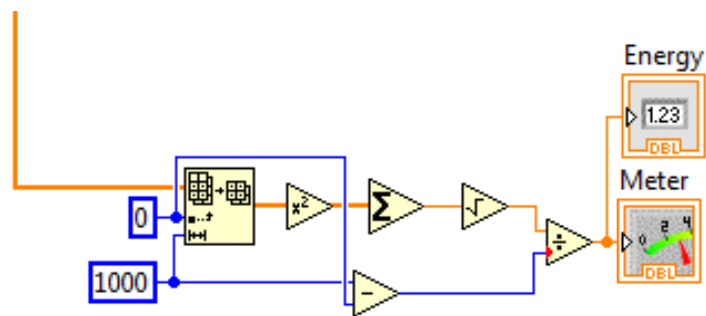
Le niveau de décomposition souhaité va dépendre des caractéristiques de la machine à être surveillé et de la fréquence fondamentale et d'échantillonnage. Donc pour le retrouver plus facilement une partie du code a été ajouté pour le calculer automatiquement en introduisant seulement les valeurs de partie et qu'on connaît.

Figure 9. Niveau de décomposition



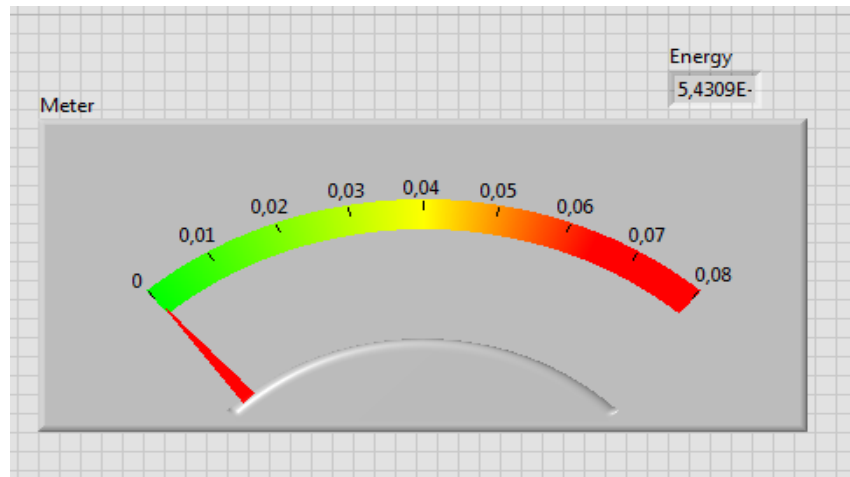
Pour finir cette partie du programme on ajoute le calcul de l'évaluation de l'énergie du signal obtenu après le traitement.

Figure 10. Evaluation de l'énergie



On a ajouté un indicateur visuel pour avoir une référence plus visible du niveau d'énergie et pouvoir identifier d'un coup d'œil la présence d'un défaut ou déséquilibre.

Figure 11. Indicateur du défaut



Une fois cette partie est fini on a suivi une démarche similaire pour implémenter le calcul des défauts statoriques à partir du courant rotorique. Dans ce cas le processus a été beaucoup plus simple et rapide à cause de n'avoir pas besoin d'enlever la composant continue de la courant analysé.

La principale différence c'est l'utilisation du coefficient du détail au lieu du ce d'approximation dans le DWT. Donc on n'ajoute pas une explication détaillé.

Figure 12. Calcul défauts statoriques 1

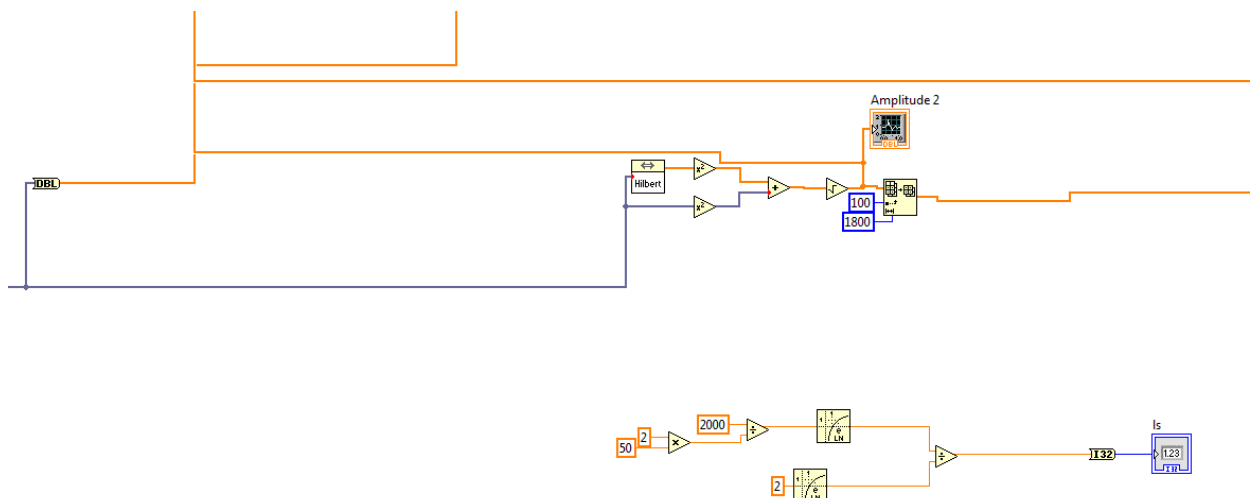
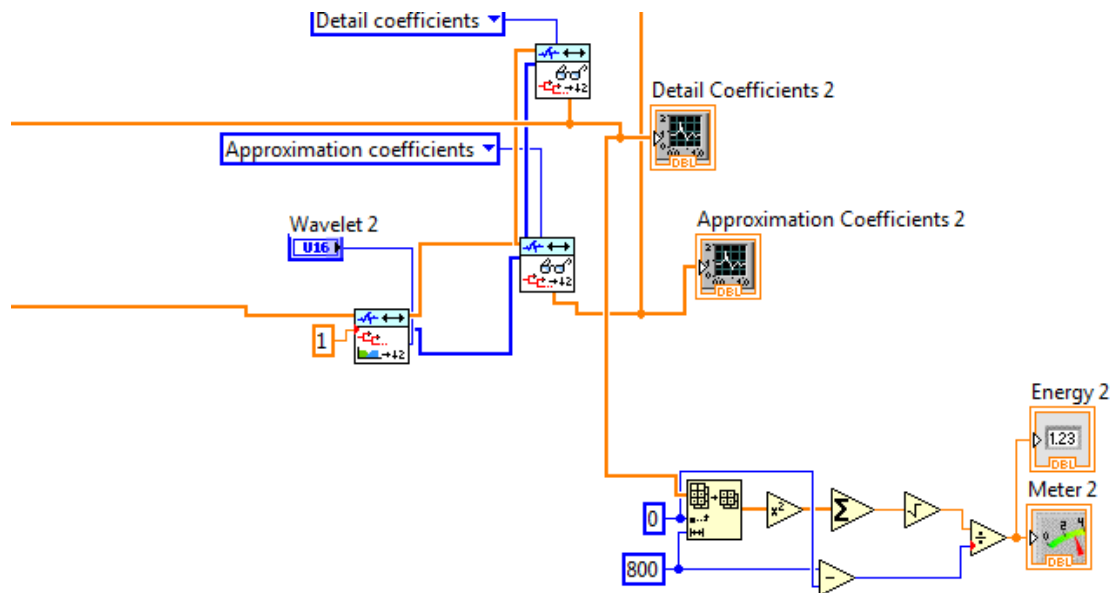
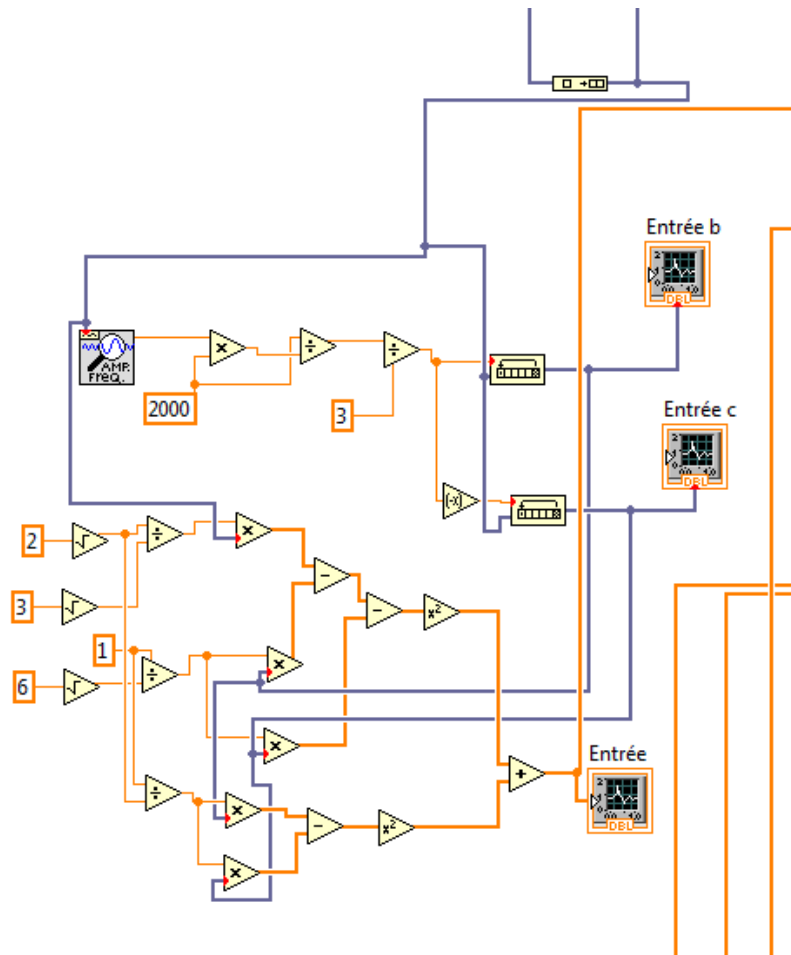


Figure 13. Calcul défauts statioriques 2



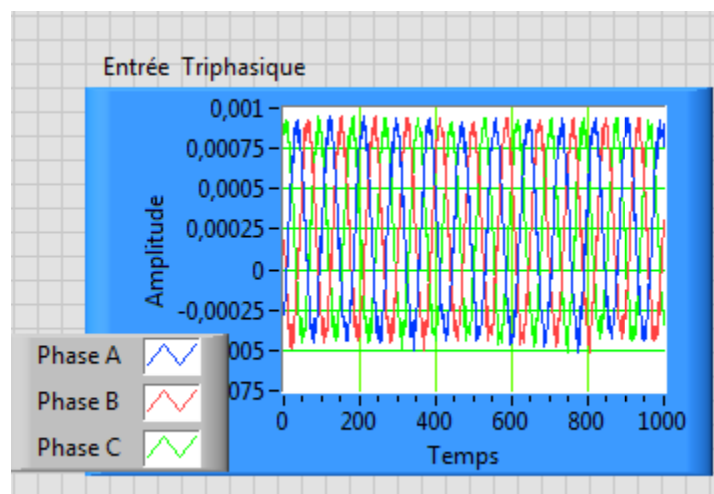
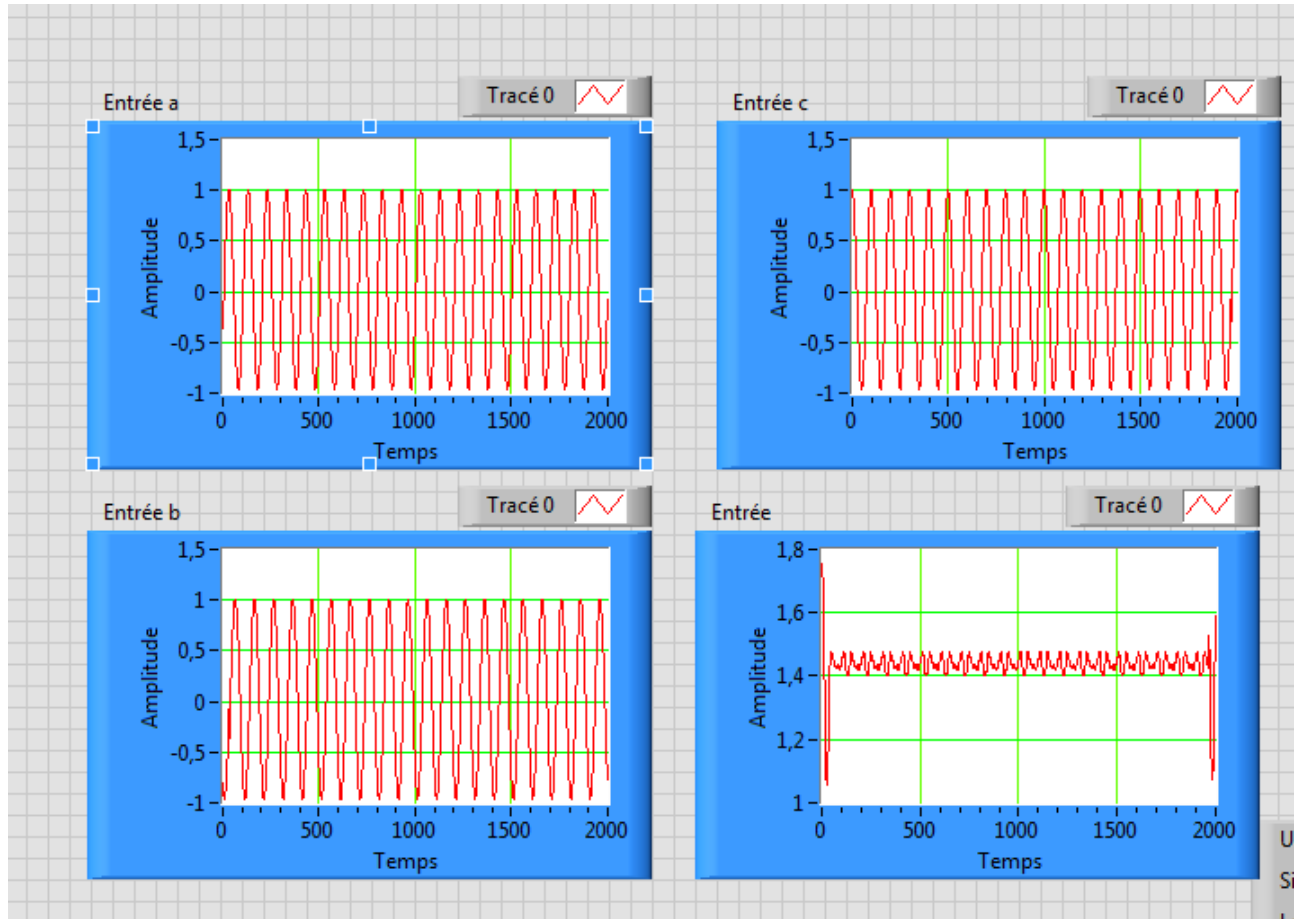
Une fois la base du programme était réalisé on a travaillé sur l'entrée du programme pour la préparer pour une entrée triphasique. Du code était ajoutée pour simuler l'entrée triphasique avec le générateur de signal et pour calculer le vecteur du courant d'entrée à partir de ce courant triphasique au lieu d'utiliser la phase directement.

Figure 14. Obtention vecteur de courant



On utilisé principalement des bloques arithmétiques pour le calcul du vecteur d'entrée et des bloques pour détecter la valeur instantanée du signal et le déplacer 120 degrés pour créer les autres phases.

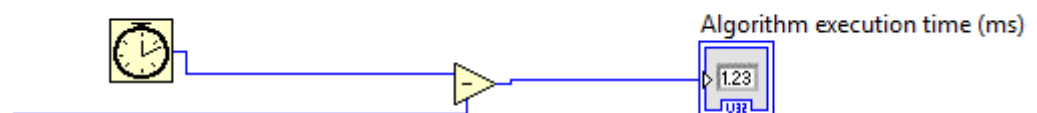
Figure 15. Courants de phase et vecteur d'entrée



Deux fonctions complémentaires ont été introduites dans le programme. La première avec l'objectif de pouvoir calculer le temps d'exécution du programme et détecter s'il y a un mal fonctionnement dans celui-ci pour exemple si l'exécution deviens trop lente pour être capable de faire les calculs en temps réel.

Pour ça on a utilisé le bloque « Tick count » :

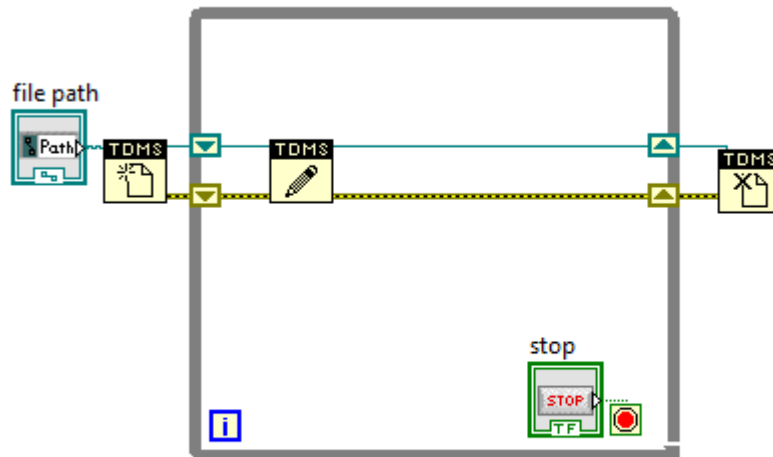
Figure 16. Calcul temps d'exécution



L'autre fonction a comme objective ajouter un système d'enregistrement des données intéressants pour pouvoir être capable de les afficher comme graphiques et effectuer un analyse à posteriori hors de la visualisation directe.

Pour ça on a utilisé les bloques TDMS pour créer un fichier dans ce format (ou ouvrir un si ceci existe déjà) écrire les différents données pendant que le boucle « while » d'exécution du programme est active, et fermer le fichier à la fin du boucle du fonctionnement.

Figure 17. Enregistrement des résultats



Pour finaliser, on a travaillé sur l'interface visuelle pour que les informations soient proprement affichées et rangés pour discerner facilement qu'est ce que ce chaque chose et séparer les parties du contrôle du routeur ou stator.

La on trouve aussi le champ pour indiquer le lien où on veut le fichier avec les donnés enregistrées, ainsi comment le déroulante où ont indique la cible laquelle on veut utiliser pour l'acquisition des donnes au moment. On peut aussi modifier la méthode utilisée pour l'analyse DWT si on trouve qu'un autre s'ajuste mieux aux nôtres conditions de travail. Finalement on trouve le bouton pour arrêter le programme, tant la partie d'acquisition comme la partie des calculs.

Figure 18. Interface Graphique 1

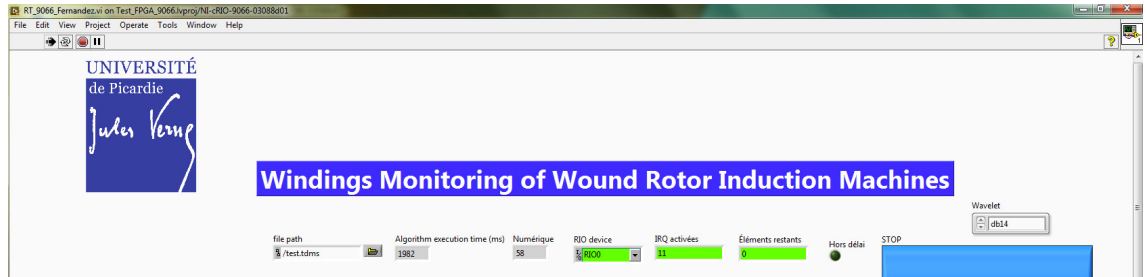


Figure 19. Interface Graphique 2

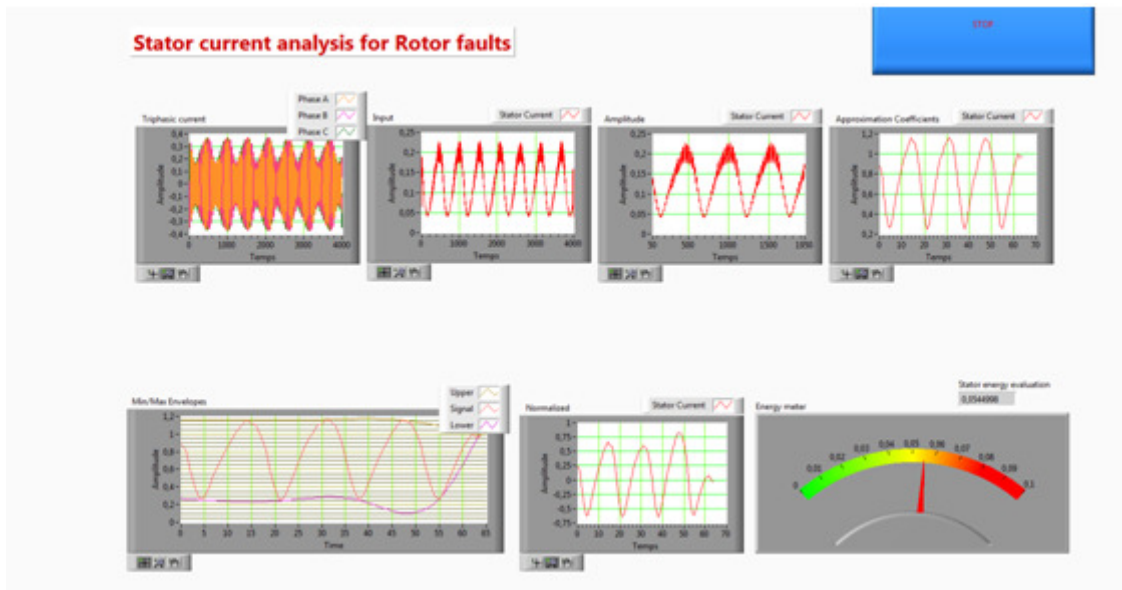
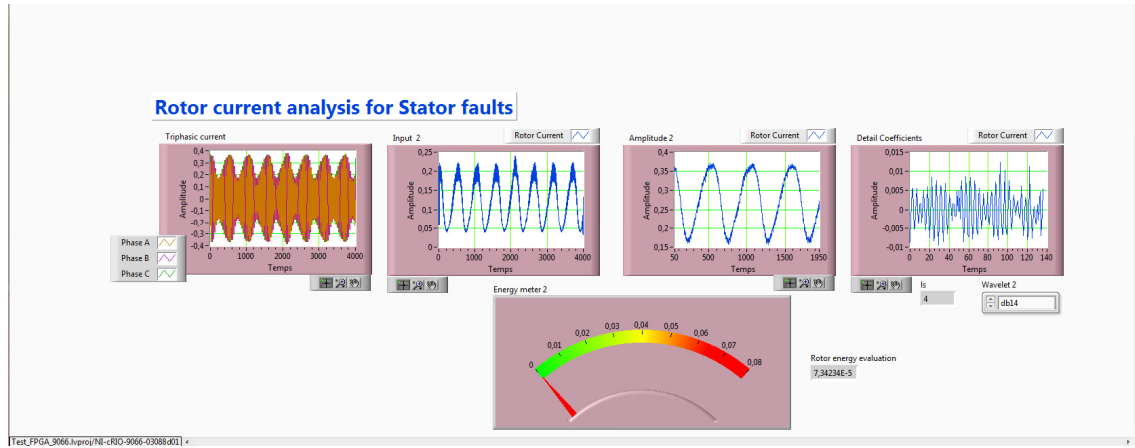


Figure 20. Interface Graphique 3



4. PROGRAMME DEFINITIF POUR DETECTION DES DEFAUTS ROTORIQUES ET STATORIQUES

Pendant les tests avec la machine d'induction plusieurs modifications ont eu lieu. Spécialement pour la configuration du calcul et construction des enveloppes dans la partie de détection de défauts rotoriques, mais des autres petites modifications ont eu lieu aussi comme l'utilisation d'un bloque qui permet obtenir à la sortie les dimensions de l'array d'entrée. Comme ca si on modifie la longueur de l'array de donnés pour nous centrer sur la partie intéressant des donnés que sont pas déformées pour la traitement cette modification est transmise au reste de programme qu'a besoin d'avoir en compte cette longueur au lieu de devoir modifier les valeur multiples fois et risquer inconsistances.

Figure 21. Programme final 1

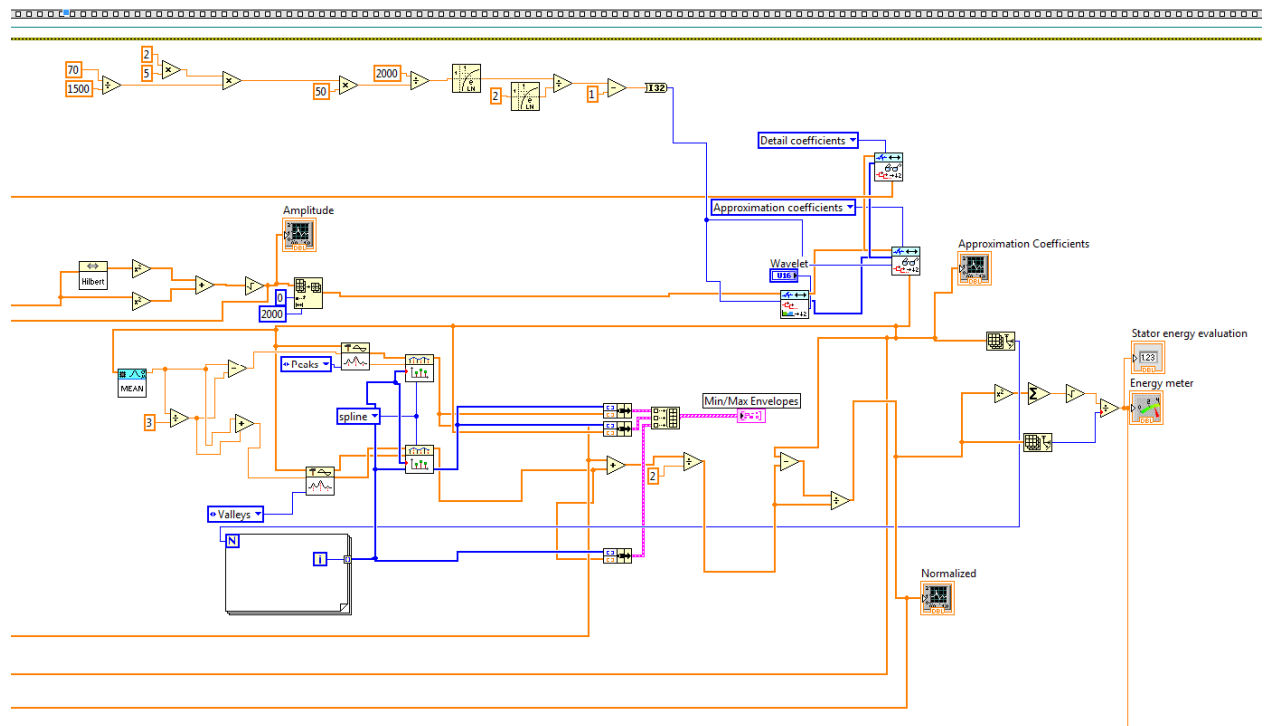
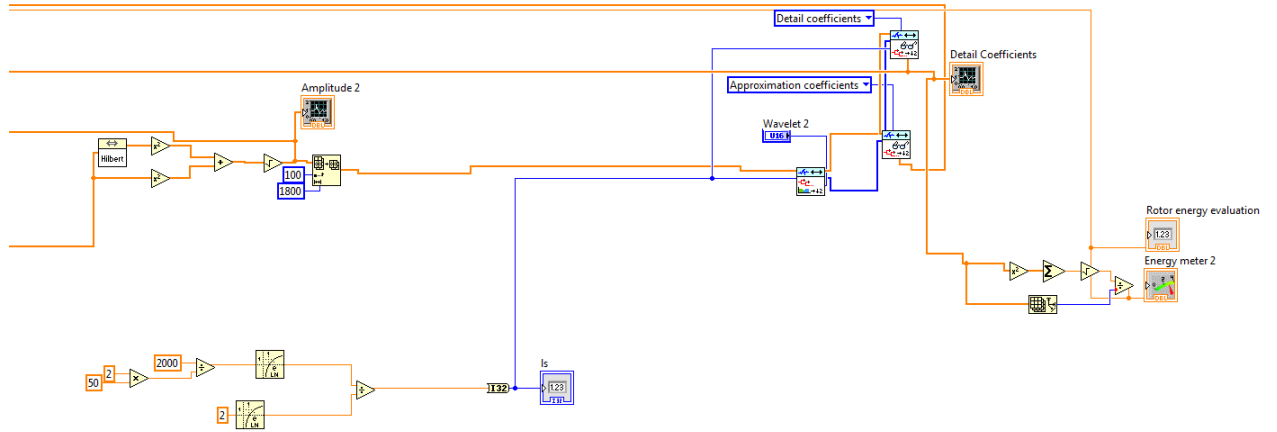


Figure 22. Programme final 2



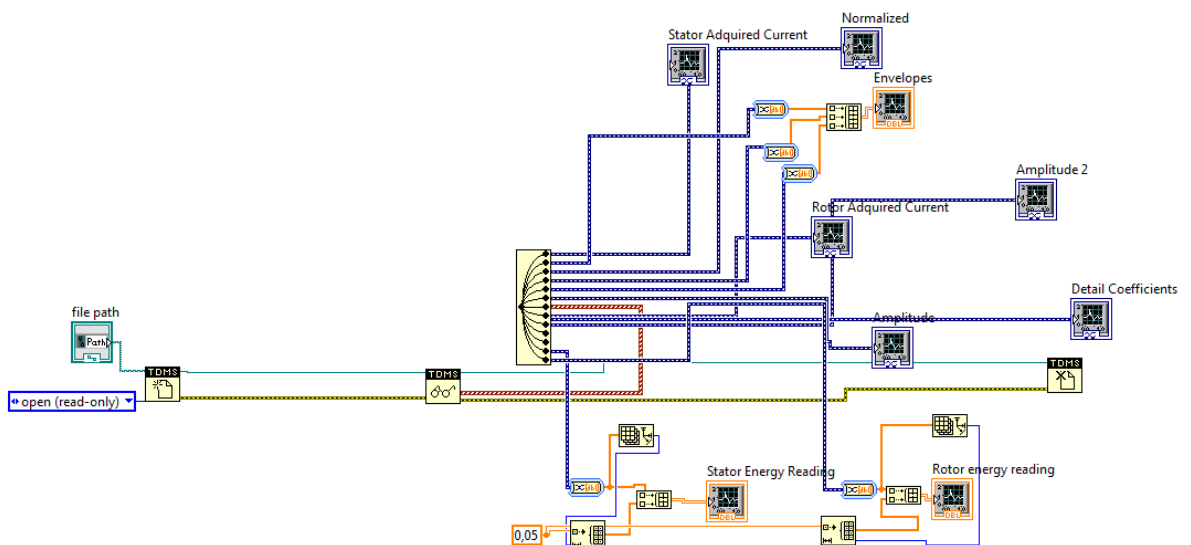
5. TEST ET RESULTATS OBSERVES

Plusieurs tests ont été réalisés, avec la machine d'induction de rouleur bobiné indiquée au début de ce document, dans lesquelles on a varié la charge de la machine autour de la vitesse nominale de fonctionnement à l'aide d'un système de freinage digitalement contrôlé.

On démarre le programme au point de fonctionnement sans charge ou avec une charge très faible et on augmente la charge en dépassant légèrement le point de fonctionnement nominale pour après retourner à la situation initiale dans des périodes de 10-15 secondes.

Pour pouvoir analyser les résultats et observer l'évolution des différents indices dans le temps, un VI complémentaire a été créé pour lire le fichier créé par le programme de analyse et afficher les différents graphiques remarquables.

Figure 23. VI de lecture de résultats



La situation de déséquilibre de la machine utilisée pour les tests à empêché d'obtenir un rapport précis de la situation sans défaut, mais on connaît par rapport à les valeurs observés pendant le développement du programme que la valeur de l'énergie normalisé évalué dans la bande de fréquence oscillé entre 0,01 et 0,02.

Par contre dans les tests finals on a observé que le déséquilibre présenté dans la machine cause ces valeurs osciller entre 0,05 et 0,06. Pourtant ca s'est qu'un déséquilibre qui n'est pas très important et la machine peut continuer à fonctionner encore longtemps même si n'est pas parfaitement. Un défaut plus sévère de ce type peut facilement dépasser le 0,1 dans l'évaluation de l'énergie.

On voit dans le test comme la machine dépassé presque permanemment le 0,5 que on a marqué comme limite de présence du défaut.

Figure 24. Test défaut rotorique

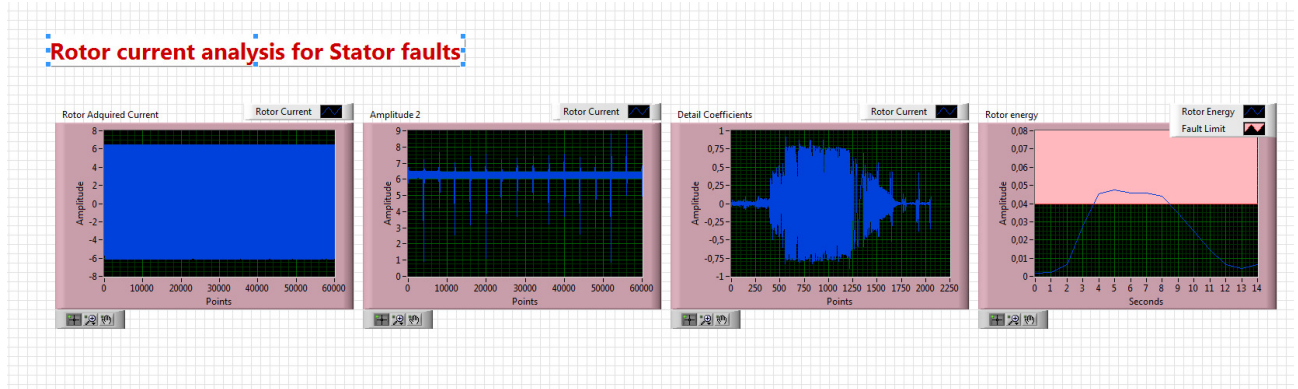


Pour le défaut rotorique, comme indiqué au début on a du utiliser un générateur de signal. On a démarré l'analyse avec une fréquence environ 5 Hz et on a monté lentement jusqu'à la bande de fréquence ou le défaut se manifeste et après encore monté après ceci.

On observe comme quand on est très loin de la bande stipulé l'énergie reste entre 0 et 0,01 et après ca commence à monter quand on s'est rapproche à la bande de défaut.

Un fois dans la bande la valeur s'stabilise autour 0,5 pour commencer à descendre une autre fois quand la fréquence dans le générateur monte encore.

Figure 25. Test défaut statorique



Donc on peut confirmer que l'application semble fonctionner correctement à faute d'avoir une machine disponible et en des conditions parfaites et sans déséquilibres ou défauts pour réaliser des tests complètes et plus détaillés.

On voit que la théorie semble être correcte et les défauts sont effectivement montrés dans les bandes de fréquences attendues.

Le limite de 0,5 dans l'énergie évalué semblé correct par rapport à les tests pour indiquer le début d'existence des problèmes, mais selon l'objectif, la présence de déséquilibres qu'on peut accepter et l'anticipation qu'on veut pour la détection de ces défauts il peut être qu'o n veut monter ce limite un peu pour les défauts rotoriques ou les descendre pour les statoriques à cause de que ces derniers ont d'habitude de valeur plus faibles

7. CONCLUSION

Le travail ici réalisé nous a permis de voir la faisabilité de l'application des méthodes de détection des défauts dans le domaine industrielle et son intérêt pratique. Un sujet sur lequel il reste encore des progrès importants à réaliser avec le futur développement des systèmes plus flexibles capables de faire des détections plus précises dans des conditions de fonctionnements plus extrêmes ainsi comme adaptées à une plus ample assortiment des machines, conditions de travail et équipements utilisées.

Personnellement, pendant ces trois mois j'avais l'opportunité de découvrir et apprendre sur cette problématique, de détection de défauts dans les machines d'induction pendant des conditions normales de fonctionnement avec charge variable, qui est présente encore aujourd'hui.

C'était une opportunité importante pour moi parce que c'a m'a permis de travailler par première fois sur une application pratique avec intérêt industrielle, ainsi comment découvrir et devoir faire face aux problèmes qui présente le développement et application pratique, quand on veut appliquer la théorie à la résolution des problèmes réelles et présentes dans les entreprises du secteur électrique.

C'est aussi une expérience très éducative, pour moi personnellement, dans le domaine de la programmation face à mon manque d'expérience dans le développement de programmes et application informatiques et aussi comme ma première prise de contact avec le sujet de détection de défauts sur les machines électriques, et spécifiquement pour les machines d'induction à rotor bobiné.